Oishi Banerjee and Cooper Raterink Summer 2020

CS109 Final Quiz

Take-Home Quiz information

This quiz will be a 48-hour open-book, open-note, open-calculator exam. Quiz material will take about 3-4 hours of active work (*before* typesetting), though it may be more depending on how you prepare your submission. Unlike the problem sets, the take-home quizzes are ***strictly individual work***. Even course staff assistance will be limited to clarifying questions of the kind that might be allowed on a traditional, in-person exam. If you have questions during the exam, please ask them via our discussion forum. We will not have any office hours for answering quiz questions during the quiz.

For each problem, briefly explain/justify how you obtained your answer at a level such that a future CS109 student would be able to understand how to solve the problem. Please derive your mathematical expressions analytically, not via computerized simulation. For example, please use mathematical principles to solve counting problems, rather than writing a brute-force program to generate all possible outcomes. (That said, you may use code to *check* your analytically-derived answers.)

It is fine for your answers to be well-defined mathematical expressions including finite summations (but not integrals or unevaluated derivatives), products, factorials, exponentials, and combinations, unless the question *specifically* asks for a numeric quantity. Where numeric answers are required, the use of fractions is fine. You may use a calculator (e.g. WolframAlpha) to evaluate your mathematical expressions.

Honor Code Guidelines for Take-Home Quizzes

The take-home exams are open-book (open lecture notes, handouts, textbooks, course lecture videos, and internet searches for conceptual information, e.g., Wikipedia). Consultation of other humans in any form or medium (e.g., communicating with classmates, asking questions on forum websites such as StackOverflow) is prohibited. All work done with the assistance of any external material in any way (other than provided CS109 course materials) must include citation (e.g., "Referred to Wikipedia page on DeMorgan's Law for Question 2."). Copying solutions is unacceptable, even with citation. If by chance you encounter solutions to the problem, navigate away from that page before you feel tempted to copy. If you become aware of any honor code violations by any student in the class, your commitments under the Stanford Honor Code obligate you to inform course staff. *Please remember that there is no reason to violate your conscience to complete a take-home exam.*

Submission

You should upload your submission as a PDF to Gradescope. We provide a LaTeX template if you find it useful, but we will accept any legible submission. You can submit multiple times; we will only grade the last submission you submit before the deadline$^{\text{th}}$. Late submissions will not be accepted. ***Please double-check that you submit the right file.*** When uploading, PLEASE assign pages to each question.

I acknowledge and accept the letter and spirit of the Honor Code:

Name (typed or written): ____

1 Potpourri [34 points]

a. (6 points) For each subpart of this question, you should select ALL of the following distributions that apply:

1. $Beta(1, 1)$
2. $Beta(2, 6)$
3. $Beta(6, 2)$
4. $Beta(4, 5)$
5. $Beta(3, 11)$
6. $Uniform(0, 1)$ - (the continuous Uniform)

You are given a two-sided coin that comes up heads with probability p.

i. Which distributions model the prior for p, if you believe p is equally likely to take any value from 0 to 1, inclusive?

ii. Which distributions model the posterior for p after seeing the coin produce 2 heads and 10 tails, if you assume a uniform prior belief?

iii. Which distributions model the prior for p, if you believe the coin has a 20% chance of coming up heads?

b. (6 points) For each subpart of this question, you should select ALL of the following distributions that apply:

1. $Bin(100, 0.5)$
2. $Bin(50, 0.5)$
3. $Bin(20, 0.5)$
4. $Poi(10)$
5. $Exp(10)$
6. $N(0, 1)$

i. You want to model the distribution that produced this data: [15,25,21]. Which of the above distributions are valid models (that is, the data could possibly be produced by that model)?

ii. Same as (i), but for this data: [-0.1,0.1,0].

iii. Same as (i), but for this data: [0.5,2,1].

c. (6 points) Let X, Y, and Z be IID Normal with means $\mu_X = 4$, $\mu_Y = 5$, and $\mu_Z = 6$ and variances $\sigma_X^2 = 16$, $\sigma_Y^2 = 25$, and $\sigma_Z^2 = 36$. Let $A = X + Y$ and $B = Y + Z$. Find the Bivariate Normal distribution of (A, B). Specifically, you should find the mean vector and covariance matrix for this distribution.

d. (12 points) Suppose student scores, R, on a 150-question exam are computed as follows:

$$R = \sum_{i=1}^{50} M_i + 0.5 \cdot \sum_{j=1}^{100} W_j$$

Where M_i and W_j are IID Beta random variables (the questions are scored on a continuous scale from 0 to 1). The parameters are $\alpha_M = 10, \beta_M = 2$ and $\alpha_W = 8, \beta_W = 4$ respectively.

If we sample 100 student scores and compute their sample average $\bar{S_R}$, what is the approximate distribution of $\bar{S_R}$?

e. (4 points) We would like to estimate the parameter n of a Binomial. We know $p = 0.5$. If we observe k datapoints X_i, where $i = 1, 2, ..., k$, we estimate $\hat{n} = \max_{i=1,2,...,k} X_i$. Is this estimator unbiased? Explain why or why not in one sentence.

2 Admissions [24 points]

You work in an admissions office. You look at each applicant's academic records and give them an academic score, A. You also look at each applicant's other activities and give them an extracurricular score, B. A and B are both distributed as continuous random variables ranging from 0 to 20. Your office rejects all students whose A and B scores are both under 18 and accept some of the remaining students.

Among accepted students, $f_{A,B}(a, b) = c * a^{.03} * b^{.05}$ for all valid pairs (a,b), and 0 otherwise. To be a valid pair of scores, both scores must fall between 0 and 20, and at least one score must be over 18. c is a mystery constant.

(*Note:* For full credit, please do not leave any unevaluated integrals in your final answers. A calculator such as WolframAlpha may be helpful here!)

a. (4 points) Among admitted students, are A and B independent? You don't have to do any in-depth calculations here, but please give a brief informal explanation.

b. (6 points) What is c?

c. (6 points) Among admitted students, what is the distribution of A scores? You may leave your answer in terms of c.

d. (4 points) How likely is an admitted student to have both A and B scores over 19? You may again leave your answer in terms of c.

e. (4 points) For an admitted student, what is $P(A = 2)/P(B = 2)$? For this problem, you can leave your answer in terms of PDFs or CDFs, such as $f_{A,B}(a, b)$, $f_A(a)$, and $f_B(b)$. A numeric answer is not required here.

3 Psuspicious Pseudocode [12 points]

While testing the efficacy of a new drug, Skylar has collected 1000 data samples. Most of the samples came from patients who were treated with the drug. The rest of the samples came from patients who only received a placebo. Skylar observed that the sample mean blood pressure in the treated group was 80, while the sample mean blood pressure in the placebo group was 86. To show this difference is statistically significant, Skylar needs a properly calculated two-tailed[1] p-value.

Skylar wrote the following bootstrapping code and got a concerning p-value! Unfortunately, their code isn't quite right:

```
import numpy as np

#a helper method to be used later
def resample(whole,num_samples):
    return np.random.choice(whole, num_samples, replace=True)

# list_treat is an ordinary 1-d numpy array
# it contains all the diastolic blood pressures
# of each patient who was treated

# list_placebo is an ordinary 1-d numpy array
# it contains the diastolic blood pressures
# of each patient who received a placebo
def pvalue(list_treat, list_placebo):
    #np.concatenate will make a 1000-element array
    #containing the elements of both list_treat and list_placebo
    whole = np.concatenate([list_treat,list_placebo])
    threshold = np.mean(list_treat)-np.mean(list_placebo)
    counter = 0
    num_trials = 100000
    for trial in range(num_trials):
        sample_treat = resample(list_treat, 500)
        sample_placebo = resample(list_placebo, 500)
        mean_treat = np.mean(sample_treat)
        mean_placebo = np.mean(sample_placebo)
        new_diff = np.abs(mean_treat-mean_placebo)
        if new_diff == threshold:
            counter += 1
    return counter/num_trials
```

[1]Don't worry if this term is unfamiliar! Two-tailed p-values are the only type of p-value we cover in this class.

(12 points) Please point out the algorithmic errors. State what Skylar should do instead, explaining why each change you would make is necessary for correct bootstrapping. By our count, there are 4 lines with algorithmic errors.

Note: For each explanation, please write at most 2 sentences. No essays required!

4 A Natural Progression of Regression [15 points]

a. (6 points) Your lab has trained a regression model to predict how someone will behave during an upcoming election. For each person, there are three possible outcomes:
   1. Event A: They will vote for Candidate A
   2. Event B: They will vote for Candidate B
   3. Event C: They will choose not to vote

   For each voter, you use four binary variables as the input. Your model outputs three probabilities: P(A), P(B) and P(C).

   Your model contains fifteen parameters. You can organize these into three seperate groups of parameters, one per event:

Event	Index 0	Index 1	Index 2	Index 3	Index 4
θ^A	0	5	5	5	-4
θ^B	-3	0	1	2	5
θ^C	-1	-2	-7	-10	3

   Here's how you predict a voter's behavior, where the input for that voter is x, a 4-element vector.
   1. First, pick an outcome of interest, I.
   2. Get θ^I. For example if we picked A as our outcome of interest, we'd look for θ^A, which is [0,5,5,5,-4].
   3. Let z_I be defined as $\theta_0^I + \sum_{i=1}^4 \theta_i^I x_i$.

   For any outcome of interest I, the probability that the outcome of interest occurs is $e^{z_I}/(e^{z_A}+e^{z_B}+e^{z_C})$. You want to predict how one particular voter will behave. This person's input vector is (1,0,0,1). According to your model, what's the probability that they will vote for Candidate A?

Your lab decides to train a new logistic regression model, which will predict whether someone volunteers with a campaign or not. $y^{(i)}$ is the ith datapoint's label, and you have n datapoints overall. $y^{(i)}$ is a binary label that equals 1 if the ith person volunteers and 0 if they don't. $\widehat{y}^{(i)}$ is a probability produced by our model, which gives our predicted probability that this person volunteers.

You want to find parameters that maximize the following objective function:

$$\sum_{i=0}^{n} y^{(i)} \log(\widehat{y}^{(i)}) + \frac{(1 - y^{(i)}) \log(1 - \widehat{y}^{(i)})}{2}$$

.

b. (3 points) This objective function inherently prioritizes performance on some datapoints over others. Which datapoints is your lab prioritizing?

c. (6 points) You want to find optimal parameters for your new model using gradient ascent. For this purpose, find the gradient update for θ_j on an arbitrary training loop, using all n datapoints in your dataset. Assume j is an integer between 1 and m, inclusive, where m is the number of features in any given datapoint.

5 Naive Bayes...Nets? [19 points]

(6 points) You're a pollster surveying voters about whether they like two different policies. Thanks to prior research, you know each voter's opinions on the two policies are conditionally independent given the voter's party affiliation. All voters are affiliated with either the Grey party or the Purple party.

Here's a table describing your results from surveying 100 randomly selected voters.

Opinions	Is Grey-Affiliated	Is Purple-Affiliated
Likes Policy 1 and 2	2	4
Likes Policy 1, Dislikes Policy 2	50	5
Dislikes Policy 1, Likes Policy 2	10	25
Dislikes Policy 1 and Policy 2	4	0

a. (6 points) Let's make a Naive Bayes model that predicts a voter's party affiliation given their opinions on the two policies. Calculate the following probabilities, using maximum likelihood estimation.

   Note: A is a random variable that equals 1 if a voter likes policy 1 and 0 otherwise. B is a random variable that equals 1 if a voter likes policy 2 and 0 otherwise. G is a random variable that equals 1 if a voter is affiliated with the Grey party and 0 otherwise.

   $P(G = 1) =$
   $P(A = 1|G = 1) =$
   $P(B = 1|G = 1) =$
   $P(A = 1|G = 0) =$
   $P(B = 1|G = 0) =$

b. (9 points) Via a Naive Bayes model using Laplace estimation[2], predict which party someone who likes both policies is more likely to belong to. Remember to apply Laplace smoothing to conditional probabilities, but not $P(G = 0)$ or $P(G = 1)$.

[2]You should use k = 1. In other words, even before you look at the data you imagine that you've seen each outcome one time.

c. (4 points) Construct a Bayesian net that perfectly represents your Naive Bayes model from part a. Your net should:

- Clearly explain what variable each node represents.
- Give the relevant table of probabilities for each node. Include the minimum number of probabilities needed to specify the entire Bayesian net. If you find yourself reusing probabilities from part a, you can just say which ones (e.g. $P(G = 1)$) without rewriting all the numbers.
- Use the minimum possible number of edges.

If you don't want to include a drawing of your net, you can instead describe all the nodes and edges via text.