Alex Tsun Winter 2022
CSE 312: Foundations of Computing II

PSet #2

With problems from several past UW CSE 312 instructors (Martin Tompa, Anna Karlin, Larry Ruzzo) and Stanford CS 109 instructors (Chris Piech, David Varodayan, Lisa Yan, Mehran Sahami)

Groups: This pset may be done in groups of **up to 2 people**. However, you can not work with the same partner you had in PSet 1. This means that only one person will submit on Gradescope to "PSet 2 [Written]" and add their partner as a collaborator. The coding part still must be done individually, so each group member will submit their own coding assignment "PSet 2 [Coding]". Individuals and groups are encouraged to discuss problem-solving strategies with other classmates as well as the course staff, but each group must write up their own solutions.

Instructions: For each problem, remember you must briefly explain/justify how you obtained your answer, as correct answers without an explanation will receive **no credit**. Moreover, in the event of an incorrect answer, we can still try to give you partial credit based on the explanation you provide. It is fine for your answers to include summations, products, factorials, exponentials, or combinations; you don't need to calculate those all out to get a single numeric answer.

Submission: You must upload your written compiled LaTeX PDF to Gradescope under "PSet 2 [Written]" (with your partner if applicable) and your two code files `gen_rvs.py` and `nb.py` to "PSet2 [Coding]". Extra credit can be submitted by groups or individuals in their "PSet 2 [Written]". You must tag your written problems on Gradescope, or you will receive **no credit** as mentioned in the syllabus. Please cite any collaboration at the top of your submission (beyond your group members, which should already be listed).

1. Match the following to the most appropriate distribution (from the Zoo of Discrete RVs), **including parameters** (your answer should be in a form like $NegBin(30, 0.2)$, or $Poi(100)$, for example). Distributions may be used more than once or not at all. Suppose there are B blue fish, R red fish, G green fish in a pond, where $B + R + G = N$. **You do not need to show work for this problem**.

   (a) How many of the next 10 fish I catch are green, if I catch and release.

   (b) How many fish I had to catch until my first blue fish, if I catch and release.

   (c) How many red fish I catch in the next **five** minutes, if I catch on average r red fish per minute, if I catch and release.

   (d) Whether or not my next fish is blue, if I catch and release.

   (e) How many fish I had to catch until my fourth red fish, if I catch and release.

   (f) How many blue fish I caught in one scoop of a net containing M fish.

2. Do these parts **independently** of each other. These are routine but necessary calculations to test your understanding of mean and variance and your ability to apply formulae.

(a) Let X be a random variable with $p_X(k) = ck$ for $k \in \{1, \ldots, 5\} = \Omega_X$, and 0 otherwise. Find the value of c that makes X a valid probability distribution and compute its mean and variance ($E[X]$ and $Var(X)$).

(b) Let X be *any* random variable with mean $E[X] = \mu$ and variance $Var(X) = \sigma^2$. Find the mean and variance of $Z = \dfrac{X - \mu}{\sigma}$. (When you're done, you'll see why we call this a "standardized" version of X!)

(c) Let X, Y be independent random variables. Find the mean and variance of $X - 4Y + 2$ in terms of $E[X], E[Y], Var(X)$, and $Var(Y)$.

(d) Let $X_1, \ldots, X_n$ be independent and identically distributed (iid) random variables each with mean μ and variance σ^2. The sample mean is $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. Find the mean and variance of $\bar{X}$. If you use the independence assumption anywhere, **explicitly label** at which step(s) it is necessary for your equalities to be true.

3. There are n different Pokemon cards available. Each is sold *individually* in identical packaging, so that you cannot tell which one you will get until you buy it and open the package. Suppose that each of the n cards is equally probable whenever you buy one. Let random variable X be the number of cards Alex buys until he has a complete set of n. What is $E[X]$? (Hint: Define random variable X_i (for $i = 1, \ldots, n$) as the number of cards bought from the first time Alex has $i - 1$ different cards up to and including the first time he has i different cards. X_i is which random variable in our zoo?)

   The n-th *harmonic number* is defined as $H_n = 1 + 1/2 + 1/3 + 1/4 + \cdots + 1/n$. For your interest, as a function of n, H_n grows very similarly to $\ln n$. If you are curious, this is because $\sum_{i=1}^{n} \frac{1}{i} \approx \int_1^{n+1} \frac{1}{x} dx = \ln(n+1)$). Give your answer as a function of one of these harmonic numbers (e.g., $\sqrt{H_5} - 2$ or $\log(H_7 + 9)$).

4. Let $X_1, \ldots, X_5$ be five independent rolls of a fair six-sided die. Define $Y = \min\{X_1, \ldots, X_5\}$, the smallest value.

   (a) Compute $P(Y \geq k)$ for $k = 1, \ldots, 6$. (Hint: You may want to start with the "easier" values of $k = 6$ and $k = 1$ before the others).

   (b) Use your answer to part (a) to compute $p_Y(k) = P(Y = k)$ for $k = 1, \ldots, 6$. (Hint: Start with $p_Y(6)$ and work downwards. What's the pattern?)

   (c) Show that if Z is a **nonnegative, integer-valued** random variable, that $E[Z]$ can also be computed using the formula $E[Z] = \sum_{k=0}^{\infty} P(Z > k) = \sum_{k=1}^{\infty} P(Z \geq k)$. (Hint: Do this by writing out each term in the sum $E[Z] = \sum_{k=0}^{\infty} k \cdot p_Z(k)$, one per line like below.)

```
1p(1) = p(1)
2p(2) = p(2)+p(2)
3p(3) = p(3)+p(3)+p(3)
4p(4) = p(4)+p(4)+p(4)+p(4)
...
```

(d) Now compute $E[Y]$ using your answers to part (a) and (c). You may wish to verify your answer by computing it the "normal way" using your answer to part (b). **Give your answer to 4 decimal places**.

5. You're playing minigolf for the very first time. You play a total of 18 holes. Your score on each hole is the number of strokes you need to get the ball in the hole. Since you're a novice, the success of each of your strokes is independent. For the first 9 holes, each stroke has 20% chance of getting the ball in the hole. For the last 9 holes, you improve slightly and each stroke has 25% chance of success.

   (a) What is the probability that you take more than 4 strokes to complete the first hole? (Hint: Use a random variable from our zoo!) **Give your answer to 4 decimal places**.

   (b) A hole-in-one happens when your score for a hole is 1; that is, you get the ball in the hole on your first attempt. What is the probability that you get exactly one hole-in-one as you play all 18 holes? (Hint: Use a random variable from our zoo, but with some massaging first!) **Give your answer to 4 decimal places**.

   (c) Given that your cumulative score after the first 9 holes is 50 strokes, what is the probability that your total score after all 18 holes will be exactly 80 strokes? (Hint: Use a random variable from our zoo, but with some massaging first!) **Give your answer to 4 decimal places**.

6. You have 8 pairs of mittens, each a different color. Left and right mittens are *distinct*. Suppose that you are fostering (possibly imaginary) kittens, and you leave them alone for a few hours with your mittens. When you return, you discover that they have hidden 4 mittens! Suppose that your kittens are equally likely to hide any 4 of your 16 distinct mittens. Let X be the number of complete, distinct pairs of mittens that you have left.

   (a) Compute the probability mass function of X, $p_X(k)$. (Hint: You may want to first figure out the range Ω_X).

   (b) Find $E[X]$ using your answer to part (a) and the definition of expectation.

   (c) Find $E[X]$ again, but using linearity of expectation this time and indicator RVs. Are these indicator RVs independent? Rhetorical (answering this next part is optional): Which method was easier? What if I increased the number of pairs of mittens?

7. Suppose that it requires at least 10 out of 12 votes from a jury to convict a defendant (i.e., if there are at least 10 guilty votes, the defendant is found guilty; but if there are 9 or less guilty votes, they are found innocent). In addition, suppose that a juror votes innocent for a guilty defendant with probability 0.2, and they vote guilty for an innocent defendant with probability 0.15. Assuming each juror votes independently, and that 75% of defendants are actually guilty, find the probability that the jury renders the correct decision. (Hint: Let C be the event the jury is correct, and consider two cases). **Give your answer to 4 decimal places**.

8. UberPool gets 1 requests every **two** minutes, on average, for a particular route. A user requests this route and Uber commits a driver to take her. Any user who also requests the route in the next **five** minutes will also be added to the car as long as there is space. The car can fit up

to five people including the driver. Uber will make \$8 for each user in the car (the revenue) minus \$11 (the operating cost). What is Uber's expected profit per ride? (Hints: The number of requests is unbounded, but the number of passengers is bounded. The answer is **not** an integer.) **Give your answer to 4 decimal places**.

9. To determine whether they have llama flu, 1000 people have their blood tested. However, rather than testing each individual separately (1000 tests is quite costly), it is decided to use a group testing procedure:

   - Phase 1: First, place people into groups of 5. The blood samples of the 5 people in each group will be pooled and analyzed together. If the test is positive (at least one person in the pool has llama flu), continue to Phase 2. Otherwise send the group home. 200 of these pooled tests are performed.
   - Phase 2: Individually test each of the 5 people in the group. 5 of these individual tests are performed per group in Phase 2.

   Suppose that the probability that a person has llama flu is 5% for all people, independently of others, and that the test has a 100% true positive rate and 0% false positive rate (note that this is unrealistic). Using this strategy, compute the expected total number of blood tests (individual and pooled) that we will have to do across Phases 1 and 2.

10. A building has n floors numbered 1, 2, ..., n, plus a basement parking level B. At the basement, m people get on the elevator together. Each gets off at one of the n floors with equal probability (independently of everybody else). No-one else gets on the elevator. What is the expected number of floors the elevators stops at (not counting the basement)? (Hint: linearity of expectation.)

11. Mia and Nia are two roommates who live in the room directly above yours. They aren't friends, so whether each is in their room is **independent** of the other being in the room.

    (a) On **Saturday** night Mia goes out with probability $\frac{3}{4}$ and Nia goes out with probability $\frac{2}{3}$. What is the probability that **exactly one** of the roommates is in their room on Saturday night?

    (b) On **Sunday** night the probability that exactly one roommate is in their room is 0.5 and the probability than both are in is 0.1. When one roommate is in, she stomps loudly on the floor at an average rate of 3 stomps per hour. When both are in, they stomp at a combined average rate of 6 stomps per hour. Assume the number of stomps per hour are Poisson-distributed. You listen for an hour on Sunday night and hear no stomps coming from upstairs. Given this information, what is the probability that **exactly one** of the roommates is in their room on Sunday night? **Give your answer to 4 decimal places**.

12. Below are two sequences of Heads and Tails, each (supposedly) representing 300 independent flips of a fair coin. One of these sequences was truly randomly created, and one was typed by a human. Both sequences have exactly 149 heads. Which one is more likely to be the "real" random sequence? (Hint: You will want to write some code!) Include any code you wrote (using Python3) below in your write-up, and justify your reasoning with evidence, including plots if you made any. There are multiple valid and correct approaches.

    (a) (Sequence 1)

    HTHHHTHTTHHTTTTTTTTHHHTTTHHTTTTHHTTHHHTTHTHTTTTTTHTHTTTTHHHH
    THTHTTHTTTHTTHTTTTHTHHTHHHHTTTTTHHHHTHHHTTTTTHTHTTHHHHTHHHHHH
    HHTTHHTHHTHHHHHHHTTHTHTTTHHTTTTHTHHTTHTTHTHTHTTTHHHHHTTHTTTHT
    HTHHTTTTHTTTTTHHTHTHHHHTTTTHTHHHHHHTHTHTHTHHHTHTTHHHTHHHHHHT
    HHHTHTTTHHHTTTHHTHTTHHTHHHTHTTHTTHTTTHHTHTHTTTTHTHTHTTHTHTHT

    (b) (Sequence 2)

    TTHHTHTTHTTTHTTTHTTTHTTHTHHTHHTHTHHTTTHHTHTHTTHTHHTTHTHHTHTT
    THHTTHHTTHHHTHHTHTTHTHTTHHTHHHTTHTHTTTHHTTHTHTHTHTHTTHTHTHHH
    TTHTHTHHTHHHTHTHTTHTTHHTHTHTHTTHHTTHTHTTHHHTHTHTHTTHTTHHTTHT
    HHTHHHTTHHTHTTHTHTHTHTHTHTHHHTHTHTHTTHTHHTHTHTTHTTTHHTHTTTHT
    HHTHHHHTTTHHTHTHTHTHHHTTHHTHTTTHTHHTHTHTHHTHTTHTTHTHHTHTHTHH

13. [**Coding**] Understanding the process that leads to different random variables is a great way to gain familiarity for what they mean. For each random variable, write a function that simulates its generation process. Your function should return a random sample of that rv, with the appropriate probability. The **only** function you **can and should** use to generate randomness is `np.random.rand()`: a function that returns a uniform random float in the range $[0, 1]$. We include a solution to (a) in the starter code. Note that a function from one part may call a function from a previous part if you wish. For more clarity, we are asking you to generate a random sample from a particular distribution; multiple calls to your function can and should return different values in its range, approximately matching that variable's probability mass function.

    Write your code for the following parts in the provided file: `gen_rvs.py`

    (a) $X \sim Ber(p)$: 1 with probability p and 0 with probability $1 - p$. Implement the function `gen_ber`.

    (b) $X \sim Bin(n, p)$: the number of heads in n independent flips of a coin with probability of heads p. Implement the function `gen_bin`.

    (c) $X \sim Geo(p)$: the number of independent flips up to and including the first head, when the probability of heads is p. Implement the function `gen_geo`.

    (d) $X \sim NegBin(r, p)$: the number of independent flips up to and including the r-th head, when the probability of heads is p. Implement the function `gen_negbin`.

    (e) $X \sim HypGeo(N, K, n)$: the number of kit kats you get when you grab n from a bag consisting of N total candies, only K of which are kit kats. Implement the function `gen_hypgeo`.

    (f) $X \sim Poi(\lambda)$: the number of events in a minute, where the historical rate is λ events per minute. (Hint: Recall how we derived the Poisson PMF, and break the minute down into 2500 ms events). Implement the function `gen_poi`.

    (g) Given an arbitrary list (or numpy array) of probabilities, like $ps = [0.1, 0.3, 0.4, 0.2]$, sample an index with the appropriate probability. That is, return 0 with probability 0.1, 1 with probability 0.3, 2 with probability 0.4, and 3 with probability 0.2. Implement the function `gen_arb`.

14. [**Coding**] Use the Naive Bayes Classifier to implement a spam filter that learns word spam probabilities from our pre-labeled training data and then predicts the label (ham or spam) of a set of emails that it hasn't seen before. See the slides from Section 2 on 1/13/22 for details on implementation!

    Write your code for the following parts in the provided file: `nb.py`.

    Some notes and advice:

    - Read about how to avoid floating point underflow using the log-trick in the notes.
    - Make sure you understand how Laplace smoothing works.
    - Remember to remove any debug statements that you are printing to the output.
    - **Do not directly manipulate file paths or use hardcoded file paths.** A file path you have hardcoded into your program that works on your computer won't work on the computer we use to test your program.
    - Needless to say, you should practice what you've learned in other courses: document your program, use good variable names, keep your code clean and straightforward, etc. Include comments outlining what your program does and how. We will not spend time trying to decipher obscure, contorted code. Your score on Gradescope is your final score, as you have unlimited attempts. **START EARLY**.
    - We will evaluate your code on data you don't have access to, in addition to the data you are given.

    Remember, it is not expected that Naive Bayes will classify every single test email correctly, but it should certainly do better than random chance! As this algorithm is deterministic, you should get a certain specific test accuracy around 90-95%, which we will be testing for to ensure your algorithm is correct. Note that we will run your code on a test dataset you haven't seen, but you will know immediately if you got full score.

    (a) Implement the function `fit`.

    (b) Implement the function `predict`.

15. **(Extra Credit)**: If you worked with a partner that you were randomly paired with during a social event or through the partner survey, attach a screenshot here to get extra credit! If it was a social event zoom call, your screenshot must include the zoom meeting information to prove it was one of our social zoom meetings.