R Notebook

```r
# Loading packages
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```r
library(rstanarm)
```

```
## Warning: package 'rstanarm' was built under R version 4.2.3

## Loading required package: Rcpp

## Warning: package 'Rcpp' was built under R version 4.2.3

## This is rstanarm version 2.21.4

## - See https://mc-stan.org/rstanarm/articles/priors for changes to default priors!

## - Default priors may change, so it's safest to specify priors, even if equivalent to the defaults.

## - For execution on a local, multicore CPU with excess RAM we recommend calling

##    options(mc.cores = parallel::detectCores())
```

10.1

Regression with interactions: Simulate 100 data points from the model, $y = b_0 + b_1 \cdot x + b_2 \cdot z + b_3 \cdot x \cdot z + error$, with a continuous predictor x and a binary predictor z, coefficients $b = (1, 2, -1, -2)$, and errors drawn independently from a normal distribution with mean 0 and standard deviation 3, as follows. For each data point i, first draw z_i, equally likely to take on the values 0 and 1. Then draw x_i from a normal distribution with mean z_i and standard deviation 1. Then draw the error from its normal distribution and compute y_i.

    a) Display your simulated data as a graph og y vs.x, using dots and circles for the points with $z = 0$ and 1, respectively

    b) Fit a regression predicting y from x and z with no interaction. Make a graph with the data and the two parallel lines showing the fitted model

    c) Fit a regression predicting y from x, z and their interaction. Make a graph with the data and the two lines showing the fitted model.

10.2

We are given the output of a linear regression with an outcome y, a pre-treatment predictor x and a treatment indicator z as well as their interaction.

    a) Write the equation of the estimated regression line of y on x for the treatment group, and the equation of the estimated regression line of y on x for the control group.

    b) Just use R to graph this exercise. Graph the two regression lines, assuming the values of x fall in the range (0, 10). On this graph also include a scatter plot of data (using open circles for treated units and dots for controls) that are consistent with the fitted model.

10.3

In this exercise and the next, you will simulate two variables that are statistically independent of each other to see what happens when we run a regression to predict one from the other. Generate 1000 data points from a normal distribution with mean 0 and standard deviation 1 by typing var1 <- rnorm(1000,0,1) in R. Generate another variable in the same way (call it var2). Run a regression of one variable on the other. Is the slope coefficient "statistically significant"? We do not recommend summarizing regressions in this way, but it can be useful to understand how this works, given that others will do so.

10.4

Continuing the previous exercise, run a simulation repeating this process 100 times. This can be done using a loop. From each simulation, save the z-score (the estimated coefficient of var1 divided by its standard error). If the absolute value of the z-score exceeds 2, the estimate is "statistically significant.

How many of these 100 z-scores exceed 2 in absolute value, thus achieving the conventional level of statistical significance?