|  |
| --- |
| **Resampling from a Dataset**  rep\_sample\_n(<NAME OF DATASET>,  size = <SIZE OF SAMPLE>,  replace = TRUE,  reps = <NUMBER OF RESAMPLES>)  ***Note:*** The value of TRUE for replace means sampling is done *with replacement*. For sampling *without replacement*, you would need to set replace equal to FALSE. |
| **Working in the infer Package Workflow for a Basic Regression** |
| **Obtaining the Sample (Observed) Slope**  obs\_slope <- <NAME OF DATASET> %>%  specify(response = <NAME OF VARIABLE>,  explanatory = <NAME OF VARIABLE>) %>%  calculate(stat = "slope")  ***Note:*** This step **must** be done **first**, before you find your confidence interval! |
| **Obtaining 1000 Bootstrap Slope Statistics**  bootstrap <- <NAME OF DATASET> %>%  specify(response = <NAME OF RESPONSE VARIABLE>,  explanatory = <NAME OF EXPLANATORY VARIABLE>) %>%  generate(reps = 1000, type = "bootstrap") %>%  calculate(stat = "slope")  ***Note:*** You choose the number of reps. I recommend choosing at least 1000, to get a good idea of the shape of the bootstrap distribution – remember we need to verify it is approximately normal. |
| **Plotting the Bootstrap Distribution**  visualize(data = bootstrap) +  labs(x = “<STATISTIC YOU CALCULATED>”)  ***Note:*** Your x-axis should declare what statistic your bootstrap distribution is visualizing (e.g., “slope statistic” or “slope statistic for the relationship between \_\_\_ and \_\_\_” ). |

|  |
| --- |
| **Obtaining a Percentile Confidence Interval from a Bootstrap Distribution**  get\_confidence\_interval(bootstrap,  level = **0.95**,  type = “percentile”)  ***Note:*** You choose the confidence level of your interval! |
| **Obtaining an SE Confidence Interval from a Bootstrap Distribution**  get\_confidence\_interval(bootstrap,  level = **0.95**,  type = “se”,  point\_estimate = obs\_slope)  ***Note:*** You choose the confidence level of your interval! |
| **Working in the infer Package Workflow for a Multiple Regression** |
| **Obtaining the Sample (Observed) Regression Fit**  obs\_fit <- <NAME OF DATASET> %>%  specify(<RESPONSE> ~ <EXPLANATORY VARIABLE 1> **\*** <EXPLANATORY VARIABLE 2>) %>%  fit()  ***Note:*** Within specify(), your syntax looks identical to what you used to fit a model with the lm() function. You use a ~ to separate the response from the explanatory variables.  ***Note:*** In the above code a **\*** separates the two explanatory variables. This would fit an interaction model! If you want to fit an additive model, you would need to use a **+** sign to separate the variables! |
| **Obtaining 1000 Bootstrap Fits**  bootstrap <- <NAME OF DATASET> %>%  specify(<RESPONSE> ~ <EXPLANATORY VARIABLE 1> **\*** <EXPLANATORY VARIABLE 2>) %>%  generate(reps = 1000, type = "bootstrap") %>%  fit()  ***Note:*** You choose the number of reps. I recommend choosing at least 1000, to get a good idea of the shape of the bootstrap distribution – remember we need to verify it is approximately normal. |

|  |
| --- |
| **Plotting the Bootstrap Distributions**  visualize(data = bootstrap)  ***Note:*** This will create **multiple** histograms, one for each variable included in the multiple regression. |
| **Obtaining a Percentile Confidence Interval from a Bootstrap Distribution**  get\_confidence\_interval(bootstrap,  level = **0.95**,  type = “percentile”,  point\_estimate = obs\_fit)  ***Note:*** You choose the confidence level of your interval! |
| **Obtaining an SE Confidence Interval from a Bootstrap Distribution**  get\_confidence\_interval(bootstrap,  level = **0.95**,  type = “se”,  point\_estimate = obs\_fit)  ***Note:*** You choose the confidence level of your interval! |