Lisa Yan and Jerry Cain
CS 109

Problem Set #5
October 26, 2020

Problem Set #5
Due: 1:00pm on Friday, November 6

With problems by Mehran Sahami, Chris Piech, David Varodayan, Anand Shankar, and Lisa Yan

- Submit on Gradescope by 1:00pm Pacific on Friday, November 6th, for a small, "on-time" bonus.
- All students have a pre-approved extension, or "grace period" that extends until Monday 1:00pm Pacific, when they can submit with no penalty. **The grace period expires on 1:00 Pacific on Monday, November 9th**, after which we cannot accept further late submissions.
- **Collaboration policy:** You are encouraged to discuss problem-solving strategies with each other as well as the course staff, but you must write up your own solutions and submit individual work. Please cite any collaboration at the top of your submission.
- **Tagging written problems**: When you submit your written PDF on Gradescope you must tag your PDF, meaning that you must assign pages of your PDF as answers to particular questions so that we can properly grade your submission. For problem sets, we are deducting **2 points** for any submissions that do not have all questions tagged.
- **For each problem, briefly explain/justify how you obtained your answer.** Brief explanations of your answer are necessary to get full credit for a problem even if you have the correct numerical answer. The explanations help us determine your understanding of the problem whether or not you got the correct answer. 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.
- If you handwrite your solutions, you are responsible for making sure that you can produce **clearly legible** scans of them for submission. You may also use any word processing software you like for writing up your solutions. On the CS109 webpage we provide a template file and tutorial for the LaTeX system, if you'd like to use it.

Written Problems

1. The joint probability density function of continuous random variables X and Y is given by:

$$f_{X,Y}(x,y) = c\frac{y}{x} \qquad \text{where } 0 < y < x < 1$$

   a. What is the value of c in order for $f_{X,Y}(x,y)$ to be a valid probability density function?
   b. Are X and Y independent? Explain why or why not.
   c. What is the marginal density function of X?
   d. What is the marginal density function of Y?
   e. What is $E[X]$?

2. A robot is located at the *center* of a square world that is 10 kilometers on each side. A package is dropped off in the robot's world at a point (x, y) that is uniformly (continuously) distributed in the square. If the robot's starting location is designated to be $(0, 0)$ and the robot can only move up/down/left/right parallel to the sides of the square, the distance the robot must travel to get to the package at point (x, y) is $|x| + |y|$. Let D = the distance the robot travels to get to the package. Compute $E[D]$.

3. Let X, Y, and Z be independent random variables, where $X \sim \mathcal{N}(\mu_1, {\sigma_1}^2)$, $Y \sim \mathcal{N}(\mu_2, {\sigma_2}^2)$, and $Z \sim \mathcal{N}(\mu_3, {\sigma_3}^2)$.

   a. Let $A = X + Y$. What is the distribution (along with parameter values) of A?
   b. Let $B = 4X + 3$. What is the *joint* distribution (along with parameter values) of B and Z? (Hint: Bivariate Normal)
   c. Let $C = aX - b^2Y + cZ$, where a, b, and c are real-valued constants. What is the distribution (along with parameter values) of C? Show how you derived your answer.

4. You are tracking the distance of a satellite from Earth by reading values from a distance measurement instrument. The instrument gives a reading that is true distance plus Gaussian noise G, where $G \sim N(0, 4)$. Before using the instrument, your belief of the true distance D of the satellite was a Gaussian distribution $D \sim N(\mu = 98, \sigma^2 = 16)$, measured in a.u. (astronomical units). Suppose the instrument reports that the satellite is 100 a.u. from Earth.

   a. What is the PDF of your prior belief of the true distance of the satellite, D?
   b. Suppose that the true distance of the satellite from Earth is t a.u. What is the likelihood (i.e., conditional PDF) that your instrument reads 100 a.u., given that the true distance of the satellite is t a.u.?
   c. What is the PDF of your posterior belief (after observing the instrument reading) of the true distance of the satellite? You may leave a constant in your PDF; you do not need to simplify.

5. Program A will run 20 algorithms in sequence, with the running time for each algorithm being independent random variables with mean = 46 seconds and variance = 100 seconds^2. Program B will run 20 algorithms in sequence, with the running time for each algorithm being independent random variables with mean = 48 seconds and variance = 200 seconds^2.

   a. What is the approximate probability that Program A completes in less than 900 seconds?
   b. What is the approximate probability that Program B completes in less than 900 seconds?
   c. What is the approximate probability that Program A completes in less time than Program B?

Coding Problems

Here are some rules that apply to **all the coding questions**:

- For coding parts, implement the corresponding functions of the provided Python 3 files, available for download from the course website. For written parts, write your answer in the PDF that gets submitted to Gradescope. Do not rename any files.
- Your code will be autograded.

- Do not use global variables.
- You may define helper functions if you wish.
- All data files are located in the `datasets` directory, so you can use the provided helper function `get_filepath(filename)` defined in `cs109_pset5_util.py` to resolve paths, e.g., `get_filepath("email.txt")` returns `datasets/email.txt`. This function is imported for you in each provided file and used by the provided `main` methods.

6. Did you know that computers can identify you not only by what you write, but also by how you write? Coursera uses Biometric Keystroke signatures for plagiarism detection. If you cannot write a sentence with the same statistical distribution of key press timings as in your previous work, they assume that you are not the person sitting behind the computer. In this problem we provide you with several data files:

    - `personKeyTimingA.txt` has keystroke timing information for a user A writing a passage. The first column is the time in milliseconds (since the start of writing) when the user hit each key. The second column is the key that the user hit.
    - `personKeyTimingB.txt` has keystroke timing information for a second user (user B) writing the same passage as the user A. Even though the content of the passage is the same the timing of how the second user wrote the passage is different.
    - `email.txt` has keystroke timing information for an unknown user. We would like to know if the author of the email was user A or user B.

    Let X and Y be random variables for the duration of time, in milliseconds, for users A and B (respectively) to type a key. Assume that each keystroke from a user has a duration that is an independent random variable with the same distribution.

    For coding problems, write your answer in the relevant function of `cs109_pset5_email.py`. Follow the same coding guidelines as listed above. For written problems, write your answer in the PDF that gets submitted to Gradescope.

    a. **[Coding]** Complete the function `get_expectation` provided in the starter code. This function takes in a `filepath` (which is a path to either `personKeyTimingA.txt` or `personKeyTimingB.txt`) and should return $E[X]$ or $E[Y]$, respectively.

    b. **[Coding]** Complete the function `get_squared_expectation` provided in the starter code. This function should return $E[X^2]$ or $E[Y^2]$, depending on which file is supplied as `filepath`.

    c. **[Written]** Use your answers to part (a) and (b) and approximate X and Y as Normal random variables with mean and variance that match their biometric data. Report both distributions.

    d. **[Written]** Calculate the ratio of the probability that user A wrote the email over the probability that user B wrote the email. You do not need to submit code, but you should include the formula that you attempted to calculate and a short description (a few sentences) of how your code works.

7. Stanford's HCI class runs a massive online class that was taken by ten thousand students. The class used peer assessment to evaluate students' work. We are going to use their data to learn more about peer graders. In the class, each student has their work evaluated by 5 peers and every student is asked to evaluate 6 assignments: five peers and the "control assignment" (the graders were unaware of which assignment was the control). All 10,000 students evaluated the same control assignment, and the scores they gave are in the file `peerGrades.csv`. You may use simulations to solve any part of this question.

   For coding problems, write your answer in the relevant function of `cs109_pset5_hci.py`. Follow the same coding guidelines as listed above. For written problems, write your answer in the PDF that gets submitted to Gradescope.

   a. **[Coding]** What is the sample mean of the 10,000 grades to the control assignment? Implement the `get_sample_mean` function, which should return this quantity as a float.

   For parts (b) and (c), you'll need to run some simulations. To get credit from the autograder, you're ***required*** to abide by the following guidelines:

   - Run the algorithm for exactly 10,000 iterations.
   - You'll need to draw random samples with replacement from an array of grades. To do so, you must use the `np.random.choice` function, which you can call like so: `sample = np.random.choice(`*name-of-array*`,` *size-of-random-sample*`, replace=True)`. Do not use any other function to generate random samples.
   - Use the `np.mean`, `np.median`, and `np.var` functions to calculate the mean, median, and variance of a list or numpy array.

   b. **[Coding]** Students could be given a final score which is the *mean* of the 5 grades given by their peers. Imagine the control experiment had only received 5 peer-grades. What is the variance of the mean grade that the control experiment would have been given? Implement the `get_mean_var` function, which should return this quantity as a float.

   c. **[Coding]** Students could be given a final score which is the *median* of the 5 grades given by their peers. Suppose the control experiment had only received 5 peer-grades. What is the variance of the median grade that the control experiment would have been given? Implement the `get_median_var` function, which should return this quantity as a float.

   d. **[Written]** Would you use the mean or the median of 5 peer grades to assign scores in the online version of Stanford's HCI class? Hint: it might help to visualize the scores. Feel free to write code to help you answer this question, but for this question we'll solely evaluate your written answer in the PDF that you upload to Gradescope.

8. In this problem you are going to learn how to use and misuse p-values for experiments that are called *A/B tests*. These experiments are ubiquitous. They are a staple of both scientific experiments and user interaction design.

   Suppose you are working at Coursera on new ways of teaching a concept in probability. You have two different learning activities `activity1` and `activity2` and you want to figure out which activity leads to better learning outcomes. Over a two-week period, you randomly assign each student to be given either activity1 or activity2. You then evaluate each student's learning outcomes by asking them to solve a set of problems.

The data (the activity shown to each student and their measured learning outcomes) are found in the file learningOutcomes.csv. For coding problems, write your answer in the relevant function of cs109_pset5_coursera.py. Follow the same coding guidelines as listed above. For written problems, write your answer in the PDF that gets submitted to Gradescope.

a. **[Coding]** What is the difference in sample means of learning outcomes between students who were given activity1 and students who were given activity2? Write your answer in the diff_sample_means function, which should return a float (i.e. the difference in sample means).

b. **[Coding]** Write code to estimate the p-value (using the bootstrap method) for the observed difference in means reported in part (a). In other words: assuming that the learning outcomes for students who had been given activity1 and activity2 were identically distributed, what is the probability that you could have sampled two groups of students such that you could have observed a difference of means as extreme, or more extreme, than the one calculated from your data in part (a)? Write your answer in the estimate_p_val function, which should return a float. Here are some guidelines to follow:

   - Just like in the previous problem, you are ***required*** to use the *np.random.choice* method with *replace=True* to generate random samples.
   - For the bootstrap algorithm, you should use 10,000 iterations, i.e. you should resample 10,000 times.
   - If you have two lists *a* and *b*, you can create a new list containing all the elements of *a* followed by all the elements of *b* by writing *a + b*

Scientific journals have traditionally accepted an experiment's result as "statistically significant" if the p-value is below 0.05. By definition, this standard means that 5% of findings published in these journals are in fact not true, but just false positives. The scientific community is beginning to move away from using arbitrary p-value thresholds to determine whether a result is publishable. For example, see this 2019 editorial in the journal *Nature*: https://www.nature.com/articles/d41586-019-00874-8.

You are now troubled by the p-value you obtained in part (b), so you decide to delve deeper. You investigate whether learning outcomes differed based on the background experience of students. The file background.csv stores the background of each student as one of three labels: more experience, average experience, less experience.

c. **[Written]** For each of the three backgrounds, calculate a difference in means in learning outcome between activity1 and activity2, and the p-value of that difference.

   You'll almost certainly need to write code in this question, and we've provided an optional_function that you can use, which gets called by our provided main method. However, we won't grade any code for this part. We'll only grade what you include in your answer PDF.

d. **[Written]** Your manager at Coursera is concerned that you have been "p-hacking," which is also known as data dredging: https://en.wikipedia.org/wiki/Data_dredging. In one sentence, explain why your results in part (c) are not the result of p-hacking.