Homework 4

Felicia Liu

09/25/2020

- Solutions will be released on Tuesday, September 29.
- This semester, homework assignments are for practice only and will not be turned in for marks.

Helpful hints:

- Every function you need to use was taught during lecture! So you may need to revisit the lecture code to help you along by opening the relevant files on Datahub. Alternatively, you may wish to view the code in the condensed PDFs posted on the course website. Good luck!
- Knit your file early and often to minimize knitting errors! If you copy and paste code for the slides, you are bound to get an error that is hard to diagnose. Typing out the code is the way to smooth knitting! We recommend knitting your file each time after you write a few sentences/add a new code chunk, so you can detect the source of knitting errors more easily. This will save you and the GSIs from frustration! **You must knit correctly before submitting.**
- If your code runs off the page of the knitted PDF then you will LOSE POINTS! To avoid this, have a look at your knitted PDF and ensure all the code fits in the file (you can easily view it on Gradescope via the provided link after submitting). If it doesn't look right, go back to your .Rmd file and add spaces (new lines) using the return or enter key so that the code runs onto the next line.

[12 points] Part 1: Simulating birth defect data and sampling from an infinitely large population

The Center for Disease Control and Prevention (CDC) estimates that 1 in every 33 infants is born with a birth defect in the United States each year.

1) [3 points] Define a random variable for "birth defect". Write down the probability model for the random variable. Round each percentage to two decimal places (e.g., 0.43224 would be rounded to 43.22%). Is the sample space discrete or continuous?

The random varible X is defined for "birth defect." The sample space is discrete.

You might want to use the table template below to write out your probability model. If not, then delete it. *Knit now* to see how this table is rendered in your PDF.

Birth Defect	No Birth Defect
P(X)	1 - P(X)
3.03%	96.97%

2) [2 points] Simulate data that equals 0 if there is no birth defect and equals 1 if there is a birth defect. Simulate this data for 200 births at a local hospital. Be sure to use the risk of birth defect from part a). Assign your simulated output the name `sim_01`. Print your simulated births to the screen.

Before you run your simulation, we will "set the seed". We all will set the seed to 100. This means that everyone's simulation will yield the exact same dataset.

```
set.seed(100)
# execute this line before you write your simulation code.
# only execute the set.seed() function one time.
```

```
sim_01 <- rbinom(n = 200, size = 1, prob = 0.0303)
sim_01
```

```
##   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [38] 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [75] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
## [112] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [149] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [186] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
```

```
check_problem2()
```

```
## [1] "Checkpoint 1 Passed: You made an integer vector"
## [1] "Checkpoint 2 Passed: Correct number of elements!"
##
## Problem 2
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

Notice that `sim_01` is not a data frame, rather it is a vector of numbers. The following code stores `sim_01` as a data frame and changes its variable name. Run this code and view `sim_01` in the Viewer pane.

```
library(dplyr)
sim_01 <- as.data.frame(sim_01) # watch what happens to sim_01 in your environment
names(sim_01) # prints the variable names in the sim_01 data frame
```

```
## [1] "sim_01"
```

```
sim_01 <- sim_01 %>% rename(birth_defect = sim_01)
```

3) [2 points] Write code to determine the number of birth defects that occurred in your simulation, and the corresponding proportion with birth defects. Assign your output the name `output_01`. Print `output_01` to the screen. Hint: Use `dplyr` functions to do this.

```
output_01 <-sim_01 %>% summarize(num_birth_defects = sum(birth_defect)) %>%
  mutate(prop_birth_defects = num_birth_defects/200)

output_01
```

```
##   num_birth_defects prop_birth_defects
## 1                 5              0.025
```

```
check_problem3()
```

```
## [1] "Checkpoint 1 Passed: Correct! Output_01 should be a dataframe"
## [1] "Checkpoint 2 Passed: Correct number of columns (two columns)"
## [1] "Checkpoint 3 Passed: Correct number of rows (one row)"
##
## Problem 3
## Checkpoints Passed: 3
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

4) [2 marks] Re-run your simulation four more times and assign the output to a unique name each time. Print to the screen the number and proportion after each run. (Basically, "recycle" the code above four times)

```
sim_02 <- rbinom(n = 200, size = 1, prob = 0.0303)
sim_02 <- as.data.frame(sim_02) %>% rename(birth_defect = sim_02)
output_02 <- sim_02 %>% summarize(num_birth_defects = sum(birth_defect)) %>%
  mutate(prop_birth_defects = num_birth_defects/200)
output_02
```

```
##   num_birth_defects prop_birth_defects
## 1                 5              0.025
```

```
sim_03 <- rbinom(n = 200, size = 1, prob = 0.0303)
sim_03 <- as.data.frame(sim_03) %>% rename(birth_defect = sim_03)
output_03 <- sim_03 %>% summarize(num_birth_defects = sum(birth_defect)) %>%
  mutate(prop_birth_defects = num_birth_defects/200)
output_03
```

```
##   num_birth_defects prop_birth_defects
## 1                10               0.05
```

```
sim_04 <- rbinom(n = 200, size = 1, prob = 0.0303)
sim_04 <- as.data.frame(sim_04) %>% rename(birth_defect = sim_04)
output_04 <- sim_04 %>% summarize(num_birth_defects = sum(birth_defect)) %>%
  mutate(prop_birth_defects = num_birth_defects/200)
output_04
```

```
##   num_birth_defects prop_birth_defects
## 1                 6               0.03
```

```
sim_05 <- rbinom(n = 200, size = 1, prob = 0.0303)
sim_05 <- as.data.frame(sim_05) %>% rename(birth_defect = sim_05)
output_05 <- sim_05 %>% summarize(num_birth_defects = sum(birth_defect)) %>%
  mutate(prop_birth_defects = num_birth_defects/200)
output_05
```

```
##   num_birth_defects prop_birth_defects
## 1                 8               0.04
```

```
check_problem4()
```

```
## [1] "Checkpoint 1 Passed: Correct! output_02 should be a dataframe"
## [1] "Checkpoint 2 Passed: Correct number of columns (two columns)"
## [1] "Checkpoint 3 Passed: Correct number of rows (one row)"
## [1] "Checkpoint 4 Passed: Correct! output_03 should be a dataframe"
## [1] "Checkpoint 5 Passed: Correct number of columns (two columns)"
## [1] "Checkpoint 6 Passed: Correct number of rows (one row)"
## [1] "Checkpoint 7 Passed: Correct! output_04 should be a dataframe"
## [1] "Checkpoint 8 Passed: Correct number of columns (two columns)"
## [1] "Checkpoint 9 Passed: Correct number of rows (one row)"
## [1] "Checkpoint 10 Passed: Correct! output_05 should be a dataframe"
## [1] "Checkpoint 11 Passed: Correct number of columns (two columns)"
## [1] "Checkpoint 12 Passed: Correct number of rows (one row)"
##
## Problem 4
## Checkpoints Passed: 12
## Checkpoints Errored: 0
```

```
## 100% passed
## --------
## Test: PASSED
```

5) [1 mark] Assign the vector p5 to the simulated proportions from each of your five simulation in *increasing* order.

```
p5 <- c(0.025, 0.025, 0.03, 0.04, 0.05)
p5
```

```
## [1] 0.025 0.025 0.030 0.040 0.050
```

```
check_problem5()
```

```
## [1] "Checkpoint 1 Passed: You made a numeric vector"
## [1] "Checkpoint 2 Passed: You inputted 5 values"
## [1] "Checkpoint 3 Passed: Correct input for p5"
##
## Problem 5
## Checkpoints Passed: 3
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

6) [1 mark] Did you get close to the true value? Explain why there is variation in the proportions across the simulations.

Yes, I did get close to the true value of 0.0303. There is variation in the proportions across the simulations because for each simulation, I am sampling 200 different births. Essentially, I sampled 200 births a total of 5 times. The different samples of births will give a different number of birth defects per sample and hence a different proportion.

7) [1 mark] Suppose that rather than simulating 5 samples of size 200, we simulated 5 samples of size 1000. In 1-2 sentences, how would you expect the group of proportion estimates from part e) to be different? Comment both on the accuracy of these values at predicting the true value, and their variance. If you're not sure, you can re-run your simulation with a larger sample size and see how the results change to deduct the difference.

The group of proportion estimates should be even closer to the true proportion when the sample size is larger. The values would be more accurate in predicting the true value and there should be less variance.

[8 points] Part 2: Probability of HIV and Hepatitis C

Approximately 1.1 million Americans have HIV and 3.5 million Americans have Hepatitis C (HCV). The number of individuals with coinfection (e.g., both HIV and HCV) is 300,000. Among individuals with HIV, approximately 25% have Hepatitis C. The total US population was approximately 321 million at the time of these statistics.

references for these stats:

- https://www.cdc.gov/hiv/basics/statistics.html
- https://www.cdc.gov/media/releases/2016/p0504-hepc-mortality.html
- https://www.cdc.gov/hepatitis/populations/hiv.htm

8) [2 points] Calculate the probability that a randomly chosen American will have HIV. Calculate the probability that a randomly chosen American will have HCV. Convert to percentages and round to two decimal places. Save these values as the vector p2a with the proportion for HIV first then HCV. Don't inclue the % in your answer.

```
p8 <- c(0.34, 1.09)
p8
```

```
## [1] 0.34 1.09
```

```
check_problem8()
```

```
## [1] "Checkpoint 1 Passed: You made a numeric vector"
## [1] "Checkpoint 2 Passed: You inputted 2 values"
## [1] "Checkpoint 3 Passed: Correct input for p8"
##
## Problem 8
## Checkpoints Passed: 3
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

9) [2 points] Without using the number of co-infections provided in the question, calculate the probability that someone will have both HIV and HCV. Convert to a percent rounded to two decimal places and save to object p2b. Don't include the percent in your answer.

```
p9 <- 0.09
p9
```

```
## [1] 0.09
```

```
check_problem9()
```

```
## [1] "Checkpoint 1 Passed: Correct! It is numeric"
## [1] "Checkpoint 2 Passed: Correct rounding"
## [1] "Checkpoint 3 Passed: Correct answer for p9"
##
## Problem 9
## Checkpoints Passed: 3
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

10) [2 points] Are HIV and HCV infections independent? Show work to support your answer. Uncomment your selection.

```
#p10 <- "independent"
p10 <- "not independent"
check_problem10()
```

```
## [1] "Checkpoint 1 Passed: Correct selection"
## [1] "Checkpoint 2 Passed: Correct"
##
## Problem 10
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

11) [2 points] In general, does P(A|B) equal P(B|A)? Calculate P(HIV | HCV) and report whether or not it is equal to P(HCV | HIV).

No, in general, P(A|B) does not equal P(B|A). P(HIV | HCV) = 8.57% while P(HCV | HIV) = 25%.

[9 points] part 3: Screening for lung cancer

Background reading: Read pages 258-261 (and optionally 261-264) of Baldi & Moore, Edition 4. (For earlier editions, look for the section on diagnostic testing in medicine or on screening which covers sensitivity, specificity, negative predictive value, and positive predictive value).

Lung cancer is a leading cause of cancer-related deaths in the United States. Researchers examined the idea of testing all Medicare-enrolled heavy smokers for lung cancer with a computed tomography (CT) scan every year. In this population, the lifetime chance of developing lung cancer is high. In any given year, approximately 3% of heavy smokers develop lung cancer. The CT scan positively identifies lung cancer 89% of the time, and it gives a negative results for 93% of individuals who do not have lung cancer.

12) [3 points] Use probability notation to express the three proabilities cited. Make sure to define each event using a capital letter (or two). Provide the terminology for the 89% and 93% values based on your readings.

H = heavy smoker, L = lung cancer, P = positive, N = negative

P(L|H) = 3%

P(P|L) = 89% (true positive)

P(N|~L) = 93% (true negative)

13) [3 points] What percent of CT scans in this target population would be positive? Answer this question by making either a probability tree or using absolute frequencies. Show your work.

(Using a probability tree)

0.03 * 0.89 + 0.97 *0.07 = 0.0267 + 0.0679 = 0.001869 = 0.19%

<Note: If you are writing your solutions in R markdown you may want to upload an image of a hand-drawn tree diagram (this is optional). If so use the following code, or delete if not using. Be sure to remove the option "eval = F" if using this code or it won't run when you knit the file!:>

14) [1 point] We will now solve for the probability that a Medicare-enrolled heavy smoker who gets a positive scan actually has lung cancer. Write out the probability statement for this amount.

P(L|P) = P(true positive)/(P(true positive)+P(false positive)) = (0.89 * 0.03)/(0.89 * 0.03 + 0.97 * 0.07) = 0.0267/(0.0267 + 0.0679) = 0.2822 = 28.22%

15) [1 point] Calculate the probability value from question 14 based on your previous work from question 13. Store the answer as a percentage rounded to one decimal place in the object p15.

```
p15 <- 28.2
p15
```

```
## [1] 28.2
```

```
check_problem15()
```

```
## [1] "Checkpoint 1 Passed: Correct, it's numeric"
## [1] "Checkpoint 2 Passed: Correct, it is a percentage"
## [1] "Checkpoint 3 Passed: Correct rounding"
## [1] "Checkpoint 4 Passed: Correct"
##
## Problem 15
## Checkpoints Passed: 4
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

16) [1 point] What term from your reading does this value represent? Store the answer in object p16.

```
p16 <- "positive predicted value"
p16
```

```
## [1] "positive predicted value"
```

```
check_problem16()
```

```
## [1] "Checkpoint 1 Passed: Correct, it is character"
## [1] "Checkpoint 2 Passed"
##
## Problem 16
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

Check your score

Click on the middle icon on the top right of this code chunk (with the downwards gray arrow and green bar) to run all your code in order. Then, run this chunk to check your score.

```
# Just run this chunk.
total_score()
```

```
##              Test Points_Possible       Type
## Problem 1  FAILED               0 autograded
## Problem 2  PASSED               2 autograded
## Problem 3  PASSED               2 autograded
## Problem 4  PASSED               2 autograded
```

```
## Problem 5  PASSED              1 autograded
## Problem 6  FAILED              0 autograded
## Problem 7  FAILED              0 autograded
## Problem 8  PASSED              2 autograded
## Problem 9  PASSED              2 autograded
## Problem 10 PASSED              2 autograded
## Problem 11 FAILED              0 autograded
## Problem 12 FAILED              0 autograded
## Problem 13 FAILED              0 autograded
## Problem 14 FAILED              0 autograded
## Problem 15 PASSED              1 autograded
## Problem 16 PASSED              1 autograded
```