Data 8 Foundations of Data Science
Spring 2016

Midterm Solutions

INSTRUCTIONS

- You have 45 minutes to complete the exam.
- The exam is closed book, closed notes, closed computer, closed calculator, except one hand-written 8.5" $\times$ 11" crib sheet of your own creation and the official study guide provided with the exam.
- Mark your answers **on the exam itself**. We will *not* grade answers written on scratch paper.

Last name	
First name	
Student ID number	
BearFacts email (`_@berkeley.edu`)	
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)**	

This page was intentionally left blank.

1. (12 points) Expressions

(a) (10 pt) An array of integers named `ca` contains the (estimated) population of California every 10 years. It has 11 items. The first item is the population of California in 1900. The last is the population in 2000.

```
array([ 1485053,  2377549,  3426861, ..., 23667902, 29760021, 33871648])
```

Write a Python expression below each of the following descriptions that computes its value. The first one is provided as an example. Do not include numbers above (e.g., 1485053) in your solutions.

- The population in 1900.

  `ca.item(0)`

- The population change from 1940 to 1960, expressed as a number of persons (not a proportion).

  `ca.item(6) - ca.item(4)`

- Whether the population ever grew by less than 500000 in a decade represented by `ca`. (`True` or `False`)

  `np.any(np.diff(ca) < 500000)` or `min(np.diff(ca)) < 500000`

- The *annual* (yearly) growth rate from 1920 to 1930.

  `(ca.item(3)/ca.item(2)) ** (1/10) - 1`

- The population in 1924, assuming a fixed exponential annual growth rate from 1920 to 1930. *You may use the name `g` for the annual growth rate from 1920 to 1930 (computed above).*

  `ca.item(2) * (1+g)**4`

- The number of items in `ca` that are at least twice as large as the population in 1960.

  `np.count_nonzero(ca >= 2 * ca.item(6))`

(b) (2 pt) You have 1000 different shirts in your huge closet. Each morning, you pick one uniformly at random, wear it, then put it back at night. Write a Python expression to compute the chance that, during a 30-day month, there is at least one shirt that you wear two or more times. (Call the fashion police!)

`1-np.prod(1-np.arange(30)/1000)`

2. (10 points) Tables

(a) (2 pt) The table named `twins` has a row for each pair of twins that contains the height of each twin in inches. The following code computes the average absolute difference in heights among the twins.

```
def diff(height1, height2):
    return abs(height1 - height2)
diffs = Table(['Absolute Differences'])
for i in np.arange(twins.num_rows):
    diffs.append([diff(twins.column(0).item(i), twins.column(1).item(i))])
np.mean(diffs.column(0))
```

Complete the expression below to compute the same result without calling `diff` or using a `for` statement.

`np.mean(np.abs(twins.column(0) - twins.column(1)))`

(b) (8 pt) Each row of the `trip` table from lecture describes a single bicycle rental in the San Francisco area. Durations are integers representing times in seconds. The first three rows out of 338343 appear below.

Start	**End**	**Duration**
Ferry Building	SF Caltrain	765
San Antonio Shopping Center	Mountain View City Hall	1036
Post at Kearny	2nd at South Park	307

Write a Python expression below each of the following descriptions that computes its value. The first one is provided for you. You *may* use up to two lines and introduce variables.

- The average duration of a rental. `total_duration = sum(trip.column(2))`
  `total_duration / trip.num_rows`
- The number of rentals that started at the `SF Caltrain` station.

  `trip.where(0, 'SF Caltrain').num_rows`

- The average duration for rentals that started and ended at different stations.

  `np.average(trip.where(trip.column(0) != trip.column(1)).column(2))`

- The name of the station where the most rentals ended (assume no ties).

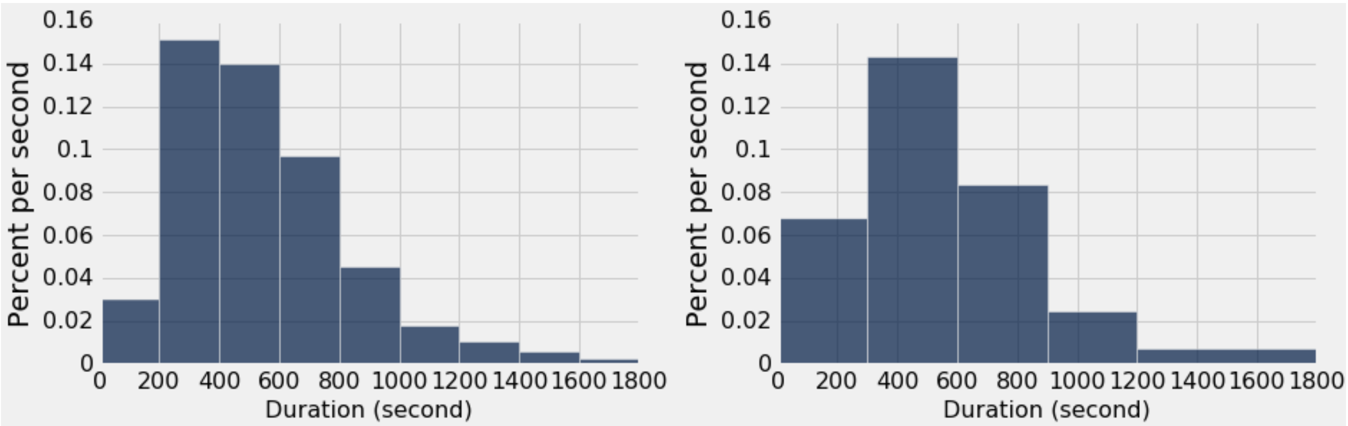
  `trip.group('End').sort('count', descending=True).column(0).item(0)`

- The number of stations for which the average duration ending at that station was at least 300 seconds.

  `np.count_nonzero(trip.select([1, 2]).group(0, np.mean).column(1) >= 300)`

3. (15 points) Distributions

The two histograms of bike trip durations below were both generated by `trip.hist(...)` using different bins.



(a) (8 pt) Write the proportion of trips that fall into each range of durations below. *Show your work.* If it is not possible to tell from the histograms, instead write **Not enough information**.

- Between 200 (inclusive) and 400 (exclusive) seconds

  `0.0015 * 200 == 0.30` or 30%

- Between 300 (inclusive) and 900 (exclusive) seconds

  `0.0014 * 300 + 0.0008 * 300 == 0.66` or 66%

- Between 400 (inclusive) and 900 (exclusive) seconds

  `0.0014 * 200 + 0.0008 * 300 == 0.52` or 52%

- Between 200 (inclusive) and 300 (exclusive) seconds

  `0.0015 * 200 + 0.0014 * 200 - 0.0014 * 300 == 0.16` or 16%
  Alternatively: `0.0007 * 300 - 0.0003 * 200 == 0.15` or 15%

(b) (4 pt) Frank Underwood conducts a poll to estimate what proportion of voters approve of his America Works program. He found that the voters he polled give the program an average grade of 6.5 out of 10, and the Standard Deviation in the grades is 1.5.

- No matter how the distribution of grades looks, what is the maximum proportion of those polled that could have given the program a grade lower than 2 according to Chebyshev's inequality?

  At most, 1/9 could be giving a grade lower than a 2, because at least 8/9 fall between 2 and 11.

- If the distribution is symmetrical and has a bell shape just like the normal distribution, about what proportion of those polled gave the program a grade lower than 2?

  About 0.14%, by the normal distribution, which is symmetric.

(c) (3 pt) A study followed 369 people with cardiovascular disease, randomly selected from hospital patients. A year later, those who owned a dog were four times more likely to be alive than those who didn't.

- Circle *True* or *False*: This study is a randomized controlled experiment. Answer: False; the experimenters did not control who owned a dog.

- Circle *True* or *False*: This study shows that dog owners live longer than cat owners on average. Answer: False; the experiment compares those who owned a dog to those who didn't (no mention of cats)

- Circle *True* or *False*: This study shows that for someone with cardiovascular disease, adopting a dog will probably cause them to live longer. Answer: False; an observational study does not show causation.

4. (8 points) Predictions

The `ball` table contains player data for some of the Golden State Warriors. Only the first five rows are shown.

Player	Minutes per Game	Points per Game
Klay Thompson	34	21
Andrew Bogut	20	5
Stephen Curry	34	29
James McAdoo	4	3
Andre Iguodala	28	7

You have computed the following summary statistics from the full `ball` table.

Expression	Value
`np.average(ball.column(1))`	24
`np.std(ball.column(1))`	10
`np.average(ball.column(2))`	13
`np.std(ball.column(2))`	8
`correlation(ball, 1, 2)`	0.75

(a) (6 pt) For each question below, answer with a number. You may show your work for partial credit.

- What is the value of Stephen Curry's points per game in standard units?

```
(29-13) / 8 == 2
```

- What is the slope of the regression line when the points per game are plotted on the vertical axis, the minutes per game are plotted on the horizontal axis, and a regression line is fit to the data? That's the slope of the regression line computed by `slope(ball, 1, 2)` in original units, $\frac{\text{points}}{\text{minute}}$.
  The `slope` function is defined on the last page of the midterm study guide.

```
0.75 * 8/10 == 0.6
```

- What is the fitted value for Stephen Curry using this regression line to estimate his points per game from his minutes per game?

```
(34-24)/10 * 0.75 * 8 + 13 == 19
```

(b) (2 pt) How would the fitted value of points per game for a player who played 34 minutes per game change if Stephen Curry were removed from the table and the regression line recomputed? *Circle one.*

(a) Increase (b) Decrease (c) Stay the same (d) Not enough information

Since the regression line minimizes the sum of squared errors, and Stephen Curry's squared error is positive at x = 34 minutes, removing this term from the objective function would result in a smaller fitted value.