Data 8 Foundations of Data Science
Fall 2019

MIDTERM SOLUTIONS

INSTRUCTIONS

- The exam is worth 100 points. You have 100 minutes to complete it.
- The exam is closed book, closed notes, closed computer/phone/tablet, closed calculator, except the official midterm exam reference guide provided with the exam.
- Write/mark your answers on the exam in the space/bubbles provided. Answers written anywhere else will not be graded. Unless the question specifically asks you to explain your answer, you do not need to do so, and if you write an explanation it will not be graded.
- If you need scratch paper, you are welcome to use the reference sheet and the back of this cover page. Scratch work will not be graded.
- For all Python code, you may assume that the statements `from datascience import *` and `import numpy as np` have been executed. Do not use features of the Python language that have not been described in this course.
- In any part, you are free to use any tables, arrays, or functions that have been defined in previous parts of the same question, and you may assume they have been defined correctly.

Last name	
First name	
Student ID number	
Calcentral email (`_@berkeley.edu`)	
Lab GSI	
← Name of the person to your left	
Name of the person to your right →	
All the work on this exam is my own. **(please sign)**	
Your room & seat number (for example, Dwinelle A1)	

0. (0 points) **W**rite your name in the space provided on one side of every page of the exam, and don't forget to fill in all your information on the first page (including your room and seat number).

1. (8 points) Python

For each of the Python expressions below, write the output when the expression is evaluated. If an error occurs, write "Error". **If you show any work or intermediate steps, make sure you circle your final answer.**

Example Expression:

```
make_array(1,2,3,4,5) == 3
```

Example Answer:

```
array([False, False, True, False, False])
```

(a) (2 pt)

```
make_array(155, 1, 2050) + make_array('Dwinelle', 'Pimentel', 'VLSB')
```

Error

(b) (2 pt)

```
np.max(np.arange(7))
```

6

(c) (2 pt)

```
def my_function(x):
    return x.item(1) - x.item(0)
my_function(make_array(10, 1, 3))
```

-9

(d) (2 pt)

```
my_table = Table().with_columns(
    'nums', make_array(4, -10, 1),
    'labels', make_array('cat', 'mouse', 'dog'))
my_table.sort('nums', descending=True).column('labels')
```

```
array(['cat', 'dog', 'mouse'])
```

2. (12 points) Visualizing the news

Celina decides to collect data about different articles on the front page of her favorite news sites. She collects data every day for one year, and stores the data in a table called `news`. Here are the first few rows:

day	site	headline font size	article length	topic
1	Washington Post	18	721	politics
1	Wall Street Journal	12	497	business
2	Daily Cal	20	230	science and technology

The table contains five columns:

- **day**: an int, the day of the year she recorded the data (from 1 to 365).
- **site**: a string, recording which site she found the article on
- **headline font size**: an int, the font size of the article headline on the front page
- **article length**: an int, the number of words in the article
- **topic**: a string, what topic Celina decided the article was about

Celina wants to answer the following questions. For parts (a) - (d), choose which kind of visualization would be the best choice to answer it. **Choose only one answer for each question.**

(a) (2 pt) Is there an association between the font size of the headline and the length of the article?

○ Line graph ○ Histogram ○ Bar chart ● Scatter plot

(b) (2 pt) How does the average length of articles written on each day change over the year?

● Line graph ○ Histogram ○ Bar chart ○ Scatter plot

(c) (2 pt) Are there more long articles than short articles? How long do articles tend to be?

○ Line graph ● Histogram ○ Bar chart ○ Scatter plot

(d) (2 pt) How do the distributions of article lengths vary between the Washington Post and the Daily Cal?

○ Line graph ● Histogram ○ Bar chart ○ Scatter plot

(e) (4 pt) Celina decides to compare how many articles different sites write about different topics. She wants a table where the first few rows look like this:

site	business	entertainment	health	politics	science and technology	sports
ABC	607	1628	926	184	1623	2597
Atlantic	231	46	834	469	280	794
Daily Beast	1348	184	693	1523	139	885
Daily Cal	1113	1208	1586	653	514	1198
FiveThirtyEight	1717	1163	1344	2085	231	138

Each number should be a count of how many articles Celina collected from that particular site on that particular topic. For example, she collected 607 articles from ABC about business, 1628 articles from ABC on entertainment, and so on.

Write a line of code that produces this table.

```
news.pivot('topic','site')
```

3. (26 points) Choosing shirts

Chidi has 4 white shirts, 9 blue shirts, and 2 gold shirts. Each day, he chooses one shirt to wear from all the shirts available in his closet. Each shirt is equally likely to be chosen.

In each part below write a mathematical expression (not Python) that evaluates to the probability described. **You do not need to simplify any arithmetic. Please do not multiply by 100 to get percents.**

For parts (a) and (b), assume that Chidi will only wear each individual shirt once in any particular week.

(a) (3 pt) What is the probability that Chidi wears blue shirts every day for the first three days of the week?

$\frac{9}{15} \times \frac{8}{14} \times \frac{7}{13}$ (multiplication rule, no replacement)

(b) (3 pt) What is the probability that out of the first two days of the week, he wears blue one day and gold on the other?

$\left(\frac{9}{15} \times \frac{2}{14}\right) + \left(\frac{2}{15} \times \frac{9}{14}\right)$ (addition rule for blue/gold and gold/blue, then multiplication rule, no replacement)

For the remainder of the question, assume that after wearing a shirt, Chidi puts it back in the closet without washing it. This means that he can wear the same shirt multiple times.

(c) (3 pt) What is the probability that Chidi wears gold shirts every day for a week?

$\left(\frac{2}{15}\right)^7$ (multiplication rule, with replacement)

(d) (3 pt) What is the probability that Chidi wears at least one white shirt during one week?

$1 - \left(\frac{11}{15}\right)^7$ (complement rule and multiplication rule, with replacement)

Chidi explains how he picks his shirts to his friend Janet. Janet says, "I wrote down your shirt color every day for the last 30 days, and you wore gold shirts on 15 of them. I think the gold shirts are your favorite, and you're choosing them more than the others!" They decide to use hypothesis testing to test Janet's claim.

(e) **(2 pt)** Fill in the blanks in the null hypothesis below. If your answer for any blank contains arithmetic, you do not need to simplify it.

Null hypothesis: Chidi chose one shirt at random every day for 30 days,

and each day, the probability he chose gold shirts is 2/15.

(f) **(2 pt)** Fill in the blank in the alternative hypothesis below.
Hint: your answer shouldn't include any arithmetic.

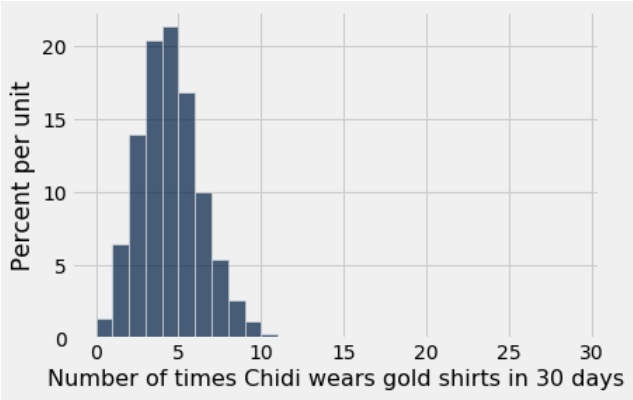
Alternative hypothesis: The probability that Chidi chose a gold shirt each day is

more than 2/15.

(g) **(6 pt)** As a test statistic, they decide to use the number of times Chidi wears gold shirts in 30 days. Fill in the blanks in the code to simulate this process 10,000 times. At the end of the simulation, the array `statistics` should contain 10,000 simulated values of the test statistic under the null hypothesis. Assume that the first line is finished correctly, and the array `shirts` contains one string for each shirt in Chidi's closet.

```
shirts = make_array('gold', 'gold', 'blue', ...)

def compute_test_statistic():

    worn_shirts = np.random.choice(shirts, 30)

    return np.count_nonzero(worn_shirts == 'gold')



statistics = make_array()

for i in np.arange(10000):

    stat = compute_test_statistic()

    statistics = np.append(statistics, stat)
```

Chidi and Janet correctly complete the simulation, and obtain the following histogram of simulated statistics. They use `np.arange(32)` as the bins for the histogram.



(h) **(2 pt)** Based on the histogram above and Janet's observation, which of the following should they conclude? **Choose only one answer.**

- ○ The data are consistent with the null hypothesis.
- ● The data are not consistent with the null hypothesis.

(i) **(2 pt)** Which of the following is **closest** to the p-value for Janet's data? **Choose only one answer.**

- ● 0
- ○ 0.05
- ○ 0.1
- ○ 0.5
- ○ 0.9
- ○ 1
- ○ 15

4. (27 points) Table operations

A table `trees` contains one row for each tree in a small forest. Here are the first few rows:

year	wood density	burned	type	height
1980	0.813	no	oak	23
1829	0.529	yes	redwood	127.47
2000	0.601	no	pine	5.2

The table contains five columns:

- **year**: an int, the year the tree was planted
- **wood density**: a float, the density of the tree's wood
- **burned**: a string, which indicates whether the tree was ever burned in a fire (`yes` or `no`)
- **type**: a string, what kind of tree it is
- **height**: a float, the height of the tree in meters

In each part below, fill in the blanks of the Python expression. **You must use ONLY the lines provided.** Some of the chained operations we might normally do in one line have been broken up into two or more lines, storing intermediate results in temporary tables. You may find the names of the temporary tables to be useful hints. Do not write any code outside the blanks provided. The expression in the last line should evaluate to the value described.

Parts (e) and (f) ask for the same table, but each one uses a different approach.

(a) (2 pt) The number of trees in the forest that were ever burned in a fire.

```
np.count_nonzero(trees.column('burned') == 'yes')
```

(b) (2 pt) The height of the tallest tree.

```
max(trees.column('height'))
```

(c) (3 pt) The type of the least dense tree.

```
sorted = trees.sort('wood density')
sorted.column('type').item(0)
```

(d) (3 pt) The table `trees`, but with only the rows for burnt trees, and without the **burned** column.

```
trees.where('burned', 'yes').drop('burned').
```

Also valid: `trees.where('burned', are.equal_to('yes')).drop('burned')`

(e) (4 pt) The table `trees` with an added column named `Growth speed` that contains the average growth speed of the tree in meters per year. For example, if a tree was planted in 1999 (twenty years ago) and is 22 meters tall, its growth speed is `22/(2019-1999) = 1.1` meters/year.

```
speeds = trees.column('height') / (2019 - trees.column('year'))


trees.with_column('Growth speed', speeds)
```

(f) (4 pt) Same table as (e), using a different approach: The table `trees` with an added column named `Growth speed` that contains the average growth speed of the tree in meters per year. For example, if a tree was planted in 1999 (twenty years ago) and is 22 meters tall, its growth speed is `22/(2019-1999) = 1.1` meters/year.

```
def compute_speed(height, year):

    return height / (2019 - year)

speeds = trees.apply(compute_speed, 'height', 'year')

trees.with_column('Growth speed', speeds)
```

We also accepted `speeds = trees.apply(compute_speed, ['height', 'year'])` in the 3rd line.

(g) (4 pt) A new table `type_heights`, which has one row per type. It should have two columns: one labeled `type` that has the name of the type, and one that has the average height of trees for that type. It should not have any other columns.

```
trees_2col = trees.select('type', 'height')

type_heights = trees_2col.group('type', np.average)

type_heights
```

(h) (5 pt) The table `trees` with an additional column `Pct of average height`. It should store each tree's percent of the average height for that type. For example, if the average height of oak trees in the forest is 20 meters, then the first row of the new column should contain the value `100 * 23/20 = 115`.

Hint: You should use the `type_heights` table from part (g). You should assume that it has been defined correctly, regardless of your answer to part (g).

```
with_h = trees.join('type', type_heights)

trees.with_column('Pct of average height',

    100 * with_h.column('height') / with_h.column('height average'))
```

5. (15 points) Tree visualizations

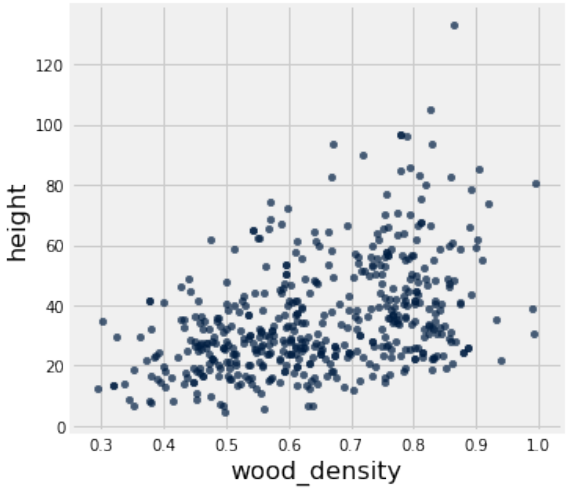
For this question, you'll be working with the `trees` table from the previous question. Here is the same table and description of its columns:

year	density	burned	type	height
1980	0.813	no	oak	23
1829	0.529	yes	redwood	127.47
2000	0.601	no	pine	5.2

The table contains five columns:

- **year**: an int, the year the tree was planted
- **density**: a float, the density of the tree's wood
- **burned**: a string, which indicates whether the tree was burned in a fire within the last few years
- **type**: a string, what kind of tree it is
- **height**: a float, the height of the tree in meters

(a) (3 pt) The following scatter plot was created using `trees.scatter('wood_density', 'height')`.



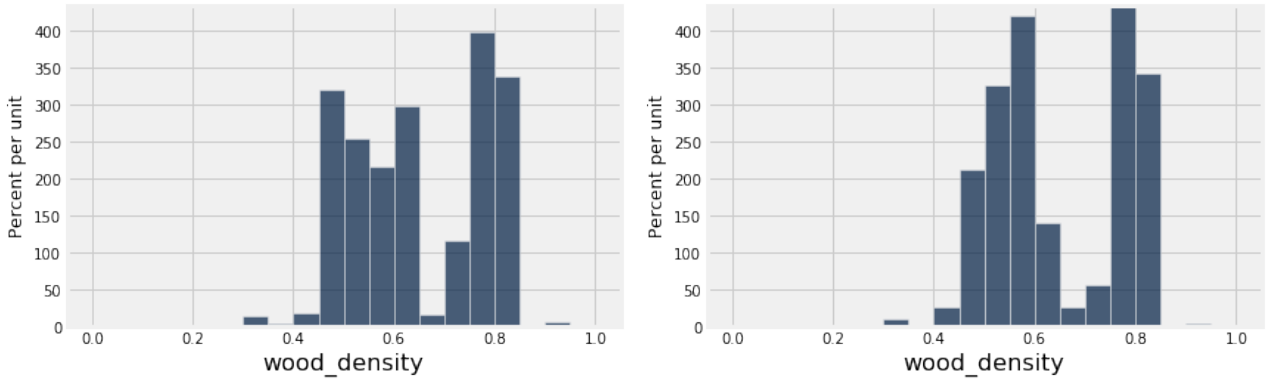
Which of the following are valid conclusions that can be drawn from this graph and the information above? **Choose all that apply.**

- ● There is a positive association between wood density and height in this forest.
- ○ There is a negative association between wood density and height in this forest.
- ○ There is no association between wood density and height in this forest.
- ● There are more short trees (less than 60 meters tall) than tall trees (more than 60 meters tall).
- ○ In this forest, higher wood density causes trees to grow taller.
- ○ In this forest, lower wood density causes trees to grow taller.
- ○ In this forest, trees growing taller causes them to have higher wood density.
- ○ In this forest, trees growing taller causes them to have lower wood density.
- ○ In this forest, the burnt trees are shorter than the unburnt trees.

(b) (2 pt) Suppose you want to visualize how common different types of trees (oak, redwood, etc.) are in this forest. Choose the line of code that generates the best visualization for this task. **Choose only one answer.**

- ○ `trees.hist('type')`
- ○ `trees.barh('type')`
- ○ `trees.plot('type')`
- ○ `trees.group('type').hist('type')`
- ● `trees.group('type').barh('type')`
- ○ `trees.group('type').plot('type')`

Consider the following two histograms of tree heights for burnt (left) and unburnt (right) trees. Assume that all data points are shown, and the area of the bars sums to 100%. For both histograms, the bins used are `np.arange(0, 1.05, .05)`.



For parts (c) - (g), mark whether the statement is a valid conclusion from the histograms above. If it is not, provide a short (one sentence or less) explanation of why. Your explanation should fit in the blank provided.

(c) (2 pt) More than half of the burnt trees have a wood density above 0.5.

- ● Valid conclusion
- ○ Not a valid conclusion:

(d) (2 pt) Getting burnt causes a tree's wood density to increase.

- ○ Valid conclusion
- ● Not a valid conclusion:

We don't know if the data came from a randomized controlled trial, so we can't determine causality.

(e) (2 pt) Among trees whose wood density is between 0.8 and 0.85 (not including 0.85), the number of burnt trees and unburnt trees is about the same.

- ○ Valid conclusion
- ● Not a valid conclusion:

The percent of burnt trees and percent of unburnt trees is the same, but we don't know how many of each there are.

(f) (2 pt) About 25% of the burnt trees have a density between 0.75 and 0.8 (not including 0.8).

- ○ Valid conclusion
- ● Not a valid conclusion:

The percentage is the width of the bar (0.05 density units) multiplied by its height (400 percent per density unit), or 0.2. We accepted True as well for this question since 20% could be "about" 25%.

(g) (2 pt) About 300 unburnt trees have a wood density between 0.6 and 0.65 (not including 0.65).

- ○ Valid conclusion
- ● Not a valid conclusion:

We don't know how many unburnt trees there are, and the histogram only shows the percent.

6. (11 points) Menu items

You find an article claiming that 15% of Bay Area restaurant menu items are vegan, 35% are vegetarian (but not vegan), and 55% contain meat (i.e., are neither vegetarian nor vegan). Your vegan friend Vera says, "I don't really care about vegetarian or meat items, but I think I've seen way more than 15% of menu items that are vegan here in the Bay Area! I think that number is too low."

Throughout this question, we'll use the word **vegetarian** to refer to menu items that are vegetarian but not vegan.

You decide to collect data on a random sample of 100 menu items from local restaurants, and use what you learned about hypothesis testing in Data 8 to check Vera's claim. You obtain the following data:

Menu item type	count
Vegan	33
Vegetarian	40
Meat	27

For this question, you'll be testing Vera's claim using your data.

(a) (3 pt) Which of the following could be a valid null hypothesis for testing Vera's claim? **Choose all that apply.**

- ○ Vegan menu items are more popular in the Bay Area than other parts of the U.S.
- ○ In a random sample of 100 menu items, there should be an equal number of vegan and non-vegan items. Any deviation from that is due to chance.
- ○ The article you read is incorrect.
- ● Menu items in your data are a random sample from a distribution with a 15% chance of being vegan, a 35% chance of being vegetarian, and a 55% chance of containing meat.
- ● Menu items in your data are a random sample from a distribution with a 15% chance of being vegan and a 85% chance of being non-vegan.
- ○ In a random sample of 100 menu items, we should see exactly 15 vegan, 35 vegetarian, and 55 containing meat.

(b) (3 pt) Which of the following could be a valid alternative hypothesis for testing Vera's claim? **Choose all that apply.**

- ○ Vegan menu items are more popular in the Bay Area than other parts of the U.S.
- ● The chance a random menu item is vegan is more than 0.15.
- ○ The chance a random menu item is vegan is less than 0.15.
- ○ The chance a random menu item is vegan is different from 0.15.
- ○ The distribution of menu items in the Bay Area is different from the one given in the article.
- ○ The distribution of menu items in the Bay Area is more than the one given in the article.
- ○ The data you collected support Vera's claim.
- ○ The data you collected do not support Vera's claim.

(c) (3 pt) What is a good choice of test statistic for this test? **Choose all that apply.**

- ○ In a sample of 100 menu items, the number of restaurants with vegan menu items
- ● In a sample of 100 menu items, the number of vegan items
- ● In a sample of 100 menu items, the number of vegan items minus 15
- ○ The absolute value of the previous option
- ○ In a sample of 100 menu items, the total variation distance between the empirical distribution of the sample and the distribution in the article

We tried to minimize cascading errors, so we graded 6c solely off of the selected hypotheses in 6b.
If option 2 and/or 3 were selected in 6b, option 2 and 3 must be selected in 6c to get credit for 6c.
If option 4 was selected in 6b, option 4 must be selected in 6c.
If option 5 was selected in 6b, option 5 must be selected in 6c.
If a combination of those options were selected in 6b, all of the corresponding options must have been selected for 6c. For example, if option 2 and 5 were selected in 6b, option 2, 3, and 5 had to be selected in 6c to get credit for 6c.

(d) (2 pt) Suppose you work with Vera to run a simulation and you obtain an array `test_statistics` with the result of $10,000$ simulations under the null hypothesis. Which of the following is true about the p-value of your experiment? **Choose all that apply.**

- ○ The p-value must be less than .05.
- ● A p-value close to 1 means that the data are consistent with the null hypothesis.
- ● A p-value close to 0.5 means that the data are consistent with the null hypothesis.
- ○ A p-value close to 0 means that the data are consistent with the null hypothesis.
- ○ The p-value is the probability that the article is correct.
- ○ The p-value is the probability that Vera is correct.
- ○ The p-value is the probability of finding a vegan menu item at a restaurant in the Bay Area.
- ● If the p-value is exactly .01, then 100 of the simulations produced test statistics greater than or equal to the one observed in the data.

7. (0 points) **W**rite your name in the space provided on one side of every page of the exam, and don't forget to fill in all your information on the first page (including your room and seat number).

You're done!