Homework 4 (STAT 5860)

Eric Pettengill

Instructions:

1. Download the `Homework4.Rmd` file from the course Elearning.
2. Open `Homework4.Rmd` in RStudio.
3. Replace the *"Your Name Here"* text in the `author` with your own name.
4. Write your answer to each problem by editing `Homework4.Rmd`.
5. After you finish all the problems, click `Knit to PDF` to create a pdf file. Upload your pdf file to Homework 4 Dropbox in the course Elearning.

Set the seed number

```
set.seed(328)
```

Problem 1. Consider the data set `scor`. The data set is available after loading the `bootstrap` package. The data set consists test score on 88 students who took examinations in five subjects. The first two tests (mechanics, vectors) were closed book and the last three tests (algebra, analysis, statistics) were open book. Each row of the data frame is a set of scores for the ith student.

(a) First obtain each of the following sample corrleations: $\hat{\rho}_{12} = \hat{\rho}(\text{mec}, \text{vec})$, $\hat{\rho}_{35} = \hat{\rho}(\text{alg}, \text{sta})$, and $\hat{\rho}_{45} = \hat{\rho}(\text{ana}, \text{sta})$.

```
library(bootstrap)
library(boot)

rho12 <- cor(scor$mec, scor$vec)
rho35 <- cor(scor$alg, scor$sta)
rho45 <- cor(scor$ana, scor$sta)

c(rho12, rho35, rho45)
```

```
## [1] 0.5534052 0.6647357 0.6071743
```

(b) Obtain bootstrap estimates of standard errors for each of the sample correlation in (a). Use 2000 bootstrap samples.

```
correlation12 <- function(x, index){
    cor(x[index,1], x[index,2])
}

correlation35 <- function(x, index){
    cor(x[index,3], x[index,5])
}

correlation45 <- function(x, index){
    cor(x[index,4], x[index,5])
}

cor12 <- boot(data = scor, statistic = correlation12, R = 2000)
cor35 <- boot(data = scor, statistic = correlation35, R = 2000)
cor45 <- boot(data = scor, statistic = correlation45, R = 2000)

list(cor12, cor35, cor45)
```

```
## [[1]]
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = scor, statistic = correlation12, R = 2000)
##
##
## Bootstrap Statistics :
##      original       bias    std. error
## t1* 0.5534052 -0.002763341  0.07504647
##
## [[2]]
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = scor, statistic = correlation35, R = 2000)
##
##
## Bootstrap Statistics :
##      original       bias    std. error
## t1* 0.6647357 -0.002022065  0.06036049
##
## [[3]]
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = scor, statistic = correlation45, R = 2000)
##
##
## Bootstrap Statistics :
##      original       bias    std. error
## t1* 0.6071743 -0.000304164  0.06879693
```

(c) Obtain the jackknife estimates of standard errors for each of the sample correlation in (a).

```
n <- nrow(scor)
cor.jack <- matrix(0, n, ncol = 3)
for(i in 1:n){
    cor.jack[i,1] <- cor(scor[-i,1], scor[-i,2])
    cor.jack[i,2] <- cor(scor[-i,3], scor[-i,5])
    cor.jack[i,3] <- cor(scor[-i,4], scor[-i,5])
}

head(cor.jack)
```

```
##           [,1]      [,2]      [,3]
## [1,] 0.5248370 0.6526963 0.5954873
## [2,] 0.5397002 0.6398614 0.5920914
## [3,] 0.5348151 0.6475054 0.5966798
## [4,] 0.5471765 0.6582374 0.5962185
## [5,] 0.5471728 0.6583760 0.5988812
## [6,] 0.5500789 0.6503933 0.5988343
```

```
jack.se <- function(data){
  sqrt(((length(data)-1)/length(data))*sum((data - mean(data))^2))
}

apply(cor.jack, 2, jack.se)
```

```
## [1] 0.07752814 0.06059588 0.06918566
```

(d) Comptue 95% percentile bootstrap confidence intervals and BC_a bootstrap confidence intervals for each of the sample correlation in (a). Use 2000 bootstrap samples.

```
boot.ci(cor12, type = "bca")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = cor12, type = "bca")
##
## Intervals :
## Level       BCa
## 95%   ( 0.3968,  0.6902 )
## Calculations and Intervals on Original Scale
```

```
boot.ci(cor12, type = "perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = cor12, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.3911,  0.6884 )
## Calculations and Intervals on Original Scale
```

```
boot.ci(cor35, type = "bca")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = cor35, type = "bca")
##
```

```
## Intervals :
## Level       BCa
## 95%   ( 0.5257,  0.7634 )
## Calculations and Intervals on Original Scale
```

```r
boot.ci(cor35, type= "perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = cor35, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.5341,  0.7700 )
## Calculations and Intervals on Original Scale
```

```r
boot.ci(cor45, type = "bca")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = cor45, type = "bca")
##
## Intervals :
## Level       BCa
## 95%   ( 0.443,  0.716 )
## Calculations and Intervals on Original Scale
```

```r
boot.ci(cor45, type = "perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = cor45, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.4650,  0.7274 )
## Calculations and Intervals on Original Scale
```

Problem 2. In this problem, we will design two simulation studies to check the performance of bootstrap estimate of standard error of sample mean. First, we will compare the bootstrap estimate of standard error of sample mean to the exact standard error of sample mean on diffrent sample sizes. Next, we will compare the bootstrap estimate of standard error of sample mean to the exact standard error of sample mean on diffrent distributions of sample. (Hint: exact standard error of sample mean is $\sigma/\sqrt{n}$ where σ is a standard deviation.)

(a) Use the `rnorm()` function to genreate random samples of size 10, 50, and 100 from the normal distribution with mean 0 and standard deviation 2. Find the bootstrap estimate of standard error of sample mean for each sample size with 2000 bootstrap samples. Repeat this $R = 100$ times and calculate the following root-mean-squared error (RMSE) to compare the peformance of bootstrap estimate of standard error of sample mean on differnt sample sizes. Make a comment on your simulation reuslt.

$$\text{RMSE} = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\text{SE}(\bar{x}) - \widehat{\text{SE}}_r(\bar{x}^*)\right)^2}$$

```r
mean.fun <- function(data, index) {
  mean(data[index])
}

norm10.sd <- replicate(100, {
  obj <- boot(data = rnorm(10, mean = 0, sd = 2), statistic = mean.fun, R = 2000)
  samp <- obj$t
  sd(samp)
})

norm50.sd <- replicate(100, {
  obj <- boot(data = rnorm(50, mean = 0, sd = 2), statistic = mean.fun, R = 2000)
  samp <- obj$t
  sd(samp)
})

norm100.sd <- replicate(100, {
  obj <- boot(data = rnorm(100, mean = 0, sd = 2), statistic = mean.fun, R = 2000)
  samp <- obj$t
  sd(samp)
})

head(data.frame(norm10.sd, norm50.sd, norm100.sd))
```

```
##   norm10.sd norm50.sd norm100.sd
## 1 0.5674244 0.3180487  0.2101218
## 2 0.3396777 0.2860573  0.1985506
## 3 0.5115942 0.3162266  0.1844000
## 4 0.6847466 0.2870810  0.2033431
## 5 0.4065980 0.2635787  0.2093606
## 6 0.7046960 0.2788271  0.1979728
```

```r
rmse <- function(boot.sd, sigma){
  sqrt(sum((sigma/sqrt(length(boot.sd)) - boot.sd)^2)/length(boot.sd))
}

rmse(norm10.sd, 2)
```

```
## [1] 0.3868348
```

```r
rmse(norm50.sd, 2)
```

```
## [1] 0.08313744
```

```r
rmse(norm100.sd, 2)
```

```
## [1] 0.0154076
```

As we can see above, as the sample size of the original normal sample generated increases, the RMSE decreases.

(b) Use the `rnorm()` function to genreate a random sample of size 100 from the standard normal distribution. Also, use the `rchisq()` function to generate a random sample of size 100 from the chi-squared distribtuion with degrees of freedom 3. Find the bootstrap estimate of standard error of sample mean for each distribution with 2000 bootstrap samples. Repeat this $R = 100$ times and calculate the root-mean-squared error (RMSE) to compare the peformance of bootstrap estimate of standard error of sample mean on differnt sample distributions. Make a comment on your simulation reuslt. (Hint: the variance of chi-squared distribution is equal to $2k$ where k is degrees of freedom.)

```r
mean.fun <- function(data, index) {
  mean(data[index])
}

normx.sd <-  replicate(100, {
  obj <- boot(data = rnorm(100), statistic = mean.fun, R = 2000)
  samp <- obj$t
  sd(samp)
})

chisq.sd <-  replicate(100, {
  obj <- boot(data = rchisq(100, df=3), statistic = mean.fun, R = 2000)
  samp <- obj$t
  sd(samp)
})

rmse <- function(boot.sd, sigma){
  sqrt(sum((sigma/sqrt(length(boot.sd)) - boot.sd)^2)/length(boot.sd))
}

rmse(normx.sd, 1)
```

```
## [1] 0.008360907
```

```r
rmse(chisq.sd, sqrt(6))
```

```
## [1] 0.03100793
```

As we can see above, the RMSE of the normal distribution is smaller than that of the chi-square distribution. Meaning the bootstrap estimates are closer to the overall mean of the bootstrap estimates while the chi-square bootstraps have a higher RMSE, meaning the estimates are further from the overall mean of the bootstrap estimates.