Caedon presentation

Caedon Ott

2022-12-14

To start out my analysis, I had to figure out how I was going to sample from such a massive data set. I used the following function to sample from a selected indicator:

```
f1 <- function(data, m,n){

  bootstrap_data <- list()
  for(i in 1:m){
    temp_sample_ind <- sample(1:length(data), n,replace=TRUE)
    bootstrap_data[[i]] <- data[temp_sample_ind]
  }
  return(bootstrap_data)
}
```

This gave me a massive data set of m's and n's, of which I took means of. Thus, for something like m=50 and n=250, I had 50 bootstrapped means with low variance due to having a large n.

Now that I had my sample, I could start computing my BCa confidence intervals. For the BCa algorithm, I wanted to do it by hand. We never did it in class, so I wanted to see what the computation would look like. I searched around on the internet, which led me to this equation provided by :

I used the equation:

$g(u) = \hat{F}^{-1}(\phi(z_0 + \frac{z_0+z_u}{1-a(z_0+z_u)}))$

Lower bound $= g(\alpha/2)$, Upper bound $= g(1-\alpha/2)$ ϕ is the standard normal cdf $\hat{F}$ is the bootstrap cdf $z_0 = \phi^{-1}F(\hat{\theta})$, quantile of mean of theta_boot <= theta_hat $z_u = \phi^{-1}(u)$, quantiles of our desired bounds

$\hat{a} = \frac{1}{6}\frac{\sum_{i=1}^{n} I_i^3}{(\sum_{i=1}^{n} I_i^2)^{3/2}}$

$I_i = (n-1)(\hat{\theta} - \hat{\theta}_{-i})$ finite sample jacknife approx.

I implemented $g(u)$ with this code:

```
bca_CI <- function(theta_boot,theta_hat,m){

  alpha <- 0.05
  u <- c(alpha/2, 1-alpha/2)
  z0 <- qnorm(mean(theta_boot <= theta_hat))
  zu <- qnorm(u)
  a <- 0
  #because the sample size of our data set is so large, the computation
  #is too intensive for rstudio to conduct. With such a large data set a should
  #be close to zero anyways, so I elected to choose zero.

  u_adjusted <- pnorm(z0 + (z0+zu)/(1-a*(z0+zu)))
  #u_adjusted <- pnorm(z0 + (z0+zu)/(1-a_approx*(z0+zu)))
```

```
  conf_int <- quantile(theta_boot,u_adjusted)
  return(conf_int)

}
```

Originally, I did not have a set to zero. I used this function to approximate a: