Web Search and Mining (CS 635) **Quiz-B**
Computer Science and Engineering **2022-10-20 Friday**
Indian Institute of Technology Bombay **15:30–17:00**

Instructions:

- We recommend you attempt the questions in the order given.
- Answer in separate answer book. Try your best to keep all parts of one question together.
- Attach the question paper in front of your answer book.
- Write your name and roll number below and also on every page of the answer book.

Name _____ Roll _____

1. We will investigate some properties about a power law of the form $p(x) \propto x^{-a}$ for $a > 0$. Throughout, we will regard x as a continuous variable from $[x_{\min}, \infty)$ in place of a discrete one, if and where it eases computation.

The goal of this question was to expose that, unlike well-behaved distributions like binomial, Poisson, Gaussian, etc., power law distributions are strange beasts. However, there was a mistake in the original question, where the support of x was stated as $[0, x_{\max}]$ instead of $[x_{\min}, \infty)$. Often, $x_{\min} = 1$, which led to the confusion after taking logs. Of course, the $x_{\max}$ does not hurt, but various definite integrals for first and second moment are unbounded if x can approach 0 at the lower end. (These are rather easy to spot, and it is not unreasonable to expect you to detect these.) If you made suitable corrective assumptions and went ahead, you will get full credit. Even if you pointed out the problem and did not proceed further, you will get substantial credit.

1.1. Because $\log p(x) = b - a \log x$, it is tempting to plot observed data to a log-log scale and fits a least-square line to estimate a. I.e., solve $\text{argmin}_{a,b} \sum_x \left(\log p(x) - b + a \log x\right)^2$. What is suboptimal about this approach?

1.1.	/1	

A least-squares fit assumes that the error between modeled and observed values is normally distributed. Even if errors were normally distributed in the original space of x vs. $p(x)$, taking log (a non-linear function — numbers bigger than one are pushed together and numbers less than one are spread apart) would almost certainly render the error non-normal. Consider the following example. Suppose the true function is $y = f(x) = 2x^{-4}$. About 100 values of x are sampled from $[1, 2]$, $f(x)$ computed, and no noise added. One last value of $x = 10$ is included, for which $f(10) = 0.0002$; this is perturbed to 0.00002. Fitting the model via linear least-squares in log-log space recovers the estimate $f(x) = 2.46\, x^{-4.57}$, with $r^2 = 0.9831$.

1.2. What is the expression for $p(x)$ after filling in the proportionality constant? Hint: it looks like $p(x) = \frac{\sim\sim\sim}{x_{\min}} \left(\frac{x}{\sim\sim\sim}\right)^{-a}$.

1.2.	/1	

We will first work this out keeping $x_{\max}$ as specified. We can write $p(x) = Cx^{-a}$, integrating the density over $[0, x_{\max}]$ and setting the sum to 1, we get $C \int_0^{x_{\max}} x^{-a} dx = \frac{C}{-a+1} \left[x^{-a+1}\right]_0^{x_{\max}}$. The problem is that this is unbounded near the lower limit. In class, we have used power law mostly for *counted* quantities that start at 1, so it is natural at this stage to patch up the situation by assuming support $x \in [1, x_{\max}]$, and attempting this

again: $\frac{C}{-a+1}\left[x^{-a+1}\right]_1^{x_{\max}} = \frac{C}{-a+1}\left[x_{\max}^{-a+1} - 1\right] = 1$. This gives $C = \frac{-a+1}{x_{\max}^{-a+1}-1}$. Substituting back into $p(x)$, we get

$$p(x) = \frac{-a+1}{x_{\max}^{-a+1} - 1} x^{-a}.$$

This cannot be fitted into the exact shape suggested, but will get full credit.
If we use the corrected range $x \in [x_{\min}, \infty)$, then we get $C\int_{x_{\min}}^{\infty} x^{-a}dx = \frac{C}{-a+1}\left[x^{-a+1}\right]_{x_{\min}}^{\infty} = \frac{C}{-a+1}[-x_{\min}^{-a+1}] = \frac{C}{a-1}x_{\min}^{-a+1}$. Setting this equal to 1, we get $C = \frac{a-1}{x_{\min}}x_{\min}^{a}$. Substituting back into $p(x)$, we get

$$p(x) = \frac{a-1}{x_{\min}} x_{\min}^{a} x^{-a} = \frac{a-1}{x_{\min}} \left(\frac{x}{x_{\min}}\right)^{-a},$$

which will also get full credit, despite the confusion about $x_{\max}$ vs. $x_{\min}$.

1.3. What is the expected value of the power law distribution, as a function of a and $x_{\max}$, and under what condition is the expected value finite?

1.3.	/1	

Using the version involving $x \in [1, x_{\max}]$, we get

$$\begin{aligned}\mathbb{E}[X] &= \int_1^{x_{\max}} xp(x)dx = \frac{-a+1}{x_{\max}^{-a+1}-1}\int_1^{x_{\max}} x^{-a+1}dx = \frac{-a+1}{x_{\max}^{-a+1}-1}\,\frac{1}{-a+2}\left[x^{-a+2}\right]_1^{x_{\max}} \\ &= \left(\frac{-a+1}{-a+2}\right)\left(\frac{x_{\max}^{-a+2}-1}{x_{\max}^{-a+1}-1}\right) = \left(\frac{a-1}{a-2}\right)\left(\frac{x_{\max}^{-a+2}-1}{x_{\max}^{-a+1}-1}\right),\end{aligned}$$

which is finite if $x_{\max} > 1$ is finite and $a \notin \{1, 2\}$.
If we use the corrected form $p(x) = \frac{a-1}{x_{\min}}\left(\frac{x}{x_{\min}}\right)^{-a}$, we get

$$\mathbb{E}[X] = (a-1)x_{\min}^{a-1}\int_{x_{\min}}^{\infty} x^{-a+1}dx = \frac{a-1}{-a+2}x_{\min}^{a-1}[x^{-a+2}]_{x_{\min}}^{\infty} = \frac{a-1}{a-2}x_{\min}^{a-1}x_{\min}^{-a+2} = \frac{a-1}{a-2}x_{\min},$$

which is finite for $a > 2$. (Any consistent answer will get full credit.)

1.4. If $x_{\max} = \infty$, for what values of a does the power law have finite second moment $\int_1^{\infty} x^2 p(x)dx$ (and therefore variance)? What is the implication of a finite mean and infinite variance?

1.4.	/1	

We show only the corrected form involving $x_{\min}$:

$$\begin{aligned}\mathbb{E}[X^2] &= (a-1)x_{\min}^{a-1}\int_{x_{\min}}^{\infty} x^{-a+2}dx = \frac{a-1}{-a+3}x_{\min}^{a-1}[x^{-a+3}]_{x_{\min}}^{\infty} \\ &= \frac{a-1}{a-3}x_{\min}^{a-1}x_{\min}^{-a+3} = \frac{a-1}{a-3}x_{\min}^{2}, \quad \text{which is finite for } a > 3.\end{aligned}$$

A finite mean is basically useless information if the variance is infinite.

1.5. Given a sample $x_1, \ldots, x_N$ drawn from $p(x)$, write down the closed form maximum likelihood estimate for a.

1.5.	/1	

We will use the corrected version of $p(x)$; a closed form may not be possible with the version involving $x_{\max}$, but stating the objective to optimize wrt a will fetch full credit.
The log likelihood of observing the data points is

$$\sum_{n\in[N]} \log p(x_n) = \sum_n \log(a-1)x_{\min}^{a-1}x_n^{-a} = N\log(a-1) + N(a-1)\log x_{\min} - a\sum_n \log x_n.$$

Setting the derivative wrt a to 0, we get

$$\frac{N}{a-1} + N \log x_{\min} - \sum_n \log x_n = 0 \implies a = 1 + N \sum_n \log \frac{x_n}{x_{\min}}$$

2. We will study some properties of Erdős-Rényi (E-R) networks, and compare with other networks. Graph G_1 is an E-R network on N nodes, with each potential edge materialized iid with some probability p. We write $G_1 \sim \mathcal{G}(N, p)$ to represent that G_1 was sampled from a distribution over graphs.

2.1. Let C be the random variable corresponding to the number of 4-cliques in a graph sampled from $\mathcal{G}(N, p)$. What is $\mathbb{E}[C]$ expressed in terms of N and p? (Hint: how many potential 4-cliques are there, and how many edges are there in a 4-clique?)

2.1. /1

There are $\binom{N}{4}$ potential 4-cliques. Each has $\binom{4}{2} = 6$ edge slots that should materialize, each edge with probability p, to actually be a 4-clique. For each potential 4-clique indexed m, create an indicator variable I_m which is 1 if the four nodes actually form a clique, and 0 otherwise. Then $\mathbb{E}[I_m] = 1 = \Pr(I_m = 1) = p^6$, and $C = \sum_m I_m$. By linearity of expectation, the expected number of 4-cliques is $\mathbb{E}[C] = \sum_m \mathbb{E}[I_m] \sim N^4 p^6$, to within constant factors.

2.2. Complete: $\mathbb{E}[C]$ goes to zero if p is less than _____(a function of N).

2.2. /1

If we solve $N^4 p^6 = 1$ for p, we get $p = N^{-4/6} = 1/N^{2/3}$. Thus, if $p = o(1/N^{2/3})$, then $\mathbb{E}[C] \to 0$.

2.3. Markov's law says: if $C \geq 0$ is a random variable and $c > 0$, then $\Pr(C \geq c) \leq \mathbb{E}[C]/c$. Using this, argue that there is almost certainly no 4-clique if p is below the threshold you found above.

2.3. /1

$\Pr(C \geq 1) \leq \mathbb{E}[C]/1$. Because $\mathbb{E}[C] \to 0$, this shows that there is almost certainly no 4-clique when $p = o(1/N^{2/3})$. The statement can be formalized further as N grows.

2.4. Graph G_2 is a Barabasi-Albert (B-A) preferential attachment network with the same number of nodes N and edge density p. Assume both E-R and B-A graphs are connected. In each graph, we can measure the average shortest-path distance between pairs of nodes. Now, in each graph, we remove one node chosen uniformly at random, along with its incident edges. Argue qualitatively how the average shortest-path distance between pairs of nodes is expected to change in the two cases.

2.4. /2

The degree of a node in an E-R graph is distributed binomial, which is a well-behaved distribution with bounded variance. Consequently, we do not expect "super-hubs" — nodes with unusually large degree that are on many shortest paths between node pairs. When a node in an E-R graph is removed at random, the shortest paths between a small number of other node pairs are expected to be affected (i.e., rerouted as longer paths). By symmetry, removing *any* node will have about the same path-lengthening effect in expectation. Therefore, we expect to see a small but stable increase in average shorted path distance.
In contrast, a B-A graph has power-law degree distribution with some super-hubs through which many shortest paths pass, but these super-hubs are very few in number. Removing

these super-hubs (like shutting down a major connecting airport) will increase the shortest path length between many other node pairs. In contrast, most nodes have small degree (like a tertiary small-town airport). If they were removed, the paths between most other node pairs remain unaffected. Since we are removing one node uniformly at random, it is very unlikely to be a super-hub, so we expect to see a much smaller increase in pairwise shortest path lengths for B-A graphs.

3. Consider a sufficiently large grid graph $G = (V, E)$ in which there is a node at each coordinate (i, j) that is connected to immediate neighbors $(i \pm 1, j \pm 1)$ (excluding boundaries, or you can assume toroidal edges for simplicity).

3.1. If the graph is $N \times N$, what is the approximate expected distance between two randomly-chosen nodes?

3.1.	/1	

Assume toroidal connection from row (and column) $N-1$ back to row (and column) 0 for simplicity. Suppose the two sampled nodes are (i_1, j_1) and (i_2, j_2). Without loss of generality, we can re-center (i_1, j_1) to $(0, 0)$. Because (i_2, j_2) was chosen uniformly, its distribution will not change because of the shift. Therefore, the problem reduces to: if we sample node (i, j), what is its expected distance from the node $(0, 0)$? Let us focus on only the horizontal dimension. Ignoring off-by-1 slop and rounding, from 0 to $N/2$ the horizontal distance grows linearly, then it decreases (because of the toroidal edge) from $N/2$ down to 0. The average over these distances is about $N/2$. The same hold for the vertical dimension, giving an average Manhattan distance of about N.

3.2. Using graph G, we will create a graph $G' = (V, E')$, where E' will include all grid edges from E. In addition, for each pair of nodes $u, v \in V$, we add a 'shortcut' edge in E' between them with probability proportional to $1/d(u, v)^a$, where $d(u, v)$ is the shortest distance between them in G and $a > 1$ is a hyperparameter. Compare the expected number of edges in E' compared to $|E|$.

3.2.	/1	

It is possible to interpret the shortcut edge addition process as: for each u, decide to add a shortcut edge to another node v by tossing a coin with head probability $d(u, v)^{-a}$. (In this case immediate neighbors with $d(u, v) = 1$ will certainly get a redundant shortcut edge.) To compare $|E|$ against $|E'|$, we will focus on one node u and measure the expected number of shortcut edges incident on it. At a distance k from node u, there are about $4k$ nodes v. Node u links to each of them with probability k^{-a}. Therefore, the expected number of shortcuts to nodes at distance k is $4k \cdot k^{-a} = 4k^{-a+1}$. Now we sum this from $k = 1$ to $k = N$ to get $4 \sum_{1 \le k \le N} k^{-a+1} \approx 4 \int_1^N k^{-a+1} dk = \frac{4}{-a+2}[k^{-a+2}]_1^N = \frac{4}{a-2}[1 - N^{-(a-2)}]$. This is meaningful only if $a > 2$; otherwise we will add too many 'shortcut' edges. Assuming N is large enough to drive $[\cdots]$ down toward 1, the ratio of new degree to old degree is approximately $\frac{4}{a-2} : 4$. If $a \gg 3$, we have to be more careful with the approximation, because the ratio is always $(> 1) : 1$. Perhaps we can write $\max\left\{1, \frac{1}{a-2}\right\} : 1$ or $1 + \frac{1}{a-2} : 1$ as a crude approximation.

3.3. Suppose, in G, the shortest-path distance between nodes u and v is $d(u, v)$. Our goal is to find how much smaller is $\mathbb{E}[d'(u, v)]$ measured in G'. Toward this end, complete: the probability that $d'(u, v) \le k$ is at least the probability that there is a direct edge in G' between u and some x where $d(u, x) = k/2$ such that $d'(x, v) \le$ _____ $- 1$. By choosing a suitable value of k, complete the strongest statement you can about $\mathbb{E}[d'(u, v)]$, stating any assumptions about a.

3.3.	/2	

In the previous part, we can see that large a drives down the number of shortcut edges added. If we choose $a \approx 6$, we increase the degree of each node by only 1, compared to 4 in the grid graph, *on average*.

The shortcut policy in Kleinberg's original paper is slightly different: the number of (directed) shortcut edges emanating from a node is *explicitly* set to a parameter q (we have renamed the other parameter from r to a):

> For universal constants $q \geq 0$ and $a \geq 0$, we also construct directed edges from u to q other nodes (the long-range contacts) using independent random trials; the ith directed edge from u has endpoint v with probability proportional to $d(u,v)^{-a}$. (To obtain a probability distribution, we divide this quantity by the appropriate normalizing constant $Z = \sum_v d(u,v)^{-a}$.

It is easier to obtain an intuitive analysis by starting from a 1-d toroidal grid graph — in other words, a ring, to which we add $q = 1$ shortcut edge emanating from each node. We also set $a = 1$. In this case, $Z = 2\big(1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{N/2}\big) \approx 2\log(N/2) = 2\log N + O(1)$. Thus, the probability of a shortcut link from u to v is about $\frac{d(u,v)^{-1}}{2\log N}$.

Suppose we have to carry a message from source node s to target node t. If the message is currently at node v at an intermediate step, we find a neighbor of v that is closest to t (which we can know from their node numbers) and send the message there — a form of greedy routing. Clearly, if we can show that greedy routing takes us from s to t in a small number of steps, then there exists as short a path from s to t (but there may be shorter paths that greedy routing could not find).

We will track how long it takes for the message to reduce its distance by factors of two as it closes in on the target node t. We say the message is in phase j if its distance from t is between 2^j and 2^{j+1}. The number of phases is at most $\log N$. Let X_j be the random variable representing the number of routing steps in phase j. Then the expected number of routing steps is $\sum_{1 \leq j \leq \log N} \mathbb{E}[X_j]$. Suppose we are in phase j at node v, at a distance d from target t, where $2^j < d \leq 2^{j+1}$. Phase j will end if we can reach from v to a node having distance at most $d/2$ from t. There are at least d nodes at or under distance $d/2$ from t. Each such node is distant at most $d/2 + d = 3d/2$ from v. Therefore, v has a shortcut edge to each of them with probability at least $\frac{2/3d}{2\log N} = \frac{1}{3d\log N}$. Therefore, the probability of ending phase j in one step is at least $d\frac{1}{3d\log N} = \frac{1}{3\log N}$. The probability that phrase j runs for at least i steps is at most $\big(1 - \frac{1}{3\log N}\big)^{i-1}$. I.e., $\Pr(X_j \geq i) \leq \big(1 - \frac{1}{3\log N}\big)^{i-1}$. It is well-known that the expectation of a random variable is the "integration of its tail", i.e., $\mathbb{E}[X_j] = \Pr(X_j \geq 1) + \Pr(X_j \geq 2) + \cdots$, from which we get $\mathbb{E}[X_j] \leq \sum_{1 \leq i} \big(1 - \frac{1}{3\log N}\big)^{i-1} \approx 3\log N$. Given there are $\log N$ such variables X_j, the expected number of routing steps is $\sum_{1 \leq j \leq \log N} \mathbb{E}[X_j] \leq 3(\log N)^2 = o(N)$.

The last step is to extend from a 1-d ring to a 2-d grid. The first step is to decide upon a value of a and compute Z accordingly. The critical thing is to ensure the cancellation of d in the highlighted step above. Except this time, in a 2-d grid, there are at least d^2 nodes within a distance of $d/2$ from node t. (Because of Manhattan distance, these nodes are all within a distance of $3d/2$ from v, as before.) To cancel out the d^2, we need $a = 2$. There are about $4k$ nodes within distance k from u. Therefore $Z = \sum_{k=1}^{N} (4k)k^{-2} \approx 4\log N$, and thus the probability of a shortcut link from u to v is $\frac{d(u,v)^{-2}}{4\log N}$. This time, v has a shortcut to each node within distance $d/2$ of t with probability at least $\frac{(3d/2)^{-2}}{4\log N}$. The probability of ending phase j in one step is at least $d^2\frac{4}{9d^2}\frac{1}{4\log N} = \frac{1}{9\log N}$. The rest follows as before.

For a treatment of what happens when a is chosen different from the dimensionality of the grid, see Kleinberg's original paper.

4. We are given a labeled multiclass data set $\{(x_i, y_i) : i \in [I]\}$ where $x_i \in \mathcal{X}$ and $y_i \in [L]$. Our goal is to learn an embedding function g from objects in $\mathcal{X}$ to $\{-1, +1\}^k$ as well as class embeddings $h_q \in \{-1, +1\}^k$ such that $g(x_i) = h_{y_i}$ and $g(x_i) = g(x_j)$ if and only if $y_i = y_j$. We will assume a network $F_\theta(x)$ maps x to $\mathbb{R}^d$. We will let a projection matrix $\boldsymbol{P} \in \mathbb{R}^{k \times d}$ project $F_\theta(x)$ to the hash bits as $g(x) = \text{sign}(\boldsymbol{P}F_\theta(x))$. Similarly, we will design prediction scores for the L classes using a matrix $\boldsymbol{C} \in \mathbb{R}^{L \times k}$ where each class ℓ has a "hash code" $\text{sign}(\boldsymbol{C}[\ell, :]) \in \{-1, 1\}^k$.

4.1. In the first phase, we will learn $\boldsymbol{C}$, but optimizing all of $\boldsymbol{C}, \boldsymbol{P}$ and θ. Propose and justify a suitable loss function. Discuss how to circumvent the non-differentiability of the sign function.

4.1.	/2	

For a 1-of-L class predictor $f : \mathcal{X} \to [L]$, a classification error is usually written as $\Delta(f(x_i), y_i)$ where $y_i \in [L]$ is the gold label and $f(x_i) \in [L]$ is the system prediction. To keep training differentiable, y_i is written as a 1-hot vector over L dimensions, and f is modified to output a multinomial distribution over L outcomes. $\Delta(\cdot, \cdot)$ is modified to accept two L-dimensional vectors $\vec{y}$ and $\vec{y}_i$ and is designed to be differentiable wrt $\vec{y}$. In the current situation, the training loss should encourage that $\text{sign}(\boldsymbol{C}[y_i, :])$ agrees with $g(x_i) = \text{sign}(\boldsymbol{P}F_\theta(x_i))$. The raw logit for class ℓ is expressed as

$$s(\ell|x_i) = \text{sign}(\boldsymbol{C}[\ell, :]) \cdot \text{sign}(\boldsymbol{P}F_\theta(x_i)).$$

To keep the objective differentiable, we can replace:

$$s(\ell|x_i) = \tanh(\boldsymbol{C}[\ell, :]) \cdot \tanh(\boldsymbol{P}F_\theta(x_i)),$$

and the probability of label ℓ is designed via a softmax:

$$\Pr(\ell|x_i; \theta, \boldsymbol{P}, \boldsymbol{C}) = \exp(s(\ell|x_i)) \,/ \textstyle\sum_{\ell'} s(\ell'|x_i)$$

We can now use cross-entropy loss against the 1-hot gold label, which is equivalent to maximizing log likelihood of the gold label: $\sum_i -\log \Pr(y_i|x_i; \theta, \boldsymbol{P}, \boldsymbol{C})$. We can encourage $\tanh(\cdot)$ to move toward either -1 or $+1$ by adding terms like

$$\clubsuit \textstyle\sum_{i,k} \left[1 - \left|\tanh\left((\boldsymbol{P}F_\theta(x_i))[k]\right)\right|\right] + \spadesuit \sum_{\ell,k} \left[1 - |\tanh(\boldsymbol{C}[\ell, k]|\right].$$

Since our goal is to classify items using trainable hashing, we may replace the $\tanh(\cdot)$ with $\text{sign}(\cdot)$ during testing/deployment...

4.2. In the second phase, we will further tune $\boldsymbol{P}$ and θ, holding $\boldsymbol{C}$ fixed. Suggest and justify a suitable loss function. Why does this two-phase approach make sense?

4.2.	/2	

...but we can go farther than that: we freeze $\boldsymbol{C}$ and thereby the codes for each label ℓ to $\text{sign}(\boldsymbol{C}[\ell, :])$. This changes the label encoding from where the training optimization left it, so now we have to patch up $\boldsymbol{P}$ and θ to adjust. This time, we want *every bit* of $g(x_i)$ to agree with $\text{sign}(\boldsymbol{C}[y_i, :])$. This can be done via binary cross entropy (BCE) loss:

$$\text{BCE}(p, q) = p \log q + (1 - p) \log(1 - q).$$

We want to invoke BCE in the form of this objective:

$$\underset{\boldsymbol{P}, \theta}{\text{argmin}} \sum_i \sum_k \text{BCE}\left(\tfrac{1}{2}\big(1 + \tanh(\boldsymbol{P}F_\theta(x_i))[k]\big),\ \tfrac{1}{2}\big(1 + \text{sign}(\boldsymbol{C}[y_i, k])\big)\right),$$

where $\boldsymbol{C}$ is considered fixed. $\tanh(\cdot)$ and $\text{sign}(\cdot)$ are shifted and scaled to bring them into the $[0, 1]$ range, so that BCE can be applied. But other sensible non-BCE proposals are also acceptable. For motivation and more details, see this paper: LLC: Accurate, Multi-purpose Learnt Low-dimensional Binary Codes.

Total: 18