CS224W: Analysis of Networks **Fall 2017**

Problem Set 3

Due 11:59pm PDT November 9, 2017

General Instructions

These questions require thought, but do not require long answers. Please be as concise as possible. You are allowed to take a maximum of 1 late period (see the information sheet at the end of this document for the definition of a late period).

Submission instructions: You should submit your answers via GradeScope and your code via the SNAP submission site.

Submitting answers: Prepare answers to your homework in a single PDF file and submit it via GradeScope. The number of the question should be at the top of each page. We recommend that you use the submission template latex file to prepare your submission.

Fill out the information sheet located at the end of this problem set or at the end of the template file and sign it in order to acknowledge the Honor Code (if typesetting the homework, you may type your name instead of signing). This should be the last page of your submission. Failure to fill out the information sheet will result in a reduction of 2 points from your homework score.

Submitting code: Upload your code at `http://snap.stanford.edu/submit`. Put all the code for a single question into a single file and upload it.

Homework survey: After submitting your homework, please fill out the Homework 3 Feedback Form. Respondents will be awarded extra credit.

Questions

1 Mob Psychology and Thresholds [10 points – Yokila]

A question that occupied sociologists and economists as early as the 1900's is how do *innovations* (e.g. ideas, products, technologies, behaviors) *diffuse* (spread) within a society. One of the prominent researchers in the field is Professor Mark Granovetter who, along with Thomas Schelling, introduced *threshold models* in sociology. In Granovetter's model, there is a population of individuals (a mob) and for simplicity two behaviors (riot or not riot).

- *Threshold model:* each individual i has a threshold t_i that determines her behavior in the following way. If there are at least t_i individuals that are rioting, then she will join the riot, otherwise she stays inactive.

Here, it is implicitly assumed that each individual has full knowledge of the behavior of all other individuals in the group. Nodes with small threshold are called innovators (early adopters) and nodes with large threshold are called laggards (late adopters). Granovetter's threshold model has explained both classical empirical adoption curves by relating them to thresholds in the underlying population and the impact that a small number of individuals might have in creating large-scale behavior.

In this problem, you are going to explore the impact of thresholds on the final number of rioters. For a mob of n individuals, define the histogram of thresholds $\mathbf{N} = (N_0, \ldots, N_{n-1})$, where N_ℓ expresses the number of individuals that have threshold $\ell \in [n]$. For example, N_0 is the number of people who riot no matter what; N_1 is the number of people who riot if one other person is rioting; and so on.

(a) **[3 points]** For a given histogram find the conditions such that individuals with threshold ℓ become active. This problem involves some simple induction.

(b) **[3 points]** For a given histogram of thresholds $\mathbf{N}$, provide an expression for the final number of rioters. You should use your result/intuition from the previous part, and your answer can involve a maximum or minimum over a set.

(c) **[4 points]** The cumulative histogram of the histogram $\mathbf{N}$ is defined as $[\mathbf{N}] = (N_{[1]}, \ldots, N_{[n-1]})$, where $N_{[k]} = \sum_{\ell=0}^{k} N_\ell$. Download the histogram at http://cs224w.stanford.edu/homeworks/hw3/data_sets/thresholds.txt, plot the cumulative histogram, and report the final number of rioters. In one to two sentences, explain how to predict the number of rioters using this plot (based on your analysis in part 2).

What to submit [all tagged together on Gradescope, preferably on one page]

Page 1:
- (part a) Give the conditions as a set of inequalities; show your reasoning.
- (part b) Give an expression for the number of rioters. Your expression can involve a maximum or minimum over a set of numbers.
- (part c) Provide the plot and report the final number of rioters. Explain how to infer the number of rioters from the plot (1–2 sentences).

2 Empirical Power Laws [25 points – Anthony, Ziyi]

In this problem, you will generate a dataset following a power-law distribution and test various methods for estimating the exponent of the distribution.

2.1 CCDF [5 points]

Recall from class that the probability density function (PDF) of a power-law distribution is:

$$P(X = x) = \frac{\alpha - 1}{x_{\min}} \left(\frac{x}{x_{\min}} \right)^{-\alpha}, \tag{1}$$

where $x_{\min}$ is the minimum value that X can be. Derive an expression of $P(X \geq x)$, the Complementary Cumulative Distribution Function (CCDF), in terms of α.

2.2 Sampling [5 points]

Now we will generate a dataset from a power-law distribution. First, show how to generate a random sample from the power law distribution in Equation 1 using the CCDF derived in part 1 and a uniform random sample $u \sim U(0, 1)$. (*Hint*: You can use inverse transform sampling.)

Next, using this sampling technique, create a dataset of 100,000 samples following the power-law distribution with exponent $\alpha = 2$ and $x_{\min} = 1$. Plot the empirical distribution on a log-log scale by first rounding each sample to the nearest integer and then plotting the empirical PDF over these rounded values. Also plot the true probability density function for the power law (this will help you verify that you generated the data correctly).

2.3 Least squares estimation with the PDF [5 points]

One way to fit a power law distribution to data is through a least-squares linear regression on a histogram of the data. Again, round your samples to the nearest integer to compute a histogram of the data. Derive a linear least-squares problem for estimating α from the empirical PDF. (Hint: Use Equation (1).) Compute and report the least-squares estimate $\alpha_{\text{ls,pdf}}$ of α using the dataset generated in part 2.

In 1–2 sentences, briefly explain how you can improve the accuracy of the estimate by ignoring some of your data. Use this procedure to compute a new estimate $\alpha'_{\text{ls,pdf}}$.

Make the same plot as in part 2 but also include the two estimated probability distributions (from $\alpha_{\text{ls,pdf}}$ and $\alpha'_{\text{ls,pdf}}$). It is possible that your estimate for α is less than 1, which does not lead to a probability distribution because the probabilities according to the PDF become negative (with the $\alpha - 1$ term). In this case, you can just plot the slope without the estimated distribution.

2.4 Maximum log-likelihood estimation [5 points]

Given randomly sampled data points $x_1, \ldots, x_n$ from a power-law distribution, derive the log-likelihood expression for the dataset in terms of α, $\{x_i\}$ and n (assume that $x_{\min} = 1$). Recall the log-likelihood is given by $\ln \mathcal{L}(\alpha\,;\, x_1, \ldots, x_n) = \sum_{i=1}^{n} \ln P(X = x_i \mid \alpha)$.

Compute and report α_{mlle} for the dataset generated in part 2. Make the same plot as in part 2 but also include the estimated probability distribution (from α_{mlle}).

2.5 Comparison [5 points]

Generate 100 datasets of 100,000 samples each following the procedure in part 2. For each dataset, compute the estimates $\alpha_{\text{ls,pdf}}$, $\alpha'_{\text{ls,pdf}}$, and α_{mlle}. Report the sample mean and standard deviation of each estimate over the 100 datasets.

What to submit

Page 2:
- Derivation of the CCDF in terms of α.

Page 3:
- Short proof showing how to sample from the power law distribution with a uniform random variable
- Plot of the empirical distribution as well as the power law distribution with true value of α

Page 4:
- Show how to set up the the least-squares linear regression.
- The computed estimate $\alpha_{\text{ls,pdf}}$.
- 1–2 sentences explaining how to improve the estimate by ignoring some data
- The improved estimate $\alpha'_{\text{ls,pdf}}$.

- Plot of the empirical distribution as well as the power law distribution with the true value of α and the two estimated values.

Page 5:
- Short derivation of the maximum log likelihood estimate.
- The computed estimate α_{mlle}.
- Plot of the empirical distribution as well as the power law distribution with the true value of α and the estimate value.

Page 6:
- Sample mean and standard deviation for each estimate.

3 The SIR Model of Disease Spreading [30 points – Poorvi, Praty]

In this question you will explore how varying the set of initially infected nodes in the SIR model can affect how a contagion spreads through a network.

For the 2005 Graph Drawing conference, a data set was provided of the IMDB movie database. We will use a reduced version of this dataset, which derived all actor-actor collaboration edges where the actors co-starred in at least 2 movies together between 1995 and 2004. Please download the required files from:
http://cs224w.stanford.edu/homeworks/hw3/data_sets/imdb_actor_edges.tsv
http://cs224w.stanford.edu/homeworks/hw3/data_sets/imdb_actors_key.tsv

We will be comparing our results to two other null models, the Erdős-Rényi graph and the Preferential Attachment graph, with the same number of nodes and expected degree. Please download the networks from:
http://cs224w.stanford.edu/homeworks/hw3/data_sets/SIR_erdos_renyi.txt
http://cs224w.stanford.edu/homeworks/hw3/data_sets/SIR_preferential_attachment.txt

Recall from lecture that under the SIR model, every node can be either susceptible, infected, or recovered and every node starts off as either susceptible or infected. Every infected neighbor of a susceptible node infects the susceptible node with probability β, and infected nodes can recover with probability δ. Recovered nodes are no longer susceptible and cannot be infected again. Algorithm 1 describes for pseudo-code of this process.

3.0 [0 points] (do not submit)

For a node with d neighbors, what is its probability of getting infected in a given round?

3.1 [10 points]

Implement the SIR model above and run 100 simulations with $\beta = 0.05$ and $\delta = 0.5$ for each of the three graphs. Initialize the infected set with a single node chosen uniformly at random. Record the total percentage of nodes that became infected in each simulation. Note that a simulation ends when there are no more infected nodes; the total percentage of nodes that became infected at some point is thus the number of *recovered* nodes at the end of your simulation divided by the total number of nodes in the network.

Inspecting the data, you should see that some simulations die out very quickly, while others manage to become *epidemics* and infect a large proportion of the networks. For all three graphs:

Algorithm 1: Pseudo-code for simulating the SIR model on a graph $G = (V, E)$

Input: initial set of infected nodes I
$S \leftarrow V \backslash I$ `// susceptible nodes`
$R \leftarrow \emptyset$ `// recovered nodes`
while $I \neq \emptyset$ **do**
  $S' \leftarrow \emptyset$ `// nodes no longer susceptible after the current iteration`
  $I' \leftarrow \emptyset$ `// newly infected nodes after the current iteration`
  $J' \leftarrow \emptyset$ `// nodes no longer infected after the current iteration`
  $R' \leftarrow \emptyset$ `// newly recovered nodes after the current iteration`
  foreach *node* $u \in V$ **do**
    if $u \in S$ **then**
      foreach $(u, v) \in E$ *with* $v \in I$ **do**
        With probability β: $S' \leftarrow S' \cup \{u\}$, $I' \leftarrow I' \cup \{u\}$, and break for loop
    else if $u \in I$ **then**
      With probability δ: $J' \leftarrow J' \cup \{u\}$ and $R' \leftarrow R' \cup \{u\}$
  $S \leftarrow S \backslash S'$
  $I \leftarrow (I \cup I') \backslash J'$
  $R \leftarrow R \cup R'$

Compute the proportion of simulations that infected at least 50% of the network; we will consider these events epidemics. To compare the likelihood of an epidemic starting across graphs, and more importantly, test whether or not the observed differences are actually significant, use pairwise Chi-Square tests. For each pair of networks, compute:

```
scipy.stats.chi2_contingency([[e1, 100-e1],[e2, 100-e2]]),
```

where `e1` is the number of trials where $\geq$50% were infected in network 1 and `e2` is the number of trials where $\geq$50% were infected in network 2. Report both the χ^2 statistic and p-values. See the documentation linked above for details on interpreting the output of the function call.

Finally, answer the following questions about the two synthetic networks:

- Does the Erdős-Rényi graph appear to be more/less susceptible to epidemics than the Preferential Attachment graph?
- In cases where an epidemic does take off, does Erdős-Rényi graph appear to have higher/lower final percentage infected?
- Overall, which of these two networks seems to be more susceptible to the spread of disease?
- Give one good reason why you might expect to see these significant differences (or lack thereof) between Erdős-Rényi and Preferential Attachment? (2–3 sentences)

We highly recommend that you debug your code with a smaller number of simulations and only run with 100 simulations once you are confident in your code. Running 100 simulations is necessary to ensure statistical significance in some of the comparisons. The reference code takes 15-20 minutes to run for the entire question (parts 3.1–3.4).

3.2 [10 points]

Repeat parts (a)–(c) from 3.1, but instead of selecting a random starting node, infect the node with the highest degree. *However, do not compute the χ^2 statistic; in this part, it is certainly possible that some networks will have epidemics (i.e., $\geq$50% infected) in all trials, which makes the χ^2 test ill-defined.*

Compute the relative increase in the average proportion of people infected for both the Erdős-Rényi graph and Preferential Attachment graphs, relative to part 3.1 (without conditioning on the epidemic spreading to $\geq$ 50% of the population). Between the Erdős-Rényi graph and Preferential Attachment graph, which one seems to be more impacted by the targeting of the highest degree node? Give one good reason why this might be. (2–3 sentences)

3.3 [5 points]

One aspect of real-world networks that most random-graph models lack is community structure (i.e., distinct clusters of highly inter-connected nodes). The movie-actor network, for example, is highly modular with distinct clusters corresponding to different genres, etc. What do the above simulations indicate about the impact of community structure on the spread of epidemics? Explain this in 3–4 sentences.

3.4 [5 points]

Now repeat the experiments from 3.1 and 3.2, but instead initialize the infected set to be 10 random nodes and the top 10 highest degree nodes, respectively. However, as with the previous part, you should skip the χ^2 test computation, since some networks will likely have epidemics in 100/100 trials. Does the relative impact of targeting high-degree nodes increase/decrease? Explain this result in 3–4 sentences.

What to submit

Page 8:
- The proportion of trials that result in epidemics in each graph and the mean proportion infected for each graph (both conditioned on an epidemic and without the conditioning).
- The results of the specified statistical test (report test statistics and p-values) and whether or not the results appear significantly different.
- Short answers to the four questions on the synthetic networks.

Page 9:
- The proportion of trials that result in epidemics in each graph, the mean proportion infected for each graph (both conditioned on an epidemic and without the conditioning).
- The relative increase in the average proportion of people infected for both synthetic networks.
- 2–3 sentences explaining why targeting the high degree node had more/less impact in the Erdős-Rényi graph compared to the the Preferential Attachment graph.

Page 10:
- 3–4 sentences explaining the impact of community structure on the spread of epidemics in the simulations

Page 11:
- The proportion of trials that result in epidemics in each graph, the mean proportion infected for each graph (both conditioned on an epidemic and without the conditioning).
- 3–4 sentences explaining why the relative impact of targeting high-degree nodes increased/decreased after increasing the number of nodes infected at the beginning.

4 Influence Maximization [25 points – Anunay, Silviana]

In class we discussed the influence maximization problem and the greedy hill-climbing approach to solving it. In the algorithm, we add nodes to the current seed set one at a time. At step 0, we have an empty set S_0. At step $i > 0$, we pick the node which maximizes the marginal gain: $S_i = S_{i-1} \cup \{\arg\max_u f(S_{i-1} \cup \{u\}) - f(S_{i-1})\}$, where $f(S)$ denotes the number of nodes influenced by the initially active set S (includes the set S itself).

As we showed in class the hill climbing algorithm cannot guarantee an optimal solution. In other words, there might exist a set T with $|T| = i$ such that $f(S_i) < f(T)$. Parts 1 and 2 of this problem ask you to construct examples where this is the case. Your answer should consist of: (1) For every node u its influence set X_u (you can describe the set or draw a directed graph where an edge from A to B indicates that node A influences node B with probability 1), (2) S_i, the set of nodes that a greedy hill climbing would choose after i iterations, and (3) T, the optimal set of i nodes.

For all the questions, you can assume: (1) The nodes in S are influencing themselves, i.e., the count of total influence $f(S)$ includes the nodes in S. (2) The influence set X_u contains all nodes that are influenced by node u, both directly and eventually. (3) When several nodes have the same level of marginal gain, we choose one of them at random.

4.1 Non-Optimal Hill-Climbing [8 points]

For $i = 2$, construct an example where $f(S_i) < f(T)$. That is, hill-climbing will only find a non-optimal solution. (Hint: the last step of the greedy approach is optimal given the $i - 1$ previous steps.)

4.2 Bounded Non-Optimal Hill-Climbing [8 points]

For $i = 3$, construct an example where $f(S_i) \leq 0.8f(T)$. That is, hill-climbing will only find a solution that is at most 80% of the optimal solution.

4.3 Optimality of Hill-Climbing [4 points]

Give a property of influence sets X_u such that $f(S_i) = f(T)$. In other words, what is a *sufficient* property of influence sets of nodes such that greedy hill-climbing always outputs the optimal solution? The property does not need to be a necessary one. It must be a property of X_u. Properties such as "the network has only i nodes" are not valid as correct answers.

There are several correct answers; we will accept all reasonable answers.

4.4 More Hill-Climbing... [5 points]

Assume that we stop hill-climbing after k steps and $|S_k| = |T| = k$. Recall that in the class we proved a bound in the form of

$$f(T) \leq f(S_k) + \sum_{i=1}^{k} \delta_i, \tag{2}$$

where $\delta_1, .., \delta_k$ are the largest k values of $\{f(S_k \cup \{u\}) - f(S_k)|u\}$ (See the course slides). Construct a family of examples for which $f(S_k) + \sum_{i=1}^{k} \delta_i - f(T)$ can be arbitrarily large.

To be more specific, given any number b you should exhibit a network (graph) such that $f(S_k) + \sum_{i=1}^{k} \delta_i - f(T) > b$.

Note: A *family* of examples is a set F of examples such that for any number b, there exists a network (graph) $E(b) \in F$ (corresponding to b) such that $f(S_k) + \sum_{i=1}^{k} \delta_i - f(T) > b$.

What to submit

Page 12:
- Submit an example: X_u, S, T.

Page 13:
- Submit an example: X_u, S, T.

Page 14:
- Submit a property and a brief explanation.

Page 15:
- Submit a family of examples and a brief explanation.

Information sheet
CS224W: Analysis of Networks

Assignment Submission Fill in and include this information sheet with each of your assignments. This page should be the last page of your submission. Assignments are due at 11:59pm and are always due on a Thursday. All students (SCPD and non-SCPD) must submit their homeworks via GradeScope (http://www.gradescope.com). Students can typeset or scan their homeworks. Make sure that you answer each (sub-)question on a separate page. That is, one answer per page regardless of the answer length. Students also need to upload their code at http://snap.stanford.edu/submit. Put all the code for a single question into a single file and upload it. Please do not put any code in your GradeScope submissions.

Late Homework Policy Each student will have a total of *two* free late periods. *Homeworks are due on Thursdays at 11:59pm PDT and one late period expires on the following Monday at 11:59pm PDT.* Only one late period may be used for an assignment. Any homework received after 11:59pm PDT on the Monday following the homework due date will receive no credit. Once these late periods are exhausted, any assignments turned in late will receive no credit.

Honor Code We strongly encourage students to form study groups. Students may discuss and work on homework problems in groups. However, each student must write down their solutions independently i.e., each student must understand the solution well enough in order to reconstruct it by him/herself. Students should clearly mention the names of all the other students who were part of their discussion group. Using code or solutions obtained from the web (github/google/previous year solutions etc.) is considered an honor code violation. We check all the submissions for plagiarism. We take the honor code very seriously and expect students to do the same.

Your name: _____

Email: _____ **SUID:** _____

Discussion Group: _____

I acknowledge and accept the Honor Code.

(Signed) _____