Chris Piech
CS109

Fall 2025
Oct 27^{th}

CS109 Midterm Exam

This is a closed calculator/computer/phone/smart-watch/smart-toothbrush exam. You are, however, allowed to use notes in the exam. You have 2 hours (120 minutes) to take the exam. The exam is 120 points, meant to roughly correspond to one point per minute of the exam. You may want to use the point allocation for each problem as an indicator for pacing yourself on the exam.

In the event of an incorrect answer, any explanation you provide of how you obtained your answer can potentially allow us to give you partial credit for a problem. For example, describe the distributions and parameter values you used, where appropriate. It is fine for your answers to include summations, products, factorials, exponentials, and combinations. You can leave your answer in terms of Φ (the CDF of the standard normal) or Φ^{-1}. For example $\Phi(3/4)$ is an acceptable final answer.

I acknowledge and accept the letter and spirit of the honor code. I pledge to write more neatly than I have in my entire life:

Signature: ____

Family Name (print): ____

Given Name (print): ____

Stanford Email (@stanford.edu): ____

1. Counting Rain Drops [18 points]

a. (6 points) If $Y \sim \text{Poisson}(\lambda = 2)$, what is the probability that $Y \leq 10$?

b. (6 points) A truncated Poisson $T \sim \text{Trunc}(\lambda = 2, n = 10)$ can only take on values 0 through n. It has the following PMF:

$$P(T = t) = \begin{cases} K \cdot \dfrac{\lambda^t e^{-\lambda}}{t!} & \text{if } 0 \leq t \leq n \\ 0 & \text{otherwise} \end{cases}$$

Solve for K when $\lambda = 2$ and $n = 10$.

c. (6 points) A microphone counts the number of raindrops. The number of drops that land in a second follows a Poisson distribution with rate $\lambda = 3$ drops per second. However, if more than 20 drops hit in a single second, the microphone experiences signal overlap, and the count for that second is not recorded. As a result, the recorded data only include values between 0 and 20. Based on this process, what is the probability that a recorded value is 1? Solve for any constants.

2. Vibrant Variables [18 points]

a. (6 points) A new training method is tested on four puppies. The method successfully gets three of the puppies to sit when told, but one puppy ignores the command. Assume a Uniform$(0, 1)$ prior for the probability that the training method succeeds. What is the variance of the posterior distribution (after observing the four puppies) for this probability?

b. (6 points) In a population of sea-horses, the number of striped offspring per cycle is observed to follow an approximately Normal distribution with mean 60 and variance 24. We suspect that the observed Normal is the result of a binomial process. What are values for the parameters of a binomial n and p that are consistent with the observed Normal.

c. (6 points) You are counting trees in a forests using satellite data. On average, there are 5.7 trees per every 10 meters squared. Assume that the locations of trees are independent of one another and that the average density of trees is the same throughout the forest. What is the probability of seeing 3 trees in a 10 meters squared patch?

3. Word Identification [16 points]

We are building a reading app! A learner has just said a word, which we have recorded as a sound file. You have a dictionary `prior_words` which stores the prior probability the learner would say each word.:

```
prior_words = {
    'love': 0.001,
    'mum': 0.01,
    'kpop': 0.0002,
    # ...
}
```

If a word is not in this dictionary the prior probability the learner said that word is 0.

You have access to an API `whisper_pr(sound_recording, test_word)` which returns the probability of observing a particular sound recording, given the learner was trying to say the `test_word`.

Implement a function `probability_from_recording` that computes and returns a dictionary mapping each word in `prior_words` to its posterior probability, given the sound recording.

```
def probability_from_recording(sound_recording, prior_words):
```

4. Variance Reduction Sampling [22 points]

This toy problem is a gentle introduction to variance reduction sampling. We will show how different sampling strategies can be used to estimate the same expected value, while having very different variances. We want to estimate $\mathrm{E}[Y-X]$ where $X \sim \mathrm{Bern}(p)$ and $Y \sim \mathrm{Bern}(p+k)$ where $p = 0.4$ and $k = 0.01$. For this learning experience we are going to think about two different sampling strategies:

```
def simulation_1(p=0.4, k=0.01):
    X = bern(p)             # samples a bernoulli with given probability
    Y = bern(p+k)           # independent sample
    return Y - X

def simulation_2(p=0.4, k=0.01):
    X = bern(p)
    if X == 1:
        Y = 1
    else:
        Y = bern(k/(1-p))
    return Y - X
```

a. (5 points) In `simulation_2`, based on the code, what is the probability that $Y = 1$? Show your work.

b. (7 points) Show that the expectation of the value returned in `simulation_2` is the same as the expectation of the value returned in `simulation_1`.

c. (7 points) What is the variance of the value returned from `simulation_2`?

d. (3 points) The expected return for both functions is 0.01. The variance of the return for `simulation_1` is 0.482 and the variance of the return for `simulation_2` is 0.0099. You collect 100 samples from each.

What is the distribution of the average calculated from the 100 samples from `simulation_1`?

What is the distribution of the average calculated from the 100 samples from `simulation_2`?

5. Outlier Detection for Code in Place [16 points]

A Code in Place student has written a Python solution to an assignment. Perhaps it's especially creative, or perhaps it is doing something different than what the problem asked. We want to quickly check the probability that their solution is an outlier!

To detect if it is an outlier we are going to use two "landmark" Python solutions to the same problem (solution 1, solution 2). For each landmark we can calculate the distance between the student code and the landmark. Distance is calculated using a modified edit distance that gives continuous distance values.

Probabilistic Model of Student Code
Each student is in one of three mutually exclusive **states**:

- L_1, they are working towards landmark 1.
- L_2, they are working towards landmark 2.
- E, they are working towards neither landmark.

When a student is working towards landmark i, the edit distance between their code and landmark i is distributed as an Exponential with $\lambda = 1$. When they are **not** working towards landmark i, the edit distance between their code and landmark i is distributed as a Normal with $\mu = 3$ and $\sigma^2 = 1$.

Your prior belief is that $P(E) = 0.1$. Of the remaining students, half are in state L_1 and the other half are in L_2.

You may assume: Each distance measure is independent once you know what **state** a student is in.

For a particular student solution, you observe the following two distances to the landmarks:

Landmark	Distance to Landmark
1	1.4
2	2.2

What is the probability the student code is in state E given the two observed distances? You may write your answer either as a math expression or as code.

(continued from the previous page)

6. Random Molecules [15 points]

There are $n = 101$ particles in a molecular system. Between every pair of particles (i, j), a bond is formed independently with probability 0.2. Bonds are "undirected". In other words there is no distinction between a bond (i, j) and a bond (j, i).

a. (7 points) What is the probability that particle 1 has at least one bond?

b. (8 points) What is the expected number of particles with at least one bond?

7. The Golden Coin [15 points]

Consider a fair gold coin and a fair silver coin. The silver coin is flipped until it gets a heads. It takes M trials. We then flip the gold coin M times.

What is the probability the gold coin comes up heads exactly 3 times?

Use the following identity:

For any positive integer x and t such that $0 < t < 1$:

$$\sum_{n=x}^{\infty} \binom{n}{x} t^n = \frac{t^x}{(1-t)^{x+1}}.$$

That's all folks! We hope you had fun. Here are some optional notes for further curiosity. If I had first pass student code outlier detection, I would use it in Code in Place. Variance reduction sampling has been a big source of breakthroughs in machine learning over the last few years. Word identification is an open problem for apps that try to teach children to read (though preferably without using an LLM).