2 Calculation Exercises

James Young

6/5/2019

# General instructions.

There are 5 exercises below. You are required to provide solutions for at least four of the five. You are required to solve at least one exercise in R, and at least one in SAS. You are required to provide five solutions, each solution will be worth 10 points. Thus, you may choose to provide both R and SAS solutions for a single exercise, or you may solve all five problems, mixing the languages as you wish. Warning - we will be reusing the formulas from the first three exercises in later homework, so if you implement them now later exercises will be easier.

#### Experimental

I’ve been arguing that this course should also include Python. To explore this idea, I’ll allow one solution (10 of your 50 points) to be implemented in Python. To get full credit for a Python solution :

* Solve one of the first three exercises.
* Explain how the Python solution differs from the corresonding R or SAS solution. Note the differences in the languages. For example, does Python use the same assignment operator? Are all math operators the same as R or SAS? Are the math or statistics functions loaded by default? For the first exercise, I’ve found three important differences between R and Python.
* You can include Python in RMarkdown by replacing r with python in the code chunk prefix.
* I won’t be teaching Python this summer, but if you’re familiar with Python, this may help understand the inner workings of R or SAS.

# Exercise 1

### Please Grade Exercise 1

Cohen gives a formula for effect size, , for the difference between two means and , as

where is a pooled standard deviation. Use the formula .

Calculate the effect size for the differences among calories per serving, 1936 versus 2006, 1936 vs 1997 and 1997 vs 2006. Use the values from Wansink, Table 1 as given in Homework 1 or in the course outline.

##From Week 1  
CookingTooMuch.dat <- data.frame(  
 Year=c(1936, 1946, 1951, 1963, 1975, 1997, 2006),  
 CaloriesPerRecipeMean = c(2123.8, 2122.3, 2089.9, 2250.0, 2234.2, 2249.6, 3051.9),  
 CaloriesPerRecipeSD = c(1050.0, 1002.3, 1009.6, 1078.6, 1089.2, 1094.8, 1496.2),  
 CaloriesPerServingMean = c(268.1, 271.1, 280.9, 294.7, 285.6, 288.6, 384.4),  
 CaloriesPerServingSD = c(124.8, 124.2, 116.2, 117.7, 118.3, 122.0, 168.3),  
 ServingsPerRecipeMean = c(12.9, 12.9, 13.0, 12.7, 12.4, 12.4, 12.7),  
 ServingsPerRecipeSD = c(13.3, 13.3, 14.5, 14.6, 14.3, 14.3, 13.0)  
)

## Answer

Enter the R code in the chunks below. If you choose SAS for this exercise, use the marked portion in the SAS homework template.

#### 1936 versus 2006

Effect size (absolute value) is < 0.8 indicating this is not a large effect.

##From Week 1  
##d = \frac{|m\_1-m\_2|}{s\_{pooled}} where s\_{pooled} = \sqrt{(s\_1^2 + s\_2^2)/2}  
##Plugging in numbers from table  
effect1 <- (384.4-268.1)/((((124.8)^2+(168.3)^2)/2)^(1/2))  
print(effect1)

## [1] 0.784987603959

#### 1936 versus 1997

Effect size (absolute value) is < 0.8 indicating this is not a large effect.

effect2 <- (288.6-268.1)/((((122.0)^2+(168.3)^2)/2)^(1/2))  
print(effect2)

## [1] 0.139470482705

#### 1997 versus 2006

Effect size (absolute value) is < 0.8 indicating this is not a large effect.

effect3 <- (384.4-288.6)/((((122.0)^2+(168.3)^2)/2)^(1/2))  
print(effect3)

## [1] 0.651769377713

To check your work, consider that Cohen recommends that be considered a small effect, a medium effect and a large effect. I don’t find any of these to be fully large effects.

# Exercise 2.

### Please Do Not Grade

Suppose you are planning an experiment and you want to determine how many observations you should make for each experimental condition. One simple formula (see Kuehl, “Design of Experiments : Statistical Principles of Research Design and Analysis”) for the required replicates is given by

where and .

Use this formula to calculate the number of replicates required to detect differences between calories per serving, 1936 versus 2006, 1936 vs 1997 and 1997 vs 2006. You will need to research how to use the normal distribution functions (\*norm in R, ). Use and for probabilities, and let mean = 0 and sd = 1 (both should be positive).

Since must be an integer, you will need to round up. Look up the built in functions for this.

## Answer

### Please Grade Exercise 1

Enter the R code in the chunks below. If you choose SAS for this exercise, use the marked portion in the SAS homework template.

#### 1936 versus 2006

sdpooled <- ((((124.8)^2+(168.3)^2)/2)^(1/2))  
Diff <- function(x,y){(x-y)/((x-y)^2)}  
diff <- Diff(168.3,124.8)  
CV <- (((((124.8)^2+(168.3)^2)/2)^(1/2))/((168.3-124.8)/2))

#### 1936 versus 1997

#### 1997 versus 2006

To check your work, use the rule of thumb suggested by van Belle (“Statistical Rules of Thumb”), where

with . How does this compare with your results? Why does this rule of thumb work? How good is this rule of thumb?

# Exercise 3

### Please Grade Exercise 3

The probablity of an observation , when taken from a normal population with mean and variance is calculated by

For values of , write code to calculate .

## Answer

Enter the R code in the chunks below. If you choose SAS for this exercise, use the marked portion in the SAS homework template.

#### 

x <- (1/((2\*pi)^(1/2))) \* exp(-((-0.1-0)^2)/((2\*1)^2))  
print(x)

## [1] 0.397946170357

#### 

x <- (1/((2\*pi)^(1/2))) \* exp(-((0.0-0)^2)/((2\*1)^2))  
print(x)

## [1] 0.398942280401

#### 

x <- (1/((2\*pi)^(1/2))) \* exp(-((0.1-0)^2)/((2\*1)^2))  
print(x)

## [1] 0.397946170357

You can confirm your results using the built in normal distribution function. Look up dnorm in R help and use the same values for x, mean and sigma as above. You should get matching results to at least 12 decimal places.

dnorm(0, mean =0, sd =1)

## [1] 0.398942280401

dnorm(0.1, mean =0, sd =1)

## [1] 0.396952547477

#dnorm of zero seems right but dnorm of 0.1 seems off

# Exercise 4

### Please Grade Exercise 4

## Part a

Write code to compute

Type this in verbatim, using only numbers, -,\* and /, with no parenthesis. Do you agree with the result? Explain why, one or two sentences.

#### Answer

7-1\*0+3/3

## [1] 8

Yes, I agree with the order of operations used in this calculation.

## Part b

According to “Why Did 74% of Facebook Users Get This Wrong?” (<https://profpete.com/blog/2012/11/04/why-did-74-of-facebook-users-get-this-wrong/>), most people would compute the result as 1. Use parenthesis ( ) to produce this result.

#### Answer

(7-1)\*0+3/3

## [1] 1

## Part c

Several respondents to the survey cited in Part 2 gave the answer 6. Add *one* set of parenthesis to produce this result.

#### Answer

7-1\*(0+3/3)

## [1] 6

# Exercise 5.

### Please Grade Exercise 5

### Part a

Quoting from Wansink and Payne

Because of changes in ingredients, the mean average calories in a recipe increased by 928.1 (from 2123.8 calories … to 3051.9 calories … ), representing a 43.7% increase.

Show how 43.7% is calculated from 2123.8 and 3051.9, and confirm W&P result.

#### Answer

x <- ((3051.9-2123.8)/2123.8)\*100 # where 100 is 100%  
print(x)

## [1] 43.6999717488

#Confirmation should print below

The resulting increase of 168.8 calories (from 268.1 calories … to 436.9 calories …) represents a 63.0% increase … in calories per serving.

### Part b

Repeat the calculations from above and confirm the reported 63.0% increase in calories per serving. Why is there such a difference between the change in calories per recipe and in calories per serving?

#### Answer

y <- ((436.9-268.1)/268.1)\*100 # where 100 is 100%  
print(y)

## [1] 62.9615814994

## Calories per serving has increased more than calories per recipe. This may be due to a relatively

### Part c

Calculate an average calories per serving by dividing average calories per recipe by average servings per recipe, for years 1936 and 2006, then calculate a percent increase. Which of the two reported increases (a or b) are consistent with this result?

#### Answer

CPS1936 <- (2123.8/12.9)  
print(CPS1936)

## [1] 164.635658915

CPS2006 <- (3051.9/12.7)  
print(CPS2006)

## [1] 240.307086614

(CPS2006 - CPS1936)/CPS1936

## [1] 0.459629634298

#

It seems like neither of the results from the paper are completely similar to my calculation, but part b is closest.