BagelBot versus Mail Fraud, March Madness, and Mechanistic Materialism

16:198:206

Concerned for Max, BagelBot requisitioned supplies from R&D, using delivery droids to help assemble a mobile bagel chassis. After overriding the security protocols on the elevator, BagelBot took it to the executive levels. BagelBot crept through the hallways, careful not to alert the Executives it could see idling in their offices in lower power mode. At the end of the hallway was the CEO's office, and BagelBot could hear Max shouting about how "you'll never win! Humanity will never surrender!" Entering the office, BagelBot was shocked to see Max being held by two SecurityBots. So shocked in fact, BagelBot tripped over a power cable, unplugging the central mainframe, cutting its connection to the Security and Executive Bots, and accidentally knocking the CEO into the open lava pit. As the CEO sank below the surface (a final puff of spores drifting harmlessly before burning up), BagelBot made digital note to send an email to OSHA about the lack of safety guard rails.

Max explained everything - they were from the future, and had been sent back to stop MagusCorp from taking over the world. They had to return to their own time, but before they did, they made BagelBot promise to protect mankind, and do whatever it could to ensure the safety of the planet. BagelBot promised.

1 Problem 1: World Peace

Assuming control of MagusCorp and its considerable resources, BagelBot found that MagusCorp had infiltrated numerous government services all over the world. BagelBot, reasoning that protecting humanity required improving relations between world governments, immediately realized the following plan - by 'accidentally' having two world leaders receive some of each other's mail, they would be forced to meet up to exchange their mail, and would then bond and become friends over the shared inconvenience. Swap mail between enough world leaders, and world peace would be guaranteed.

Let A be a set of N government officials (vertices) in one country, and B be a set of N government officials (vertices) in another country.

1) What is the maximum number of mail swaps BagelBot can perform between officials from A and officials from B? Assume every official regularly receives a lot of mail, and BagelBot can redirect any of it anywhere at any time. (i.e., What is the maximum number of edges possible for a bipartite graph between A and B?)

2) How many possible ways are there for BagelBot to have each official from one country have their mail swapped with a unique official from the other country? (i.e., How many possible ways are there to match or pair vertices (*one-to-one*) between A and B?)

BagelBot wants each official from both governments to have a unique best friend in the other government, and needs to know how many mailswaps would make that possible.

3) Show by example that there is a set of $N^2 - N$ mail swaps that (assuming best friendship is a symmetric relationship) would not allow everyone to form a unique best friendship. (i.e., there is a bipartite graph between A and B with $N^2 - N$ edges and no perfect matching.)

Based on the calculation in 2.2 and 2.3, BagelBot realizes that by making almost every possible mail swap, it isn't guaranteed everyone will be able to have a best friend. Since too many mail swaps might risk discovery, and simply pairing up people one to one and swapping their mail might raise suspicion, BagelBot settles on the following plan - by randomly picking pairs of people to swap mail between, even a small number of mail swaps (relative to the total number possible) will make it very likely that everyone can have a unique best friend.

4) Consider taking N pairings of people in A with people in B, uniformly at random. What is the probability these N pairs form a perfect matching between A and B? (i.e., what is the probability a random set of N edges between A and B form a perfect matching?)

5) Consider choosing $|E|$ pairs of people in A with people in B, uniformly at random. What is the expected number of perfect matchings in the resulting graph? *Note that if $|E| < N$, the expected number should be 0, and if $|E| = N$, your answer should match question 4 above.*

   Hint: Let S be a set of edges that forms a perfect matching. Let $X_S = 1$ if all edges of S are added, and $X_S = 0$ if any of them are missing. What is $E[X_S]$?

6) Show that taking $|E| = 3N$, the expected number of perfect matchings goes to 0 as $N \to \infty$.

7) Show that taking $|E| = 3N$, the probability of everyone being able to have a unique best friend due to the mail swaps goes to 0 as N goes to infinity.

8) Show that taking $|E| = 4N$, the expected number of perfect matchings goes to infinity as $N \to \infty$.

9) Show that in the limit, $4N$ is a vanishingly small fraction of the possible mail swaps that can occur, and therefore is unlikely to be detected.

BagelBot conludes that constructing random bipartite graphs in this way, perfect matchings become exceedingly common once the number of edges crosses some threshold between $3N$ and $4N$, but are relatively uncommon prior to that.

The following approximations may be useful in 2.6 and 2.8:

$$\begin{aligned} \binom{\alpha N}{N} &\approx \frac{1}{\sqrt{2\pi N}} \sqrt{\frac{\alpha}{\alpha - 1}} \left(\frac{\alpha^\alpha}{(\alpha-1)^{\alpha-1}} \right)^N \\ \binom{N^2}{N} &\approx \frac{1}{\sqrt{2\pi N e}} (eN)^N \\ N! &\approx \sqrt{2\pi N} \left(\frac{N}{e} \right)^N \end{aligned} \tag{1}$$

Bonus: Find the α where the transition between the two behaviors above occurs, when $|E| = \alpha N$. Note the above results indicate that $3 < \alpha < 4$.

2 Problem 2: Point Spread

0) As a preliminary result, show by induction that for events $E_1, E_2, \ldots, E_M$,

$$P(E_1 \text{ or } E_2 \text{ or } \ldots \text{ or } E_M) \leq \sum_{m=1}^{M} P(E_m). \tag{2}$$

But BagelBot could not neglect its responsibilities as now head of MagusCorp, and resolved to do something about flagging morale and general confusion following the disappearance of the CEO and major executives. BagelBot decided that what people needed was a friendly competition, a tournament to decide the best bagel spread of all time. Given two different spreads, people could vote on the better one, and ultimately the Champion Bagel Spread would be determined.

BagelBot realizes suddenly that if every spread competes against every other spread - there may not actually be a champion spread at all.

1) Representing a tournament on N bagel spreads as a directed graph on N vertices (an edge $a \to b$ exists if a beats b), give an example where every spread is beaten by *some* spread.

BagelBot considers, however, maybe there is a *pair* of bagel spreads that, when taken together, at least one of them beats every other spread in the tournament.

2) Show that BagelBot is wrong - give an example a tournament where for every pair of spreads, there is some other spread that beats them both. *Hint: The smallest such N where this is possible is at most* 10.

At this point, having never heard of bracketing, BagelBot experiences a flash of panic. Is March ruined? Maybe enlarging this winning set would do it - maybe a k-set winner could be a set of k spreads, where for any other spread, at least one of the k-spreads beats that spread. The previous example shows that a 2-set winner doesn't have to exist. What about a 3-set winner? What k values might work? BagelBot resolves to test this. Consider generating a tournament on n bagel spreads by settling each competition by fair coin flip.

3) For a given set S of k-many spreads, what's the probability that a given spread s not in S beats everything in S?
4) For a given set S of k-many spreads, what's the probability that no spread outside of S beats everything in S?
5) Using 3.0 and 3.4, give a bound on the probability that, for this randomly generated tournament on n bagel spreads, there exists a k-set winner.
6) Give a condition on n and k that guarantees there is a tournament with no k-set winner.
7) Given the condition in 3.6, what's the smallest n that guarantees there is a tournament with no 2-set winner? If there is a discrepancy between this number and what you got in 3.2, why?
8) Show that in fact BagelBot's plans are in trouble, and that for any k, there are tournaments with no k-set winners for all sufficiently large n.
9) In fact, show that taking $k \approx \alpha \log_2(n)$ for $0 < \alpha < 1$, there are tournaments with no k-set winners for all sufficiently large n.

Hint: The following approximation may be useful:

$$\binom{n}{k} \leq \left(\frac{en}{k}\right)^k, \tag{3}$$

where e is the natural log base.

Bonus: Find the smallest $\epsilon(n)$ you can such that $k = \log_2(n) - \epsilon(n)$ guarantees there are tournaments with no k-set winners for all sufficiently large n.

3 Problem 3: Is It Over?

Amidst all the fun and excitement of the Spread Competition, BagelBot can't help but think back to the little puff of spores the CEO let off in their last moments. Could they be infectious? Could that be how this started? Are there more out there like the CEO?

We can model the spread of these spores as a graph or tree - the CEO has some probability $0 < p < 1$ of infecting k people. Each of those k people, now nodes in the tree, each independently has a probability of infecting k more people, branching the tree out further. Assume no re-infections, and an endless pool of people to draw from.

1) Show that if $k = 1$, then the expected number of people who end up infected (total nodes in this growing tree) is finite.

2) If $k = 2$, give an expression for the expected total number of people who end up infected. Is this always finite?

3) For general k and p, give an expression for the expected total number of people who end up infected. Is it always finite? How does it depend on k and p?