|  |
| --- |
| **Simulation-based Confidence Interval 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! |
| **Simulation-based Hypothesis Test for a Basic Regression** |
| **Obtaining 1000 Permuted Slope Statistics**  null\_dist <- <NAME OF DATASET> %>%  specify(response = <NAME OF RESPONSE VARIABLE>,  explanatory = <NAME OF EXPLANATORY VARIABLE>) %>%  hypothesize(null = “independence”) %>%  generate(reps = 1000, type = "permute") %>%  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 Null Distribution**  visualize(null\_distribution)  ***Note:*** You can add axis labels to this plot! All you need to do is connect the visualize() step to labs() using a **+** sign. |
| **Shading the p-value**  visualize(null\_distribution)+  shade\_p\_value(obs\_stat = obs\_slope,  direction = “two-sided”)  ***Note:*** You can add axis labels to this plot! All you need to do is connect the visualize() step to labs() using a **+** sign. |
| **Obtaining a p-value**  get\_p\_value(null\_dist,  obs\_stat = obs\_slope,  direction = “two-sided”) |
| **Parametric Methods for Obtaining a p-value for Basic Regression** |
| my\_model <- lm(<NAME OF RESPONSE VARIABLE> ~ <NAME OF EXPLANATORY VARIABLE>,  data = <NAME OF DATASET>)  get\_regression\_table(my\_model,  conf.level = **0.95**)  ***Note:*** If you want a 90% confidence interval (or an alpha of 0.1), you change conf.level to 0.90 |

|  |
| --- |
| **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! |
| **Simulation-based Confidence Interval 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! |