Lab 11: The Relationship Between the Chi-Square Test for Independence and the Two Sample Z-Test for Proportions

Name and Student ID

Today's Date

```
BEGIN ASSIGNMENT
requirements: requirements.R
generate: true
files:
 - data
 - turn_in.py
 - src
```

Run this chunk of code to load the autograder package!

Instructions

- Due date: Tuesday, April 26th, at 10:00pm PST with 2 hour grace period.
- Late penalty: 50% late penalty if submitted within 24 hours of due date, no marks for assignments submitted thereafter.
- This assignment is graded on **correct completion**, all or nothing. You must pass all public tests and submit the assignment for credit.
- Submission process: Follow the submission instructions on the final page. Make sure you do not remove any `\newpage` tags or rename this file, as this will break the submission.

The Western Collaborative Group Study Dataset

The data we will look at for this week's lab comes from a cohort study that began in the 1960s. These data were collected prospectively to assess the effects of behavior type on coronary heart disease (CHD). At the beginning of the study, researchers enrolled 3524 men aged 39-59 who worked at a subset of corporations in California. Each individual's behavior type was assessed during an interview and all individuals were followed for 8.5 years (until 1969). Full data is available for 3142 participants. Of these, 257 (8.2%) had a CHD event.

Overview of the lab

The purpose of this lab is to investigate the relationship between the chi-square test of independence and the two sample z-test for proportions. To do this, we will look at the relationship between personality type (`dibpat`) and CHD outcome (`chd69`) in a random sample of WCGS participants.

Read in the data from the sample:

```
## # A tibble: 6 x 13
##      id  age0 height0 weight0  sbp0  dbp0 chol0 behpat0 ncigs0 dibpat0 chd69
##   <dbl> <dbl>   <dbl>   <dbl> <dbl> <dbl> <dbl>   <dbl>  <dbl>   <dbl> <dbl>
## 1  6092    45      70     168   118    84   275       3     14       0     0
## 2  3579    40      75     163   116    72   199       2      0       1     0
## 3 12671    48      70     173   138    88   197       1      0       1     0
## 4 13074    39      72     170   110    76   259       1     40       1     0
```

```
## 5 10366    49      69     182   122    82   238       3      0       0     0
## 6  3496    40      66     145   126    70   195       4      0       0     0
## # ... with 2 more variables: arcus0 <dbl>, cigs <dbl>
```

In this sample, `chd69 = 1` implies that a CHD event occurred. `dibpat0=1` codes participants with a "Type A" personality and `dibpat0 = 0` codes participants with a "Type B" personality. Here, CHD is the response variable and personality type is the explanatory variable.

1. State the null and 2 sided alternative hypotheses in words and using probability notation.

```
BEGIN QUESTION
name: p1
manual: true
```

H_0 : Behavior type and CHD are independent in this population. Stated another way: $H_0 : P(CHD = 1|TypeA) = P(CHD = 1|TypeB)$

H_A : Behavior type and CHD are dependent in this population. $H_A : P(CHD = 1|TypeA) \neq P(CHD = 1|TypeB)$

2. [1 point] Calculate a two sample z-test statistic to test the null hypothesis that behavior type is independent of CHD. Report the p-value rounded to 4 decimal places.

```
BEGIN QUESTION
name: p2
manual: false
points: 1
```

```
. = " # BEGIN PROMPT
p_value <- NULL # YOUR CODE HERE
p_value
" # END PROMPT

# BEGIN SOLUTION NO PROMPT

# First calculate the number of people and the proportion with CHD for those of
# Type A and Type B personalities:

summary_stats <- dat %>% group_by(dibpat0) %>% summarise(n = n(),
                                                         propCHD = mean(chd69))

summary_stats
```

```
## # A tibble: 2 x 3
##   dibpat0     n propCHD
##     <dbl> <int>   <dbl>
## 1       0   100    0.03
## 2       1   100    0.16
```

```
# To perform this test, you also need an estimate of the pooled proportion, because
# under the null hypothesis the two proportion are the same. Our best estimate of
# this takes all the "successes" (CHD) from both groups divided by the total sample size
# This is equivalent to asking for the mean of CHD using all the data:

pooled_p <- dat %>% summarise(p_under_null = mean(chd69))
pooled_p
```

```
## # A tibble: 1 x 1
##   p_under_null
##          <dbl>
## 1        0.095
```

```
# Use the pooled_p to calculate the SE under the null hypothesis
SE <- sqrt(pooled_p*(1-pooled_p)*(1/100 + 1/100))

SE_2 <- sqrt(pull(pooled_p, p_under_null)*(1-pull(pooled_p, p_under_null))*(1/100 + 1/100))

# Calculate the z statistic
z_stat <- ((0.16 - 0.03) - 0)/0.04146685
z_stat <- ((0.16 - 0.03) - 0)/SE
z_stat <- 3.135034

#The z_statistic is equal to 3.135034

## 2-sided p-value
p_value <- round(pnorm(q = z_stat, lower.tail = F)*2, 4)
p_value <- 0.0017
```

```
# END SOLUTION
```

```
## Test ##
test_that("p2a", {
  expect_true(p_value < 1 & p_value > 0)
  print("Checking: range of p-value")
})
```

```
## [1] "Checking: range of p-value"
## Test passed
```

```
## Test ##
test_that("p2b", {
  expect_true(all.equal(p_value, 0.0017))
  print("Checking: value of p-value")
})
```

```
## [1] "Checking: value of p-value"
## Test passed
```

3. [1 point] Check your p-value using the R function for a two-sample z test for proportions. Hint: to obtain the same p-value that you calculated by hand, you need to use `correct = F` as an argument.

```
BEGIN QUESTION
name: p3
manual: false
points: 1
```

```
. = " # BEGIN PROMPT
p_value_using_code <- NULL # YOUR CODE HERE
p_value_using_code
" # END PROMPT

# BEGIN SOLUTION NO PROMPT
prop.test(x = c(3, 16), n = c(100, 100), correct = F)
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  c(3, 16) out of c(100, 100)
## X-squared = 9.8284, df = 1, p-value = 0.001718
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.2092514 -0.0507486
## sample estimates:
## prop 1 prop 2
##   0.03   0.16
```

```
p_value_using_code <- 0.0017
# END SOLUTION
```

```
## Test ##
test_that("p3a", {
  expect_true(p_value_using_code < 1 & p_value_using_code > 0)
  print("Checking: range of p-value")
})
```

```
## [1] "Checking: range of p-value"
## Test passed
```

```
## Test ##
test_that("p3b", {
  expect_true(all.equal(p_value_using_code, 0.0017, tol = 0.0001))
  print("Checking: value of p-value")
})
```

```
## [1] "Checking: value of p-value"
## Test passed
```

In the previous questions, you calculated the two sample z-test comparing two proportions. We've recently learned about the chi-square test applied to one or two categorical variables. When we have two categorical variables, we can use the chi-square test of independence.

4. Make a 2 by 2 table of CHD vs. personality type and calculate the chi-square test statistic by hand. Calculate the p-value using an R function.

```
BEGIN QUESTION
name: p4
manual: true
```

CHD | No CHD | Total

———-|———|———-|———| Type A | | | | ———-|———|———-|———| Type B | | | | ———-|———|———-|———| Total | | | |

```
. = " # BEGIN PROMPT
p4 <- NULL # YOUR CODE HERE
p4
" # END PROMPT

# BEGIN SOLUTION NO PROMPT
p4 <- pchisq(9.828, df = 1, lower.tail = F)
# END SOLUTION
```

| | CHD | No CHD | Total |

———-|———|———-|———| Type A | 3 | 97 | 100 | ———-|———|———-|———| Type B | 16 | 84 | 100 | ———-|———|———-|———| Total | 19 | 181 | 200 |

(9.5-3)^2/9.5 + (90.5-97)^2/90.5 + (9.5-16)^2/9.5 + (90.5-84)^2/90.5 = 9.828438

5. [1 point] Compare the chi-square test statistic you calculated by hand to the test statistic output by the `chisq.test()` function. Report your p-value rounded to 4 decimal places.

```
BEGIN QUESTION
name: p5
manual: false
points: 1
```

```
. = " # BEGIN PROMPT
p_value_chisq <- NULL # YOUR CODE HERE
p_value_chisq
" # END PROMPT

# BEGIN SOLUTION NO PROMPT
# get the counts
dat %>% group_by(dibpat0) %>% count(chd69)
```

```
## # A tibble: 4 x 3
## # Groups:   dibpat0 [2]
##   dibpat0 chd69     n
##     <dbl> <dbl> <int>
## 1       0     0    97
## 2       0     1     3
## 3       1     0    84
## 4       1     1    16
```

```
library(tibble)
two_way <- tribble(~ chd, ~ no_chd,
                   3,        97, #row for Type B personality
                   16,       84) #row for Type A personality
two_way
```

```
## # A tibble: 2 x 2
##     chd no_chd
##   <dbl>  <dbl>
## 1     3     97
## 2    16     84
```

```
chisq.test(two_way, correct = F) #not using Yates'
```

```
##
##  Pearson's Chi-squared test
##
## data:  two_way
## X-squared = 9.8284, df = 1, p-value = 0.001718
```

```
p_value_chisq <- 0.0017
# END SOLUTION
```

```
## Test ##
test_that("p5a", {
  expect_true(p_value_chisq < 1 & p_value_chisq > 0)
  print("Checking: range of p-value")
})
```

```
## [1] "Checking: range of p-value"
## Test passed
```

```r
## Test ##
test_that("p5b", {
  expect_true(all.equal(p_value_chisq, 0.0017, tol = 0.0001))
  print("Checking: value of p-value")
})
```

```
## [1] "Checking: value of p-value"
## Test passed
```

6. Compare the chi-squared test statistic to the z-test statistic and the (z-test statistic)^2. What do you notice? What do you notice about the p-values for the two tests?

```
BEGIN QUESTION
name: p6
manual: true
```

- chi-square stat = 9.8284 (with `correct = F`)
- z-statistic = 3.135034, implying that the squared z-stat = 9.828441, which is equal to the chi-square test stat!
- The p-values of the two test are the same.

In summary, the chi-square test for independence and the two sample z-test for two proportions give rise to the same p-value. Their test statistics are different, where the chi-square is the z-statistic squared.

Submission

For assignments in this class, you'll be submitting using the **Terminal** tab in the pane below. In order for the submission to work properly, make sure that:

1. Any image files you add that are needed to knit the file are in the `src` folder and file paths are specified accordingly.
2. You **have not changed the file name** of the assignment.
3. The file knits properly.

Once you have checked these items, you can proceed to submit your assignment.

1. Click on the **Terminal** tab in the pane below.
2. Copy-paste the following line of code into the terminal and press enter.

cd; cd ph142-sp22/lab/lab11; python3 turn_in.py

3. Follow the prompts to enter your Gradescope username and password.
4. If the submission is successful, you should see "Submission successful!" appear as the output. **Check your submission on the Gradescope website to ensure that the autograder worked properly and you received credit for your correct answers. If you think the autograder is incorrectly grading your work, please post on piazza!**
5. If the submission fails, try to diagnose the issue using the error messages–if you have problems, post on Piazza under the post "Datahub Issues".

The late policy will be strictly enforced, **no matter the reason**, including submission issues, so be sure to submit early enough to have time to diagnose issues if problems arise.