DSC 40A - Homework 1
due Thursday, April 11th at 11:59PM

Write your solutions to the following problems by either typing them up or handwriting them on another piece of paper. Homeworks are due to Gradescope by 11:59PM on the due date. You can use a slip day to extend the deadline by 24 hours; you have four slip days to use in total throughout the quarter.

Homework will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should **always explain and justify** your conclusions, using sound reasoning. Your goal should be to convince the reader of your assertions. If a question does not require explanation, it will be explicitly stated.

Homeworks should be written up and turned in by each student individually. You may talk to other students in the class about the problems and discuss solution strategies, but you should not share any written communication and you should not check answers with classmates. You can tell someone how to do a homework problem, but you cannot show them how to do it. We encourage you type your solutions in LATEX, using the Overleaf template on the course website.

For each problem you submit, you should **cite your sources** by including a list of names of other students with whom you discussed the problem. Instructors do not need to be cited.

This homework will be graded out of 40 points. The point value of each problem or sub-problem is indicated by the number of avocados shown.

Note: For Problems 6(a) and 6(c), you'll need to code your answers in Python. More detailed instructions are provided in Problem 6. Note that to submit the homework, you'll have to submit your answers PDF to the Homework 1 assignment on Gradescope, and submit your completed notebook `hw01-code.ipynb` to the Homework 1, Problems 6(a) and 6(c) autograder on Gradescope.

Problem 1. Welcome Survey

Make sure to fill out the Welcome Survey, linked here for two points on this homework!

Problem 2. The Proof is in the Pudding

In this problem, you'll prove or disprove various statements about a dataset of numbers, $y_1, y_2, ..., y_n$. But first, let's discuss the general approach. (Note that this problem looks long, but most of it is us explaining *how* to answer it!

To prove that a statement is always **true**, you must provide some sort of reason as to why it is always true, no matter what the values $y_1, y_2, ..., y_n$ are. For example, consider the statement:

"Suppose we add 5 to each of $y_1, y_2, ..., y_n$. The mean of the new dataset must be greater than the mean of the original dataset."

This statement is always true, but it's not enough just to say "This statement is always true; since we're adding a positive number to each value, the mean will also increase." That's good intuition to have, but we need to provide a more rigorous justification. Here's what a more rigorous justification might look like:

"The mean of the original dataset is $\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$. The mean of the new dataset is:

$$\frac{1}{n} \sum_{i=1}^{n} (y_i + 5) = \frac{1}{n} \left(\sum_{i=1}^{n} y_i + \sum_{i=1}^{n} 5 \right) = \frac{1}{n} \left(\sum_{i=1}^{n} y_i + 5n \right) = \frac{1}{n} \sum_{i=1}^{n} y_i + 5 = \bar{y} + 5$$

Therefore, the mean of the new dataset is equal to the original dataset's mean plus 5, so the mean of the new dataset is greater than the mean of the original dataset, and so the statement is always true."

Note that in the argument above, we didn't assume anything specifically about the numbers in the original dataset — we didn't use a specific example. Just because a statement holds true for one example, doesn't mean it always holds true!

On the other hand, to disprove a statement, what you need to show is that it is **not** always true. The easiest way to do this is to provide a counterexample, i.e. a set of values $y_1, y_2, ..., y_n$ where the statement is false. For example, consider the statement:

"The smallest number in the dataset must be less than the mean."

Valid justification might look like:

"This statement is not always true. For example, consider the case where our dataset only contains one unique number, like $8, 8, 8$. Here, the mean is 8 and the smallest number is 8, so the smallest number is not less than the mean, and so the statement is not always true."

This is a counterexample, and is a sufficient disproof. (Fun fact: there exist entire books about counterexamples!)

Note that in both of the examples above, our answers clearly stated whether or not we thought the statement was always true. Your answers should do the same.

Now it's your turn! Consider a dataset of numbers $y_1, y_2, ..., y_n$. Prove or find a counterexample to disprove each of the following statements.

a) At least half of the numbers in the dataset must be smaller than the mean.

b) Suppose that all of the elements in the dataset are unique. Then, removing the largest element in the dataset must increase the mean.

c) Suppose that all of the elements in the dataset are unique, that n is odd, and that the mean of the dataset is not equal to the median of the dataset. Then, if we remove the median value from the dataset, the median of the new dataset must be different from the median of the original dataset.

d) Suppose we introduce a new number to the dataset that is greater than the mean of the existing dataset. The mean of the new dataset must be greater than the mean of the original dataset.

Problem 3. Linear Functions

Consider the linear function $f(x) = 9x - 4$.

a) 🥑 If $a \leq b$, show that $f(a) \leq f(b)$.

b) 🥑🥑 Both of the statements below are true, but only one is a consequence of the property you proved in part (a). Which is it? Show that this statement is true, using the result of part (a).

1. $\text{Mean}(f(x_1), \ldots, f(x_n)) = f(\text{Mean}(x_1, \ldots, x_n))$

2. $\text{Median}(f(x_1), \ldots, f(x_n)) = f(\text{Median}(x_1, \ldots, x_n))$

c) 🥑🥑 Now, prove the other statement.

d) Suppose we consider a different linear function $g(x) = -2x + 5$. Prove or find a counterexample to disprove each of the following:

1. 🥑 If $a \leq b$, then $g(a) \leq g(b)$.

2. 🥑🥑 $\text{Mean}(g(x_1), \ldots, g(x_n)) = g(\text{Mean}(x_1, \ldots, x_n))$

3. 🥑🥑 $\text{Median}(g(x_1), \ldots, g(x_n)) = g(\text{Median}(x_1, \ldots, x_n))$

Problem 4. Prata's Idea

In Lecture 2, we found that $h^* = \text{Median}(y_1, y_2, ..., y_n)$ is the constant prediction that minimizes mean absolute error:

$$R_{\text{abs}}(h) = \frac{1}{n} \sum_{i=1}^{n} |y_i - h|$$

Your friend Prata thinks that instead of minimizing the mean absolute error, we'd get a better constant prediction if we minimized the *product of the absolute errors*:

$$P_{\text{abs}}(h) = \prod_{i=1}^{n} |y_i - h|$$

The above formula is written using *product notation*, which is similar to summation notation, except terms are multiplied and not added. For example,

$$\prod_{i=1}^{n} a_i = a_1 \cdot a_2 \cdot a_3 \cdot \ldots \cdot a_n$$

In this problem, we'll see if Prata has a good idea.

a) 🥑🥑 Without using a calculator or computer, graph $P_{\text{abs}}(h)$ for the dataset $y_1 = -3, y_2 = 4$.

b) 🥑🥑🥑🥑 For an arbitrary data set $y_1, y_2, \ldots, y_n$, what value(s) h^* minimize $P_{\text{abs}}(h)$? Discuss the pros and cons of using Prata's prediction strategy. What factors about the data set or application will influence whether this prediction strategy gives good predictions?

Problem 5. An Alternative

In Lecture 2, we found that $h^* = \text{Mean}(y_1, y_2, ..., y_n)$ is the constant prediction that minimizes mean squared error:

$$R_{\text{sq}}(h) = \frac{1}{n} \sum_{i=1}^{n} (y_i - h)^2$$

To arrive at this result, we used calculus: we took the derivative of $R_{\text{sq}}(h)$ with respect to h, set it equal to 0, and solved for the resulting value of h, which we called h^*.

In this problem, we will minimize $R_{\text{sq}}(h)$ in a way that **doesn't** use calculus. The general idea is this: if $f(x) = (x - c)^2 + k$, then we know that f is a quadratic function that opens upwards with a vertex at (c, k), meaning that $x = c$ minimizes f. As we saw in class (see Lecture 2, slide 8), $R_{\text{sq}}(h)$ is a quadratic function of h!

Throughout this problem, let $y_1, y_2, ..., y_n$ be an arbitrary dataset, and let $\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$ be the mean of the ys.

a) 🥑🥑 What is the value of $\sum_{i=1}^{n} (y_i - \bar{y})$? Justify your answer.

b) 🥑🥑 Show that:

$$R_{\text{sq}}(h) = \frac{1}{n} \sum_{i=1}^{n} \left((y_i - \bar{y})^2 + 2(y_i - \bar{y})(\bar{y} - h) + (\bar{y} - h)^2 \right)$$

Hint: To proceed, start by rewriting $y_i - h$ in the definition of $R_{sq}(h)$ as $(y_i - \bar{y}) + (\bar{y} - h)$. Why is this a valid step? Make sure not to expand unnecessarily.

c) 🥑🥑 Show that:

$$R_{\text{sq}}(h) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \bar{y})^2 + (\bar{y} - h)^2$$

Hint: At some point, you will need to use your result from part (a).

d) 🥑 Why does the result in (c) prove that $h^* = \text{Mean}(y_1, y_2, ..., y_n)$ minimizes $R_{\text{sq}}(h)$?

4

Problem 6. Gaussian Location

Historical background (not necessary to solve the problem, but interesting):

Carl Friedrich Gauss was a German mathematician born in the 18th century, who is credited for several key ideas in mathematics that you're familiar with. For one, Gaussian elimination (also known as row reduction) in linear algebra is named after him. He's also credited for being the first person to minimize mean squared error — he did so to build a model to predict the locations of planets in the night sky.

Perhaps the most famous story involving Gauss is one from when he was just a child. His teacher, supposedly, asked him to add the integers from 1 through 100, expecting it to take him a while. However, within just a few seconds, he gave the answer 5050. In this problem, you'll use his insights to help you solve a DSC 20-style algorithmic problem efficiently.

The actual problem:

Here, we'll solve the "missing value problem." Suppose `vals` is a list of length $n-1$, containing the integers from 1 to n, inclusive, with no duplicates and in unsorted order, but with exactly 1 value missing. Your job is to write a function that finds the missing integer in `vals`. For example, if `vals = [1, 2, 5, 3]`, the missing integer is `4`.

Parts (a) and (c) of this problem will require you to write code in this supplementary Jupyter Notebook. **The code that you write in that notebook is autograded, both using public test cases that you can see in the notebook and hidden test cases that will only be run after you submit on Gradescope.**

To submit your homework, in addition to submitting your answers PDF to the Homework 1 assignment on Gradescope, also submit `hw01-code.ipynb` to the Homework 1, Problems 6(a) and 6(c) autograder on Gradescope and wait until you see all public test cases pass!

a) 🥑🥑 In the linked supplementary notebook, complete the implementation of the function `missing_value_naive`. There's nothing you need to include in your answers PDF for this part.

b) 🥑🥑🥑 Your implementation of `missing_value_naive` didn't need to be particularly efficient. To make our solution to the missing value problem more efficient, we'll use the fact that the missing value is the difference between the sum we'd expect if none of the values were missing, and the sum of the values we actually have. For example, if `vals = [1, 2, 5, 3]`, then $n = 5$, so the sum we'd expect is $1+2+3+4+5 = 15$, and so the missing value is $(1+2+3+4+5)-(1+2+5+3) = 15-11 = 4$.

To find the expected sum, instead of using a `for`-loop or something like `sum(range(1, n+1))`, we can turn to Gauss to make things even faster. Let's look at how he quickly derived that $1+2+...+99+100 = 5050$. Your job will then be to generalize what he did to $1 + 2 + ... + n$, and find a formula for the sum of the first n positive integers.

Let $S_{100} = 1 + 2 + ... + 99 + 100$. Gauss wrote out S_{100} in two different ways:

$$S_{100} = \quad 1 \quad + \quad 2 \quad + \quad 3 \quad + \quad \cdots \quad + \quad 98 \quad + \quad 99 \quad + \quad 100$$
$$S_{100} = \quad 100 \quad + \quad 99 \quad + \quad 98 \quad + \quad \cdots \quad + \quad 3 \quad + \quad 2 \quad + \quad 1$$

Then, he noticed that each vertical pair of terms — 1 and 100, 2 and 99, 3 and 98, and so on, until 100 and 1 — each summed to 101. By adding the two lines above, he saw:

$$2S_{100} = 101 + 101 + 101 + ... + 101 + 101 + 101$$

Since there were 100 terms in each of the original equations for S_{100}, there were 100 terms equal to 101 in the equation above for $2S_{100}$. This let him solve for S_{100}:

$$2S_{100} = 101 + 101 + 101 + \ldots + 101 + 101 + 101$$
$$2S_{100} = 100 \cdot 101$$
$$S_{100} = \frac{100 \cdot 101}{2} = 50 \cdot 101 = 5050$$

Now, it's your turn. Let $S_n = 1 + 2 + \ldots + n$. Find a closed-form expression for S_n, for any integer $n \geq 1$, and show your work. Your answer will be an arithmetic expression involving n; for example, an incorrect answer in the correct format could be $4n^3$.

Hint: This problem may seem daunting at first, but most of the work has already been done for you. What you need to do is repeat Gauss' work, but with an arbitrary integer n instead of 100. When you expand S_n twice, the way that Gauss expanded S_{100} twice, what is the sum of each vertical pair of terms? How many such terms are there? Verify that your answer is correct by testing it out on $1 + 2$, $1 + 2 + 3$, $1 + 2 + 3 + 4$, etc.

c) 🥑🥑 In the linked supplementary notebook, complete the implementation of the function `missing_value_fast`. There's nothing you need to include in your answers PDF for this part.