STAT/CS 94 Fall 2015 Adhikari HW03, Due: 9/16/15

NAME: SID:

Please write your answers on the printed page, in the space provided. If you print this assignment yourself, please print it *double-sided* on a single sheet of paper. (There will be a small penalty for not following this instruction; it makes the grader's job more difficult.)

Problem 1 (Tables)

In this question we'll review basic operations on tables. Refer to Section 1.5 of the course text (`data8.org/text/1_data.html#tables`). As usual, please try to work out the answers on your own, then use a Jupyter notebook (on `ds8.berkeley.edu`) to check yourself.

Suppose we are analysts at a company that makes widgets at two factories (located in Berkeley and Palo Alto), and we're investigating the productivity of our factories. The factories have reported how many widgets they produced on several days. We have created a table named `production` from this data using the following Python code:

```
production = Table()
production['Factory'] = ['Berkeley', 'Palo Alto', 'Berkeley', 'Palo Alto', 'Berkeley', 'Palo Alto']
production['Date'] = [4, 5, 6, 4, 5, 6]
production['Widgets Produced'] = [112, 65, 130, 84, 91, 105]
```

(a) Using `sort`, give a Python expression whose value is the `production` table sorted in ascending order by number of widgets produced.

(b) What is the value of the following expressions? (For any array-valued expressions, it's okay to abbreviate by writing, for example, `[True, False]` instead of `array([True, False], dtype=bool`.)

(a) `len(production.rows)`

(b) `len(production.columns)`

(c) `len(production['Date'])`

(d) `len(production['Date'] >= 5)`

(e) `production['Date'] >= 5`

(f) `len(production['Date'] >= 7)`

(g) `len(production.where(np.array([True, False, False, False, True, False])).rows)`

(h) `len(production.where(production['Date'] >= 5).rows)`

(i) `len(production.where(production['Factory'] == 'Berkeley').rows)`

(c) Write a short description, in English, of the value of the following Python expression:

```
production.select(['Factory','Widgets Produced']).group('Factory', collect=np.sum)
```

(d) Give a Python expression whose value is the total number of widgets produced on date 5. (There are at least 2 reasonable ways to do this using the tools we've seen so far.)

(e) Now we would like to assign bonuses to the factory bosses according to the number of widgets produced at their factories. Say that we have a second table, `bosses`, defined as follows:

```
bosses = Table()
bosses['Factory'] = ['Berkeley', 'Palo Alto']
bosses['Name'] = ['Belinda', 'Patricia']
```

Write a short description, in English, of the value of the following Python expression. (Ignore the backslash; it's used to put single-line Python expressions on multiple lines.)

```
production.select(['Factory','Widgets Produced']).group('Factory', collect=np.sum) \
  .join('Factory', bosses, 'Factory').select(['Name', 'Widgets Produced sum'])
```

Answer:

(a) `production.sort('Widgets Produced', descending=False)`

(b) (a) 6. The value of `production.rows` is a list of all the rows in the table, and there are 6 rows.

(b) 3. The value of `production.columns` is a list of all 3 columns in the table. (Each column is an array of length 6.)

(c) 6. The value of `production['Date']` is an array of dates, one for each for the rows in the table.

(d) 6. `production['Date'] >= 5` is an array of boolean values, where the 0th is `False` because `production['Date'][0]` is less than 5, the 1st is `True` because `production['Date'][1]` is greater than or equal to 5, etc. (Each element of the array of numbers `production['Date']` is being compared with the number 5.) Each element counts as part of the array, even the `False` ones.

(e) `[False, True, True, False, True, True]`.

(f) 6. Even though all the values in the array `production['Date'] >= 7` are `False`, that array is still of length 6.

(g) 2. Generically, `production.where(x)` is a Table with a subset of the rows of `production`. If `x[0]` is `True` then the 0th row is included, if `x[1]` is `True` then the 1st row is included, etc. So the number of rows is just the number of `True`s in the given array. *Note*: There was a bug in the first version of this problem – the given array had only 5 entries, not 6. Technically this would have resulted in an error, but we will accept either 2, 3, or "error" as answers to this problem.

(h) 4. There are 4 `True` values in the array `production['Date'] >= 5`, since there are 4 dates that are 5 or larger.

(i) 3. There are 3 `True` values in the array `production['Factory'] == 'Berkeley'`.

(c) The value is a Table with one row per factory, containing each factory's name and the total number of widgets produced at that factory. *Note*: There was a bug in the first version of this problem – `select` was called like `select('Name', 'Widgets Produced sum')` rather than `select(['Name', 'Widgets Produced sum'])`. So we will also accept an answer that says there will be an error. (If we saw this question on a problem set, we would write about both the error and the likely *intended* value of the expression.)

(d) `np.sum(production.where(production['Date'] == 5)['Widgets Produced'])`

(e) The value is a Table with one row per factory boss, containing each boss's name and the total number of widgets produced at factories they managed.

Problem 2 Histograms

This problem is about a graphic that appears in the Visualizations chapter (Chapter 2) of the textbook. It is the pair of figures, one blue and one green, presented by Statistics Canada. In this problem we will only focus on the blue one, as the green one can be handled in a similar manner.

The blue figure attempts to show the distribution of after-tax income in Canadian families in 2005. The data are from the Canadian census of 2006.

(a) Is the graph a histogram? Why or why not?

(b) Based on the figure, you can assume that the percent of families with incomes in the range $[100,000, 125,000)$ Canadian dollars is 6%. If you were to draw a histogram of the incomes using the density scale (total area = 1), the height of the bar over the $[100,000, 125,000)$ bin would be:

(i) $6/25,000$ per thousand dollars

(ii) $6/25,000$ per dollar

(iii) $0.06/25,000$ per thousand dollars

(iv) $0.06/25,000$ per dollar

(v) $6/25$ per thousand dollars

(vi) $6/25$ per dollar

(vii) $0.06/25$ per thousand dollars

(viii) $0.06/25$ per dollar

Pick all that are correct (there may be more than one), and explain your choices below.

Answer:

(a) No, it is not a histogram. To be a proper histogram, a graph must have the property that the width of the interval covered by each bar (in the horizontal axis units) times its height equals the proportion (or perhaps the percentage or count) of elements in that interval. In this plot, \$10,000 times 12.5% is not the proportion, percentage, or count of people with income in the range \$10,000 to \$19,999.

Also, the visual widths of the bins in a histogram should be proportional to their actual widths, but this graph has bins of width \$10,000, \$25,000, and \$50,000 with equal visual width.

(b) (iv) and (vii).

Each bin in a density histogram must have the property that

`(proportion of elements in bin) = (bin height) * (bin width).`

Solving for bin height, we have:

`(bin height) = (proportion of elements in bin) / (bin width).`

Since the percent of families in the bin $[100000, 125000)$ is 6, the proportion is .06. And the bin width is $\$125000 - \$100000 = \$25000$. So one answer is .06/\$25000. (Note the units in the denominator!) In English, "per" means essentially "divided by", so we can say this as ".06 per 25 thousand dollars" or, equivalently, ".06/25000 per dollar" or ".06/25 per thousand dollars". So (iv) and (vii) are correct. All the others are not equivalent to that. (i) is too low by a factor of 1000; (ii) is too large by a factor of 100; (iii) is too low by a factor of 100000; (v) is too large by a factor of 100; (vi) is too large by a factor of 1000; and (viii) is too large by a factor of 10.