STAT641HW1

Brandon Keck

Exercise 1.

```
# non-parametric bootstrap for theta
library(boot)
set.seed(123) # set seed for reproducibility
x <- rgamma(200, shape = 2, scale = 5) # original data

# non-parametric bootstrap
theta_boot <- function(x, indices) {
  mean(x[indices]) / 2
}

boot(data = x, statistic = theta_boot, R = 999)
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = x, statistic = theta_boot, R = 999)


Bootstrap Statistics :
    original        bias    std. error
t1* 4.578726 0.007596511   0.2201269
```

```
# parametric bootstrap for theta
library(boot)
set.seed(123) # set seed for reproducibility
x <- rgamma(200, shape = 2, scale = 5) # original data
```

```r
boot_theta <- function(x) mean(x) / 2

gamma_boot <- function(x, mle){
  rgamma(length(x), shape = 2, scale = mle)
}
(b <- boot(x, boot_theta, R = 1000, sim = "parametric", ran.gen = gamma_boot,
           mle = mean(x) / 2))
```

```
PARAMETRIC BOOTSTRAP


Call:
boot(data = x, statistic = boot_theta, R = 1000, sim = "parametric",
    ran.gen = gamma_boot, mle = mean(x)/2)


Bootstrap Statistics :
    original      bias    std. error
t1* 4.578726 0.005143687   0.2135533
```

```r
# mean(b$t) # bootstrap estimate of theta

# mean(b$t) - b$t0 # bootstrap bias estimate

# sd(b$t) # parametric bootstrap SE
```

Exercise 2.

```r
library(resampledata)
library(dplyr)
data("Bangladesh")
head(Bangladesh)
```

```
  Arsenic Chlorine Cobalt
1    2400      6.2   0.42
2       6    116.0   0.45
3     904     14.8   0.63
```

```
4     321      35.9   0.68
5    1280      18.9   0.58
6     151       7.8   0.35
```

```r
chl <- with(Bangladesh, Chlorine[!is.na(Chlorine)])
```

(a)

```r
# 95% confidence interval for mean chlorine level mu
library(bootstrap)
set.seed(123)

bsample <- bootstrap(chl, 5000, mean)
quantile(bsample$thetastar, probs = c(0.025, 0.975))
```

```
     2.5%     97.5% 
 54.24833 104.65179 
```

(b)

```r
boot.bangladesh <- function(data, indices){
  mean(data[indices])
}
set.seed(123)
(bangladesh.boot <- boot(chl, boot.bangladesh, 5000))
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = chl, statistic = boot.bangladesh, R = 5000)


Bootstrap Statistics :
    original      bias    std. error
t1* 78.08401 0.04634684     12.9565
```

```r
boot.ci(bangladesh.boot, index = 1, type = c("perc", "bca"))
```

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 5000 bootstrap replicates

CALL :
boot.ci(boot.out = bangladesh.boot, type = c("perc", "bca"),
    index = 1)

Intervals :
Level     Percentile            BCa
95%   ( 54.23, 104.66 )   ( 56.79, 109.13 )
Calculations and Intervals on Original Scale
```

```r
# bootstrap-t
pacman::p_load(boot, bootstrap)
# Statistic: mean + SE
mean_stat <- function(data, idx){
  d <- data[idx]
  est <- mean(d)
  se <- sd(d) / sqrt(length(d))
  c(est, se)
}

set.seed(123)
boot_bangladesh <- boot(chl, statistic = mean_stat, R = 5000)
boot.ci(boot_bangladesh, type = "stud")
```

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 5000 bootstrap replicates

CALL :
boot.ci(boot.out = boot_bangladesh, type = "stud")

Intervals :
Level    Studentized
95%   ( 54.46, 106.96 )
Calculations and Intervals on Original Scale
```

All four intervals are similar towards the lower end. However, inspecting the BCa interval is shifted upward and has a larger endpoint compared to the other intervals. I would report the

BCa interval because it has second order accuracy and corrects for both bias and skewness in the bootstrap distribution.