5.1

Consider the samples 1-6. Use a six-sided die to obtain three different bootstrap samples and their corresponding means.

```
pop <- seq(from = 1, to = 6, by = 1)

n <- 6

s1 <- mean( sample(pop, n, replace = T) )
s2 <- mean( sample(pop, n, replace = T) )
s3 <- mean( sample(pop, n, replace = T) )
```

$\bar{x}_1^* = 3.3333333, \bar{x}_2^* = 3.1666667, \bar{x}_3^* = 4$

5.2

Consider the samples 1, 3, 4, and 6 from some distribution.

```
pop <- c(1, 3, 4, 6)

samples <- permutations(n = 4, r = 4, pop, repeats.allowed = T)
```

a.) For one random bootstrap sample, find the probability that the mean is one.

```
means <- apply(samples, 1, mean)

p <- mean( means == 1 )
```

Probability: **0.39%**

b.) For one random bootstrap sample, find the probability that the maximum is 6.

```
maxes <- apply(samples, 1, max)

p <- mean( maxes == 6 )
```

Probability: **68.36%**

c.) For one random bootstrap sample, find the probability that exactly two elements in the sample are less than 2.

```
lt2 <- apply(t(apply(samples, 1, function(x) { x < 2})), 1, sum)

p <- mean( lt2 == 2 )
```

Probability: **21.09%**

5.3

Consider the sample 1-3.

a.) List all the (ordered) bootstrap samples from this sample. How many are there?

```
samples <- permutations(n = 3, r = 3, 1:3, repeats.allowed = T)

n <- nrow(samples)
```

Samples: = 3^3 = **27**

b.) How many unordered bootstrap samples are there? For example, {1, 2, 2} and {2, 1, 2} are considered to be the same.

```
samples <- combinations(n = 3, r = 3, 1:3, repeats.allowed = T)

n <- nrow(samples)

assertthat::are_equal(n, choose(3 + 3 - 1, 3))
```

```
[1] TRUE
```

Samples: = $\binom{5}{3}$ = **10**

c.) How many ordered bootstrap samples have one occurrence of 1 and two occurences of 3?

```
samples <- permutations(n = 3, r = 3, 1:3, repeats.allowed = T)

n.ones <- apply(t(apply(samples, 1, FUN = function(x) { x == 1 })), 1, function(x) sum(x) )
n.threes <- apply(t(apply(samples, 1, FUN = function(x) { x == 3 })), 1, function(x) sum(x) )

sum((n.ones == 1 & n.threes == 2) == T)
```

```
[1] 3
```

Is this the same number of bootstrap samples that have each of 1, 2 and 3 occuring exactly once?

```
n.ones <- apply(t(apply(samples, 1, FUN = function(x) { x == 1 })), 1, function(x) sum(x) )
n.twos <- apply(t(apply(samples, 1, FUN = function(x) { x == 2 })), 1, function(x) sum(x) )
n.threes <- apply(t(apply(samples, 1, FUN = function(x) { x == 3 })), 1, function(x) sum(x) )

sum((n.ones == 1 & n.twos == 1 & n.threes == 1) == T)
```

```
[1] 6
```

No, 3 != 6.

d.) Is the probability of obtaining a bootstrap sample with one 1 and two 3's the same as the probabiliity of obtaining a bootstrap sample with each of 1, 2 and 3 occuring exactly once?

```
( sum((n.ones == 1 & n.threes == 2)) / n ) == ( sum((n.ones == 1 & n.twos == 1 &  n.threes == 1
```

```
[1] FALSE
```

No, 3% and 6% chances respectfully.

5.4

Consider the samples 1, 3, 3, and 5 from some distribution.

```r
samples <- c(1, 3, 3, 5)
```

a.) How many bootstrap samples are there?

```r
boot <- permutations(n = 3, r = 4, v = samples, repeats.allowed = T)

n <- nrow(boot)
```

3 unique items to pick from, 4 places to put each item.

Number of permutations: 3^4 = *81*

b.) List the distinct bootstrap samples assuming order does not matter.

```r
combinations(n = 3, r = 4, v = samples, repeats.allowed = T)
```

```
      [,1] [,2] [,3] [,4]
 [1,]    1    1    1    1
 [2,]    1    1    1    3
 [3,]    1    1    1    5
 [4,]    1    1    3    3
 [5,]    1    1    3    5
 [6,]    1    1    5    5
 [7,]    1    3    3    3
 [8,]    1    3    3    5
 [9,]    1    3    5    5
[10,]    1    5    5    5
[11,]    3    3    3    3
[12,]    3    3    3    5
[13,]    3    3    5    5
[14,]    3    5    5    5
[15,]    5    5    5    5
```

```r
choose(4 + 3 - 1, 4)
```

```
[1] 15
```

5.5

We determine the number of distinct bootstrap samples from a finite set.

a.) A bakery sells five types of cookies: sugar, chocolate chip, oatmeal, peanut butter, and ginger snap. Show that the number of ways to order 5 cookies is $\binom{9}{5}$

Unordered samping with replacement: $\binom{n+k-1}{k}, n = 5, k = 5$

```r
choose(5 + 5 - 1, 5)
```

```
[1] 126
```

b.) Show that the number of sets of size n (order does not matter) drawn with replacement from the (distinct) $a_1, a_2, \ldots, a_n$ is $\binom{2n-1}{n}$

Conclude that the number of distinct bootstrap samples from the set $[a_1, a_2, \ldots, a_n]$ is $\binom{2n-1}{n}$

5.6

Let $k_1, k_2, \ldots, k_n$ denote non-negative integers satisfying $k_1 + k_2 + \ldots + k_n = n$, and suppose the elements in the set $a_1, a_2, \ldots, a_n$ are distinct.

a.) Show that the number of bootstrap samples with k_1 occurrences of a_1, k_2 occurrences of $a_2, \ldots, k_n$ occurrences of a_n is $\binom{n}{k_1, k_2, \ldots, k_n}$

b.) Compute the probability that a randomly drawn bootstrap sample will have k_i occurrences of $a_i, i = 1, 2, \ldots, n$

5.7

Refer to Example 5.4 and the remark at the end of the example.

a.) What might account for the fact that there were more missing values for the men who skateboarded in front of the male experimenter? How might this bias the outcome?

It could be that the approached skateboarders refused to participate in the study of performing tricks in front of other men.

b.) Why do you suppose it was important that the two experimenters were blinded to the purpose of the study?

The female could have flurted or otherwise influenced skateboarders who were performing tricks if they knew the intent of the study.

5.8

Consider a population that has a normal distribution with mean $\mu = 36$, standard deviation $\sigma = 8$.

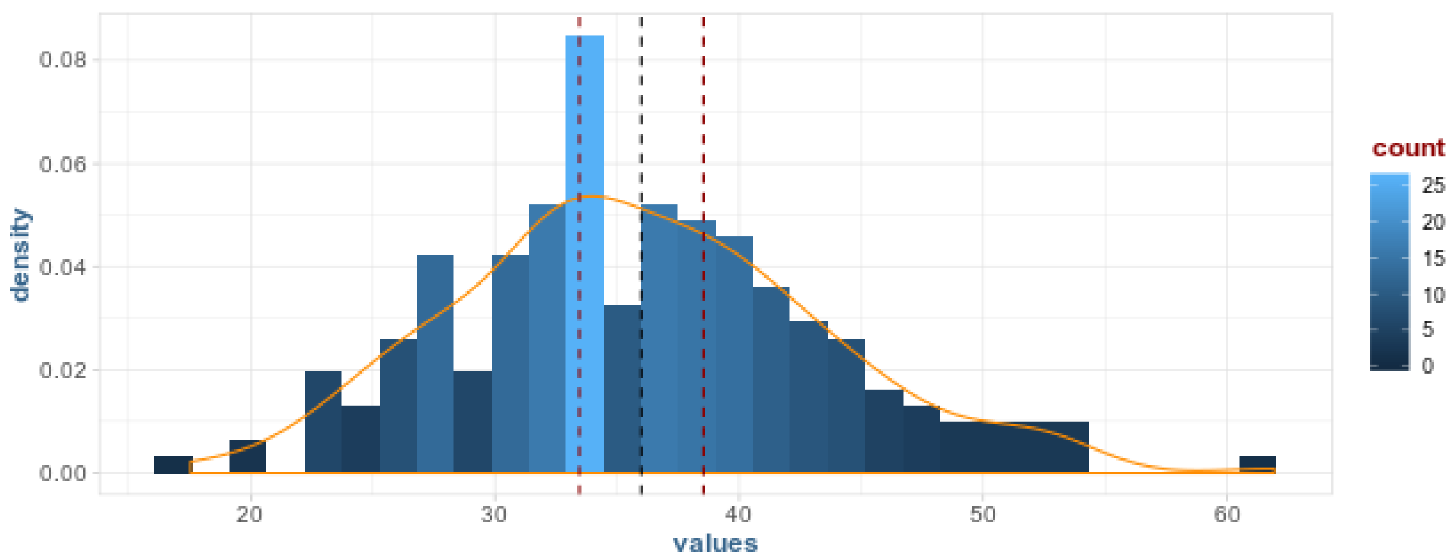
```r
mu <- 36; sigma <- 8; n <- 200

se <- mu / sqrt(n)
```

a.) The sampling distribution of $\bar{X}$ for samples of size 200 will have what mean, standard error, and shape?

$\mu = 36$, *SE = 36 / sqrt(200) = 2.5455844, shape will be approximately normal (CLT).*

b.) Use R to draw a random sample of size 200 from this population. Conduct EDA on your sample.

```
set.seed(123)

samp <- data.table(values = rnorm(200, mean = 36, sd = 8))

ggplot(samp, aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(values), color = "darkorange") +
   geom_vline(xintercept = mu, col = "black", lty = 2) +
   geom_vline(xintercept = mu - se, col = "darkred", lty = 2) +
   geom_vline(xintercept = mu + se, col = "darkred", lty = 2)
```



c.) Compute the bootstrap distribution for your sample, and note the bootstrap mean and standard error.

```
boot.fn <- function(data, index){
   mean(data[index]$values)
}

I <- 10e3

boot(samp, boot.fn, R = I) # boot pkg
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
```

```
boot(data = samp, statistic = boot.fn, R = I)


Bootstrap Statistics :
    original      bias    std. error
t1* 35.93144 0.006716095   0.5392066
```

```
cst.boot <- function(values, n, I = 10e3, alpha = 0.05) {
   bootstrap <- numeric(I)

   for(i in 1:I)
   {
      bootstrap[i] <- mean( sample(values, n, replace = T) )
   }

   observed <- mean(values)

   boot.mean <- mean(bootstrap)
   boot.bias <- boot.mean - observed
   boot.se <- sd(bootstrap)

   list(bootstrap = bootstrap,
         observed = observed,
         mean = boot.mean,
         bias = boot.bias,
         se = boot.se,
         conf = quantile(bootstrap, c(alpha/2, 1 - alpha/2)))
}
```

d.) Compare the bootstrap distribution to the theoretical sampling distribution by creating a table like Table 5.2:

```
n.200 <- cst.boot(samp$values, 200)

tbl <- data.table(Data = c("Population", "Sampling Distribution", "Sample", "Bootstrap Distribu
                  Mean = c(mu, mu, n.200$observed, n.200$mean),
                  SD = c(sigma, mu/sqrt(n), sd(samp$values), n.200$se))

pretty_kable(tbl, "Sampling Statistics")
```

Table 1: Sampling Statistics

Data	Mean	SD
Population	36.00	8.00
Sampling Distribution	36.00	2.55
Sample	35.93	7.55
Bootstrap Distribution	35.94	0.53

e.) Repeat for sample sizes $n = 50$ and $n = 10$. Carefully describe yyour observations about the effects of sample size on the bootstrap distribuiton.

```
n <- 50
samp <- data.table(values = rnorm(n, mean = mu, sd = sigma))
n.50 <- cst.boot(samp$values, n)

tbl <- data.table(Data = c("Population", "Sampling Distribution", "Sample", "Bootstrap Distribu
                  Mean = c(mu, sigma, n.50$observed, n.50$mean),
                  SD = c(sigma, mu/sqrt(n), sd(samp$values), n.50$se))

pretty_kable(tbl, "Sampling Statistics")
```

Table 2: Sampling Statistics

Data	Mean	SD
Population	36.00	8.00
Sampling Distribution	8.00	5.09
Sample	36.94	7.78
Bootstrap Distribution	36.96	1.09

```
n <- 10
samp <- data.table(values = rnorm(n, mean = mu, sd = sigma))
n.10 <- cst.boot(samp$values, 10)

tbl <- data.table(Data = c("Population", "Sampling Distribution", "Sample", "Bootstrap Distribu
                  Mean = c(mu, sigma, n.10$observed, n.10$mean),
                  SD = c(sigma, mu/sqrt(n), sd(samp$values), n.10$se))

pretty_kable(tbl, "Sampling Statistics")
```

Table 3: Sampling Statistics

Data	Mean	SD
Population	36.00	8.00
Sampling Distribution	8.00	11.38
Sample	32.42	10.58
Bootstrap Distribution	32.40	3.15

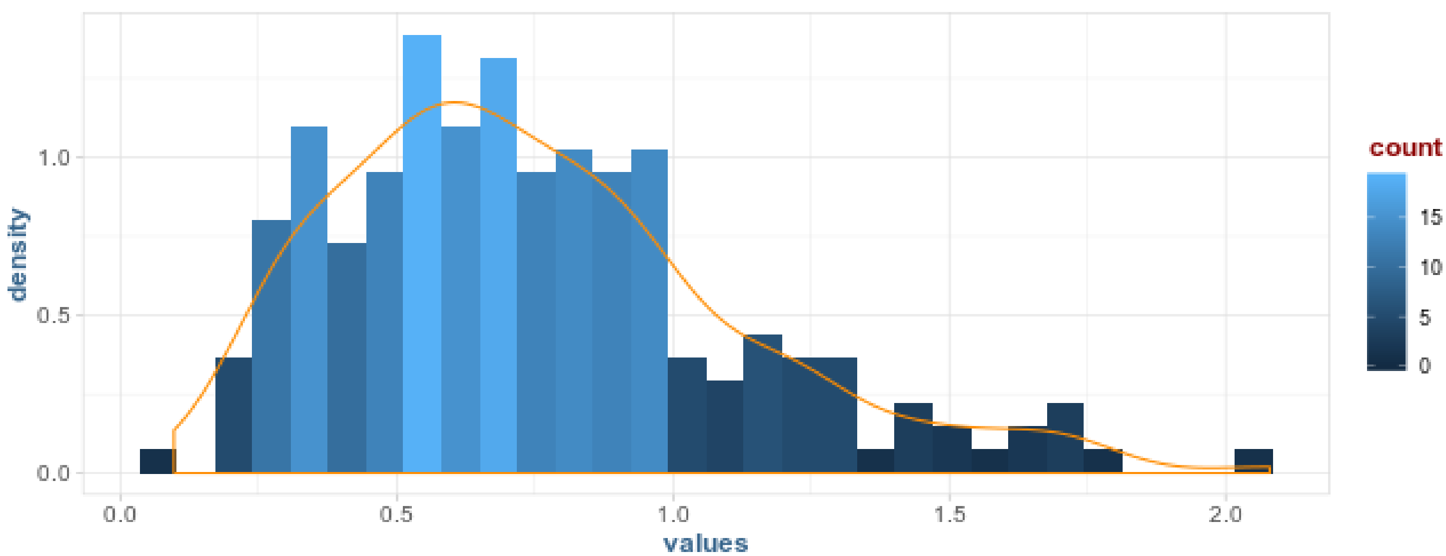
The center of the bootstrap distribution doesn't vary much with smaller n, however, confidence intervals (the sd of the bootstrap dist) vary wildy.

5.9

Consider a population that has a gamma distribution with parameters r = 5, $\lambda = 4$.

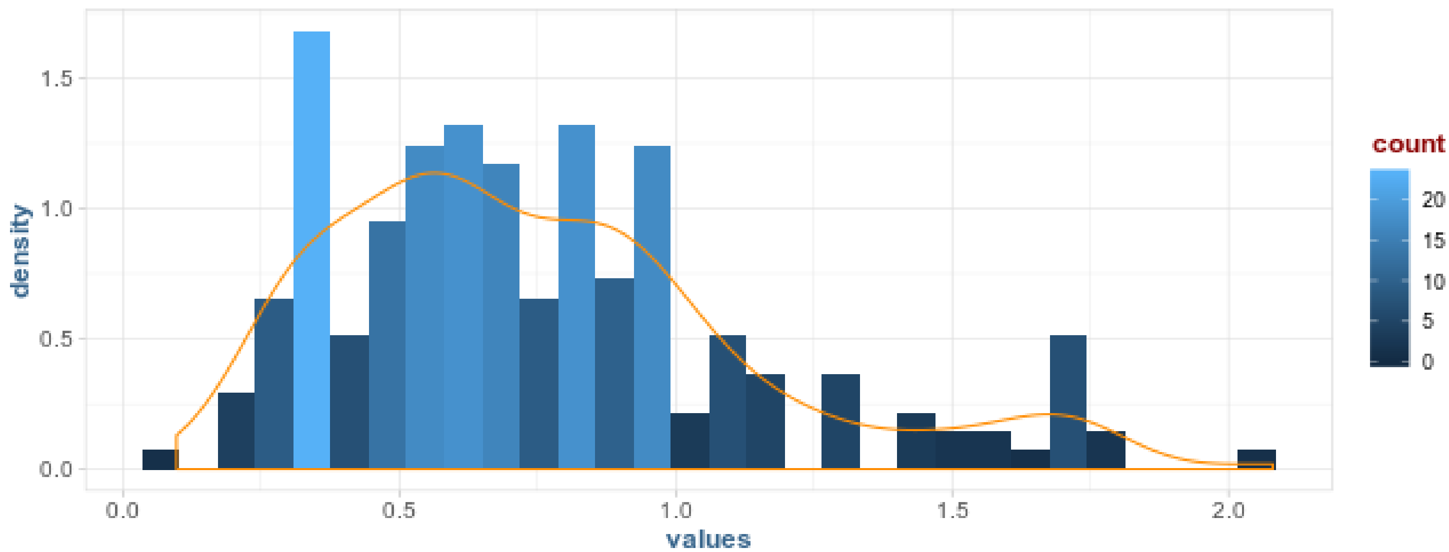
a.) Use simulation (with $n = 200$) to generate an approximate sampling distribution of the mean; plot and describe the distribution.

```
set.seed(123)

n <- 200; r <- 5; lambda <- 4; mu <- lambda/r

pop <- data.table(values = rgamma(n, shape = lambda, rate = r))

ggplot(pop, aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(values), col = "darkorange")
```



b.) Now, draw one random sample of size 200 from this population. Create a histogram of your sample, and find the mean and standard deviation.

```
samp <- data.table(values = sample(pop$values, n, replace = T))

ggplot(samp, aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(values), col = "darkorange")
```




```
xbar <- mean(samp$values); sd <- sd(samp$values)

xbar; sd
```

```
[1] 0.7571211
```

```
[1] 0.3859094
```

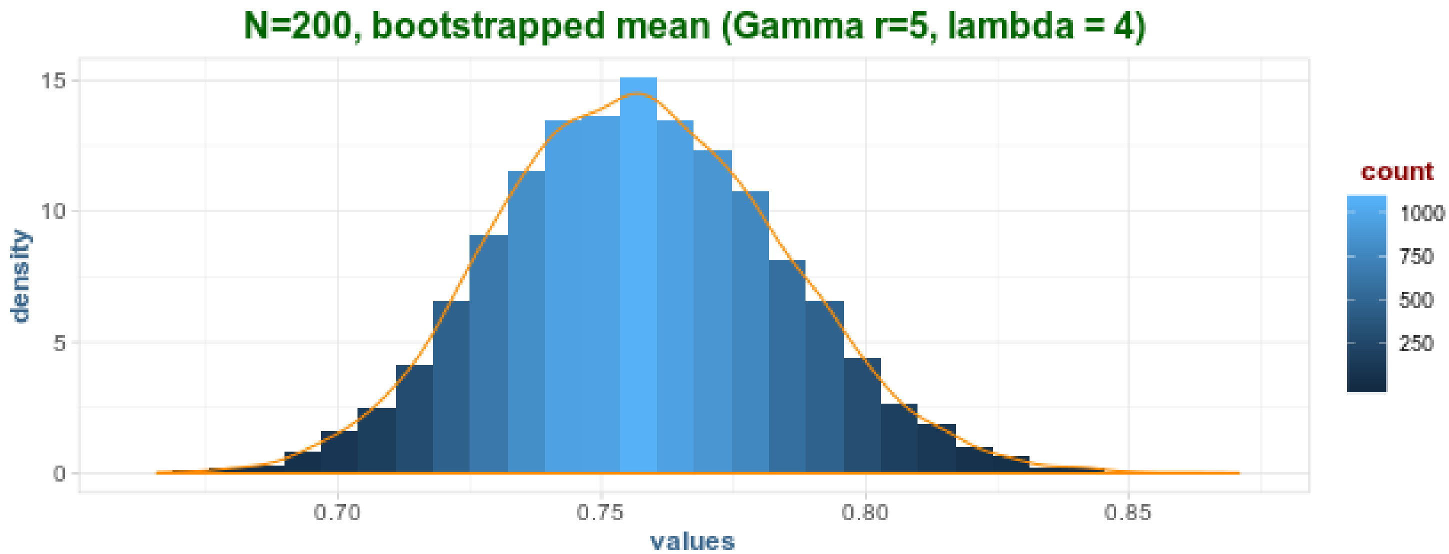
c.) Compute the bootstrap distribution of the mean for you sample, plot it, and note the bootstrap mean and standard error.

```
I <- 10e3

boot.fn <- function(data, index) {
   mean(data[index]$values)
}

boot(samp, boot.fn, R = I)
```

```


ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = samp, statistic = boot.fn, R = I)


Bootstrap Statistics :
     original        bias    std. error
t1* 0.7571211 0.0003439594  0.02732601
```

```
n.200 <- cst.boot(samp$values, n)
```

```r
ggplot(data.table(values = n.200$bootstrap), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(values), col = "darkorange") +
   labs(title = "N=200, bootstrapped mean (Gamma r=5, lambda = 4)")
```



d.) Compare the bootstrap distribution to the approximate theoretical sampling distribution by creating a table like Table 5.2.

```r
tbl <- data.table(Data = c("Population", "Sampling Distribution", "Sample", "Bootstrap Distribu
                  Mean = c(mu, sd(samp$values), n.200$observed, n.200$mean),
                  SD = c(mu, mu/sqrt(n), sd(samp$values), n.200$se))

pretty_kable(tbl, "Sampling Statistics")
```

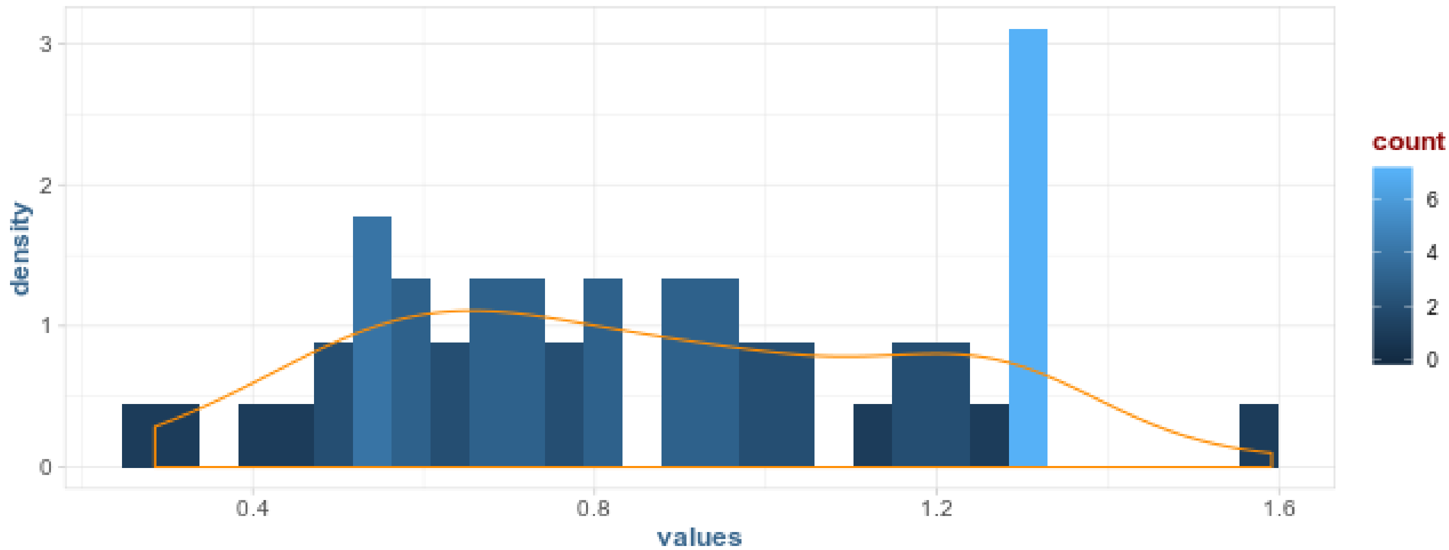
Table 4: Sampling Statistics

Data	Mean	SD
Population	0.80	0.80
Sampling Distribution	0.39	0.06
Sample	0.76	0.39
Bootstrap Distribution	0.76	0.03

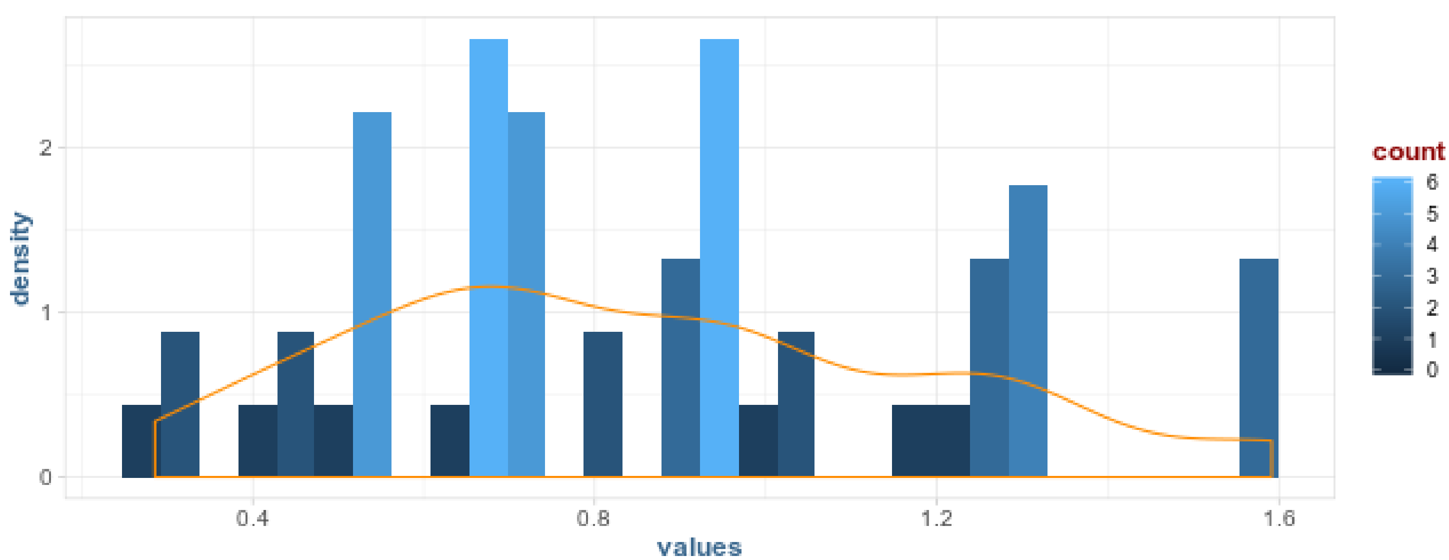
e.) Repeat (a-e) for sample sizes of n = 50, and n = 10. Describe carefully your observations about the effects of sample size on the bootstrap distribution.

```r
n <- 50

pop <- data.table(values = rgamma(n, shape = lambda, rate = r))

ggplot(pop, aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
```

```
  geom_density(aes(values), col = "darkorange")
```




```
samp <- data.table(values = sample(pop$values, n, replace = T))

ggplot(samp, aes(values)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
  geom_density(aes(values), col = "darkorange")
```

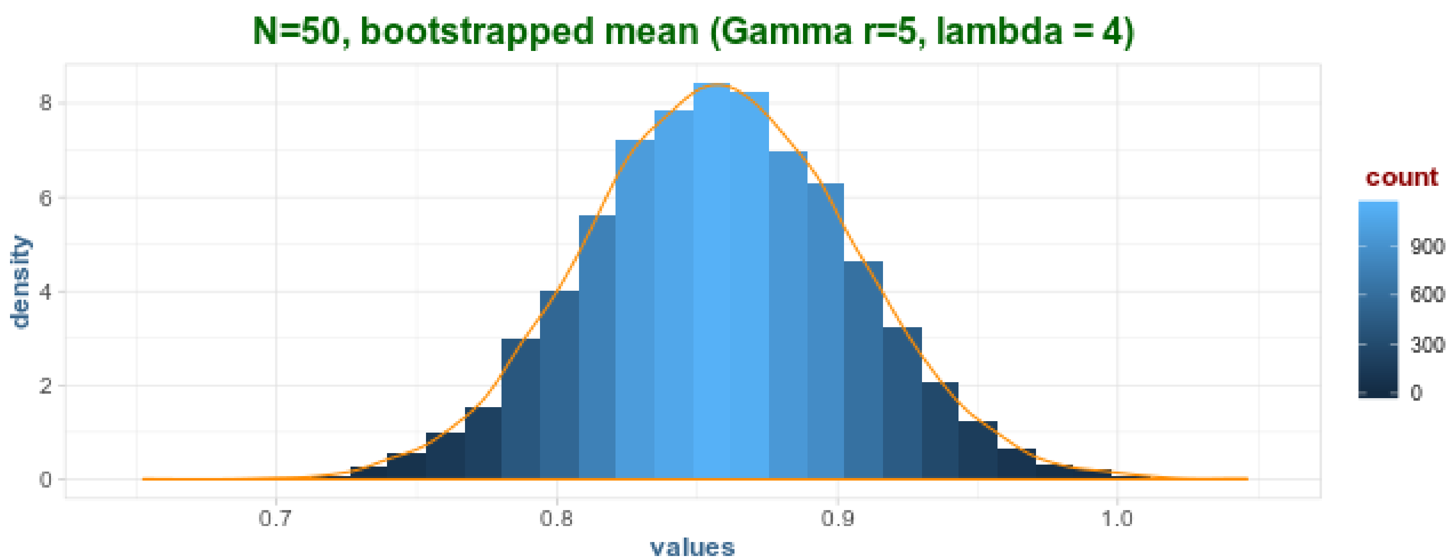



```
xbar <- mean(samp$values); sd <- sd(samp$values)

xbar; sd
```

```
[1] 0.8573007
```

```
[1] 0.3391036
```

```r
n.50 <- cst.boot(samp$values, n)

ggplot(data.table(values = n.50$bootstrap), aes(values)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
  geom_density(aes(values), col = "darkorange") +
  labs(title = "N=50, bootstrapped mean (Gamma r=5, lambda = 4)")
```

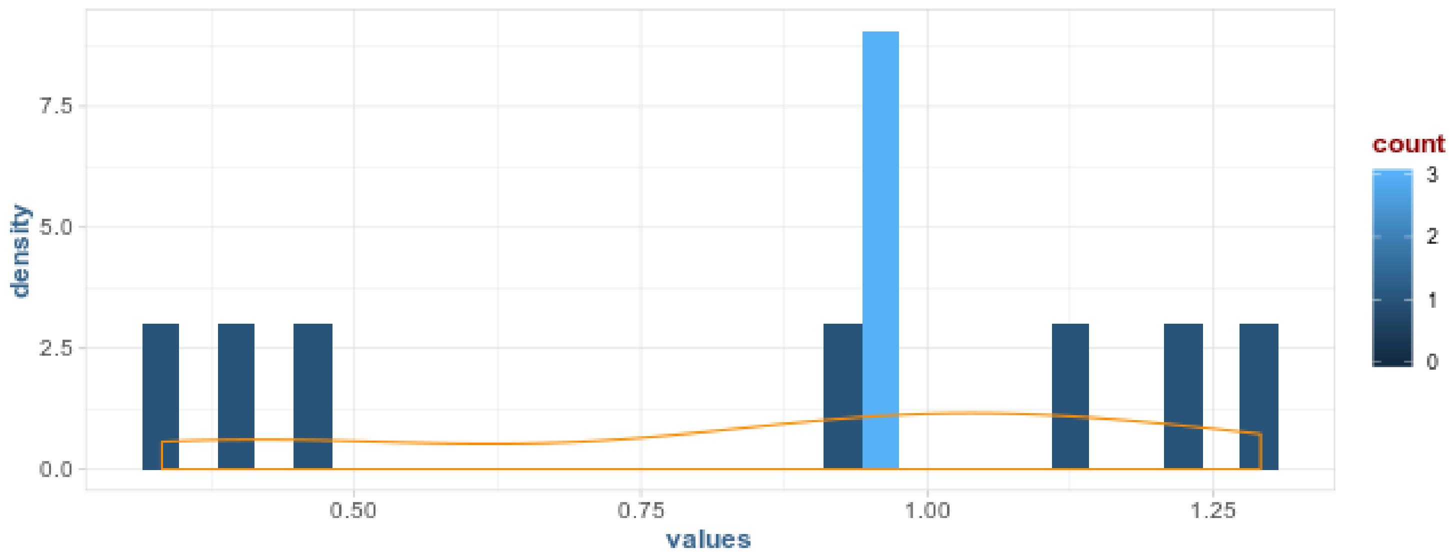



```r
tbl <- data.table(Data = c("Population", "Sampling Distribution", "Sample", "Bootstrap Distribu
                  Mean = c(mu, sd(samp$values), n.50$observed, n.50$mean),
                  SD = c(mu, mu/sqrt(n), sd(samp$values), n.50$se))

pretty_kable(tbl, "Sampling Statistics")
```

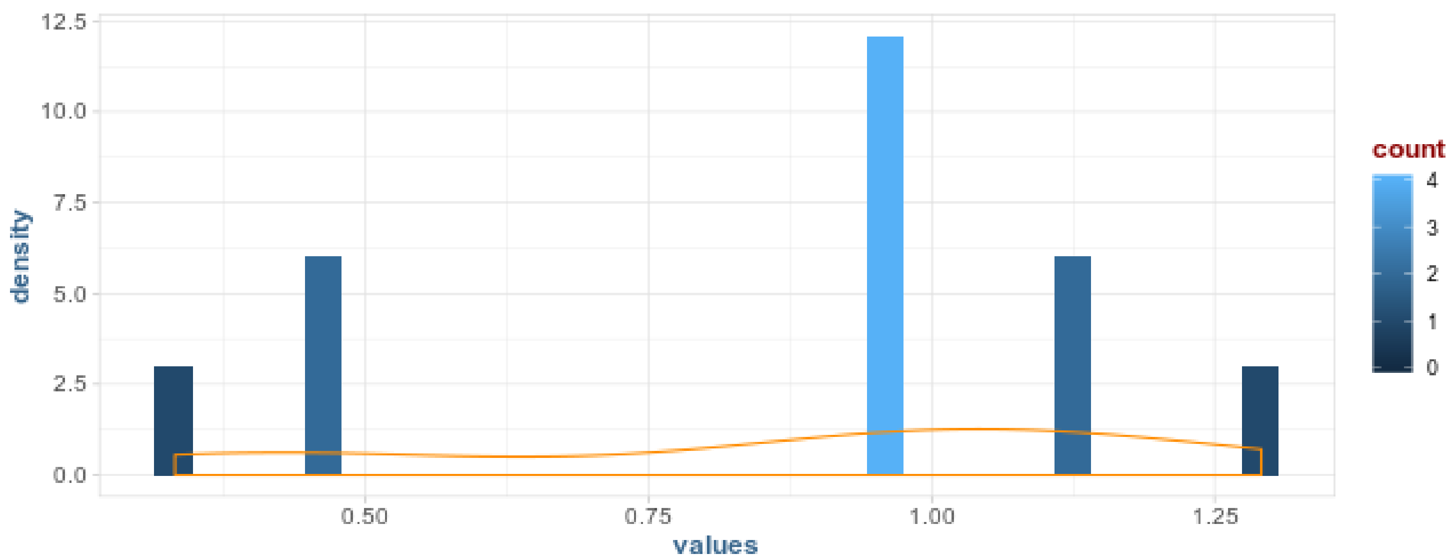
Table 5: Sampling Statistics

Data	Mean	SD
Population	0.80	0.80
Sampling Distribution	0.34	0.11
Sample	0.86	0.34
Bootstrap Distribution	0.86	0.05

```r
n <- 10

pop <- data.table(values = rgamma(n, shape = lambda, rate = r))

ggplot(pop, aes(values)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
  geom_density(aes(values), col = "darkorange")
```




```
samp <- data.table(values = sample(pop$values, n, replace = T))

ggplot(samp, aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(values), col = "darkorange")
```

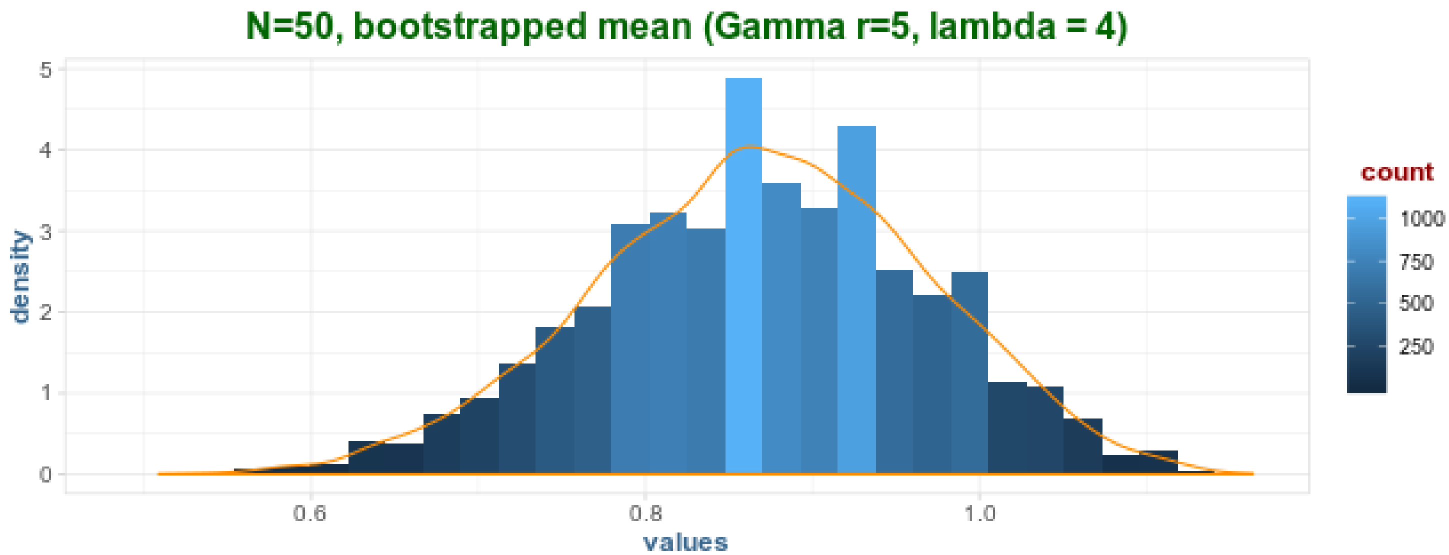



```
xbar <- mean(samp$values); sd <- sd(samp$values)

xbar; sd
```

```
[1] 0.8682445
```

```
[1] 0.3325161
```

```
n.50 <- cst.boot(samp$values, n)

ggplot(data.table(values = n.50$bootstrap), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(values), col = "darkorange") +
   labs(title = "N=50, bootstrapped mean (Gamma r=5, lambda = 4)")
```




```
tbl <- data.table(Data = c("Population", "Sampling Distribution", "Sample", "Bootstrap Distribu
                  Mean = c(mu, sd(samp$values), n.50$observed, n.50$mean),
                  SD = c(mu, mu/sqrt(n), sd(samp$values), n.50$se))

pretty_kable(tbl, "Sampling Statistics")
```

Table 6: Sampling Statistics

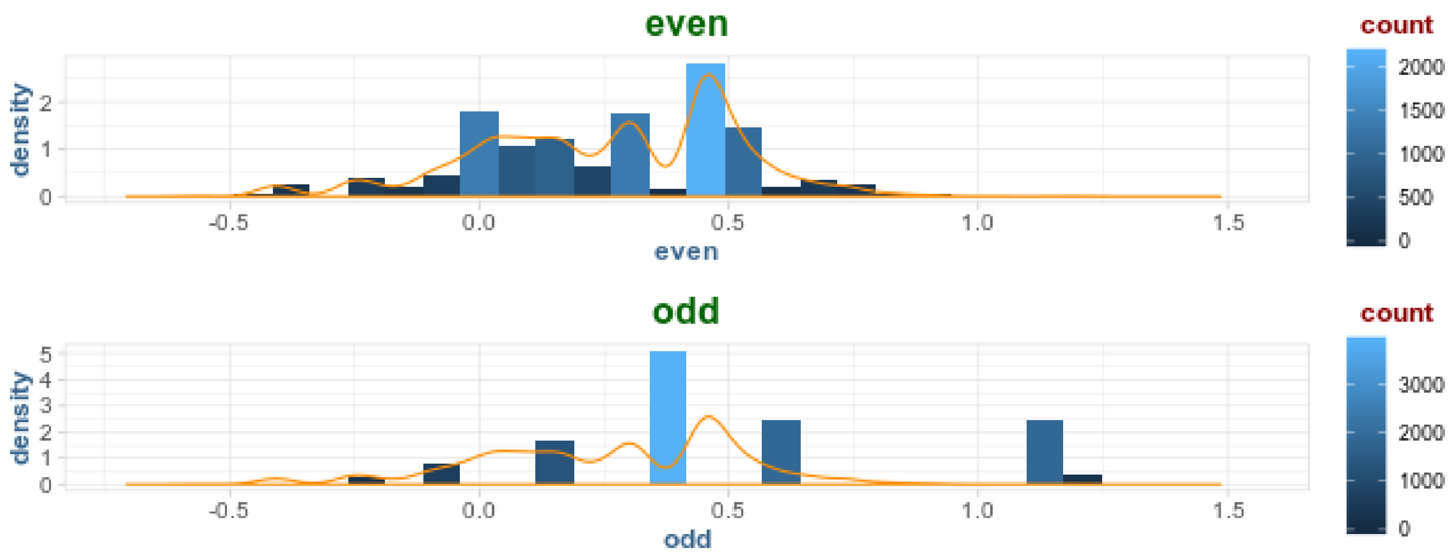
Data	Mean	SD
Population	0.80	0.80
Sampling Distribution	0.33	0.25
Sample	0.87	0.33
Bootstrap Distribution	0.87	0.10

5.10

We investigate the bootstrap distribution of the median. Create random sample of size n for various n and bootstrap the median. Describe the bootstrap distribution.

```
ne <- 14 # n even
no <- 15 # n odd
```

```r
wwe <- rnorm(ne) # draw samples of size ne
wwo <- rnorm(no) # draw random samples of size no

N <- 10^4

even.boot <- numeric(N) # save space
odd.boot <- numeric(N)

for(i in 1:N)
{
   x.even <- sample(wwe, ne, replace = T)
   x.odd <- sample(wwo, no, replace = T)

   even.boot[i] <- median(x.even)
   odd.boot[i] <- median(x.odd)
}

boot <- data.table(even = even.boot, odd = odd.boot)

p1 <- ggplot(boot, aes(even)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange") +
   labs(title = "even")

p2 <- ggplot(boot, aes(odd)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange") +
   labs(title = "odd")

gridExtra::grid.arrange(p1, p2)
```

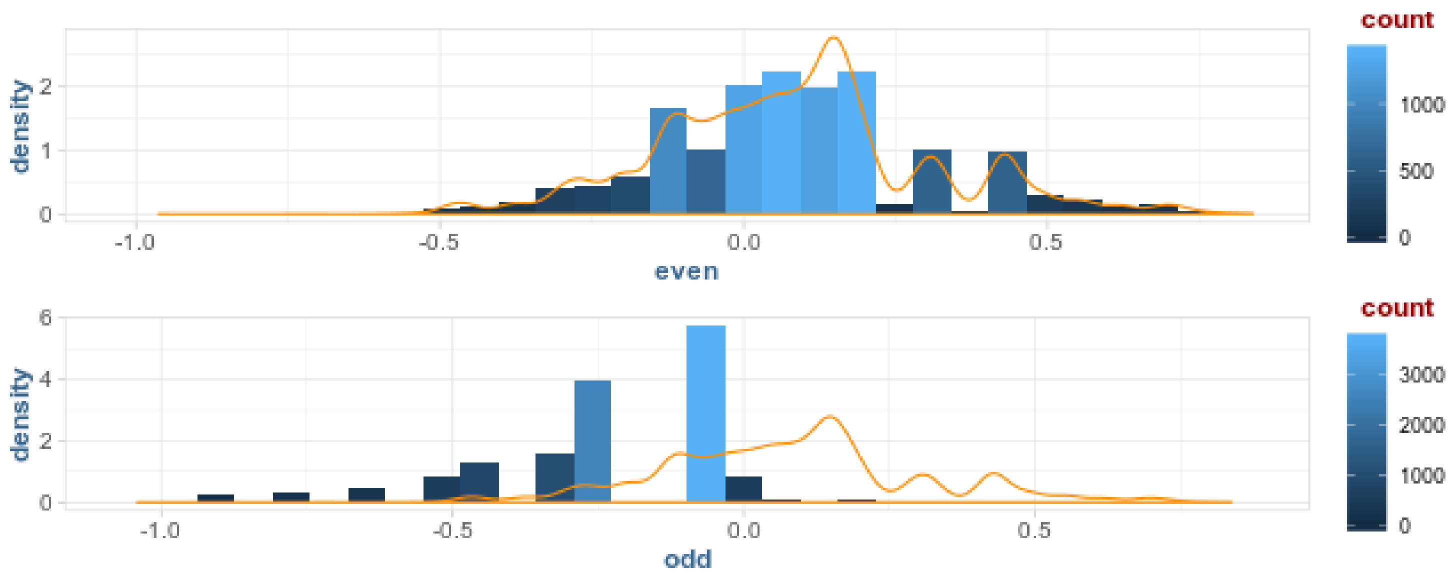


Change the sample sizes to 36 and 37; 200 and 201; and 10,000 and 10,001.

Note the similarities/dissimalarities, trends, and so on. Why does the parity of the sample size matter? (*Note: Adjust the x limits in the plots as needed.*)

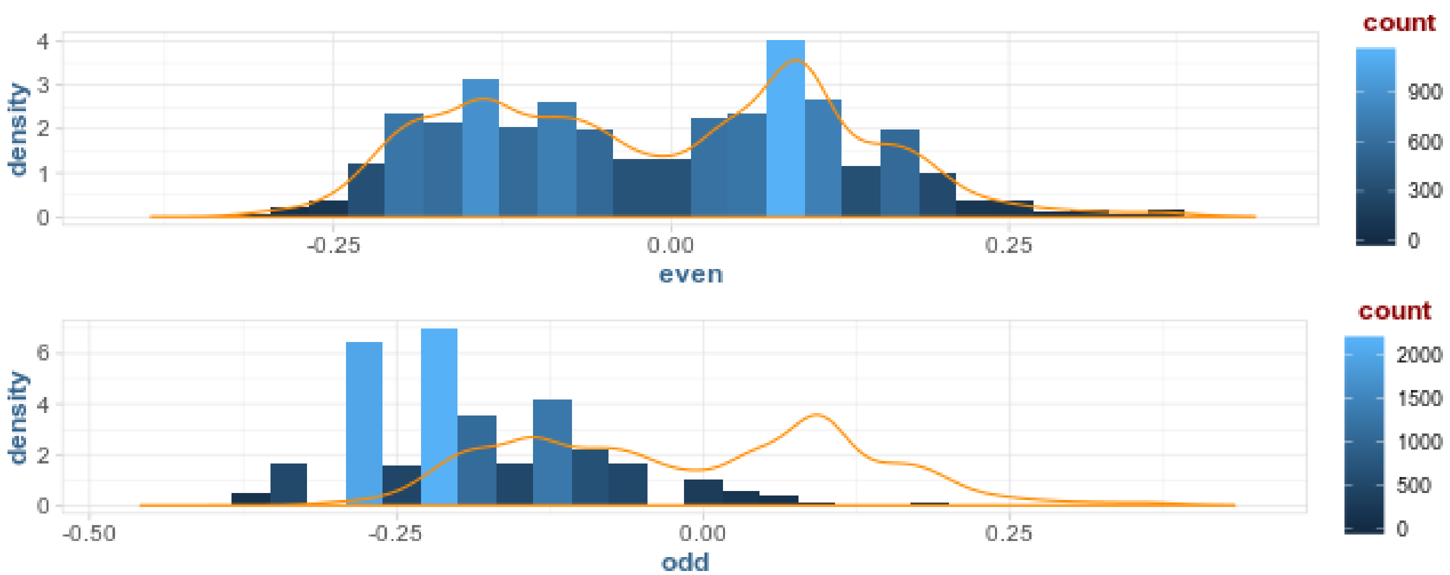
```
ne <- 36 # n even
no <- 37 # n odd

wwe <- rnorm(ne) # draw samples of size ne
wwo <- rnorm(no) # draw random samples of size no

N <- 10^4

even.boot <- numeric(N) # save space
odd.boot <- numeric(N)

for(i in 1:N)
{
   x.even <- sample(wwe, ne, replace = T)
   x.odd <- sample(wwo, no, replace = T)

   even.boot[i] <- median(x.even)
   odd.boot[i] <- median(x.odd)
}

boot <- data.table(even = even.boot, odd = odd.boot)

p1 <- ggplot(boot, aes(even)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange")
```

```
p2 <- ggplot(boot, aes(odd)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange")

gridExtra::grid.arrange(p1, p2)
```



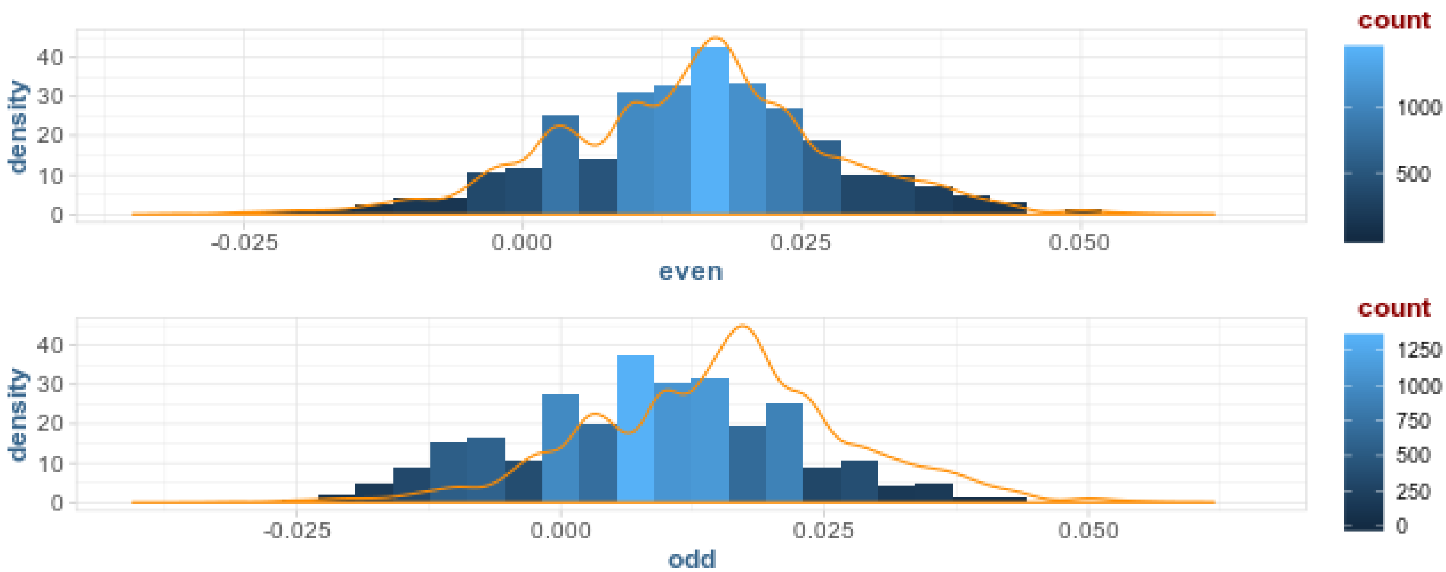

```
ne <- 200 # n even
no <- 201 # n odd

wwe <- rnorm(ne) # draw samples of size ne
wwo <- rnorm(no) # draw random samples of size no

N <- 10^4

even.boot <- numeric(N) # save space
odd.boot <- numeric(N)

for(i in 1:N)
{
   x.even <- sample(wwe, ne, replace = T)
   x.odd <- sample(wwo, no, replace = T)

   even.boot[i] <- median(x.even)
   odd.boot[i] <- median(x.odd)
}

boot <- data.table(even = even.boot, odd = odd.boot)
```

```r
p1 <- ggplot(boot, aes(even)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange")

p2 <- ggplot(boot, aes(odd)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange")

gridExtra::grid.arrange(p1, p2)
```




```r
ne <- 10000 # n even
no <- 10001 # n odd

wwe <- rnorm(ne) # draw samples of size ne
wwo <- rnorm(no) # draw random samples of size no

N <- 10^4

even.boot <- numeric(N) # save space
odd.boot <- numeric(N)

for(i in 1:N)
{
   x.even <- sample(wwe, ne, replace = T)
   x.odd <- sample(wwo, no, replace = T)

   even.boot[i] <- median(x.even)
   odd.boot[i] <- median(x.odd)
}
```

```r
boot <- data.table(even = even.boot, odd = odd.boot)

p1 <- ggplot(boot, aes(even)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange")

p2 <- ggplot(boot, aes(odd)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(even), col = "darkorange")

gridExtra::grid.arrange(p1, p2)
```



For odd n, median will be one of the sample points. For smaller n, there will be only n possible values for the median, so the sampling distribution is more “granular” than when n is even.

5.11

Import the data from data set Bangladesh. In addition to aresnic concentrations for 271 wells, the data set contains cobolt and chlorine concentrations.

a.) Conduct EDA on the chlorine concentrations and describe the salient features.

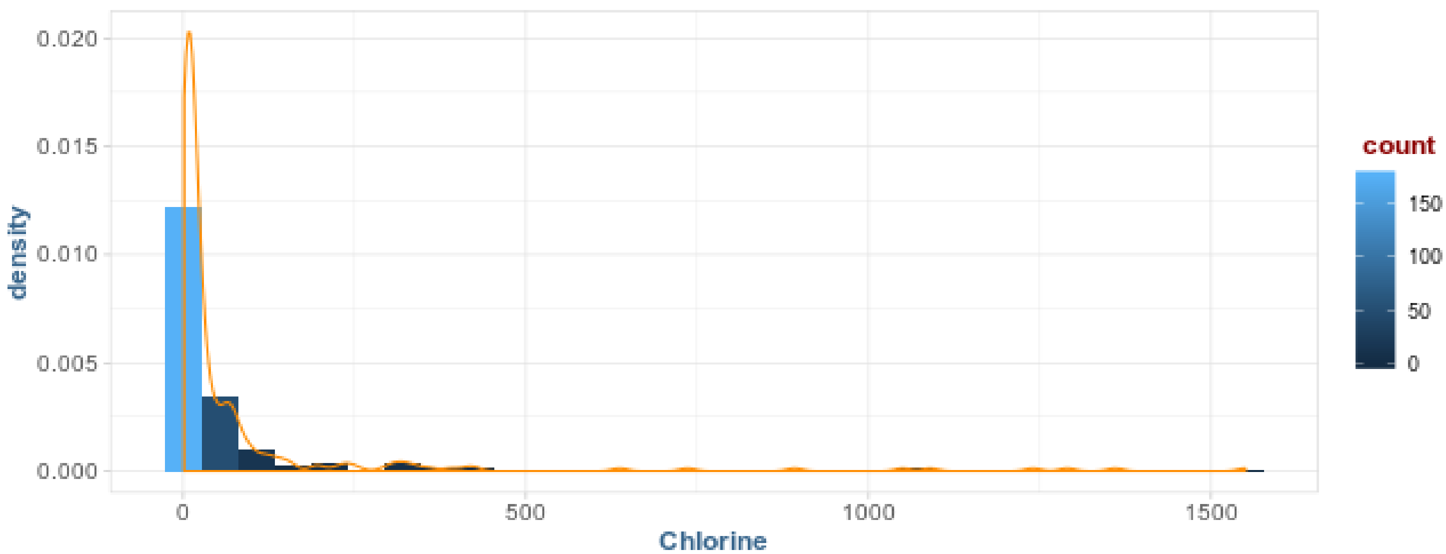
```r
Bangladesh <- data.table(read.csv(paste0(data.dir, "Bangladesh.csv"),
                                  header = T))
head(Bangladesh)
```

```
   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
```

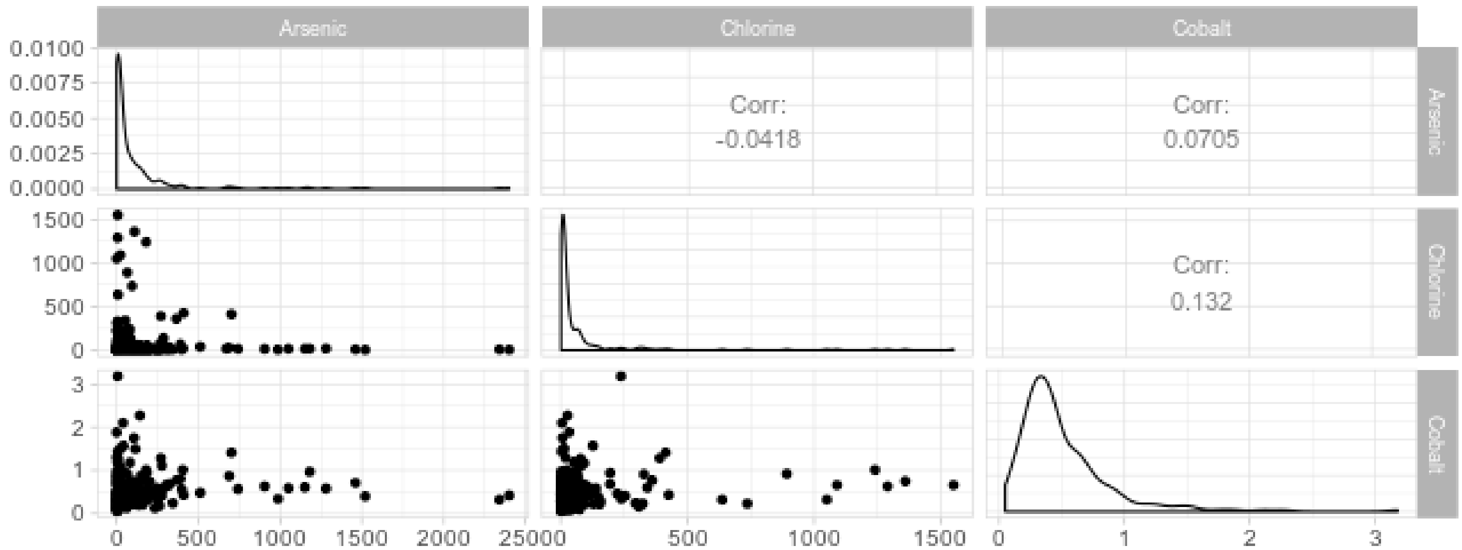
```
ggplot(Bangladesh, aes(Chlorine)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(aes(Chlorine), col = "darkorange")
```

```
Warning: Removed 2 rows containing non-finite values (stat_bin).

Warning: Removed 2 rows containing non-finite values (stat_density).
```




```
GGally::ggpairs(Bangladesh)
```

```
Warning in (function (data, mapping, alignPercent = 0.6, method = "pearson", :
Removed 2 rows containing missing values

Warning in (function (data, mapping, alignPercent = 0.6, method = "pearson", :
Removing 1 row that contained a missing value

Warning: Removed 2 rows containing missing values (geom_point).

Warning: Removed 2 rows containing non-finite values (stat_density).

Warning in (function (data, mapping, alignPercent = 0.6, method = "pearson", :
Removed 3 rows containing missing values

Warning: Removed 1 rows containing missing values (geom_point).

Warning: Removed 3 rows containing missing values (geom_point).

Warning: Removed 1 rows containing non-finite values (stat_density).
```



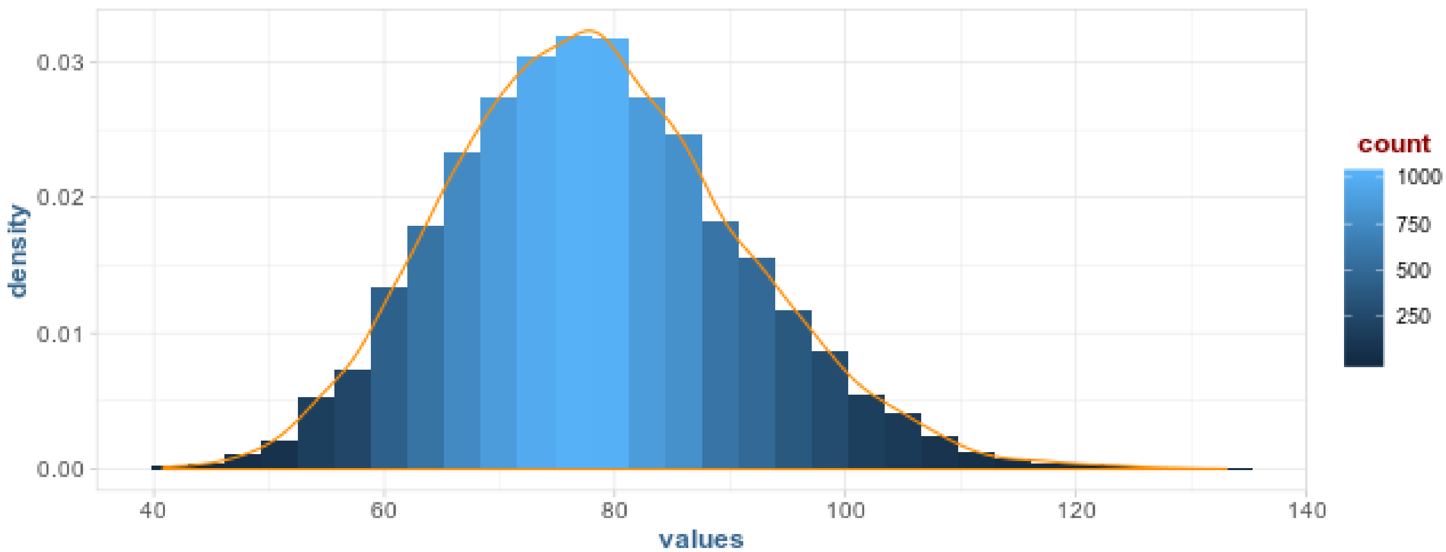
b.) Bootstrap the mean.

```r
N <- 10e3

boot.fn <- function(data, index){
  mean(data[index], na.rm = T)
}

boot(Bangladesh$Chlorine, boot.fn, R = N)
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = Bangladesh$Chlorine, statistic = boot.fn, R = N)


Bootstrap Statistics :
    original      bias    std. error
t1* 78.08401 0.04234321    12.72407
```

```r
observed <- mean(Bangladesh$Chlorine, na.rm = T)

bootstrap <- numeric(N)

for(i in 1:N)
{
  bootstrap[i] <- mean(sample(Bangladesh$Chlorine, size = nrow(Bangladesh), replace = T), na.r
}
```

```r
ggplot(data.table(values = bootstrap), aes(values)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
  geom_density(aes(values), col = "darkorange")
```



c.) Find and interpret the 95% bootstrap percentile confidence interval.

```r
alpha <- 0.05
quantile(bootstrap, c(alpha/2, 1 - alpha/2))
```

```
     2.5%     97.5% 
 55.13463 104.69678 
```

The spread on the confidence interval is extremely large, which is unsuprising given the heavly skewed distribtion of the sample.

d.) What is the bootstrap estimate of the bias? What fraction of the bootstrap standard error does it represent?

```r
bias <-  mean(bootstrap) -observed

bias / sd(bootstrap)
```

```
[1] 0.01124304
```

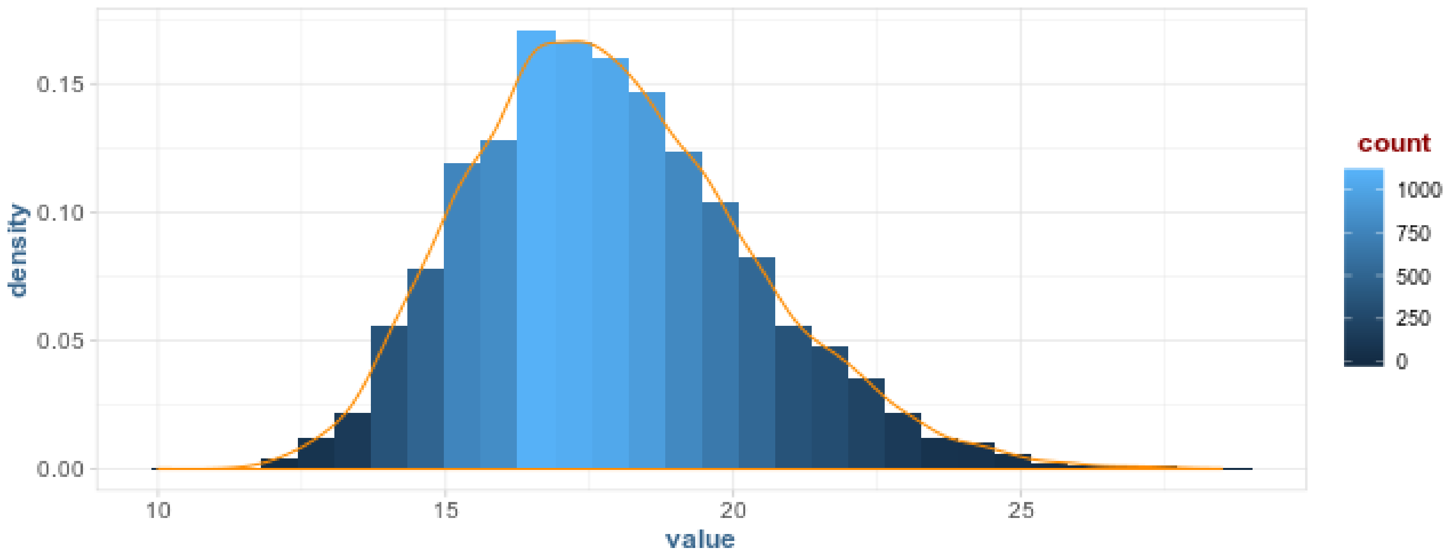
roughly 1% of the standard error.

5.12

Consider Bangladesh chlorine (concentration). Bootstrap the trimmed mean (say, trim the upper and lower 25%), and compare your results with the usual mean (previous result).

```r
values <- Bangladesh[!is.na(Chlorine)]$Chlorine
```

```
n <- length(values); N <- 10e3; trim <- .25

observed <- mean(values, trim = trim)
bootstrap <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   bootstrap[i] <- mean( sample(values, n, replace = T), trim = trim )
}

ggplot(data.table(value = bootstrap), aes(value)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange")
```




```
alpha <- 0.05

boot.mean <- mean(bootstrap)
boot.bias <- boot.mean - observed
boot.se <- sd(bootstrap)

quantile(bootstrap, c(alpha/2, 1 - alpha/2))
```

```
    2.5%    97.5%
13.70583 23.14009
```

```
# boot pkg

boot.fn <- function(data, index){
   mean( data[index], trim = trim)
}
```

```
boot(values, boot.fn, R = N)
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = values, statistic = boot.fn, R = N)


Bootstrap Statistics :
    original      bias    std. error
t1*  17.6363 0.2632425    2.469784
```

The bootstrap 20% trimmed mean is substantially smaller than the regular mean. The confidence intervals are also tighter, which is unsuprising due to the heavy presence of outliers in our sample.

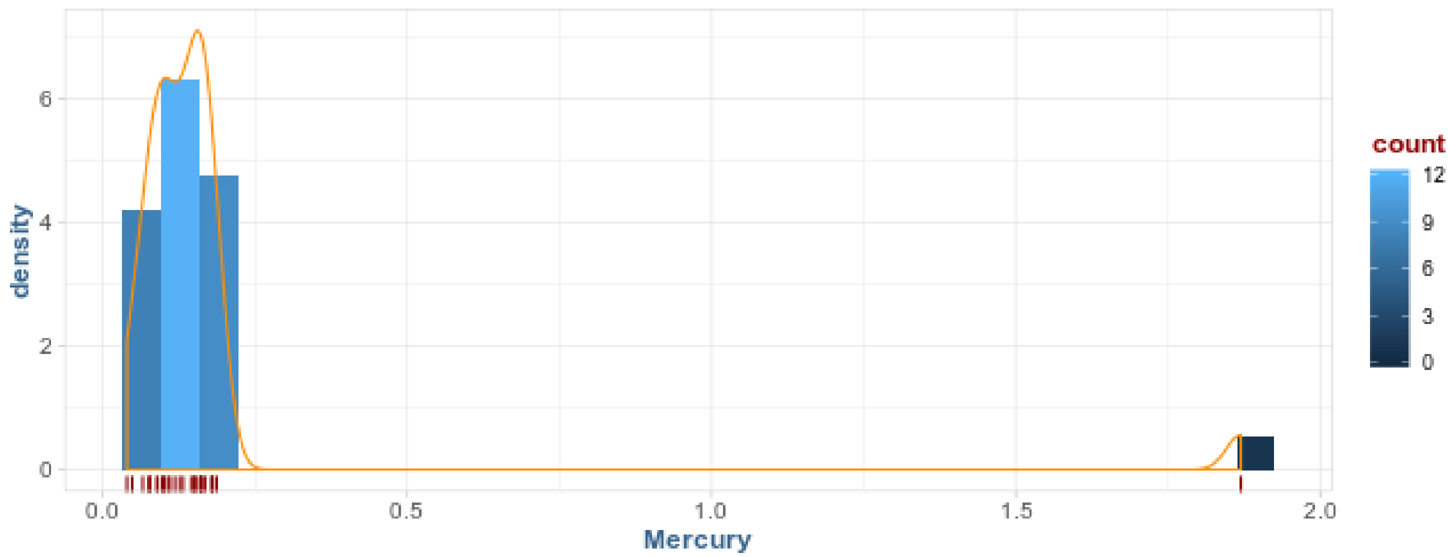
5.13

The data set *FishMercury* contains mercury levels (parts per million) for 30 fish caught in lakes in Minnesota.

```
FishMercury <- data.table(read.csv(paste0(data.dir, "FishMercury.csv"),
                                   header = T))
```

a.) Create a histogram or boxplot of the data. What do you observe?

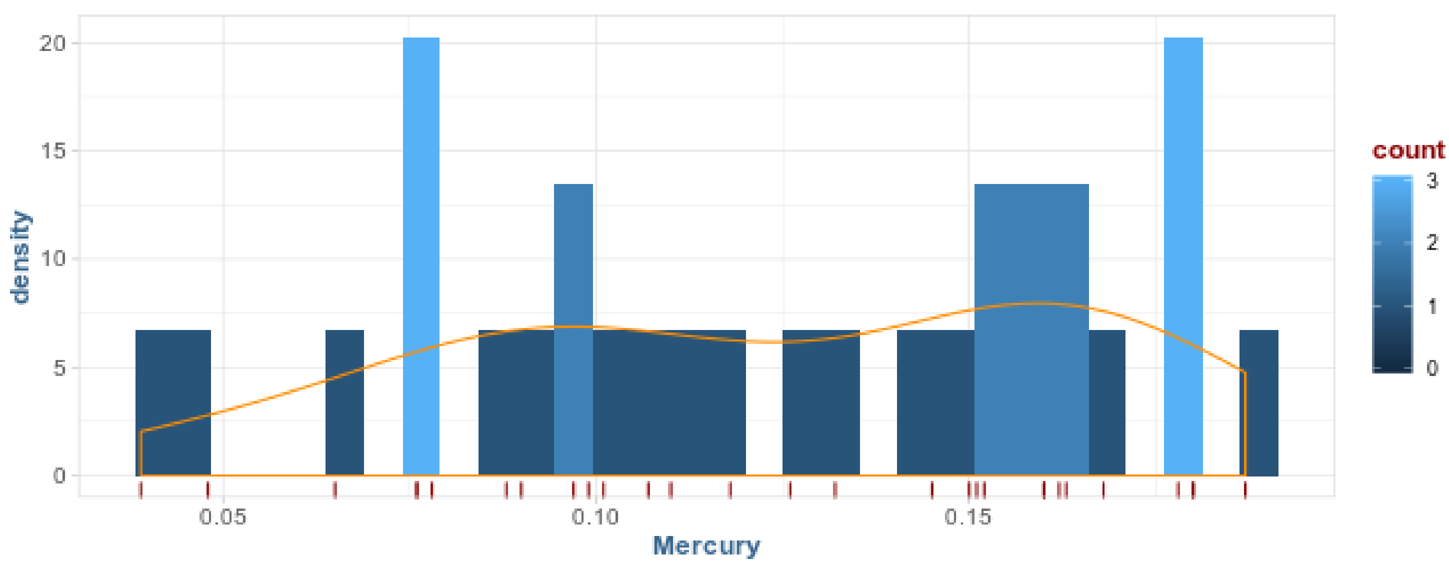
```
ggplot(FishMercury, aes(Mercury)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_rug(col = "darkred")
```




```r
FishMercury[Mercury > 1.5]
```

```
   Mercury
1:    1.87
```

```r
ggplot(FishMercury[Mercury < 1.5], aes(Mercury)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_rug(col = "darkred")
```




```r
FishMercury[Mercury > 1.5] / sd(FishMercury$Mercury) # 6 sd outlier
```

```
   Mercury
1: 5.81458
```

One extreme (6 SD) outlier in the data.

b.) Bootstrap the mean, and record the bootstrap standard error and the 95% bootstrap percentile interval.

```
n <- nrow(FishMercury); N <- 10e3

observed <- mean(FishMercury$Mercury)

bootstrap <- vector(mode = "numeric", length = n)

for(i in 1:N)
{
   bootstrap[i] <- mean( sample(FishMercury$Mercury, n, replace = T) )
}

boot.mean <- mean(bootstrap)
boot.bias <- boot.mean - observed
boot.se <- sd(bootstrap)

alpha <- 0.05

quantile(bootstrap, c(alpha/2, 1 - alpha/2))
```

```
     2.5%     97.5%
0.1130650 0.3075342
```

```
# boot pkg

boot.fn <- function(data, index) {
   mean( data[index] )
}

boot(FishMercury$Mercury, boot.fn, R = N)
```

```

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = FishMercury$Mercury, statistic = boot.fn, R = N)


Bootstrap Statistics :
     original       bias    std. error
t1* 0.1818667 -0.00048678  0.05752905
```

c.) Remove the outlier and bootstrap the mean of the remaining data. Record the bootstrap standard error and the 95% bootstrap percentile interval.

```
mercury <- FishMercury[Mercury < 1.5]$Mercury

n <- length(mercury)
bootstrap.values <- vector(mode = "numeric", length = n)

observed <- mean(mercury)

for( i in 1:N)
{
   bootstrap.values[i] <- mean( sample(mercury, size = n, replace = T) )
}

mean(bootstrap.values)
```

```
[1] 0.1236523
```

```
boot(mercury, boot.fn, R = N)
```

```

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = mercury, statistic = boot.fn, R = N)


Bootstrap Statistics :
     original       bias    std. error
t1* 0.1236552 8.709655e-05 0.007844452
```

```
quantile(bootstrap.values, c(alpha/2, 1 - alpha/2))
```

```
     2.5%     97.5%
0.1079302 0.1391379
```

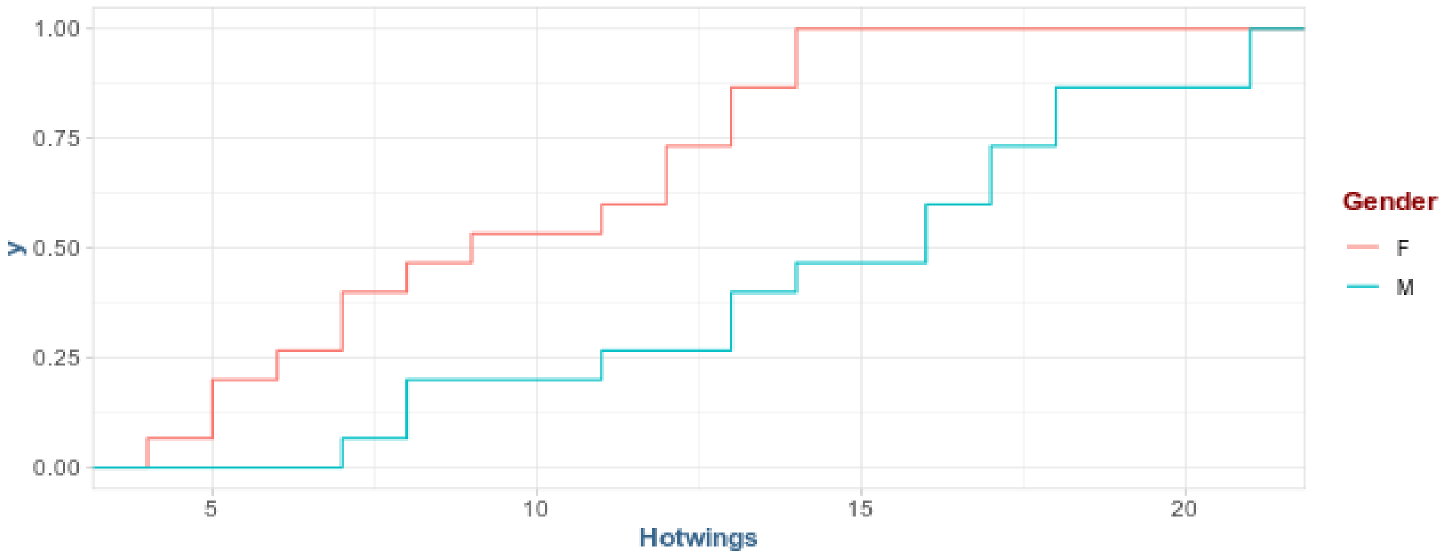
The standard error reduced drastically, and the confidence intervals are much more narrow.

5.14

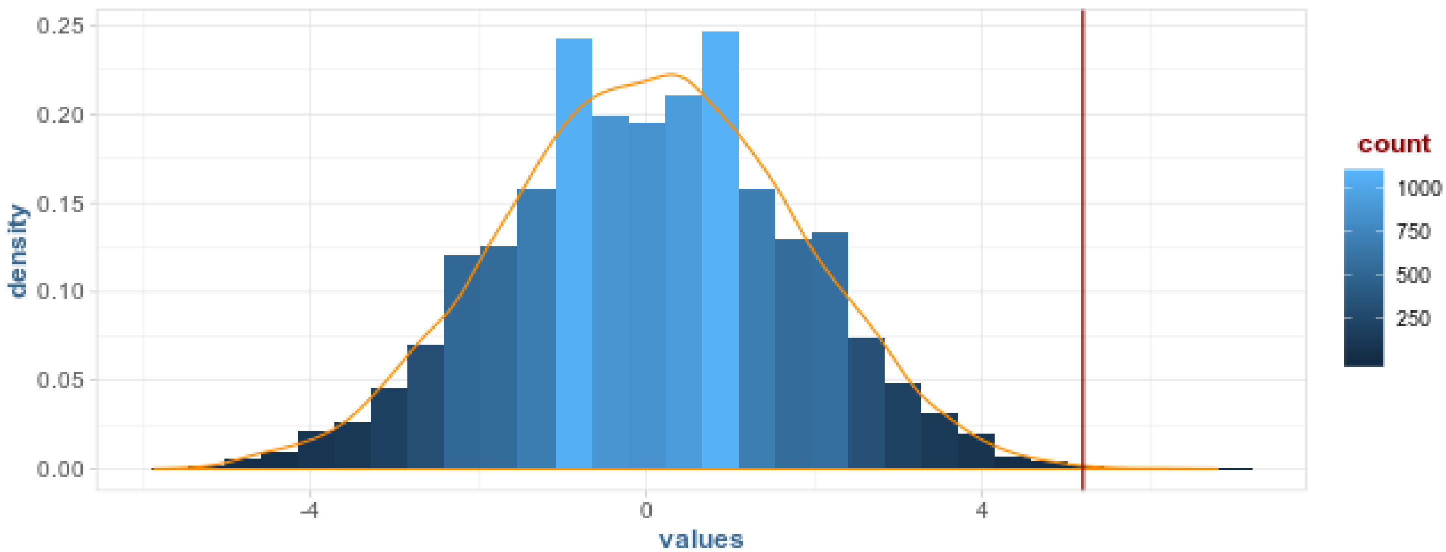
In Section 3.3, we performed a permutation test to determine if men and women consumed, on average, different amounts of hot wings.

```
BeerWings <- data.table(read.csv(paste0(data.dir, "Beerwings.csv"),
                                 header = T))
```

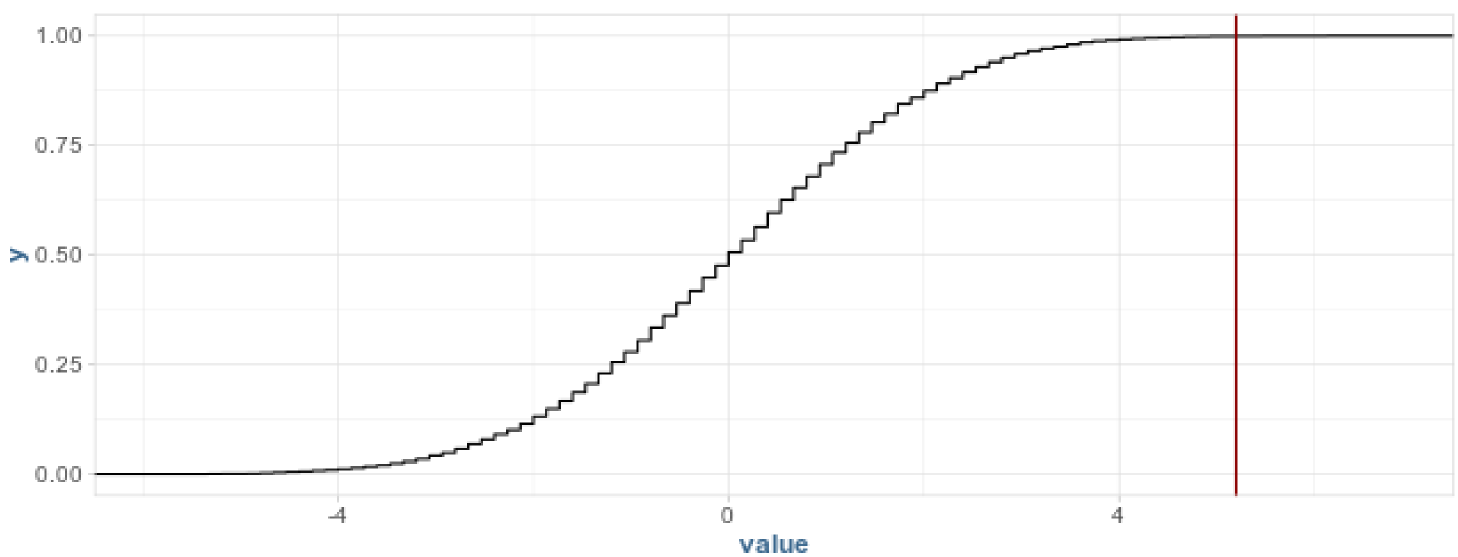
```r
wings <- BeerWings$Hotwings

?stat_ecdf

ggplot(BeerWings, aes(Hotwings, group = Gender, col = Gender)) +
   stat_ecdf(geom = "step")
```




```r
N <- 10e3; alpha <- 0.05; n <- length(wings)

wings.m <- BeerWings[Gender == "M"]$Hotwings
wings.f <- BeerWings[Gender == "F"]$Hotwings

observed <- mean(wings.m) - mean(wings.f)

permutation.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp <- sample(1:n, length(wings.m), replace = F)

   samp.m <- wings[samp]; samp.w <- wings[-samp]

   permutation.values[i] <- mean(samp.m) - mean(samp.w)
}

ggplot(data.table(values = permutation.values), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_vline(xintercept = observed, col = "darkred")
```




```r
ggplot(data.table(value = permutation.values), aes(value)) +
   stat_ecdf(geom = "step") +
   geom_vline(xintercept = observed, col = "darkred") +
   scale_y_continuous(labels = comma)
```

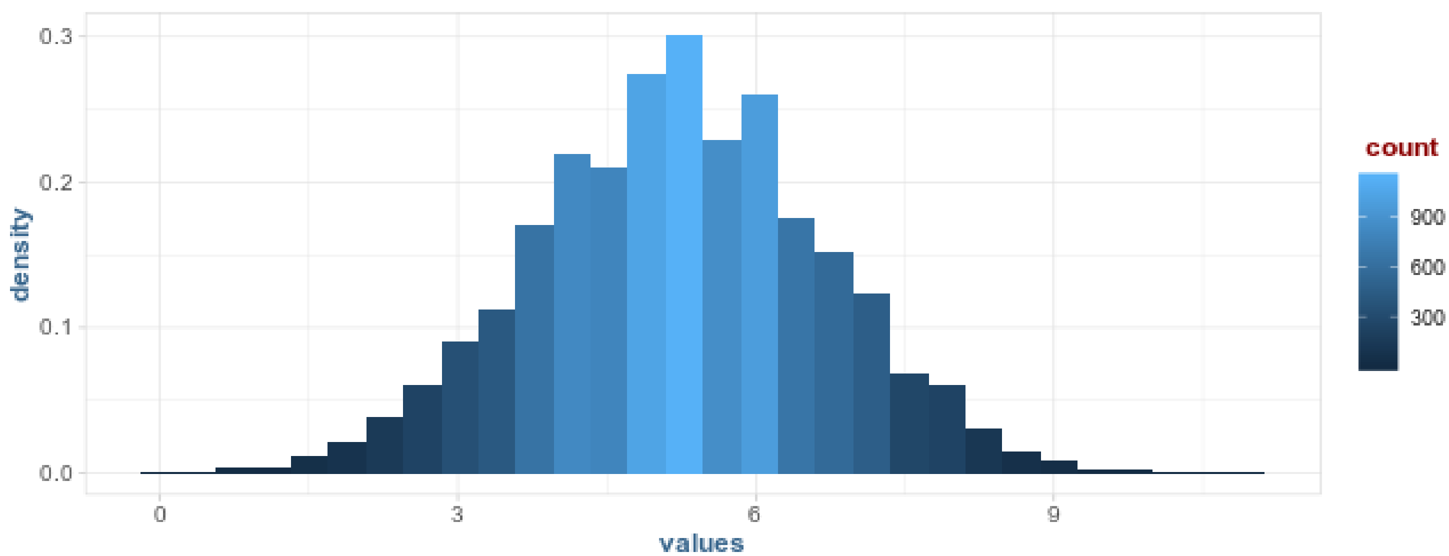



```r
p <- (sum(permutation.values >= observed) + 1) / (N + 1)

var <- p*(1 - p)/N
```

a.) Bootstrap the difference in means and describe the bootstrap distribution.

```r
bootstrap.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
```

```
{
  samp.m <- sample(wings.m, size = length(wings.m), replace = T)
  samp.f <- sample(wings.f, size = length(wings.f), replace = T)

  bootstrap.values[i] <- mean(samp.m) - mean(samp.f)
}

ggplot(data.table(values = bootstrap.values), aes(values)) +
  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30)
```




```
boot.mean <- mean(bootstrap.values)
boot.se <- sd(bootstrap.values)

boot.bias <- boot.mean - observed

boot.se; boot.bias
```

```
[1] 1.463113
```

```
[1] 0.02240667
```

b.) Find a 95% bootstrap percentile confidence interval for the difference of means, and give a sentence interpreting this interval.

```
quantile(bootstrap.values, c(alpha/2, 1 - alpha/2))
```

```
    2.5%    97.5%
2.333333 8.066667
```

The 95% confidence interval does not contain 0, further supporting our permutation test results.

c.) How do the bootstrap and permutation distributions differ?

The permutation distribution is sampled without replacement, and the bootstrap distribution is sampled using replacement.

5.15

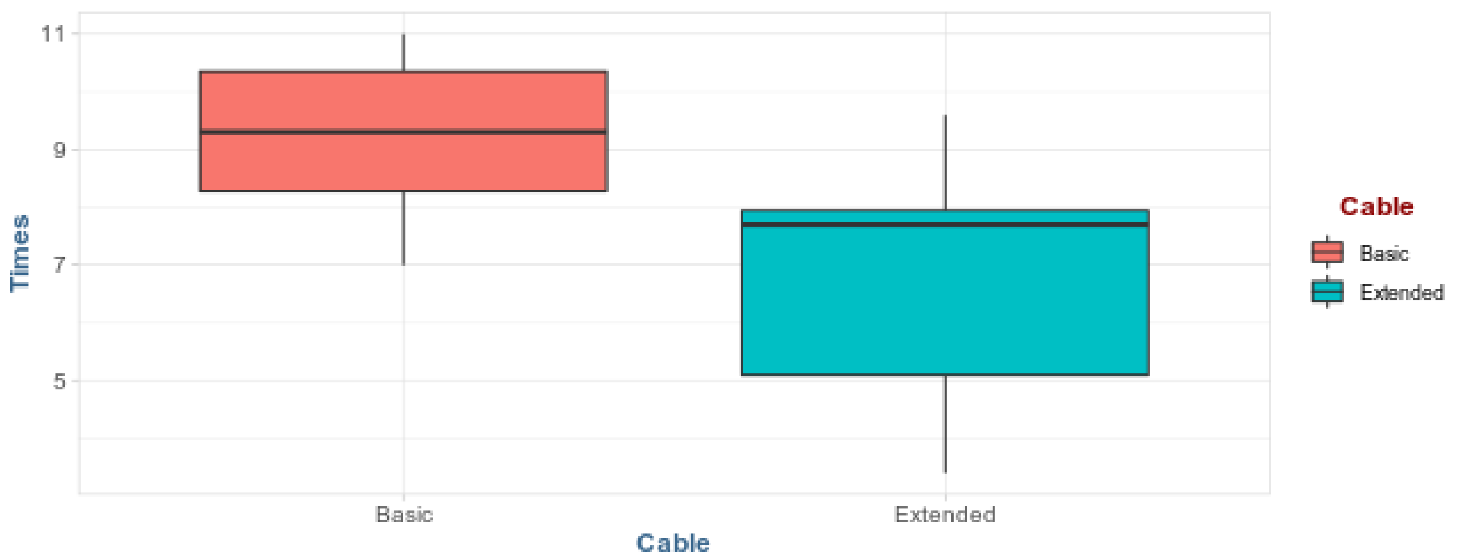
A high school student was curious about the total number of minutes devoted to commercials during any given half-hour time period on basic and extended cable TV channels. (B. Rodgers and T. Robinson, personal communication).

Import the TV data.

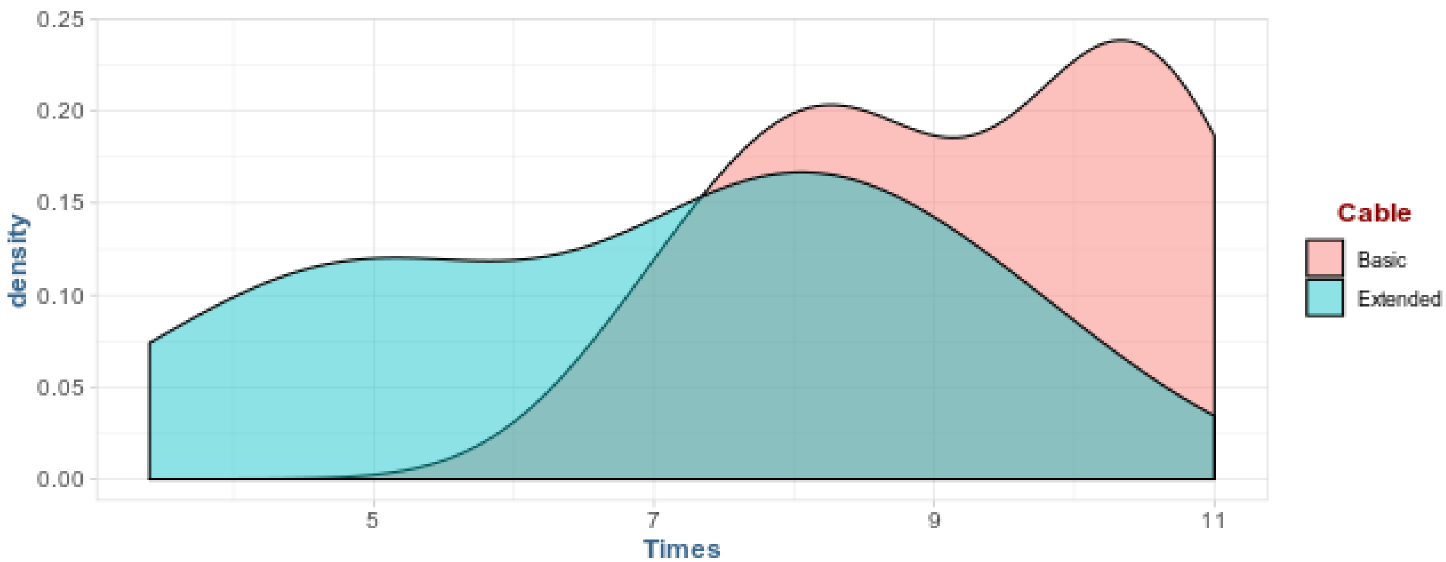
```
TV <- data.table(read.csv(paste0(data.dir, "TV.csv"),
                                  header = T))
```

a.) Perform some exploratory data analysis and obtain summary statistics on the commercial times on basic and extended cable TV channels (do seperate analysis for each type of channel).

```
ggplot(TV, aes(Cable, Times, fill = Cable)) +
   geom_boxplot()
```




```
ggplot(TV, aes(Times, group = Cable)) +
   geom_density(aes(y = ..density.., fill = Cable), alpha = .45)
```




```r
aov(TV$Times ~ TV$Cable)
```

```
Call:
   aov(formula = TV$Times ~ TV$Cable)

Terms:
                TV$Cable Residuals
Sum of Squares    27.378    57.330
Deg. of Freedom        1        18

Residual standard error: 1.784657
Estimated effects may be unbalanced
```

```r
tv.basic <- TV[Cable == "Basic"]$Times
tv.extended <- TV[Cable == "Extended"]$Times
tv.pooled <- c(tv.basic, tv.extended)

observed <- mean(tv.basic) - mean(tv.extended)

mean(tv.basic); mean(tv.extended)
```

```
[1] 9.21
```

```
[1] 6.87
```

b.) Bootstrap the difference in mean times, plot the distribution, and give summary statistics of the bootstrap distribution. Obtain a 95% bootstrap percentile confidence interval, and interpret this interval.

```r
N <- 10e3; n <- length(TV$Times); alpha <- 0.05

bootstrap.values <- vector(mode = "numeric", N)
```

```
for(i in 1:N)
{
   samp.basic <- sample(tv.basic, size = length(tv.basic), replace = T)
   samp.extended <- sample(tv.extended, size = length(tv.extended), replace = T)

   bootstrap.values[i] <- mean(samp.basic) - mean(samp.extended)
}

boot.mean <- mean(bootstrap.values)
boot.se <- sd(bootstrap.values)
boot.bias <- boot.mean - observed

quantile(bootstrap.values, c(alpha/2, 1 - alpha/2)) # bootstrap interval
```

```
 2.5% 97.5%
 0.86  3.85
```

The 95% boostrap interval does not contain 0, suggesting there is in fact a difference in the times between basic and extended (basic being longer).

c.) What is the bootstrap estimate of the bias?

```
boot.bias
```

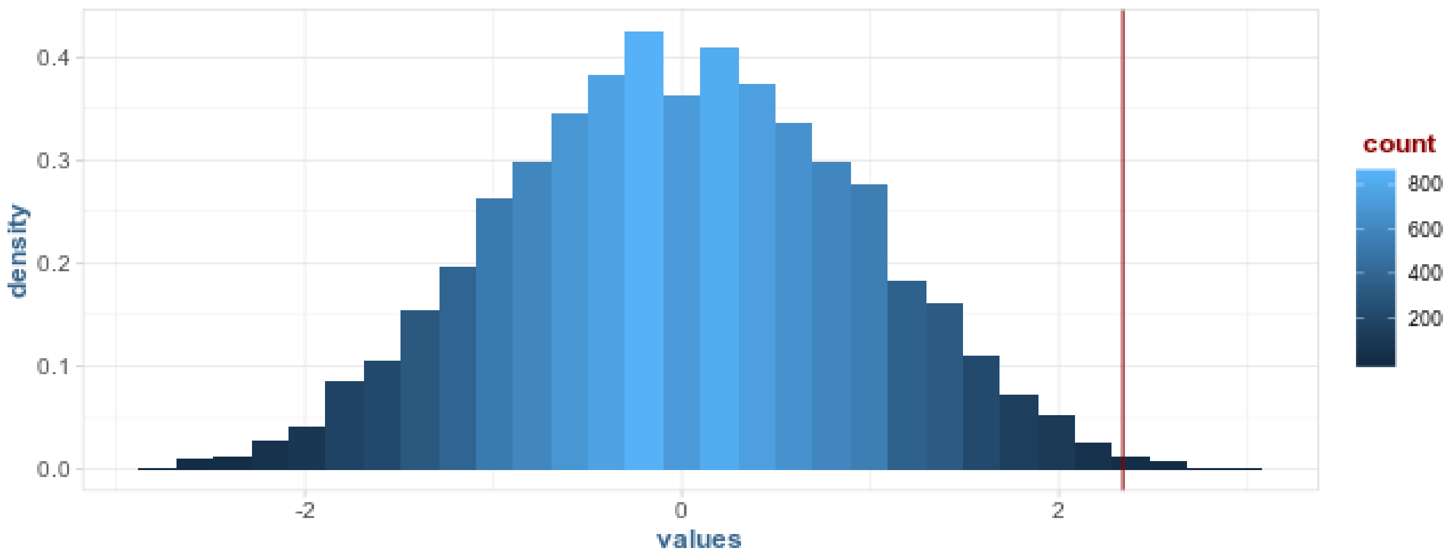
```
[1] -0.014552
```

```
boot.bias / boot.se # about 1% of the standard error
```

```
[1] -0.01894882
```

d.) Conduct a permutation test to see if the difference in mean commercial times is statistically significant, and state your conclusion.

```
permutation.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp <- sample(length(tv.pooled), length(tv.basic), replace = F)

   samp.basic <- tv.pooled[samp]; samp.extended <- tv.pooled[-samp]

   permutation.values[i] <- mean(samp.basic) - mean(samp.extended)
}

ggplot(data.table(values = permutation.values), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_vline(xintercept = observed, col = "darkred")
```




```
p <- ( sum(permutation.values >= observed) + 1) / ( N + 1)
var <- p*(1 - p)/N
```

Permutation test supports the alternative hypothesis that there is a difference in commercial times.

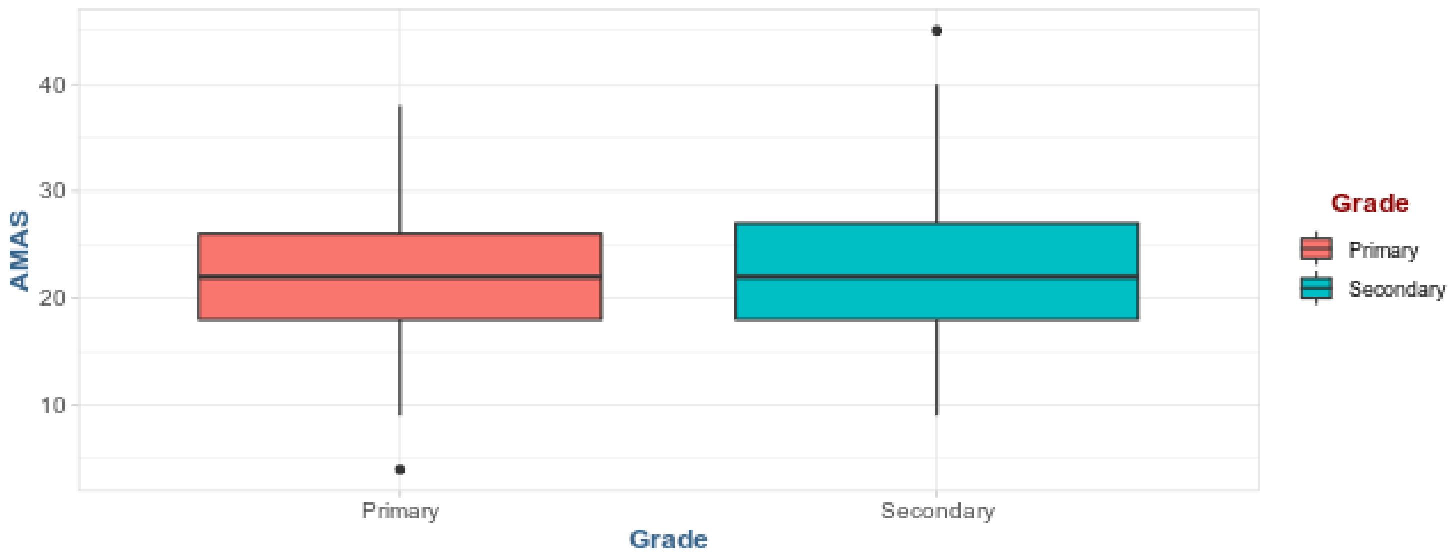
5.16

Researchers conducted a study of primary and early secondary school children in Italy to examine gender differences in math anxiety. One of the measures used to understand math anxiety is the *Abbreviated Math Anxiety Scale (AMAS)*, a self-reported math anxiety questionnaire. A higher score indicates more max anxiety. The data set MathAnxiety contains the results for a subset of the children in the original study.

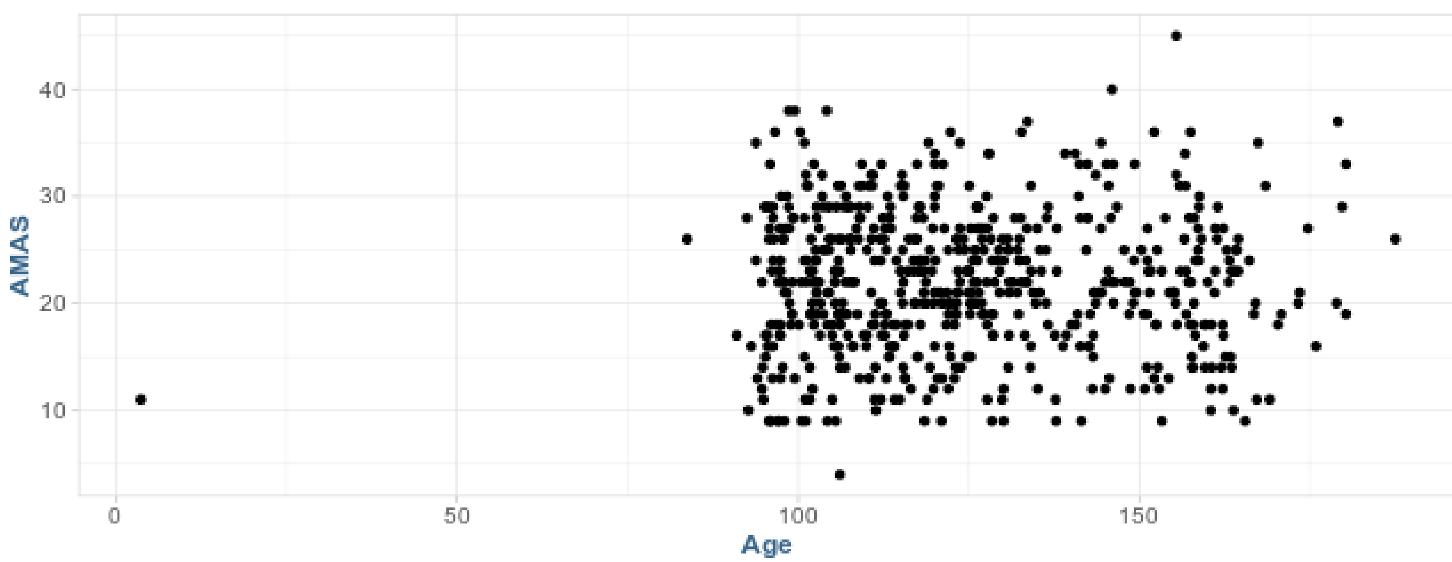
```
MathAnxiety <- data.table(read.csv(paste0(data.dir, "MathAnxiety.csv"),
                                   header = T))
```

a.) Perform some exploratory analysis and obtain summary statistics of the AMAS scores for the boys and girls (do seperate analysis for each gender).

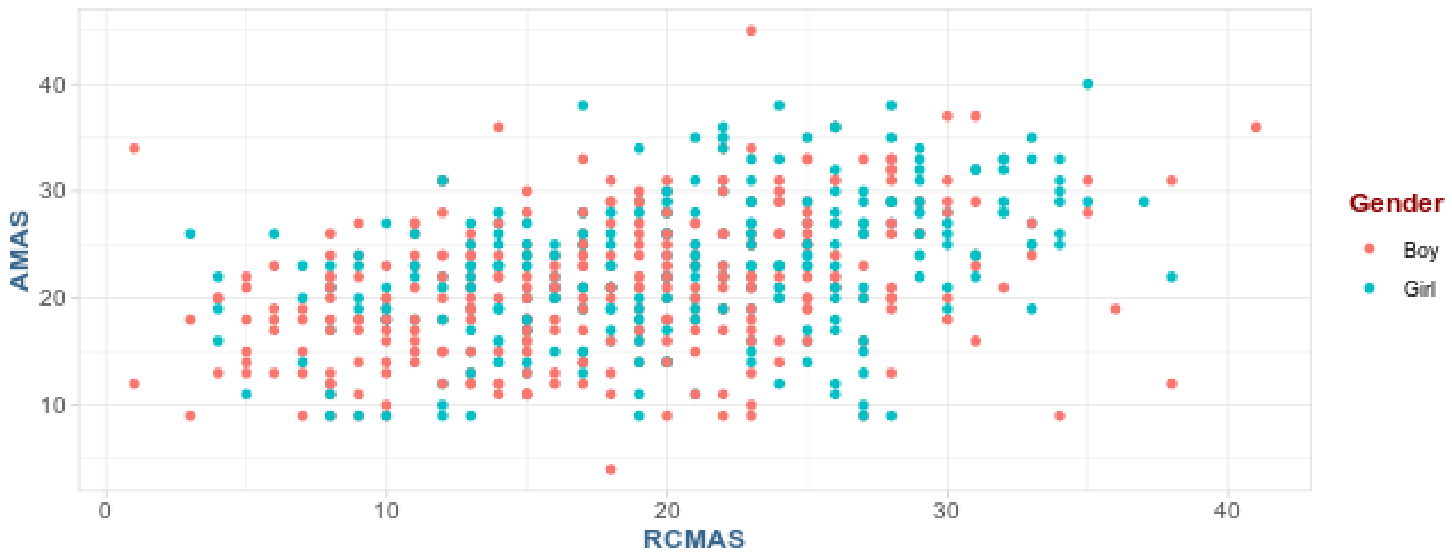
```
ggplot(MathAnxiety, aes(Grade, AMAS)) +
  geom_boxplot(aes(fill = Grade))
```




```r
ggplot(MathAnxiety, aes(Age, AMAS)) +
  geom_point()
```



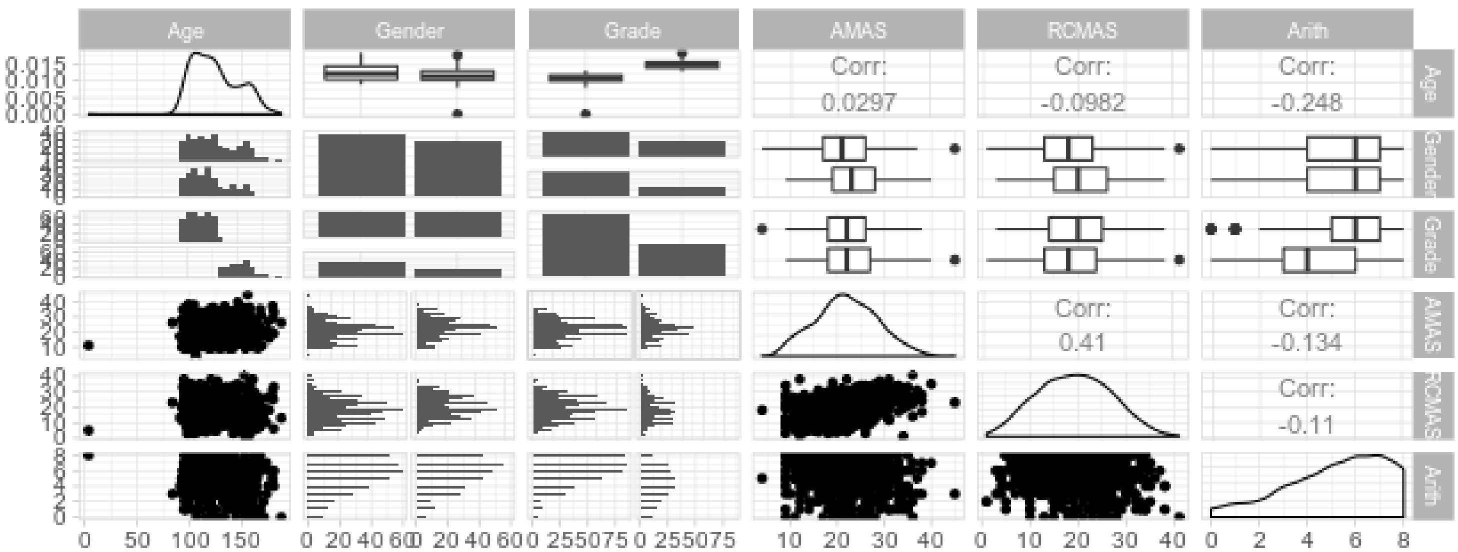

```r
ggplot(MathAnxiety, aes(RCMAS, AMAS)) +
  geom_point(aes(color = Gender))
```




```r
ggplot(MathAnxiety, aes(Gender, AMAS)) +
   geom_boxplot(aes(color = Gender))
```

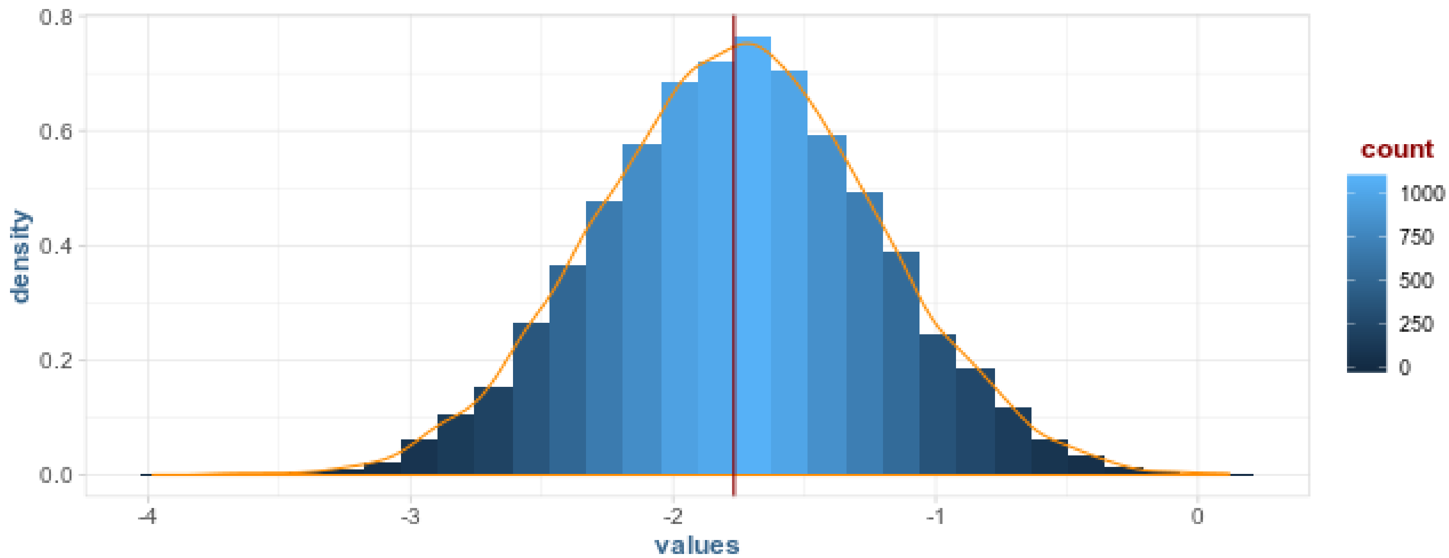

```r
ggpairs(MathAnxiety)
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



b.) Bootstrap the difference in mean times, plot the distribution, and give summary statistics of the bootstrap distribution. Obtain a 95% confidence interval, and interperet this interval.

```
amas_boy <- MathAnxiety[Gender == "Boy"]$AMAS
amas_girl <- MathAnxiety[Gender == "Girl"]$AMAS

N <- 10e3; n <- nrow(MathAnxiety); alpha <- 0.05

observed <- mean(amas_boy) - mean(amas_girl)

bootstrap.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp_boy <- sample(amas_boy, length(amas_boy), replace = T)
   samp_girl <- sample(amas_girl, length(amas_girl), replace = T)

   bootstrap.values[i] <- mean(samp_boy) - mean(samp_girl)
}

boot.mean <- mean(bootstrap.values)
boot.bias <- boot.mean - observed
boot.se <- sd(bootstrap.values)

ggplot(data.table(values = bootstrap.values), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(color = "darkorange") +
   geom_vline(xintercept = observed, col = "darkred")
```




```r
boot.bias; boot.se
```

```
[1] 0.01291534
```

```
[1] 0.5332156
```

```r
quantile(bootstrap.values, c(alpha/2, 1 - alpha/2)) # conf interval
```

```
      2.5%      97.5%
-2.8032698 -0.7139092
```

The 95% confidence interval is -2.8 ~ -0.7, which does not include zero. The data suggests that there is a statistical difference between the self-reported AMAS scores between boys and girls (boys being lower).

c.) What is the bootstrap estimate of the bias? What fraction of the bootstrap; standard error does this represent?

```r
boot.bias
```

```
[1] 0.01291534
```

```r
boot.bias / boot.se
```

```
[1] 0.02422161
```

Bootstrap bias is .006, which is less than 1% of the se.

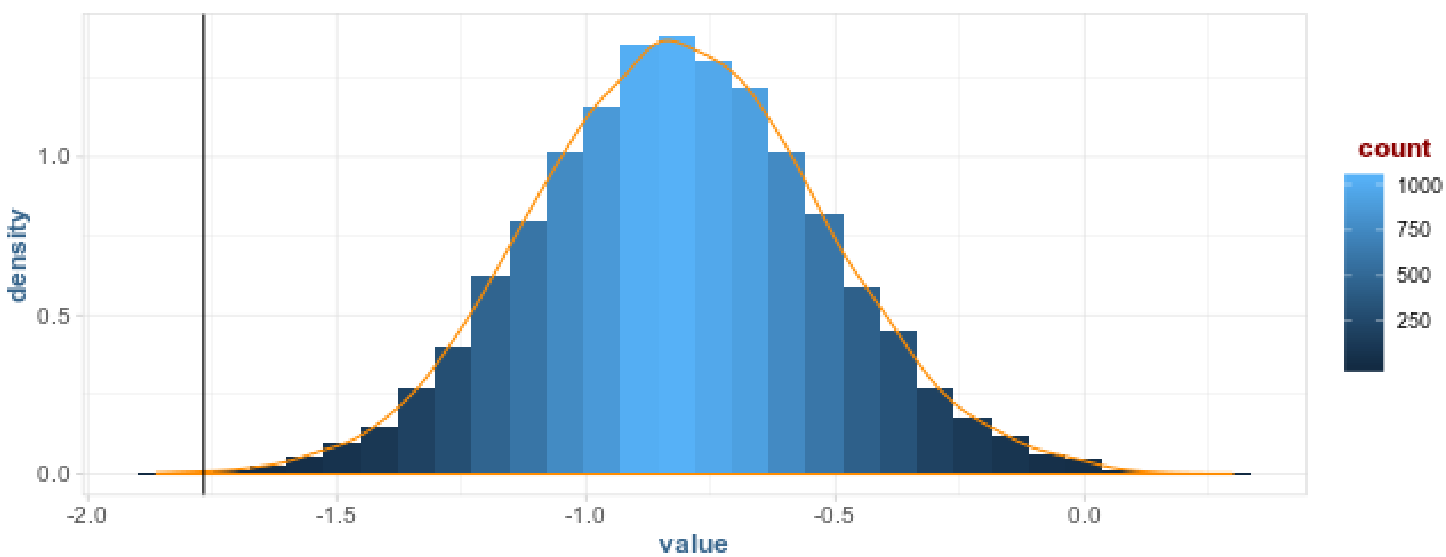
d.) Conduct a permutation test to see if the difference in mean AMAS scores is statistically significant, and state your conclustion.

```r
amas_pooled <- c(amas_boy, amas_girl)

n <- length(amas_pooled); observed <- mean(amas_boy) - mean(amas_girl)

permutation.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
```

```
{
   samp <- sample(1:n, length(samp_boy), replace = F)

   samp_boys <- amas_pooled[samp]; samp_girl <- amas_pooled[-samp]

   permutation.values[i] <- mean(samp_boy) - mean(samp_girl)
}

mean(permutation.values)
```

```
[1] -0.8142083
```

```
ggplot(data.table(value = permutation.values), aes(value)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_vline(xintercept = observed)
```




```
p <- ( sum(permutation.values <= observed) + 1 ) / (N + 1)
var <- p*(1 - p)/ N

t.test(amas_boy, amas_girl, alternative = "less")
```

```

    Welch Two Sample t-test

data:  amas_boy and amas_girl
t = -3.2918, df = 580.2, p-value = 0.0005279
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
       -Inf -0.8829609
```

```
sample estimates:
mean of x mean of y
 21.16718  22.93478
```

The p-value for the permutation test is less than 1%, we reject the null hypothesis that there is no difference in self-reported AMAS scores between boys and girls. This conclustion is in-line with our results from the bootstraped difference in mean test.

5.17

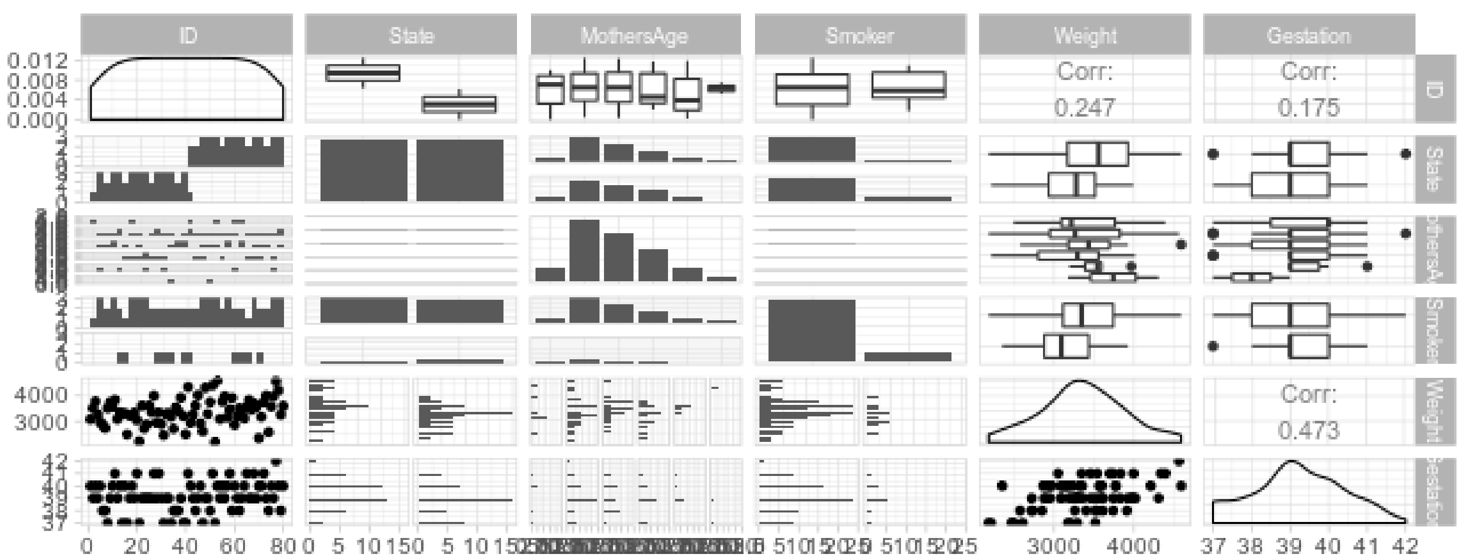
Import the data from Girls2004.

```r
Girls2004 <- data.table(read.csv(paste0(data.dir, "Girls2004.csv"),
                                 header = T))
```

a.) Perform some exploratory data analysis, and obtain summary statistics on the weights of baby girls born in Wyoming and Alaska.

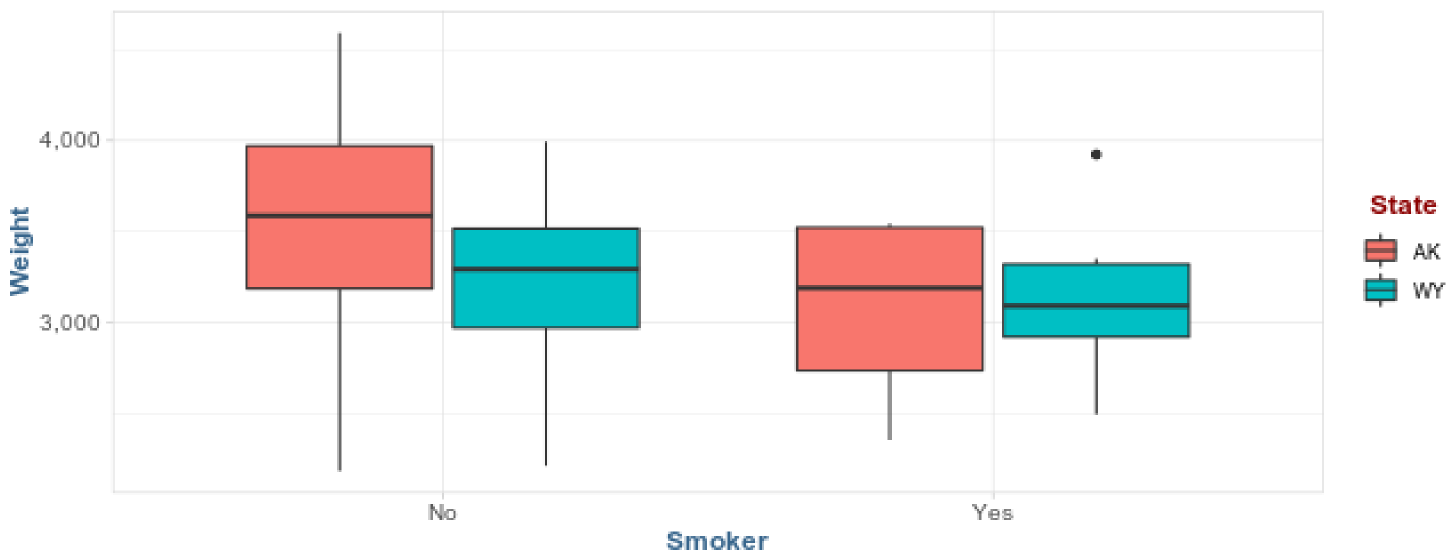
```r
ggpairs(Girls2004)
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

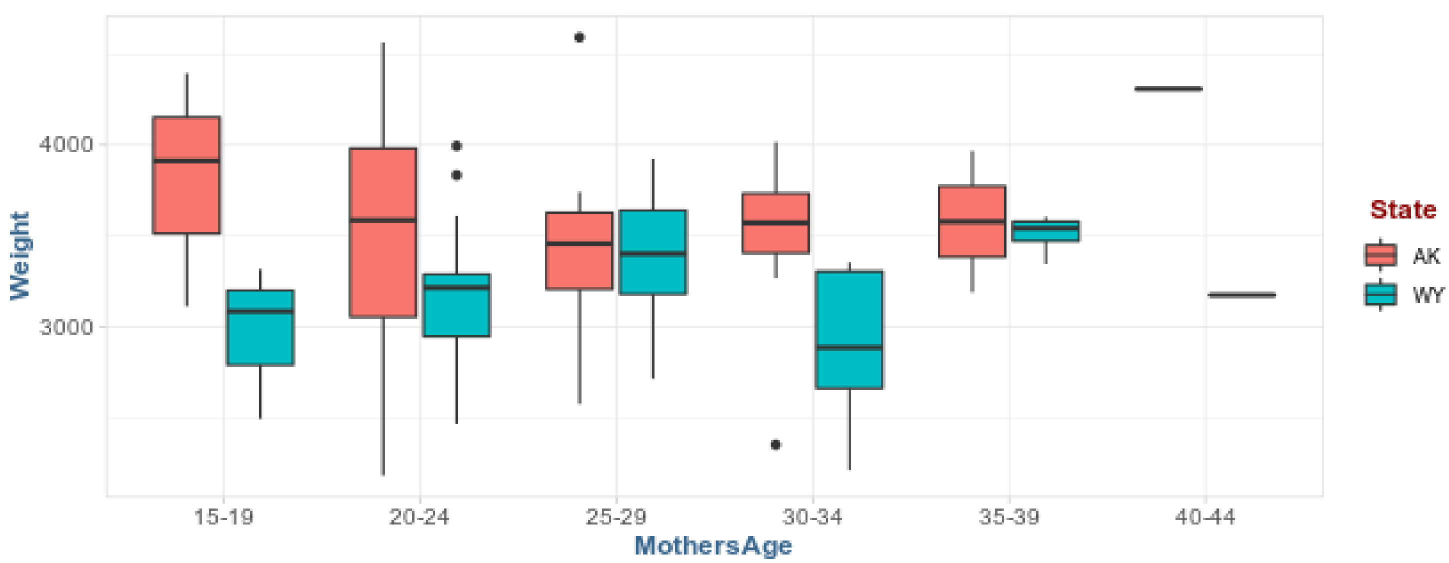



```r
ggplot(Girls2004, aes(Smoker, Weight, fill = State)) +
  geom_boxplot() +
```

```
  scale_y_continuous(labels = scales::comma)
```



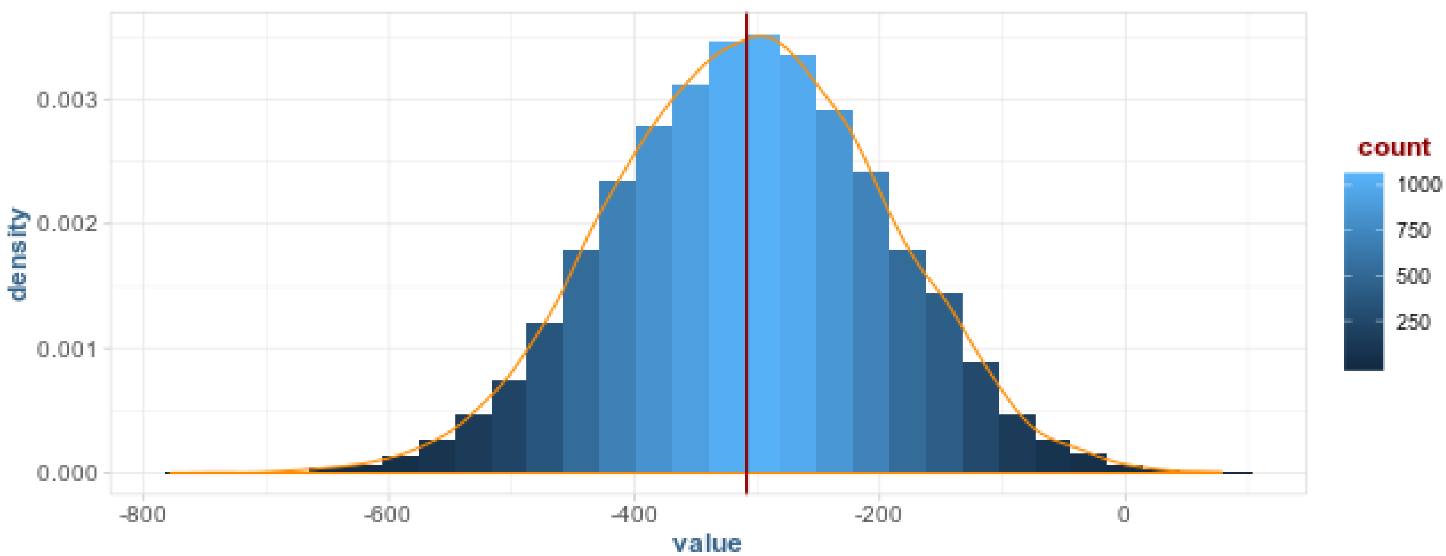

```
ggplot(Girls2004, aes(MothersAge, Weight, fill = State)) +
  geom_boxplot()
```



b.) Bootstrap the difference in means, plot the distribution, and give the summary statistics. Obtain a 95% confidence boostrap percentile confidence interval and interpret this interval.

```
weights_wy <- Girls2004[State == "WY"]$Weight
weights_ak <- Girls2004[State == "AK"]$Weight

observed <- mean(weights_wy) - mean(weights_ak); N <- 10e3
alpha <- 0.05
```

```
bootstrap.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp_wy <- sample(weights_wy, size = length(weights_wy), replace = T)
   samp_ak <- sample(weights_ak, size = length(weights_ak), replace = T)

   bootstrap.values[i] <- mean(samp_wy) - mean(samp_ak)
}

bootstrap.mean <- mean(bootstrap.values)
bootstrap.bias <- bootstrap.mean - observed
bootstrap.se <- sd(bootstrap.values)

ggplot(data.table(value = bootstrap.values), aes(value)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_vline(xintercept = observed, col = "darkred")
```




```
quantile(bootstrap.values, c(Lower = alpha/2, Upper = 1- alpha/2))
```

```
      2.5%      97.5%
-525.10125  -96.62063
```

There is a statistically significant difference in the mean birth weights of girls born in Wyoming and Alaska (Wyoming girls weighing less on average), by between 530 and 96 oz less.

c.) What is the bootstrap estimate of the bias? What fraction of thh boostrap standard error does it represent?

```
bootstrap.bias
```

```
[1] 0.6590725
```

```
bootstrap.bias / bootstrap.se
```

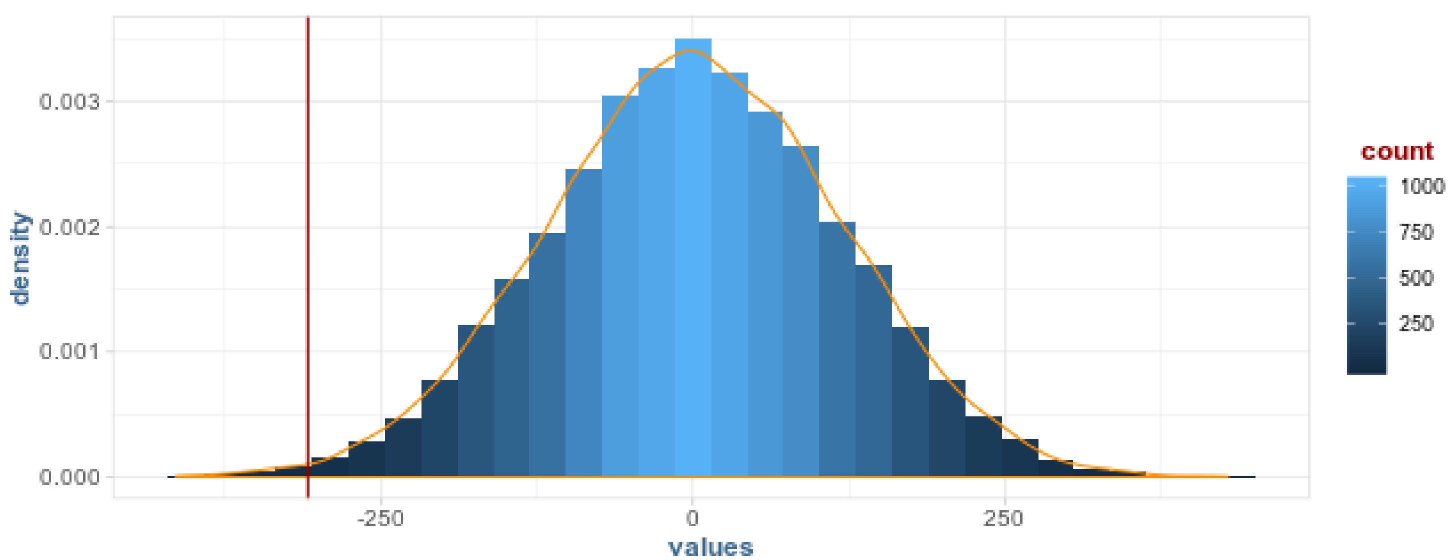
```
[1] 0.005922581
```

d.) Conduct a permutation test to see if the difference in mean weights is statistically significant, and state your conclusion.

```
observed <- mean(weights_wy) - mean(weights_ak)
weights_pooled <- c(weights_wy, weights_ak)

permutation.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp <- sample(length(weights_pooled), size = length(weights_ak), replace = F)

   samp_wy <- weights_pooled[samp]; samp_ak <- weights_pooled[-samp]

   permutation.values[i] <- mean(samp_wy) - mean(samp_ak)
}

mean(permutation.values)
```

```
[1] 0.802395
```

```
ggplot(data.table(values = permutation.values), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_vline(xintercept = observed, col = "darkred")
```



```
p <- (sum(permutation.values <= observed) + 1) / (N + 1)

var <- p*(1 - p)/N
```

The permutation test further supports the claim that the weights of baby girls born in Wyoming and Alaska are different. We reject the null hypothesis that the means are the same at the 0.05% level (p = 0.003).

e.) For what population(s), if any, does this conclusion hold?

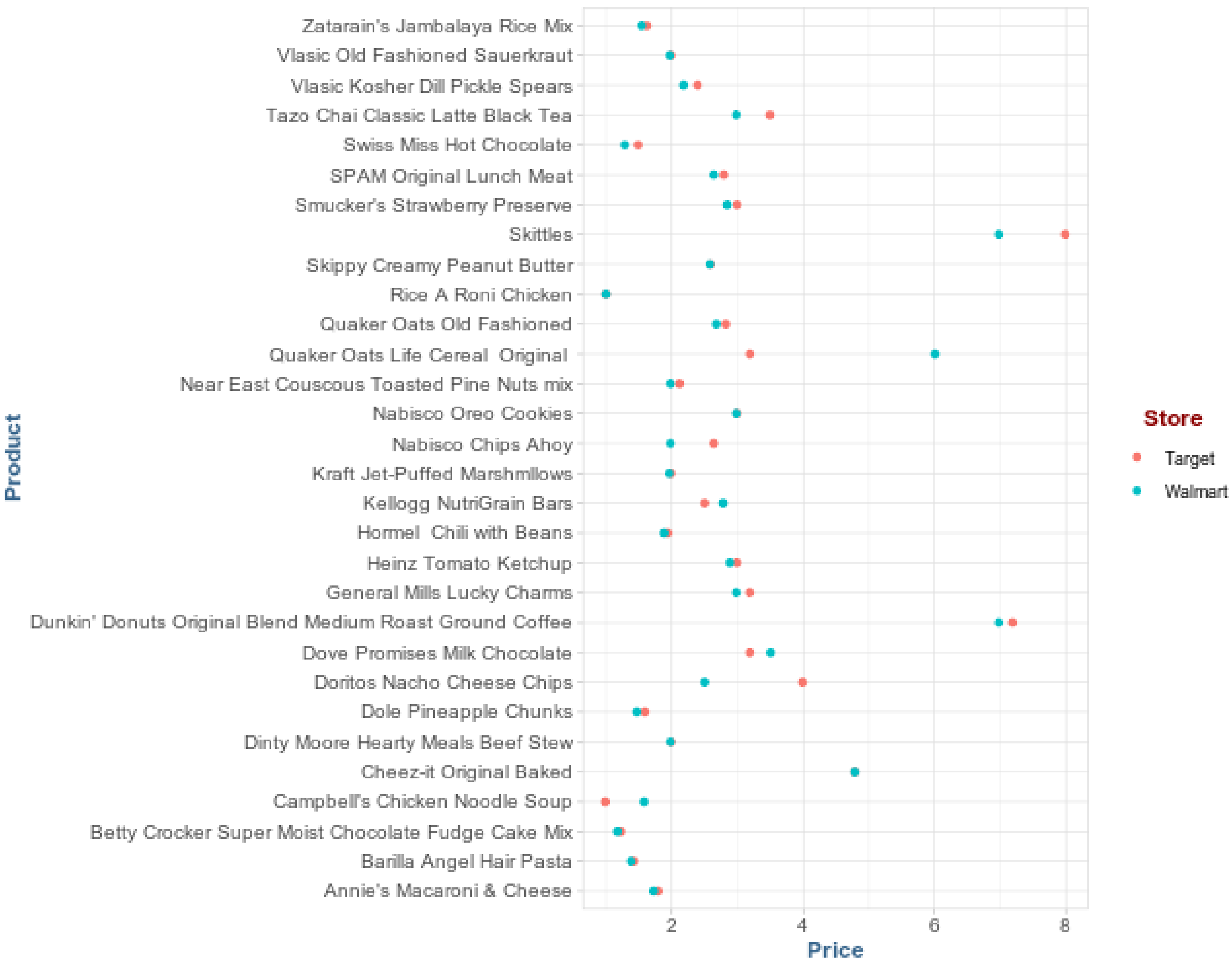
_I do not ### 5.18

Is there a difference in the price of groceries sold by the two retailers Target and Walmart? The data set *Groceries* contain, a sample of grocery items and their prices advertised on their respective websites on one specific day.

```
Groceries <- data.table(read.csv(paste0(data.dir, "Groceries.csv"),
                                 header = T))
```

a.) Compute summary statistics of the prices for each store.

```
groceries <- melt(Groceries %>% select(-Size),
                  id.vars = c("Product"),
                  value.name = "Price",
                  variable.name = "Store")

setorder(groceries, -Product)

ggplot(groceries, aes(Product, Price, col = Store, group = Store)) +
   geom_point(aes(color = Store)) +
   coord_flip()
```



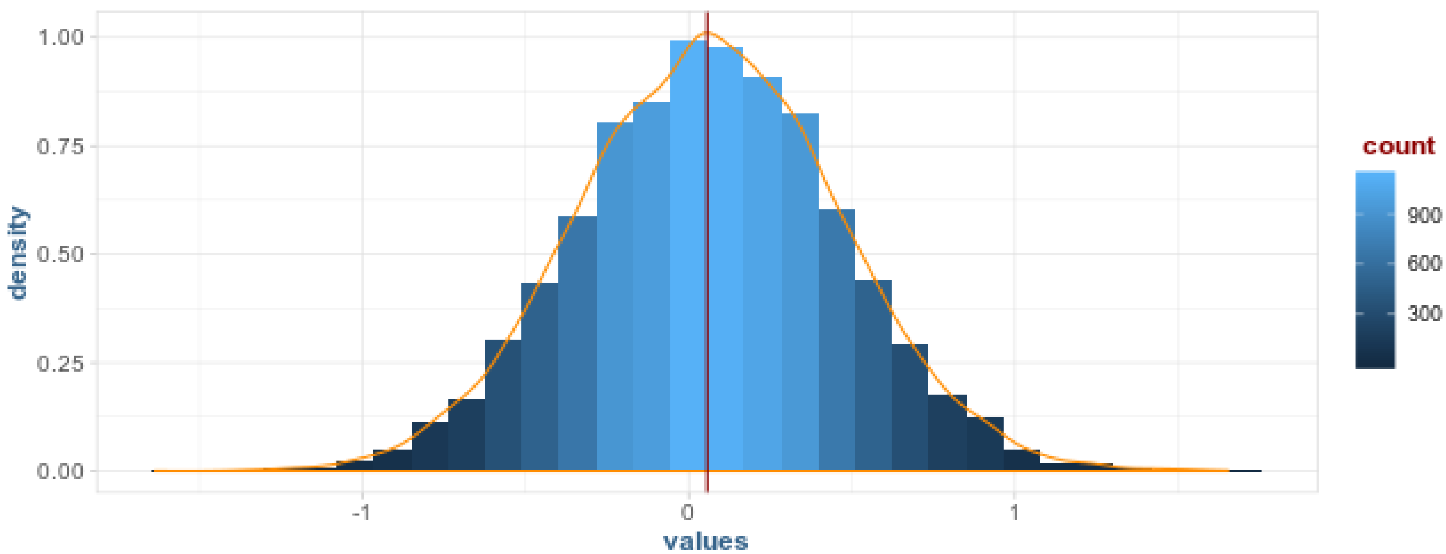
b.) Use the bootstrap to determine whether or not there is a difference in the mean prices.

```
prices.target <- groceries[Store == "Target"]$Price
prices.walmart <- groceries[Store == "Walmart"]$Price

N <- 10e3

observed <- mean(prices.target) - mean(prices.walmart)
bootstrap.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp.target <- sample(prices.target, size = length(prices.target), replace = T)
   samp.walmart <- sample(prices.walmart, size = length(prices.walmart), replace = T)
```

```
   bootstrap.values[i] <- mean(samp.target) - mean(samp.walmart)
}

boot.mean <- mean(bootstrap.values)
boot.bias <- boot.mean - observed
boot.se <- sd(bootstrap.values)

boot.bias / boot.se
```

```
[1] 0.01530084
```

```
ggplot(data.table(values = bootstrap.values), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_vline(xintercept = observed, col = "darkred")
```




```
quantile(bootstrap.values, c(Lower = alpha/2, Upper = 1 - alpha/2))
```

```
   2.5%   97.5%
-0.7227  0.8580
```

There doesn't appear to be a statistically significant difference in the average prices at the stores.

c.) Create a histogram of the difference in prices. What is unusal about Quaker Oats Life cereal?

```
Groceries[Product == "Quaker Oats Life Cereal  Original ",]
```

```
                               Product Size Target Walmart
1: Quaker Oats Life Cereal  Original  18oz   3.19    6.01
```

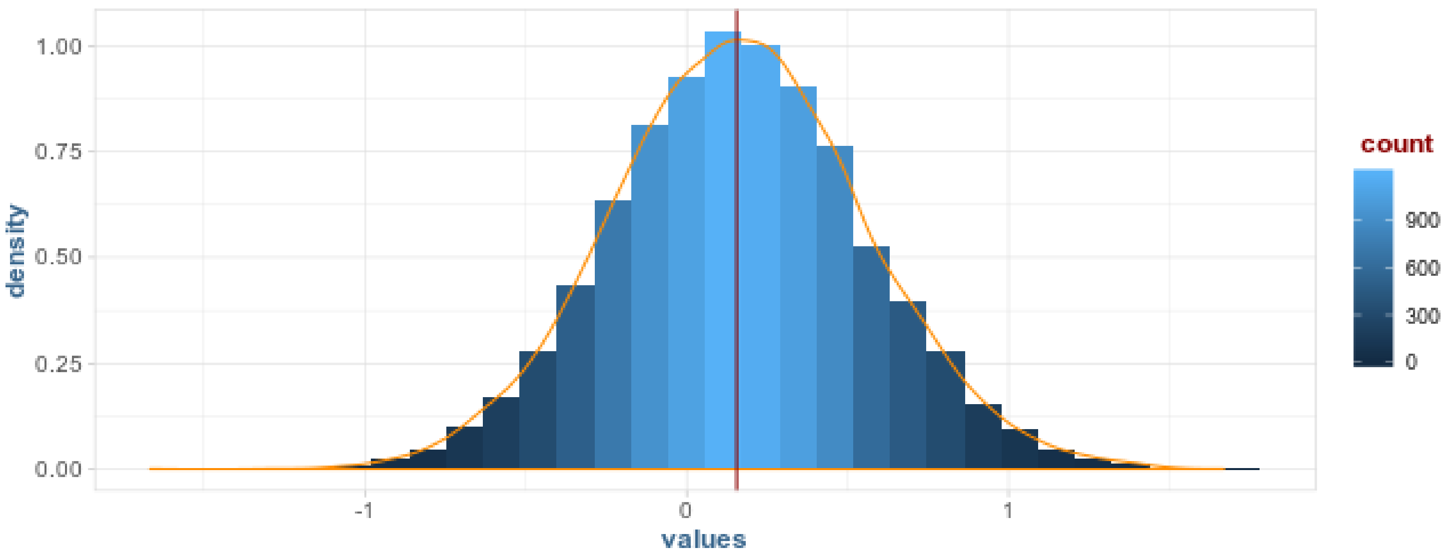
It's almost double the price.

d.) Recompute the bootstrap percentile interval without this observation. What do you conclude?

```r
groceries_sans_quaker <- groceries[Product != "Quaker Oats Life Cereal  Original ",]

stopifnot(nrow(groceries_sans_quaker) == nrow(groceries) - 2) # ensure removeed

prices.target <- groceries_sans_quaker[Store == "Target"]$Price
prices.walmart <- groceries_sans_quaker[Store == "Walmart"]$Price

N <- 10e3

observed <- mean(prices.target) - mean(prices.walmart)
bootstrap.values <- vector(mode = "numeric", length = N)

for(i in 1:N)
{
   samp.target <- sample(prices.target, size = length(prices.target), replace = T)
   samp.walmart <- sample(prices.walmart, size = length(prices.walmart), replace = T)

   bootstrap.values[i] <- mean(samp.target) - mean(samp.walmart)
}

boot.mean <- mean(bootstrap.values)
boot.bias <- boot.mean - observed
boot.se <- sd(bootstrap.values)

boot.bias / boot.se
```

```
[1] 0.01509251
```

```r
ggplot(data.table(values = bootstrap.values), aes(values)) +
   geom_histogram(aes(y = ..density.., fill = ..count..), bins = 30) +
   geom_density(col = "darkorange") +
   geom_vline(xintercept = observed, col = "darkred")
```




```
quantile(bootstrap.values, c(Lower = alpha/2, Upper = 1 - alpha/2))
```

```
      2.5%      97.5%
-0.6100345  0.9510517
```