hw3 #6

Akhil Havaldar

10/11/2022

Question 6

```
library(ISLR2)
```

```
## Warning: package 'ISLR2' was built under R version 4.1.3
```

```
dat <- Wage

#a)
samp.mean <- mean(dat$wage)
samp.sd <- sd(dat$wage)
print(samp.mean)            # 111.7036
```

```
## [1] 111.7036
```

```
print(samp.sd)              # 41.7286
```

```
## [1] 41.7286
```

```
#b)
samp.se <- samp.sd/(sqrt(nrow(dat)))
print(samp.se)               # 0.7618564
```

```
## [1] 0.7618564
```

```
#c)
t.mult <- qt(0.975, df=2999)
lower <- samp.mean - (t.mult * samp.se)
upper <- samp.mean + (t.mult * samp.se)
ci <- c(lower, upper)
print(ci)                   # (110.2098, 113.1974)
```

```
## [1] 110.2098 113.1974
```

```r
#d.i)
set.seed(4630)
boot.10 <- replicate(10, sample(dat$wage, replace=TRUE))

#d.ii)
mean.boot.10 <- array(0,10)    # initializing array
for (i in 1:10){
  m <- mean(boot.10[,i])       # finding the mean for each boot
  mean.boot.10[i] <- m
}

#d.iii)
se.boot.10 <- sd(mean.boot.10)
print(se.boot.10)            # 0.89869
```

```
## [1] 0.7720473
```

d.iv)

- The two values are relatively similar. In this instance, I would increase the number of bootstrap samples to improve the SE.

```r
#e.100)
set.seed(4630)
boot.100 <- replicate(100, sample(dat$wage, replace=TRUE))
mean.boot.100 <- array(0,100)    # initializing array
for (i in 1:100){
  m <- mean(boot.100[,i])       # finding the mean for each boot
  mean.boot.100[i] <- m
}

se.boot.100 <- sd(mean.boot.100)
print(se.boot.100)            # 0.8897307
```

```
## [1] 0.8897307
```

```r
#e.1000)
set.seed(4630)
boot.1000 <- replicate(1000, sample(dat$wage, replace=TRUE))

mean.boot.1000 <- array(0,1000)    # initializing array
for (i in 1:1000){
  m <- mean(boot.1000[,i])       # finding the mean for each boot
  mean.boot.1000[i] <- m
}

se.boot.1000 <- sd(mean.boot.1000)
print(se.boot.1000)            # 0.8021534
```

```
## [1] 0.8059771
```

```r
#e.10000)
set.seed(4630)
boot.10000 <- replicate(10000, sample(dat$wage, replace=TRUE))

mean.boot.10000 <- array(0,10000)    # initializing array
for (i in 1:10000){
  m <- mean(boot.10000[,i])       # finding the mean for each boot
  mean.boot.10000[i] <- m
}

se.boot.10000 <- sd(mean.boot.10000)
print(se.boot.10000)          # 0.7700513
```

```
## [1] 0.7650972
```

- When looking at the different bootstrap SE in relation to the sample SE, we can see that the bootstrap estimate gets closer and closer to the true value in (6b). With B=10000, the SE is only 0.01 off the actual SE.

f)

- Since the values of SE, as the bootstrap sizes get larger, converge to the estimation calculated in (6b) we know that we calculated the SE in (6b) correctly.

```r
#g)
set.seed(4630)
delta <- mean.boot.10000 - samp.mean      # calculating delta
conf <- quantile(delta, c(0.025,0.975))       # calculating 95% CI
ci <- c(samp.mean-conf[2], samp.mean-conf[1])
print(ci)         # (110.1761 ,113.2084)
```

```
##    97.5%     2.5%
## 110.1761 113.2084
```

- We can see that the bootstrap CI is very close to the CI we found from the sample.