S&DS 123 YData: An Introduction to Data Science
Spring 2019 MIDTERM

INSTRUCTIONS

- You have 50 minutes to complete the exam.
- The exam is closed book, closed notes, closed computer, closed phone, and closed calculator, except one hand-written 8.5" × 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.

First name	
Last name (Surname)	
NetID	

Problem	Score	Out of
1		8
2		2
3		4
4		6
5		10
Total		30

1. **(8 points) Tables**

   (a) Melbourne (Australia) has been collecting data on the number of pedestrians per hour in various sites around the city since 2009. A data table named `pedestrian` contains a summary of this information, with the total number of pedestrians per site each month for the years 2009 through the beginning of 2019. The first four rows of the table are displayed below. `Sensor_Name` is the name of the sensor location, `Year` and `Month` are the year and month of the observation, and `Ped_Counts` are the total number of pedestrians counted.

Sensor_Name	Year	Month	Monthly_Counts
Australia on Collins	2009	August	489810
Australia on Collins	2009	December	551491
Australia on Collins	2009	July	530208
Australia on Collins	2009	June	484893

Provide the Python expressions to compute the values described below. You can assume the statements `from datascience import *` and `import numpy as np` have been executed.

- The total number of pedestrians counted.
  `sum(pedestrian.column("Monthly_Counts"))`

- The `Sensor_Name`, `Year`, `Month` that has had the highest `Monthly_Counts`.
  `pedestrian.sort("Monthly_Counts", descending = True)~~.select(0)~~.drop("Monthly_Counts")`
  .take(0)

- The `Sensor_Name` that has had the fewest total pedestrians, across all months and years.
  `pedestrian.group("Sensor_Name",sum).sort(3).column(0).item(0)`

- One of the `Sensor_Names` is `New Quay`. Create a table that includes only the total number of pedestrians per year for `New Quay`. There should only be two columns in your table. Call this table `nq`.
  `nq = pedestrian.where("Sensor_Name","New Quay").drop("Sensor_Name").drop("Month")`
  `.group("Year",sum).sort(0)`

2. **(2 points) Causality**

In order to assess the effects of exercise on reducing cholesterol, a researcher sampled 50 people from a local gym who exercise regularly and 50 people from the surrounding community who do not exercise regularly. Each subject reported to a clinic to have their cholesterol measured. The subjects were unaware of the purpose of the study, and the technician measuring the cholesterol was not aware of whether the subject exercises regularly or not. What type of study is this? (Circle your response)

A Randomized Controlled Experiment An Observational study

An Observational study because subjects were not randomly assigned to the Exercise Regularly vs. Not Exercise Regularly groups

3. **(4 points) Probability**

(a) A college basketball player makes 60% of his free throws. Right at the end of a game, with the score tied (both teams have the same number of points), he is fouled attempting a three-pointer and hence is awarded three free throws. To win the game, he only needs to make one. What is the probability that he wins the game? Write your answer as a Python expression.

To win the game, the player needs to make at least one basket:

`1 - (1-.6)**3 = 1 - (.4)**3 ≈ 0.936`

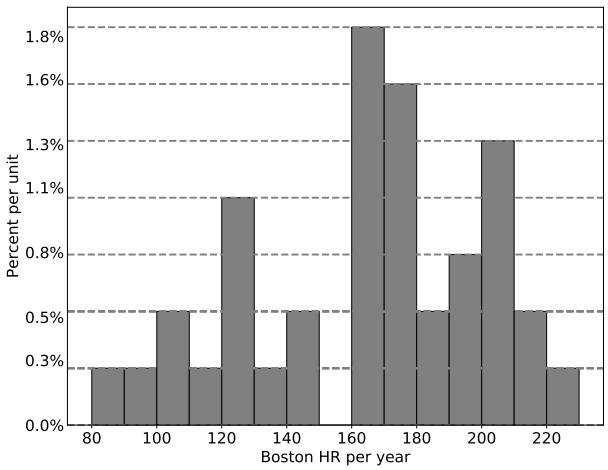
(b) Assume that birthdays are uniformly distributed across 365 days; that is, the probability of a birthday on any given date is $\frac{1}{365}$, ignoring leap years. It then turns out that in a room with 23 randomly selected people there is more than a 50% chance that at least two people will share the same birthday! Write a Python expression to compute this probability.

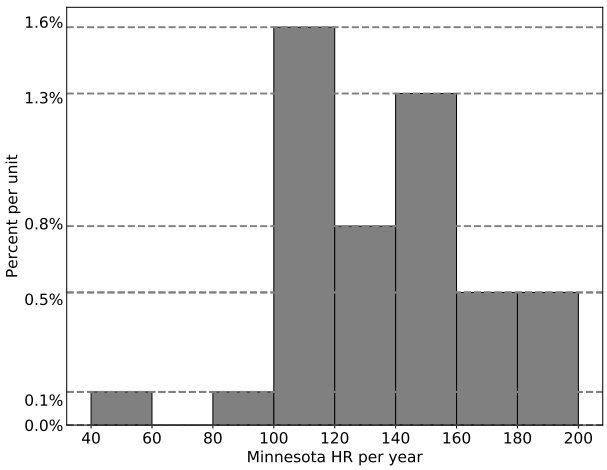
```
1 - np.prod(1 - np.arange(23)/365)
```

4. **(6 points) Distributions**

The Boston Red Sox and the Minnesota Twins are two professional baseball teams in the United States. The total number of home runs (HR) for each team from 1980 to 2018 (39 years) has been tracked, and a histogram is displayed below for each team. The bin widths for Boston is defined as `bins = np.arange(80,240,10)` and for Minnesota as `bins = np.arange(40,220,20)`.

Calculate the quantities specified below or write "Unknown" if there is not enough information to express the quantity as a single number (not a range). Be sure to show your work.





- The percentage of Boston home runs that are greater than the maximum number of Minnesota home runs.
  Unknown. The maximum Minnesota bin goes to 200, but the maximum number of home runs could be any value between 180 and 200.

- The percentage of home runs below 100 for each team.
  Boston: 10*(.3+.3) = 6 %

  Minnesota: 20*(.1+.1) = 4 %

- If the Boston histogram is redrawn so that the bins from 160 to 170 and 170 to 180 are combined into a single bin from 160 to 180, what would be the height of the bin?
  10*(1.8+1.6)/20 = 1.7 %

5. **(10 points) Hypothesis testing.** In a test of ESP (extrasensory perception), the experimenter looks at cards that are hidden from the subject who claims to have ESP. Each card contains a star, a circle, or a square. An experimenter looks at each of 100 cards in turn, and the subject tries to name the shape on the card. Of the 100 cards, the subject was able to get 38 correct.

   - State a null hypothesis to investigate the question of whether or not the subject has ESP.
     The null hypothesis is that the subject does not have ESP and would just be randomly guessing the image on the cards.

   - Fill in the blank below to calculate the observed test statistic, where `observed` is the observed number correct out of 100 and `expected` is the expected number correct out of 100 under the null hypothesis.

     ```
     observed = 38
     expected = _____________
     observed_test_statistic = observed - expected
     ```

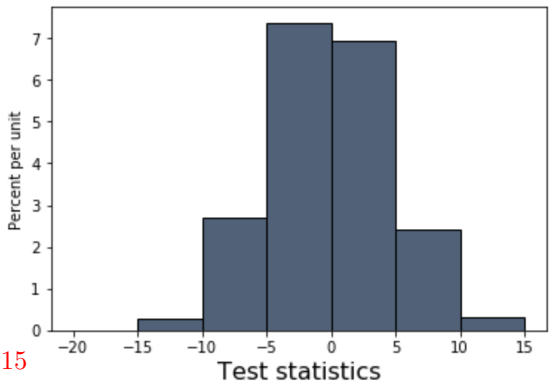
     ```
     1/3*100
     ```

   - A simulation is carried out to test the hypothesis. Fill in the blanks below for generating a sample of the test statistics under the assumption that the null hypothesis is true.

     ```
     sampled_test_statistics = _____________

     for i in np.arange(20000):

         new_test = 100*sample_proportions(_____________, _____________).item(0) - _____________

         _____________ = np.append(_____________, _____________)
     ```

     ```
     sampled_test_statistics = make_array()
     for i in np.arange(20000):
         new_test = 100*sample_proportions(100, [1/3, 2/3]).item(0) - expected
         sampled_test_statistics = np.append(sampled_test_statistics, new_test)
     ```

   - Below is a histogram using the final `sampled_test_statistics` from the previous question. What is the approximate p-value of the test using an observed test statistic rounded to 5, and what is your conclusion about the test?

     p-value $\approx$ 5*2.5 + 5*0.5 = 0.15

     Since the p-value is greater than 0.05, we would not reject the null hypothesis.