

Assignment 4: Writing functions

Due Sunday, November 26, 2023 @ 11:59pm

Intro to R for Public Health AS.280.346, Fall 2023 Your assignment for this module is to write a series of functions for calculating 95% confidence intervals for different population parameters. To help with this, relevant entries from the formula sheet in Public Health Biostatistics are reproduced at the end of this assignment.

To complete your assignment, do the following items. You can use the `Assignment4_template.Rmd` file to do your work. When completed, please submit your assignment through Github.

- (1) Write a function called `mean_ci()` that takes the arguments listed below and returns either a 95% confidence interval for a single mean or a 95% confidence interval for the difference in means depending on those arguments. I've given default values for these arguments as well.

- `x` (no default value)
- `y` (default value is `NULL` which corresponds to `one.mean` option for the `type` argument)
- `type` (default value is `one.mean`, other value is `two.means`)
- ... (to be able to pass the `na.rm` option through to any functions as needed)

Include a way to check whether the vectors of `x` and `y` are numeric and thus such a confidence interval is appropriate.

If you would like a way to check that your confidence intervals are correct, here is the code I will use to check that your function works appropriately:

```
set.seed(8675309)
x <- rnorm(50)
y <- rnorm(30)

mean_ci(x)      # should get [1] -0.1404677  0.3458514

mean_ci(x, y, type="two.means")  # should get [1] -0.1875458  0.7320355

# checking the na.rm piece
x2 <- c(x, NA)
y2 <- c(y, NA)
mean_ci(x2, y2, type="two.means") # should get [1] NA NA
mean_ci(x2, y2, type="two.means", na.rm=TRUE)  # should get [1] -0.1875458  0.7320355

# checking the numeric piece
x3 <- c("hi", "how", "are", "you", "?")
mean_ci(x3)      # should get an informative error
mean_ci(x3, y2, type = "two.means") # should get an informative error

# check the type piece
mean_ci(x2, y2, type = "two.groups") # should get an informative error
```

- (2) Write a function called `prop_ci()` that takes the arguments listed below and returns either a 95% confidence interval for a single proportion or a 95% confidence interval for the difference in proportions depending on those arguments. I've given default values for these arguments as well.

- `x` (no default value)
- `y` (default value is `NULL` which corresponds to `one.mean` option for the `type` argument)
- `type` (default value is `one.prop`, other value is `two.props`)
- ... (to be able to pass the `na.rm` option through to any functions as needed)

Include a way to check whether the vectors of `x` and `y` are 0/1 vectors and thus such a confidence interval is appropriate.

If you would like a way to check that your confidence intervals are correct, here is the code I will use to check that your function works appropriately:

```
set.seed(8675309)
x <- rbinom(50, 1, .5)
y <- rbinom(30, 1, .2)

prop_ci(x)      # should get [1] 0.4196006 0.7003994

prop_ci(x, y, type="two.props")  # should get [1] 0.1978071 0.5888596

# checking the na.rm piece
x2 <- c(x, NA)
y2 <- c(y, NA)
prop_ci(x2, y2, type="two.props") # should get [1] NA NA or an error message depending
                                   # on how checked for binary vectors
prop_ci(x2, y2, type="two.props", na.rm=TRUE)  # should get [1] 0.1978071 0.5888596

# checking the binary piece
x3 <- c(3,4,5,2,0,1)
prop_ci(x3)                                # should get an informative error message
prop_ci(x3, y2, type = "two.props")        # should get an informative error message

# check the type piece
prop_ci(x2, y2, type = "two.groups")      # should get an informative error message
```

(3) Extra credit: Write a function called `ratio_ci()` that takes the arguments listed below and returns either a 95% confidence interval for the odds ratio or a 95% confidence interval for the relative risk between two groups depending on those arguments. I've given default values for these arguments as well.

- `p1` the sample proportion in group 1 (no default value)
- `n1` the sample size in group 1 (no default value)
- `p2` the proportion in group 2 (no default value)
- `n2` the sample size in group 2 (no default value)
- `type` (default value is `odds.ratio`, other value is `risk.ratio`)

Include a way to check whether the values of `p1` and `p2` that are provided are proportions between 0 and 1.

If you would like a way to check that your confidence intervals are correct, here is the code I will use to check that your function works appropriately:

```
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = .7)  # should get [1] 0.08275144 2.21958041
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = .7, type = "odds.ratio")  # should get [1] 0.08275144 2.21958041
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = .7, type = "risk.ratio")  # should get [1] 0.3883215 1.313870
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = .7, type = "hello")      # should get error message

# checking the proportions check
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = 1.3)  # should get error message
ratio_ci(n1 = 20, n2 = 10, p1 = 1.5, p2 = .3)  # should get error message
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = 1.3, type = "risk.ratio") # should get error message
```

```
ratio_ci(n1 = 20, n2 = 10, p1 = 1.5, p2 = .3, type = "risk.ratio") # should get error message
ratio_ci(n1 = 20, n2 = 10, p1 = .5, p2 = 1.3, type = "odds.ratio") # should get error message
ratio_ci(n1 = 20, n2 = 10, p1 = 1.5, p2 = .3, type = "odds.ratio") # should get error message
```

The information on the next page of this assignment is part of the formula sheet for Public Health Biostatistics, which gives the standard errors for many statistics. It may be helpful as you complete this assignment!

Standard Errors

Mean

$$se_{\bar{x}} = \sqrt{\frac{s^2}{n}}$$

Proportion

$$se_{\hat{p}} = \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}}$$

Difference of two Means

$$se_{\bar{x}_1 - \bar{x}_2} = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

Difference of two Proportions

$$se_{\hat{p}_1 - \hat{p}_2} = \sqrt{\frac{\hat{p}_1(1 - \hat{p}_1)}{n_1} + \frac{\hat{p}_2(1 - \hat{p}_2)}{n_2}}$$

Log Relative Risk

$$se_{LRR} = \sqrt{\left(\frac{1}{\hat{p}_1 n_1} + \frac{1}{\hat{p}_2 n_2}\right) - \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}$$

Log Odds Ratio

$$se_{LOR} = \sqrt{\frac{1}{\hat{p}_1 n_1} + \frac{1}{(1 - \hat{p}_1) n_1} + \frac{1}{\hat{p}_2 n_2} + \frac{1}{(1 - \hat{p}_2) n_2}}$$