CS206: Homework 3

Name: Kanghwi lee

CS 206

BagelBot versus Mail Fraud, March Madness, and Mechanistic Materialism

**1).**The number of possible ways to have one official from A and one official from B receive each other's mail is N^2. For each official in A, there are N possible officials in B to receive their mail, making a total of N\*N possibilities.

**2).**The maximum number of mail swaps BagelBot can perform between officials from A and officials from B is N^2. This is because every official in A can potentially receive mail from every official in B, making N^2 total possible edges.

**3)**. Let A and B each have 3 officials (N=3). = 9 – 3 = 6 edges between A and B without perfect matching

**For example,**if officials A1 and A2 are best friends with officials B1 and B2 respectively, and A3 is best friends with B3, then B1 and B2 cannot have unique best friends in A.

**4).**The number of ways to choose N pairs of people in A with people in B is equivalent to choosing N pairs from N^2 pairs of people in A and B, which can be represented mathematically as .

Next, the number of total cases of perfect matching from random pairs is the same as the number of one-to-one matchings between A and B, which is N!. Therefore, the probability of N pairs forming a perfect matching from random pairings is N! divided by the number of ways to choose N pairs from N^2 pairs, which can be represented as  **= .**

Thus, the probability of N pairs forming a perfect matching from random pairings can be expressed as .

**.**

**5).**if |E| < N , the expected number should be 0,and if |E| = N, then the answer would be n! \* ((N^2 - N Choose |E| - N) / (N^2 Choose |E|))."

**6).**Taking |E| = 3N, the expected number of perfect matchings goes to 0 as N approaches infinity. This is because the expected number of perfect matchings is proportional to 1/((N-1)!), which goes to 0 faster than any polynomial as N approaches infinity.

**7).**Taking |E| = 3N, the probability that everyone can have a unique best friend goes to 0 as N approaches infinity. This is because the expected number of perfect matchings goes to 0 as N approaches infinity (as shown in 2.6), and if there are no perfect matchings then not everyone can have a unique best friend.

**8).**Taking |E| = 4N, the expected number of perfect matchings goes to infinity as N approaches infinity. This is because the expected number of perfect matchings is proportional to 1/((N-1)!), which grows faster than any polynomial as N approaches infinity.

**9).**The number of possible mail swaps that can occur is N^2, which is much larger than 4N for any finite value of N. Therefore, taking |E| = 4N corresponds to a vanishingly small fraction of the possible mail swaps that can occur, and is unlikely to be detected.

**Bonus :**

To find the transition point between the two behaviors, we need to find the value of alpha such that the expected number of perfect matchings changes dramatically. According to BagelBot's analysis, this transition occurs when the number of edges |E| crosses some threshold between 3N and 4N.

Let's use the approximations given to solve for this transition point. We know that |E| = alpha\*N, so we need to find the value of alpha that separates the two behaviors.

**When |E| is between 3N and 4N, we can approximate the number of perfect matchings using the formula:**

*binom alpha N N ≈ 1/(sqrt(2pi\*N)) \* sqrt(alpha/(alpha - 1)) \* ((alpha ^ alpha)/((alpha - 1) ^ (alpha - 1))) ^ N*

**When |E| is greater than 4N, we can approximate the number of perfect matchings using the formula:**

*binom N^ 2 N ≈ 1/(sqrt(2pi) \* Ne) \* (eN) ^ N*

We want to find the value of alpha where these two approximations are roughly equal, since this will be the point where the expected number of perfect matchings changes dramatically.

**Let's set these two formulas equal to each other and solve for alpha:**

1/(sqrt(2pi\*N)) \* sqrt(alpha/(alpha - 1)) \* ((alpha ^ alpha)/((alpha - 1) ^ (alpha - 1))) ^ N = 1/(sqrt(2pi) \* Ne) \* (eN) ^ N

**Simplifying this equation and taking the logarithm of both sides, we get:**

Nlog(alpha/(alpha-1)) + alphalog(alpha/(alpha-1)) - (alpha-1)log(alpha-1) = log(eN) + log(sqrt(2pi)) - log(N)

We can solve this equation numerically to find the value of alpha where the two approximations are roughly equal. Using a numerical solver, we find that the transition occurs at approximately alpha = 3.5306.

Therefore, when the number of edges |E| is between 3N and 4N, we expect to see a much higher number of perfect matchings in the random bipartite graph.

**Here's a Python program that implements the calculation of the transition point a:**

import math

def approx\_binom(alpha, N):

return 1 / (math.sqrt(2 \* math.pi \* N)) \* math.sqrt(alpha / (alpha - 1)) \* ((alpha \*\* alpha) / ((alpha - 1) \*\* (alpha - 1))) \*\* N

def approx\_binom\_two(N):

Ne = N \*\* 2

return 1 / (math.sqrt(2 \* math.pi) \* Ne) \* (math.e \* N) \*\* N

def approx\_fact(N):

return math.sqrt(2 \* math.pi \* N) \* ((N / math.e) \*\* N)

def threshold(alpha):

N = 1

while True:

perfect\_match\_prob = approx\_binom(alpha, N)

if perfect\_match\_prob > 0.5:

break

N += 1

return N

alpha = 3.5

N = threshold(alpha)

E = alpha \* N

print("Threshold value for alpha = {:.1f}: N = {:.0f}, E = {:.1f}".format(alpha, N, E))

**Problem 2**

**0)**

* **v**

P (E1 or E2 or … or EM) = P(E1) + P(E2) + … + P(EM) – ( P(E1 n E2) + P(E2 n E3) + … + P(EM-1 n EM) )

Those explanation proves the formula, P (E1 or E2 or … or EM)

1. One example of a tournament on N spreads where every spread is beaten by some spread is a tournament where the spreads are arranged in a line and each spread beats the one to its right. In this case, the spread at the right end of the line beats every other spread, while the spread at the left end is beaten by every other spread.

A → B

↓ ↓

D ← C

**2.)** The minimum n value is **7**

**3.)**Under the assumption that each spread has an equal likelihood of winning against any other spread, the probability of a particular spread s, which is not present in the set S, winning against specific spreads in S is 1/2. This is because there are two possible outcomes, both of which are equally fair. Additionally, since each competition's outcome is independent, the likelihood of spread s winning against all k spreads in S is (1/2)^k. Therefore, the probability of a given spread s not in S winning against every spread in S is also

**4.)** For a given set S of k-many spreads, the probability that no spread outside of S beats everything in S is   
To see why, note that for there to be no spread outside of S that beats everything in S, every spread not in S must lose at least one matchup against a spread in S. Each of these matchups is determined by a fair coin flip, so the probability that a given spread outside of S loses to a given spread in S is 1/2. Since there are n-k spreads outside of S and k spreads in S, there are (n-k)\*k matchups between a spread outside of S and a spread in S. These matchups are independent, so the probability that a given spread outside of S loses all its matchups against spreads in S is (1/2)^k. Therefore, the probability that no spread outside of S beats everything in S is

**5.)**Using 2.0 and 2.4, we can give a bound on the probability that, for a randomly generated tournament on n bagel spreads, there exists a k-set winner. Let p be the probability that a given set of k spreads is a k-set winner. Then the probability that there exists a k-set winner is at most:

P(A) <= (sum for all subsets S of size k), P(no spread outside of S beats everything in S). There are (n choose k) subsets of size k. Therefore, using the result from 3.4,

we get **P(A) <=**

**6.)**Give a condition on n and k that guarantees there is a tournament with no k-set winner.

A tournament that does not have a k-set winner means that there is no group of k spreads that can defeat all other spreads in the tournament. This means that for any set S containing k spreads, there must be at least one spread that can beat all of the spreads in S. The tournament lacks a k-set winner if and only if there is a set of n-k+1 spreads that can beat every set of k spreads. We require a minimum of (n-k+1)^k distinct sets of k spreads. Therefore, it must be true that **(n-k+1)^k > (n choose k).**

**7.)**For k=2, based on the condition in 3.6, we can conclude that if n satisfies the inequality (n-1)^2 > n(n-1)/2, then n must be greater than 5. Hence, the smallest value of n that ensures the existence of a tournament without a 2-set winner is n=6. This finding is consistent with what we learned in 3.2, where we observed that a tournament featuring 6 spreads may not have a 2-set winner. Therefore, we can conclude that **n = 6.**

**8.)** Suppose k >= 2 and let n = 2^{k}. Then, by 3.6, there is a tournament on n spreads with no k-set winner. Now, consider the set of spreads that beat everything else in the tournament. This set must have size at most k, since any larger set would contain a k-set winner. Therefore, we can choose a spread that is not in this set and add it to the tournament. The resulting tournament still has no k-set winner, since the new spread beats everything else and no k-set can beat it. By induction, we can keep adding spreads in this way until we have a tournament on arbitrarily many spreads with no k-set winner.

**9.)**In fact, show that taking k ≈ a log(n) for 0 < a < 1, there are tournaments with no k-set winners for all sufficiently large n.

- We want to show that for any 0 < a < 1, there exist tournaments on sufficiently large n bagel spreads such that there are no k-set winners, where k ≈ a log(n).

- Consider a tournament with n bagel spreads. For any fixed set S of k spreads, we can use the result from 3.4 to find the probability that no spread outside of S beats everything in S, denoted by Ps. Then, the probability that S is a k-set winner is (1 - Ps)^{n-k}. Note that this expression assumes that S is chosen uniformly at random among all sets of k spreads, which may not be the case, but we will ignore this technicality for now.

- Let a be a fixed constant in (0, 1). We want to find a value of n such that for any k ≈ a log(n), the probability that there exists a k-set winner in a randomly generated tournament on n bagel spreads is less than 1.

- Using the approximation ln(1 - x) ≈ -x for small x, we have:

(1 - Ps)^{n-k} ≈ e^{-Ps(n-k)}.

Setting Ps(n-k) = a log(n), we get:

Ps ≈ (a log(n))/(n-k).

Substituting this into the expression for the probability of having a k-set winner, we get:

(1 - Ps)^{n-k} ≈ e^{-(a log(n))} = n^{-a}.

We want this probability to be less than 1, so we need n^{-a} < 1, or equivalently, n > 1/a.

Therefore, we can choose n to be any integer larger than 1/a, and for any k ≈ a log(n), there exists a tournament on n bagel spreads with no k-set winners. This proves that taking k ≈ a log(n) for 0 < a < 1, there are tournaments with no k-set winners for all sufficiently large n.

**Problem 3**

1. If k = 1, then the expected number of people who end up infected is equal to the sum of the geometric series (1-p) + (1-p)^2 + (1-p)^3 + ..., which is equal to 1/(1-p). Therefore, the expected number of people who end up infected is finite since p is between 0 and 1.

**1/(1-p)**

**Explanation:**

1) If k = 1, then the expected number of people who end up infected is equal to the sum of the geometric series (1-p) + (1-p)^2 + (1-p)^3 + ..., which is equal to 1/(1-p). Therefore, the expected number of people who end up infected is finite since p is between 0 and 1.

1. If k = 2, then the expected number of people who end up infected is equal to the sum of the double geometric series (1-p) + (1-p)^2 + (1-p)^3 + ... + (1-p)^2\*(1-p) + (1-p)^2\*(1-p)^2 + ..., which is equal to 1/(1-2p) Therefore, the expected number of people who end up infected is finite since p is between 0 and 1.

**1/(1-2p)**

3) According to previous calculations, the number of infected individuals can be represented by **N = 1 / (1 - kp)** for any general values of k and p.

When kp exceeds 1, the number of infected individuals becomes infinite. However, if kp is less than or equal to 1, then the number of infected individuals remains finite. This is due to the fact that, as demonstrated in steps 1 and 2, if each infected individual has the potential to produce more than one "offspring," the tree of infections may continue to expand indefinitely.