Final Exam - DSC 10, Spring 2023

Instructions:

- This exam consists of 11 questions, worth a total of 120 points.
- Write your PID in the top right corner of each page in the space provided.
- Please write **clearly** in the provided answer boxes; we will not grade work that appears elsewhere. Completely fill in bubbles and square boxes; if we cannot tell which option(s) you selected, you may lose points.

  ◯ A bubble means that you should only **select one choice**.

  ☐ A square box means you should **select all that apply**.
- You may refer to the Reference Sheet that was provided to you. Other than that, you may not refer to any other resources or technology during the exam (including calculators).

Full Name:

PID:

Lecture Section: ◯ A (Sitting in Center 109) ◯ B (Sitting in Center 105)

Seat you are in:

By signing below, you are agreeing that you will behave honestly and fairly during and after this exam.

Signature:

Version A

Please do not open your exam until instructed to do so.

Welcome to the Final Exam for DSC 10 in Spring 2023!

You may have noticed that San Diego was quite cloudy in May. In fact, according to the National Weather Service, San Diego was the single cloudiest city in the contiguous United States in May, with clouds covering the sky 82% of the time. (Only a remote town in Alaska was cloudier!)

In this exam, we will work with the DataFrame `sun`, which describes the number of sunshine hours per month in various cities around the world. Each number in `sun` is an average across multiple years and multiple sensors.

The first 2 columns in `sun` are `"Country"` and `"City"`, which are strings describing a particular city. The next 12 columns are `"Jan"`, `"Feb"`, `"Mar"`, ..., `"Dec"`, which describe the number of sunshine hours seen each month. The last column, `"Year"`, is the sum of the month-specific columns.

The first few rows of `sun` are shown below, though `sun` has many more rows than are shown below.

	Country	City	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Year
0	New Zealand	Hamilton	229.8	192.9	193.3	165.1	138.3	112.8	126.4	144.1	147.5	174.8	187.1	207.6	2019.6
1	Uzbekistan	Tashkent	117.8	127.1	164.3	216.0	303.8	363.0	384.4	365.8	300.0	226.3	150.0	105.4	2823.9
2	United States	Honolulu	213.5	212.7	259.2	251.8	280.6	286.1	306.2	303.1	278.8	244.0	200.4	199.5	3035.9
3	Iceland	Reykjavik	20.0	60.0	109.0	164.0	201.0	174.0	168.0	155.0	120.0	93.0	41.0	22.0	1326.0
4	Nigeria	Lagos	164.3	169.5	173.6	180.0	176.7	114.0	99.2	108.5	114.0	167.4	186.0	192.2	1845.4

For instance, we see that Tashkent, Uzbekistan sees 164.3 sunshine hours in March.

Throughout the exam, assume that we have already run `import babypandas as bpd` and `import numpy as np`.

Advice: Read through all of the questions on the exam first before starting. The questions are not sorted by difficulty!

PID: _____

Question 1 (17 pts)

a) (5 pts) Complete the implementation of the function `most_sunshine`, which takes in `country`, the name of a country, and `month`, the name of a month (e.g. `"Apr"`), and returns the name of the city (as a string) in `country` with the most sunshine hours in `month`, among the cities in `sun`. Assume there are no ties.

```
def most_sunshine(country, month):
    country_only = __(a)__
    return country_only.__(b)__
```

What goes in blank (a)?

What goes in blank (b)?

b) (4 pts) In this part only, assume that all `"City"` names in `sun` are unique.
Consider the DataFrame `cities` defined below.

```
cities = (sun.drop(columns=["Country"])
              .groupby("City").mean().reset_index())
```

Fill in the blanks so that the DataFrame that results from the sequence of steps described below is identical to `cities`.

"Sort `sun.drop(columns=["Country"])` by __(c)__ in __(d)__ order __(e)__."

What goes in blank (c)?

◯ `"Country"` ◯ `"City"` ◯ `"Jan"` ◯ `"Year"`

What goes in blank (d)?

◯ ascending ◯ descending

What goes in blank (e)?

◯ and drop the `"Country"` column
◯ and drop the `"Country"` and `"City"` columns
◯ and reset the index
◯ , drop the `"Country"` column, and reset the index
◯ , drop the `"Country"` and `"City"` columns, and reset the index
◯ Nothing, leave blank (e) empty

c) (2 pts) True or False: In the code below, `Z` is guaranteed to evaluate to `True`.

```
x = sun.groupby(["Country", "Year"]).count().shape[0]
y = sun.groupby("Country").count().shape[0]
z = (x >= y)
```

◯ True ◯ False

In the next few parts, consider the following answer choices.

A. The name of the country with the most cities.
B. The name of the country with the fewest cities.
C. The number of cities in the country with the most cities.
D. The number of cities in the country with the fewest cities.
E. The last city, alphabetically, in the first country, alphabetically.
F. The first city, alphabetically, in the first country, alphabetically.
G. Nothing, because it errors.

d) (1.5 pts) What does the following expression evaluate to?

```
sun.groupby("Country").max().get("City").iloc[0]
```

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F ◯ G

e) (1.5 pts) What does the following expression evaluate to?

```
sun.groupby("Country").mean().get("City").iloc[0]
```

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F ◯ G

f) (1.5 pts) What does the following expression evaluate to?

```
sun.groupby("Country").count().sort_values("Jan").index[-1]
```

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F ◯ G

g) (1.5 pts) What does the following expression evaluate to?

```
sun.groupby("Country").count().sort_values("City").get("City").iloc[-1]
```

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F ◯ G

PID: _____

Question 2 (7 pts)

Vanessa is a big Formula 1 racing fan, and wants to plan a trip to Monaco, where the Monaco Grand Prix is held. Monaco is an example of a "city-state" — that is, a city that is its own country. Singapore is another example of a city-state.

We'll say that a row of `sun` corresponds to a city-state if its `"Country"` and `"City"` values are equal.

a) (4 pts) Fill in the blanks so that the expression below is equal to the total number of sunshine hours in October of all city-states in `sun`.

```
sun[__(a)__].__(b)__
```

What goes in blank (a)?

What goes in blank (b)?

b) (3 pts) Fill in the blanks below so that the expression below is also equal to the total number of sunshine hours in October of all city-states in `sun`.

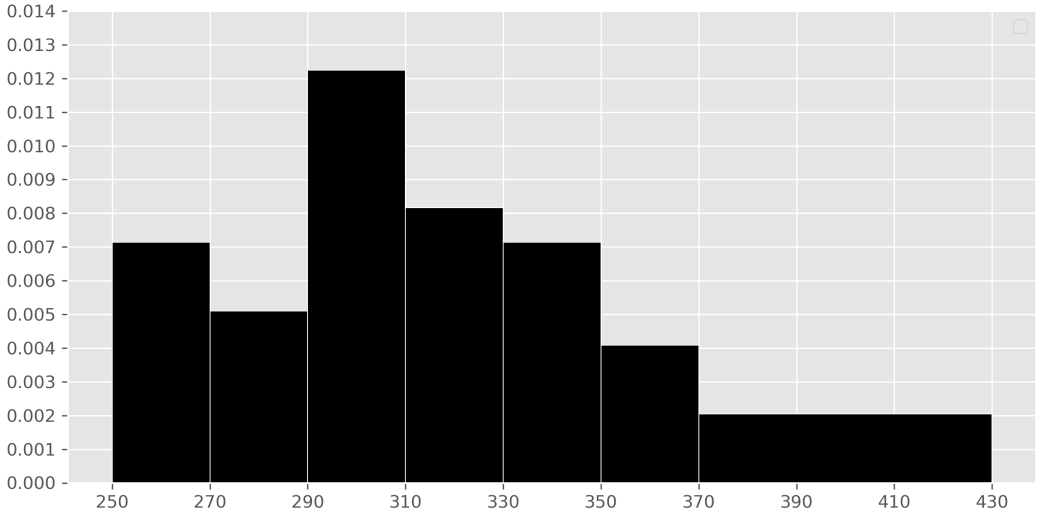
Note: What goes in blank (b) is the same as what goes in blank (b) above.

```
sun.get(["Country"]).merge(__(c)__).__(b)__
```

What goes in blank (c)?

Question 3 (13 pts)

This summer, Zoe wants to explore parts of the United States that she hasn't been to yet. In her process of figuring out where to go, she creates a histogram depicting the distribution of the number of sunshine hours in July across all cities in the United States in `sun`.



Suppose `usa` is a DataFrame with all of the columns in `sun` but with only the rows where `"Country"` is `"United States"`.

a) (3 pts) What is the value of `mystery` below?

```
cond = (usa.get("Jul") >= 370) & (usa.get("Jul") < 430)
mystery = 100 * np.count_nonzero(cond) / usa.shape[0]
```

◯ 2 ◯ 8 ◯ 12 ◯ 16 ◯ 18 ◯ 20

b) (4 pts) There are 3 more cities with between 370 and 430 sunshine hours in July than there are cities with between 270 and 290 sunshine hours in July.

How many cities in the United States are in `sun`? Give your answer as a positive integer, rounded to the nearest multiple of 10 (that is, your answer should end in a 0).

Now, suppose we convert the number of sunshine hours in July for all cities in the United States (i.e., "US cities") in `sun` from their original units (hours) to standard units.

c) (1.5 pts) Let m be the mean number of sunshine hours in July for all US cities in `sun`, in standard units. Select the true statement below.

◯ $m = -1$ ◯ $-1 < m < 0$ ◯ $m = 0$
◯ $0 < m < 1$ ◯ $m = 1$ ◯ $m > 1$

d) (1.5 pts) Let s be the standard deviation of the number of sunshine hours in July for all US cities `sun`, in standard units. Select the true statement below.

◯ $s = -1$ ◯ $-1 < s < 0$ ◯ $s = 0$
◯ $0 < s < 1$ ◯ $s = 1$ ◯ $s > 1$

e) (1.5 pts) Let d be the median of the number of sunshine hours in July for all US cities in `sun`, in standard units. Select the true statement below.

◯ $d = -1$ ◯ $-1 < d < 0$ ◯ $d = 0$
◯ $0 < d < 1$ ◯ $d = 1$ ◯ $d > 1$

f) (1.5 pts) True or False: The distribution of the number of sunshine hours in July for all US cities in `sun`, in standard units, is roughly normal.

◯ True ◯ False ◯ Impossible to tell

Question 4 (8 pts)

For each city in `sun`, we have 12 numbers, corresponding to the number of sunshine hours it sees in January, February, March, and so on, through December. (There is also technically a 13th number, the value in the `"Year"` column, but we will ignore it for the purposes of this question.)

We say that a city's number of sunshine hours **peaks gradually** if both of the following conditions are true:

- Each month from February to June has a number of sunshine hours greater than or equal to the month before it.
- Each month from August to December has a number of sunshine hours less than or equal to the month before it.

For example, the number of sunshine hours per month in Doris' hometown of Guangzhou, China peaks gradually:

$$62, 65, 71, 104, 118, 202, 181, 173, 172, 170, 166, 140$$

However, the number of sunshine hours per month in Charlie's hometown of Redwood City, California does not peak gradually, since $325 > 311$ and $247 < 271$:

$$185, 207, 269, 309, 325, 311, 313, 287, 247, 271, 173, 160$$

Complete the implementation of the function `peaks_gradually`, which takes in an array `hours` of length 12 corresponding to the number of sunshine hours per month in a city, and returns `True` if the city's number of sunshine hours peaks gradually and `False` otherwise.

```
def peaks_gradually(hours):
    for i in np.arange(5):
        cur_left = hours[5 - i]
        next_left = hours[__(a)__]
        cur_right = hours[__(b)__]
        next_right = hours[6 + i + 1]

        if __(c)__:
            __(d)__

    __(e)__
```

PID: _____

a) (2 pts) What goes in blank (a)?

b) (2 pts) What goes in blank (b)?

c) (2 pts) What goes in blank (c)?

◯ `next_left < cur_left or next_right < cur_right`
◯ `next_left < cur_left and next_right < cur_right`
◯ `next_left > cur_left or next_right > cur_right`
◯ `next_left > cur_left and next_right > cur_right`

d) (1 pt) What goes in blank (d)?

◯ `return True` ◯ `return False`

e) (1 pt) What goes in blank (e)?

◯ `return True` ◯ `return False`

Question 5 (11 pts)

In some cities, the number of sunshine hours per month is relatively consistent throughout the year. São Paulo, Brazil is one such city; in all months of the year, the number of sunshine hours per month is somewhere between 139 and 173. New York City's, on the other hand, ranges from 139 to 268.

Gina and Abel, both San Diego natives, are interested in assessing how "consistent" the number of sunshine hours per month in San Diego appear to be. Specifically, they'd like to test the following hypotheses:

- **Null Hypothesis**: The number of sunshine hours per month in San Diego is drawn from the uniform distribution, $\left[\frac{1}{12}, \frac{1}{12}, ..., \frac{1}{12}\right]$. (In other words, the number of sunshine hours per month in San Diego is equal in all 12 months of the year.)
- **Alternative Hypothesis**: The number of sunshine hours per month in San Diego is not drawn from the uniform distribution.

As their test statistic, Gina and Abel choose the total variation distance. To simulate samples under the null, they will sample from a categorical distribution with 12 categories — January, February, and so on, through December — each of which have an equal probability of being chosen.

a) (2 pts) In order to run their hypothesis test, Gina and Abel need a way to calculate their test statistic. Below is an incomplete implementation of a function that computes the TVD between two arrays of length 12, each of which represent a categorical distribution.

```
def calculate_tvd(dist1, dist2):
    return np.mean(np.abs(dist1 - dist2)) * ____
```

Fill in the blank so that `calculate_tvd` works as intended.

◯ `1 / 6` ◯ `1 / 3` ◯ `1 / 2` ◯ `2` ◯ `3` ◯ `6`

Moving forward, assume that `calculate_tvd` works correctly.

Now, complete the implementation of the function `uniform_test`, which takes in an array `observed_counts` of length 12 containing the number of sunshine hours each month in a city and returns the p-value for the hypothesis test stated at the start of the question.

```
def uniform_test(observed_counts):
    # The values in observed_counts are counts, not proportions!
    total_count = observed_counts.sum()
    uniform_dist = __(b)__
    tvds = np.array([])
    for i in np.arange(10000):
        simulated = __(c)__
        tvd = calculate_tvd(simulated, __(d)__)
        tvds = np.append(tvds, tvd)
    return np.mean(tvds __(e)__ calculate_tvd(uniform_dist, __(f)__))
```

b) (2 pts) What goes in blank (b)? *(Hint: The function* ***`np.ones(k)`*** *returns an array of length* ***`k`*** *in which all elements are* ***`1`****.)*

c) (2 pts) What goes in blank (c)?

◯ `np.random.multinomial(12, uniform_dist)`
◯ `np.random.multinomial(12, uniform_dist) / 12`
◯ `np.random.multinomial(12, uniform_dist) / total_count`
◯ `np.random.multinomial(total_count, uniform_dist)`
◯ `np.random.multinomial(total_count, uniform_dist) / 12`
◯ `np.random.multinomial(total_count, uniform_dist) / total_count`

d) (2 pts) What goes in blank (d)?

e) (1 pt) What goes in blank (e)?

◯ `>` ◯ `>=` ◯ `<` ◯ `<=` ◯ `==` ◯ `!=`

f) (2 pts) What goes in blank (f)?

Question 6 (17 pts)

Oren's favorite bakery in San Diego is Wayfarer. After visiting frequently, he decides to learn how to make croissants and baguettes himself, and to do so, he books a trip to France.

Oren is interested in estimating the mean number of sunshine hours in July across all 10,000+ cities in France. Using the 16 French cities in `sun`, Oren constructs a 95% Central Limit Theorem (CLT)-based confidence interval for the mean sunshine hours of all cities in France. The interval is of the form $[L, R]$, where L and R are positive numbers.

a) (3 pts) Which of the following expressions is equal to the standard deviation of the number of sunshine hours of the 16 French cities in `sun`?

◯ $R - L$ ◯ $\frac{R-L}{2}$ ◯ $\frac{R-L}{4}$ ◯ $R + L$ ◯ $\frac{R+L}{2}$ ◯ $\frac{R+L}{4}$

b) (1.5 pts) True or False: There is a 95% chance that the interval $[L, R]$ contains the mean number of sunshine hours in July of all 16 French cities in `sun`.

◯ True ◯ False

c) (1.5 pts) True or False: If we collected 1,000 new samples of 16 French cities and computed the mean of each sample, then about 95% of the new sample means would be contained in $[L, R]$.

◯ True ◯ False

d) (1.5 pts) True or False: If we collected 1,000 new samples of 16 French cities and created a 95% confidence interval using each one, then chose one of the 1,000 new intervals at random, the chance that the randomly chosen interval contains the mean sunshine hours in July across all cities in France is approximately 95%.

◯ True ◯ False

e) (1.5 pts) True or False: The interval $[L, R]$ is centered at the mean number of sunshine hours in July across all cities in France.

◯ True ◯ False

In addition to creating a 95% CLT-based confidence interval for the mean sunshine hours of all cities in France, Oren would like to create a 76% bootstrap-based confidence interval for the mean sunshine hours of all cities in France.

Oren resamples from the 16 French sunshine hours in `sun` 10,000 times and creates an array named `french_sunshine` containing 10,000 resampled means. He wants to find the left and right endpoints of his 76% confidence interval:

```
boot_left = np.percentile(french_sunshine, __(a)__)
boot_right = np.percentile(french_sunshine, __(b)__)
```

f) (4 pts) Fill in the blanks so that `boot_left` and `boot_right` evaluate to the left and right endpoints of a 76% confidence interval for the mean sunshine hours in July across all cities in France.

What goes in blank (a)?

What goes in blank (b)?

Suppose we are interested in testing the following pair of hypotheses.

- **Null Hypothesis**: The mean number of sunshine hours of all cities in France in July is equal to 225.
- **Alternative Hypothesis**: The mean number of sunshine hours of all cities in France in July is not equal to 225.

g) (1.5 pts) Suppose that when Oren uses `[boot_left, boot_right]`, his 76% bootstrap-based confidence interval, he fails to reject the null hypothesis above. If that's the case, then when using $[L, R]$, his 95% CLT-based confidence interval, what is the conclusion of his hypothesis test?

◯ Reject the null ◯ Fail to reject the null ◯ Impossible to tell

h) (1.5 pts) Suppose that Oren also creates a 76% CLT-based confidence interval for the mean sunshine hours of all cities in France in July using the same 16 French cities in `sun` he started with. When using his 76% CLT-based confidence interval, he fails to reject the null hypothesis above. If that's the case, then when using $[L, R]$, what is the conclusion of his hypothesis test?

◯ Reject the null ◯ Fail to reject the null ◯ Impossible to tell

i) (1 pt) True or False: The significance levels of both hypothesis tests described in part (h) are equal.

◯ True ◯ False

Question 7 (8 pts)

Gabriel is originally from Texas and is trying to convince his friends that Texas has better weather than California. Sophia, who is originally from San Diego, is determined to prove Gabriel wrong.

Coincidentally, both are born in February, so they decide to look at the mean number of sunshine hours of all cities in California and Texas in February. They find that the mean number of sunshine hours for California cities in February is 265, while the mean number of sunshine hours for Texas cities in February is 250. They decide to test the following hypotheses:

- **Null Hypothesis**: The distribution of sunshine hours in February for cities in California and Texas are drawn from the same population distribution.
- **Alternative Hypothesis**: The distribution of sunshine hours in February for cities in California and Texas are not drawn from the same population distribution; rather, California cities see more sunshine hours in February on average than Texas cities.

The test statistic they decide to use is:

mean sunshine hours in California cities – mean sunshine hours in Texas cities

To simulate data under the null, Sophia proposes the following plan:

1. Count the number of Texas cities, and call that number `t`. Count the total number of cities in both California and Texas, and call that number `n`.
2. Find the total number of sunshine hours across all California and Texas cities in February, and call that number `total`.
3. Take a random sample of `t` sunshine hours from the entire sequence of California and Texas sunshine hours in February in the dataset. Call this random sample `t_samp`.
4. Find the difference between the mean of the values that are not in `t_samp` (the California sample) and the mean of the values that are in `t_samp` (the Texas sample).

a) (1.5 pts) What type of test is this?

◯ Hypothesis test ◯ Permutation test

Complete the implementation of the function `one_stat`, which takes in a DataFrame `df` that has two columns — `"State"`, which is either `"California"` or `"Texas"`, and `"Feb"`, which contains the number of sunshine hours in February for each city — and returns a single simulated test statistic using Sophia's plan.

```
def one_stat(df):
    # You don't need to fill in the ...,
    # assume we've correctly filled them in so that
    # texas_only has only the "Texas" rows from df.
    texas_only = ...
    t = texas_only.shape[0]
    n = df.shape[0]

    total = df.get("Feb").sum()

    t_samp = np.random.choice(df.get("Feb"), t, __(b)__)

    c_mean = __(c)__
    t_mean = t_samp.sum() / t
    return c_mean - t_mean
```

b) (1.5 pts) What goes in blank (b)?

◯ `replace=True` ◯ `replace=False`

c) (3 pts) What goes in blank (c)? *(Hint: Our solution uses 4 of the variables that are defined before* `c_mean`*.)*

Fill in the blanks below to accurately complete the provided statement.

"If Sophia and Gabriel want to test the null hypothesis that the mean number of sunshine hours in February in the two states is equal using a different tool, they could use bootstrapping to create a confidence interval for the true value of the test statistic they used in the above test and check whether __(d)__ is in the interval."

d) (2 pts) What goes in blank (d)? Your answer should be a specific number.

Question 8 (10 pts)

Australia is in the southern hemisphere, meaning that its summer season is from December through February, when we have our winter. As a result, January is typically one of the sunniest months in Australia!

Arjun is a big fan of the movie Kangaroo Jack and plans on visiting Australia this January. In doing his research on where to go, he found the number of sunshine hours in January for the 15 Australian cities in `sun` and sorted them in **descending** order.

$$356, 337, 325, 306, 294, 285, 285, 266, 263, 257, 255, 254, 220, 210, 176$$

Throughout this question, use the mathematical definition of percentiles presented in class.

a) (2 pts) What is the 80th percentile of the collection of numbers above?

◯ 254 ◯ 255 ◯ 294 ◯ 306 ◯ 325 ◯ 337

b) (3 pts) What is the **largest** positive integer p such that 257 is the pth percentile of the collection of numbers above?

c) (3 pts) What is the **smallest** positive integer p such that 257 is the pth percentile of the collection of numbers above? (Make sure your answer to (c) is smaller than your answer to (b)!)

Teresa also wants to go to Australia, but can't take time off work in January, and so she plans a trip to The Land Down Under (Australia) in February instead. She finds that the mean number of sunshine hours in February for all 15 Australian cities in `sun` is 250, with a standard deviation of 15.

d) (2 pts) According to Chebyshev's inequality, at least what percentage of Australian cities in `sun` see between 200 and 300 sunshine hours in February?

◯ 9% ◯ 30% ◯ 33.3% ◯ 91% ◯ 95% ◯ 99.73%

PID: _____

Question 9 (12 pts)

Suhani's passport is currently being renewed, and so she can't join those on international summer vacations. However, her last final exam is today, and so she decides to road trip through California this week while everyone else takes their finals.

The chances that it is sunny this Monday and Tuesday, in various cities in California, are given below. The event that it is sunny on Tuesday in Los Angeles depends on the event that it is sunny on Monday in Los Angeles, but other than that, all other events in the table are independent of one another.

City	Prob. Sunny Monday	Prob. Sunny Tuesday
San Diego	$\frac{7}{10}$	$\frac{2}{5}$
Los Angeles	$\frac{3}{5}$	$\frac{3}{4}$ if it's sunny on Monday, $\frac{3}{8}$ if it's not sunny on Monday
San Francisco	$\frac{3}{4}$	$\frac{1}{10}$

a) (2.5 pts) What is the probability that it is not sunny in San Diego on Monday and not sunny in San Diego on Tuesday? Give your answer as a **positive integer percentage** between 0% and 100%.

b) (2.5 pts) What is the probability that it is sunny in at least one of the three cities on Monday?

◯ 3% ◯ 31.5% ◯ 40% ◯ 68.5% ◯ 75% ◯ 97%

c) (3 pts) What is the probability that it is sunny in Los Angeles on Tuesday?

◯ 15% ◯ 22.5% ◯ 40% ◯ 45% ◯ 60% ◯ 88.8%

d) (4 pts) Fill in the blanks so that `exactly_two` evaluates to the probability that exactly two of San Diego, Los Angeles, and San Francisco are sunny on Monday.

Hint: If ***`arr`*** *is an array, then* ***`np.prod(arr)`*** *is the product of the elements in* ***`arr`****.*

```
monday = np.array([7 / 10, 3 / 5, 3 / 4]) # Taken from the table.
exactly_two = __(a)__
for i in np.arange(3):
    exactly_two = exactly_two + np.prod(monday) * __(b)__
```

What goes in blank (a)?

What goes in blank (b)?

◯ `monday[i]`

◯ `1 - monday[i]`

◯ `1 / monday[i]`

◯ `monday[i] / (1 - monday[i])`

◯ `(1 - monday[i]) / monday[i]`

◯ `1 / (1 - monday[i])`

PID: _____

Question 10 (6 pts)

Costin, a San Francisco native, will be back in San Francisco over the summer, and is curious as to whether it is true that about $\frac{3}{4}$ of days in San Francisco are sunny.

Fast forward to the end of September: Costin counted that of the 30 days in September, 27 were sunny in San Francisco. To test his theory, Costin came up with two pairs of hypotheses.

Pair 1:

- **Null Hypothesis**: The probability that it is sunny on any given day in September in San Francisco is $\frac{3}{4}$, independent of all other days.
- **Alternative Hypothesis**: The probability that it is sunny on any given day in September in San Francisco is **not** $\frac{3}{4}$.

Pair 2:

- **Null Hypothesis**: The probability that it is sunny on any given day in September in San Francisco is $\frac{3}{4}$, independent of all other days.
- **Alternative Hypothesis**: The probability that it is sunny on any given day in September in San Francisco is **greater than** $\frac{3}{4}$.

For each test statistic below, choose whether the test statistic could be used to test Pair 1, Pair 2, both, or neither. Assume that all days are either sunny or cloudy, and that we cannot perform two-tailed hypothesis tests. (If you don't know what those are, you don't need to!)

a) (1.5 pts) The difference between the number of sunny days and number of cloudy days

◯ Pair 1 ◯ Pair 2 ◯ Both ◯ Neither

b) (1.5 pts) The absolute difference between the number of sunny days and number of cloudy days

◯ Pair 1 ◯ Pair 2 ◯ Both ◯ Neither

c) (1.5 pts) The difference between the proportion of sunny days and $\frac{1}{4}$

◯ Pair 1 ◯ Pair 2 ◯ Both ◯ Neither

d) (1.5 pts) The absolute difference between the proportion of cloudy days and $\frac{1}{4}$

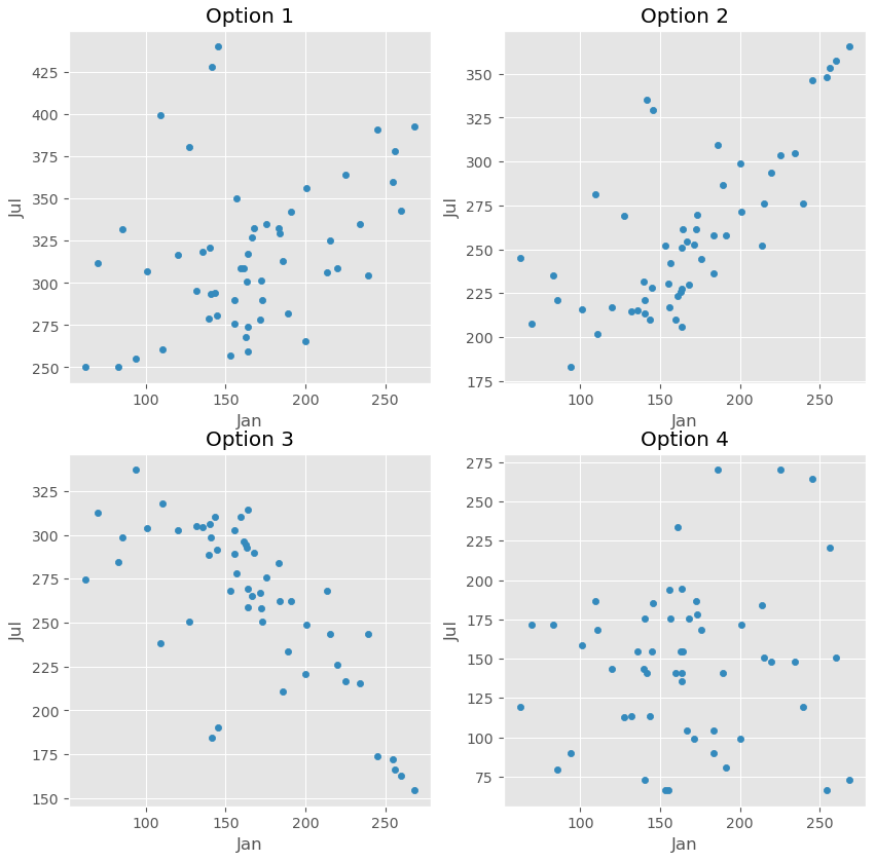
◯ Pair 1 ◯ Pair 2 ◯ Both ◯ Neither

Question 11 (11 pts)

Raine is helping settle a debate between two friends on the "superior" season — winter or summer. In doing so, they try to understand the relationship between the number of sunshine hours per month in January and the number of sunshine hours per month in July across all cities in California in `sun`.

Raine finds the regression line that predicts the number of sunshine hours in July (y) for a city given its number of sunshine hours in January (x). In doing so, they find that the correlation between the two variables is $\frac{2}{5}$.

a) (1.5 pts) Which of these could be a scatter plot of number of sunshine hours in July vs. number of sunshine hours in January?



◯ Option 1 ◯ Option 2 ◯ Option 3 ◯ Option 4

b) (3 pts) Suppose the standard deviation of the number of sunshine hours in January for cities in California is equal to the standard deviation of the number of sunshine hours in July for cities in California.

Raine's hometown of Santa Clarita saw 40 more sunshine hours in January than the average California city did. How many **more sunshine hours than average** does the regression line predict that Santa Clarita will have in July? Give your answer as a positive integer. *(Hint: You'll need to use the fact that the correlation between the two variables is $\frac{2}{5}$.)*

As we know, San Diego was particularly cloudy this May. More generally, Anthony, another California native, feels that California is getting cloudier and cloudier overall.

To imagine what the dataset may look like in a few years, Anthony subtracts 5 from the number of sunshine hours in both January and July for all California cities in the dataset — i.e., he subtracts 5 from each x value and 5 from each y value in the dataset. He then creates a regression line to use the new xs to predict the new ys.

c) (2 pts) What is the slope of Anthony's new regression line?

d) (3 pts) Suppose the intercept of Raine's original regression line — that is, before Anthony subtracted 5 from each x and each y — was 10. What is the intercept of Anthony's new regression line?

◯ -7 ◯ -5 ◯ -3 ◯ 0 ◯ 3 ◯ 5 ◯ 7

Jasmine is trying to get as far away from Anthony as possible and has a trip to Chicago planned after finals. Chicago is known for being very warm and sunny in the summer but cold, rainy, and snowy in the winter. She decides to build a regression line that uses month of the year (where 1 is January, 2 is February, 12 is December, etc.) to predict the number of sunshine hours in Chicago.

e) (1.5 pts) What would you expect to see in a residual plot of Jasmine's regression line?

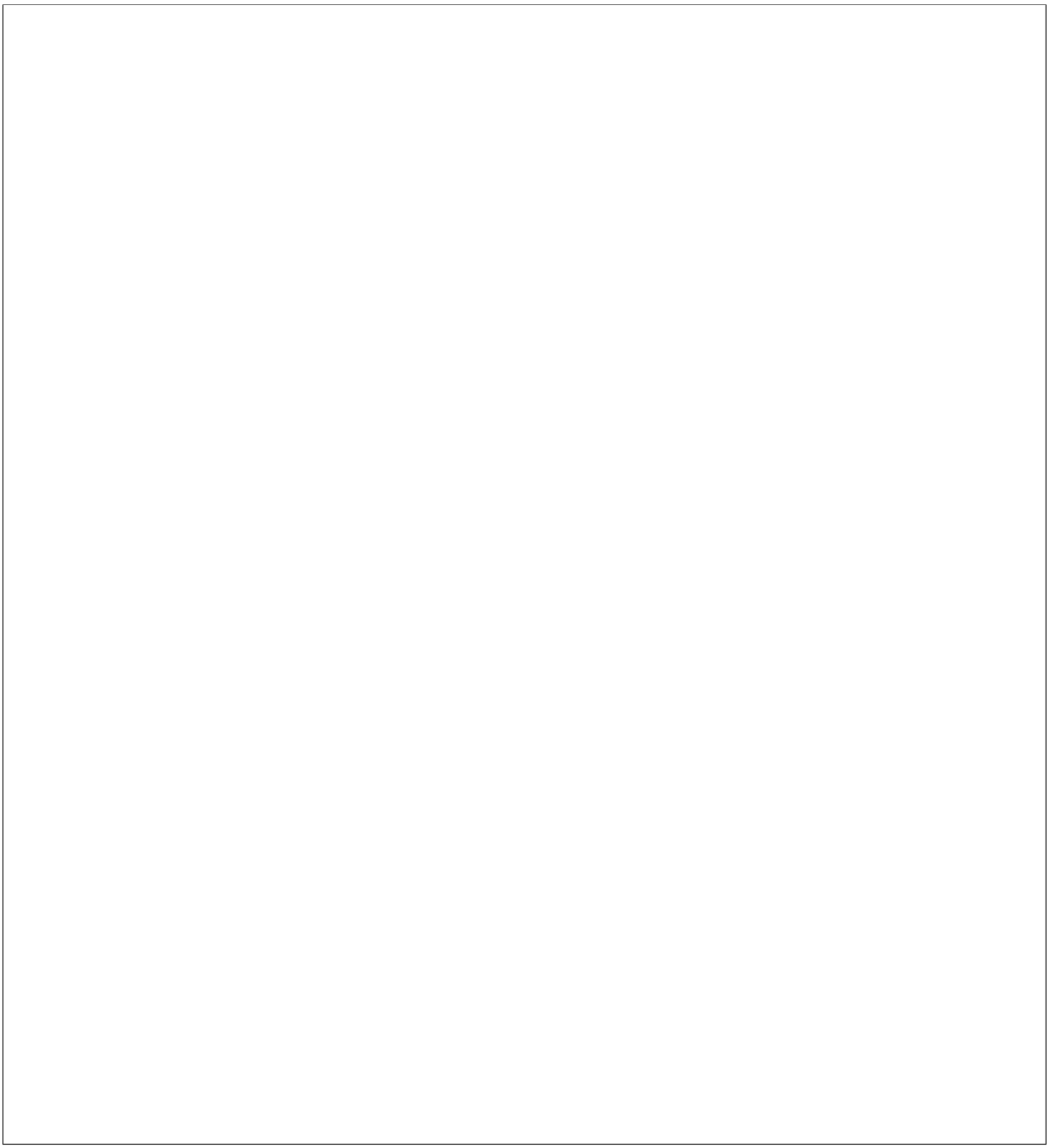
◯ A patternless cloud of points

◯ A distinctive pattern in the residual plot

◯ Heteroscedasticity (residuals that are not evenly vertically spread)

Feel free to draw us a picture or tell us about your fondest memory from DSC 10 this quarter.

(Fun fact: "Suraj" in Hindi means "sun", though that's not how Suraj chose the theme of this exam.)



Before turning in your exam, please make sure that your PID is on every page.

Congratulations on finishing DSC 10!