CS 109: Probability for Computer Scientists
Problem Set #1

Adonis Pugh

January 17, 2020

1. Introduce yourself! Fill out this Google form to tell me a bit about you:

    `https://forms.gle/tRvuhnyXfV1LXX7t5`

    (No need to copy the answers into your Gradescope submission; you can select an arbitrary page or write "done" so there is something to select.)

Answer. DONE

2. 12 computers are brought in for servicing (and machines are serviced one at a time). Of the 12 computers, 3 are PCs, 5 are Macs, and 4 are Linux machines. Assume that all computers of the same type are indistinguishable (i.e., all the PCs are indistinguishable, all the Macs are indistinguishable, etc.).

   a. In how many distinguishable ways can the computers be ordered for servicing?

   b. In how many distinguishable ways can the computers be ordered if the first 5 machines serviced must include all 4 Linux machines?

   c. In how many distinguishable ways can the computers be ordered if 1 PC must be in the first 3 and 2 PCs must be in the last 3 computers serviced?

Answer.

a. There are $12!$ ways of permuting 12 distinct computers, but there are groups of 3, 4, and 5 indistinct computer types, so we must divide out the ways the subgroups can permute, which gives $\frac{12!}{3!4!5!} = 27,7720$ orderings.

b. There are two cases: the first 5 computers contain (1) 4 Linux and 1 Mac, or (2) 4 Linux and 1 PC. With path 1 or 2, there are $5!/4!$ permutations that can occur between these groups. After the possibilities for the first 5 computers are considered, path 1 and 3 have 7 total computers left over, path 1 with 3 PCs and 4 Macs and path 2 with 2 PCs and 5 Macs. We compute these subgroup permutations just like the others. For path 1, $\frac{5!}{4!} \cdot \frac{7!}{3!4!} = 5 \cdot 35 = 175$, and for path 2, $\frac{5!}{4!} \cdot \frac{7!}{2!5!} = 5 \cdot 21 = 105$. In total, there are 175 + 105 = 280 servicing orders.

c. This is essentially a longer version of part b. The condition that 1 PC must be in the first 3 means that this must be our first step. The condition that 2 PCs must be in the last 3 could equivalently mean that there must 2 PCs in the next group of 3. Now we have our steps: (1) Consider the possibilities and permutations of the first group of 3 given there is must be 1 PC, (2) consider the possibilities and permutations of the second (last) group of 3 given it must contain 2 PCs, and (3) consider the possibilities and permutations of the remaining computer options. There are six potential pathways: (1) start with PML, end with PPL, (2) start with PML, end with PPM, (3) start with PMM, end with PPL, (4) start with PMM, end with PPM, (5) start with PPL, end with PPL, (6) start with PLL, end with PPM. Using the rules of counting in the same way as the previously problem, the sum ends up being 270 + 630 + 180 + 135 + 54 + 135 = 1134 servicing orders.

3. At the local zoo, a new exhibit consisting of 3 different species of birds and 3 different species of reptiles is to be formed from a pool of 7 bird species and 5 reptile species. How many exhibits are possible if

   a. there are no additional restrictions on which species can be selected?

   b. 2 particular bird species cannot be placed together (e.g., they have a predator-prey relationship)?

   c. 1 particular bird species and 1 particular reptile species cannot be placed together?

Answer.

a. We must consider the number of ways to choose 3 birds from 7 and 3 reptiles from 5, so there are $\binom{7}{3} \cdot \binom{5}{3} = 35 \cdot 10 = 350$ possible exhibits.

b. The forbidden cases are the two conflicting species being placed together. There are $\binom{7-2}{3-2} = \binom{5}{1} = 5$ ways of placing the two species together. Subtracting this from the number of ways to choose 3 birds from 7, we get $\binom{7}{3} - 5 = 30$ bird combinations. Now there are $30 \cdot 10 = 300$ possible exhibits.

c. The forbidden cases are the two conflicting species being placed together from different groups. Assuming the conflicting bird and reptile species have been picked, there are $\binom{7-1}{3-1} \cdot \binom{5-1}{3-1} = \binom{6}{2} \cdot \binom{4}{2} = 90$ ways of doing so. The gives us 350 - 90 = 260 possible exhibits.

4. A substitution cipher is derived from orderings of the alphabet. How many ways can the 26 letters of the English alphabet (21 consonants and 5 vowels) be ordered if each letter appears exactly once and:

   a. There are no other restrictions?

   b. The letters Q and U must be next to each other (but in any order)?

   c. All five vowels must be next to each other?

   d. No two vowels can be next to each other?

Answer.

a. Using the rules of counting, there are $26!$ orderings of letters.

b. Let us treat QU as a single letter, in which case there are $25!$ permutations. We must also account for permuting between Q and U between themselves, so the answer is $25!2!$ orderings.

c. Let us treat AEIOU as a single letter, in which case there are $22!$ permutations. We must also account for permuting of A, E, I , O, U between themselves, so the answer is $22!5!$ orderings.

d. Let us first disregard the vowels, in which case there are $21!$ orderings of the consonants. Now, let us place in the 5 vowels one after the other. For the first placement, there are 22 options for its placement, for the second - 21 options, third - 20 options, fourth - 19 options, fifth - 18 options. Mathematically, this works out to $\frac{21!22!}{17!}$ orderings.

5. Given all the start-up activity going on in high-tech, you realize that applying combinatorics to investment strategies might be an interesting idea to pursue. Say you have $20 million that must be invested among 4 possible companies. Each investment must be in integral units of $1 million, and there are minimal investments that need to be made if one is to invest in these companies. The minimal investments are $1, $2, $3, and $4 million dollars, respectively for company 1, 2, 3, and 4. How many different investment strategies are available if

   a. an investment must be made in each company?

   b. investments must be made in at least 3 of the 4 companies?

Answer.

a. There is a minimum of 1 + 2 + 3 + 4 = 10 million dollars that must be invested, leaving another 10 million for further investment. Now, we must consider the ways to group 10 items in 4 buckets. Using the divider method, this works out to $\frac{(10+4-1)!}{10!(4-1)!} = \frac{13!}{10!3!} = 286$ strategies.

b. There are $\binom{4}{3} = 4$ possible combinations of 3 companies. Each combination has a different minimum investment total, ranging from 6 - 9 million dollars with million dollar increments. There are then $\frac{(14+3-1)!}{14!(3-1)!} + \frac{(13+3-1)!}{13!(3-1)!} + \frac{(12+3-1)!}{12!(3-1)!} + \frac{(11+3-1)!}{11!(3-1)!} =$ $120+105+91+78 = 393$ three-company strategies. Summing this with possible four-company strategies yields 393 + 286 = 680 investment strategies.

6. How many ways can you split a class of 99 students into 33 project groups of 3 students each? Neither the order of the groups nor the order of students within groups matters.

Answer.

This problem is choosing 33 groups of 3 from 99, which is mathematically expressed as $\binom{99}{3_1, 3_2, ..., 3_{33}} = \frac{99!}{3!^{33}}$

7. You are counting cards in a card game that uses two standard decks of cards. There are 104 cards total. Each deck has 52 cards (13 values each with 4 suits). Cards are only distinguishable based on their suit and value, not which deck they came from.

   a. In how many distinct ways can the cards be ordered?

   b. You are dealt two cards. How many distinct pairs of cards can you be dealt? Note: the order of the two cards you are dealt does not matter.

Answer.

a. Assuming the cards are all indistinct, there are $104!$ orderings. In actuality, there are 52 groups of 2, so we must account for this in the solution, which yields $\frac{104!}{2!^{52}}$ ways to order 104 cards with 52 pairs.

b. Since the order of the cards dealt does not matter, this becomes a problem of choosing 2 from 52 distinct cards. We must also account for the possibility that a pair of identical cards are dealt, in which case there are 52 possibilities. This works out to $\binom{52}{2} + 52 = 1326 + 52 = 1378$ pairs of cards.

8. Determine the number of vectors $(x_1, x_2, \ldots, x_n)$ such that each x_i is a non-negative integer and $\sum_{i=1}^{n} x_i \leq k$, where k is some constant non-negative integer. Note that you can think of n (the size of the vector) and k as constants that can be used in your answer.

Answer. This is a standard grouping problem, but the sum of the vector can be less than or equal to k, so let us add an imaginary element to the vector to not be counted, allowing us to generate all possible vectors whose non-negative integer sum is less than or equal to k. Using the divider method, there are $\frac{(k+(n-1+1))!}{k!(n-1+1)!} = \frac{(k+n)!}{k!n!}$ possible vectors.

9. Consider an array x of integers with k elements (e.g., `int x[k]`), where each entry in the array has a distinct integer value between 1 and n, inclusive, and the array is sorted in increasing order. In other words, $1 \leq$ `x[i]` $\leq n$, for all $i = 0, 1, 2, \ldots, k-1$, and the array is sorted, so `x[0]` $<$ `x[1]` $< \ldots <$ `x[k-1]`. How many such sorted arrays are possible?

Answer. At first glance, this problem has many constraints, but its many constraints make its solution simple. Suppose k = 3 and n = 8. The first step is to pick 3 numbers from the set 1, 2, 3, 4, 5, 6, 7, 8. Since numbers cannot be reused, this is the entire set. Generalizing this for n and k, there are $\binom{n}{k}$ ways to select the group to be used in the vector. Out of all the ways each selected group of numbers can be permuted, only one is in fully sorted order, so $\binom{n}{k}$ is our answer.

10. You are running a web site that receives 8 hits (in a particular second of time). Your web site is powered by 5 computers, and each hit to your web site can be serviced by any one of the 5 computers you have, where each computer is capable of processing as many (or as few) requests as it is given.

    a. In how many distinct ways could the 8 hits to your website be distributed among the 5 computers if all hits are considered identical?

    b. In how many distinct ways could the 8 hits to your website be distributed among the 5 computers if the hits consisted of 5 identical requests for web page A and 3 identical requests for web page B (note that requests for web page A are distinguishable from requests for web page B)?

Answer.

a. If all hits are identical, we can use the divider method to compute the number of ways 8 hits can be grouped between 5 computers, which works out to be $\frac{(8+5-1)!}{8!(5-1)!} = \frac{12!}{8!4!} = 495$ distributions.

b. Since there are groups of 5 hits and 3 hits, let us first divide the 5 hits amongst the 5 computers, then divide the 3 hits amongst the five computers. This works out to be $\frac{(5+5-1)!}{5!(5-1)!} \cdot \frac{(3+5-1)!}{3!(5-1)!} = \frac{9!}{5!4!} \cdot \frac{7!}{3!4!} = 4410$ distributions.

11. Say a university is offering 3 programming classes: one in Java, one in C++, and one in Python. The classes are open to any of the 100 students at the university. There are:

- a total of 27 students in the Java class;
- a total of 28 students in the C++ class;
- a total of 19 students in the Python class;
- 12 students in both the Java and C++ classes (note: these students are also counted as being in each class in the numbers above);
- 4 students in both the Java and Python classes;
- 11 students in both the C++ and Python classes; and
- 3 students in all three classes (note: these students are also counted as being in each pair of classes in the numbers above).

a. If a student is chosen randomly at the university, what is the probability that the student is not in any of the 3 programming classes?

b. If a student is chosen randomly at the university, what is the probability that the student is taking *exactly one* of the three programming classes?

c. If two different students are chosen randomly at the university, what is the probability that at least one of the chosen students is taking at least one of the programming classes?

Answer.

a. This is the probability of the complement of the union of these events. Using the axioms of probability and the inclusion-exclusion principle

$$\begin{aligned}(J \cup C \cup P)^C &= 1 - P(J \cup C \cup P \cup) \\ = 1 - [P(J) + P(C) + P(P) - P(JC) - P(JP) - P(CP) + P(JCP)] \\ = 1 - (0.27 + 0.28 + 0.19 - 0.12 - 0.04 - 0.11 + 0.03) \\ \boxed{P(J \cup C \cup P)^C = 0.50}\end{aligned}$$

b. If this situation were imagined as a Venn diagram, we want to compute the non-overlapping regions of the Venn diagram, that is, the union of the three class probabilities minus their intersections. Using the inclusion-exclusion principle

$$\begin{aligned}P(exactly\,one) &= P(J \cup C \cup P \cup) - P(J \cap C) + P(J \cap C \cap P) \\ -P(J \cap P) + P(J \cap C \cap P) - P(C \cap P) + P(J \cap C \cap P) - P(J \cap C \cap P) \\ P(exactly\,one) &= 0.50 - 0.12 + 0.03 - 0.11 + 0.03 - 0.04 + 0.03 - 0.03 \\ \boxed{P(exactly\,one) = 0.29}\end{aligned}$$

c. It is easier to compute this answer as the complement of the probability that neither student selected is in a programming class. The probability that neither student is in a programming class is $\frac{\binom{50}{2}}{\binom{100}{2}}$. This makes sense because the sample space is the total number of ways to choose 2 out of 100 students and the event space is the number of ways to choose 2 out of 50 non-programming students. The answer is then the complement, which is $1 - \frac{\binom{50}{2}}{\binom{100}{2}} \approx 0.75$.

12. To get good performance when working binary search trees (BST), we must consider the probability of producing completely degenerate BSTs (where each node in the BST has at most one child). See Lecture Notes # 2, Example 2, for more details on binary search trees.

    a. If the integers 1 through n are inserted in arbitrary order into a BST (where each possible order is equally likely), what is the probability (as an expression in terms of n) that the resulting BST will have completely degenerate structure?

    b. Using your expression from part (a), determine the smallest value of n for which the probability of forming a completely degenerate BST is less than 0.001 (i.e., 0.1%).

Answer.

a. To guarantee a binary search tree is created, we must start with either 1 or n to guarantee only one child, eliminate 1 and n, and continue selecting the extremes (either the min or the max) until our choices are exhausted. We must do this process a total of 2^{n-1} times. The sample space is n! binary search trees, so the event space divided by the sample space is $\frac{2^{n-1}}{n!}$, the probability that a degenerate binary search tree is made.

b. Guessing and checking, for $\frac{2^{n-1}}{n!} < 0.001$, the smallest value of n = 11.

13. Suppose that m strings are hashed (randomly) into N buckets, assuming that all N^m arrangements are equally likely. Find the probability that exactly k strings are hashed to the first bucket.

Answer. The total ways to hash m distinct strings into N buckets is N^m. Now let us consider the first bucket to already contain exactly k strings. The number of ways to hash the remaining strings is $(N-1)^{m-k}$. Dividing the event space by the sample space, the answer is $\dfrac{(N-1)^{m-k}}{N^m}$.

14. Say a hacker has a list of n distinct password candidates, only one of which will successfully log her into a secure system.

    a. If she tries passwords from the list at random, deleting those passwords that do not work, what is the probability that her first successful login will be (exactly) on her k-th try?

    b. Now say the hacker tries passwords from the list at random, but does **not** delete previously tried passwords from the list. She stops after her first successful login attempt. What is the probability that her first successful login will be (exactly) on her k-th try?

Answer.

a. Let us consider the case where k = 3. Her probability is then $\frac{n-1}{n} \times \frac{n-2}{n-1} \times \frac{1}{n-2} = \frac{1}{n}$. This can be generalized for all k, because the terms will continue to cancel out. This makes sense because there is no good reason that she should be more or less likely to guess the correct password on any given attempt. The answer is then $\frac{1}{n}$.

b. Let us again consider the case where k = 3. Her probability is $\frac{n-1}{n} \times \frac{n-1}{n} \times \frac{1}{n}$. This can also be generalized for any k, because the numerator (n - 1) will be multiplied k - 1 times and the denominator n will be multiplied k times, which works out to be $\frac{(n-1)^{k-1}}{n^k}$.

15. Say we send out a total of 20 distinguishable emails to 12 distinct users, where each email we send is equally likely to go to any of the 12 users (note that it is possible that some users may not actually receive any email from us). What is the probability that the 20 emails are distributed such that there are 4 users who receive exactly 2 emails each from us and 3 users who receive exactly 4 emails each from us?

Answer. The sample space is the total number of permutations of 20 distinct emails, which is 12^{20}. To create our event space, let us first choose 4 users from 12 to receive 2 emails. Next let us choose 3 users from the remaining 8 to receive 4 emails. The last 5 users get no emails. This is mathematically expressed as $\binom{12}{4} \cdot \binom{8}{3} = \binom{12}{4, 3}$. There are $20!$ ways to order the emails between users, but this over-counts because we do not care how the 4 groups of 2 emails can be permuted, nor do we care how the 3 groups of 4 emails can be permuted. This works out to be a $\frac{\binom{12}{4,3} 20!}{12^{20} 4!^3 2!^4}$ probability.