Jared Yu

Computational Statistics

Problem Set 9

NOTE: The code for the assignment is in the Appendix.

1. Problem 9.5 with these additional comments for each part:
   a. Only find the bootstrap t confidence interval and use the bootstrap variance estimate presented in Lecture 9B and in Gentle 13.2. Don't forget to work with the data on the log scale.

$\hat{\theta}_{\log stomach}$	4.9679
$bootstrap\ t_{\log stomach}$	$(4.7729, 5.1950)$
$\hat{\theta}_{\log breast}$	6.5586
$bootstrap\ t_{\log breast}$	(5.8912,6.8036)

   b. This will be an application of the randomization method presented in Module 8 and discussed in Gentle 12.1. It is referred to as a permutation test in the text and is discussed in Section 9.8 of the Givens and Hoeting Text.

Let the log stomach data from the table be represented by $X = \{x_1, \cdots, x_{n_1}\}$ where $n_1 = 13$ and let the log breast data from the table be presented by $Y = \{y_1, \cdots, y_{n_2}\}$ where $n_2 = 11$.

$$H_0: \mu_X = \mu_Y \text{ vs. } H_1: \mu_X \neq \mu_Y$$

Using R, the statistic t_0, where $t_0 = \bar{x} - \bar{y}$, is calculated to be -1.590684. After doing the permutation test where $m = \binom{n_1 + n_2}{n_2} = \binom{13 + 11}{11} = 2{,}496{,}144$ combinations of the data from X and Y were created and the corresponding $t_i = \bar{x}_i - \bar{y}_i$ for $i = 1, \cdots, m$ were calculated. The value of t_0 is ranked as the $k = 2{,}477{,}469'th$ value. Therefore, we reject H_0 at the $\frac{k}{m} = 0.9925185$ significance level. The conclusion then is that $\mu_X \neq \mu_Y$ at significance level $\alpha = 0.01$.

   c. Use the percentile method to find the required confidence intervals. You will need to exponentiate the intervals from (a) in order to compare them.

	Breast data	*Stomach data*
Mean	1395.909	286
Mean of log data	6.5586	4.9679
Percentile C.I. (log data)	$(5.6173, 7.3759)$	$(4.3526, 5.6318)$
Exponentiated percentile C.I. (log data)	$(275.1347, 1{,}596.9896)$	$(77.6780, 279.1669)$
Percentile C.I. (original data)	$(778.1273, 2{,}145.3841)$	$(133.30, 492.25)$
Bootstrap t C.I.	$(5.8912, 6.8036)$	$(4.7729, 5.1950)$
Exponentiated bootstrap t C.I.	$(361.8524, 901.1214)$	$(118.2606, 180.3738)$

Using the percentile method and exponentiating the boundaries (exponentiated percentile C.I. (log data)), the resulting bounds of $(275.1347, 1{,}596.9896)$ for the breast data and $(77.6780, 279.1669)$ for the stomach data are obtained. In the breast data, the boundary fits around the original mean of 1395.909, however, in the stomach data the bounds are to the left of the original mean of 286. When exponentiating the bounds of the bootstrap confidence intervals (exponentiated bootstrap t C.I.), they both (breast and stomach data) have the same problem of being to the left of the original mean. However, when the percentile method is used on the original data (percentile C.I. (original data)), there's no issue with either dataset having the confidence intervals fit around the mean of the original dataset.

It seems then that there's an issue with taking the log of the data, finding a confidence interval, and then exponentiating the bounds to look back at the original values. Then a possible issue is that it's important to stay within the log scale and to perform re-calculations if changing from log scale to original scale.

2. Problem 9.7.

Cauchy example

The Cauchy distribution is a heavy-tailed distribution. Below in Figure 1 is a plot of the standard Normal distribution in black and a Cauchy distribution in red. The parameters for the Cauchy are $\alpha = 0$ (location) and $\beta = 2$ (scale). Compared to the standard normal with $\mu = 0$ and $\sigma^2 = 1$, the tails are visibly much heavier.

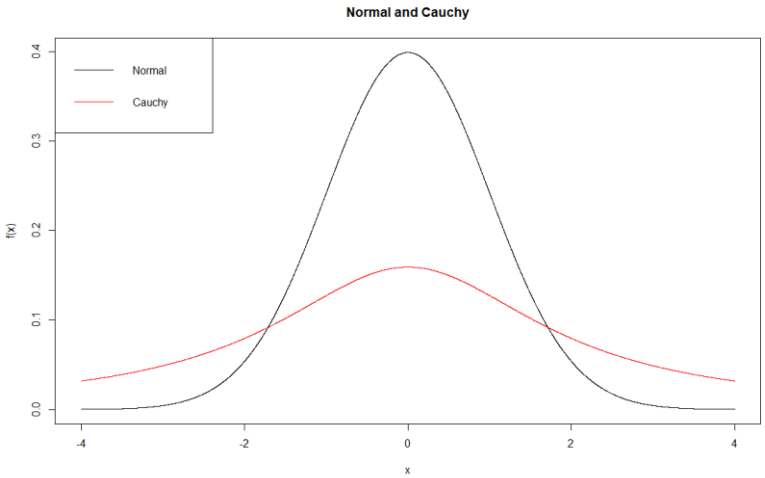
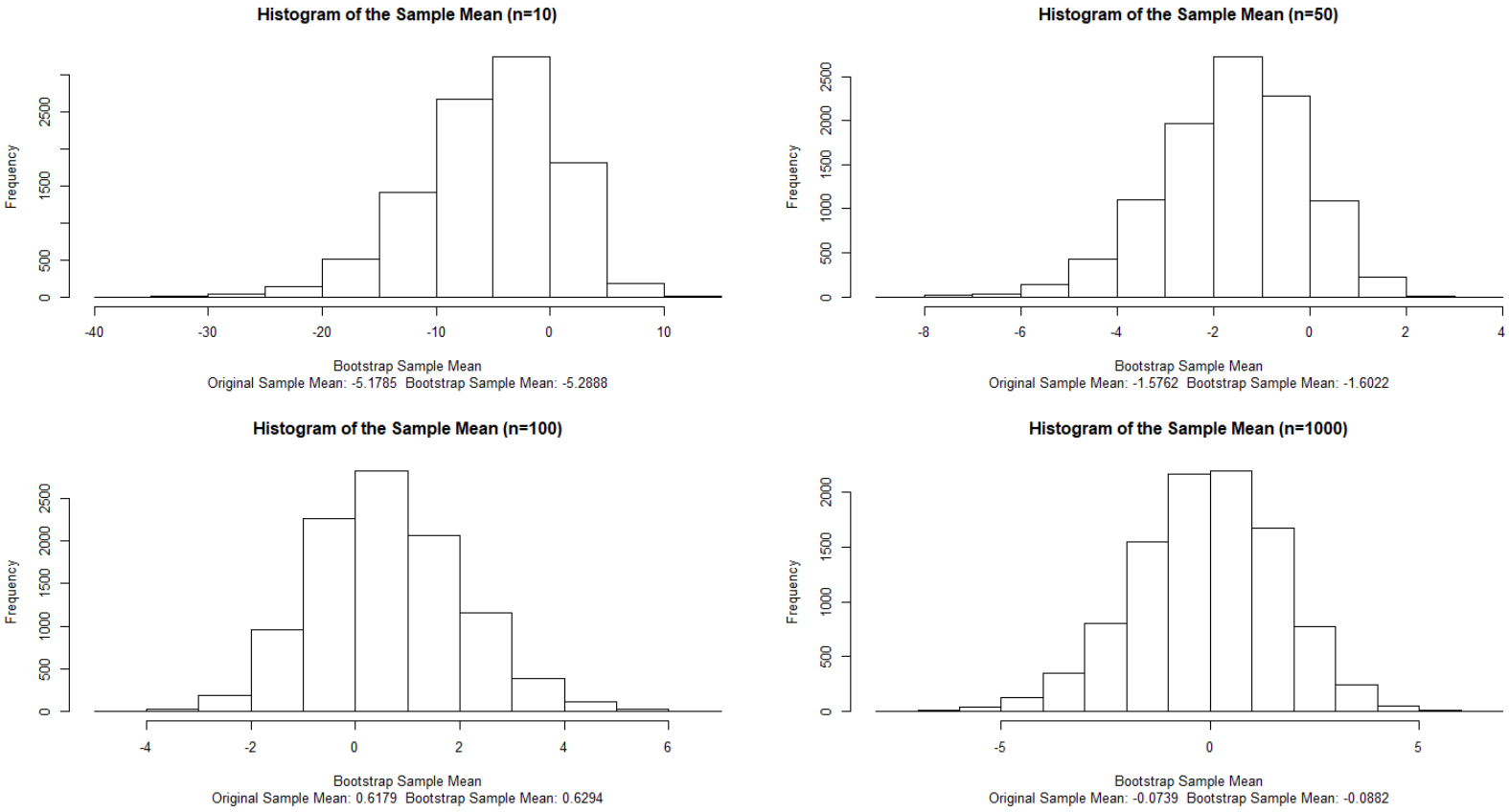


Figure 1

The problem with the principle of bootstrapping is that the given sample is ideally reflective of the population. Therefore, if a sample derived from the Cauchy distribution isn't large enough, then it may lead to poor bootstrapping results due to the (increased) possibility of observations from the heavy tails over-representing the data in the sample.

The following examples will sample $n = 10, 50, 100$ and $1{,}000$ points from a Cauchy distribution with the same parameters as those shown in the graph (the seed is 999). The

bootstrap process used $B = 10{,}000$ for the total number of bootstrap samples. In Figure 2 on the top-left is the histogram of the bootstrap sample means for when $n = 10$. In this case, it's evident that the sample means are skewed towards the left and are further away from where the above graph in Figure 1 indicated they instinctively would be (around 0). However, due to the heavy tails, the sample size of 10 isn't large enough to generate a representative sample that can be thought of as similar to the true population. The sample mean of this Cauchy sample is approximately -5.1785. Such a value shows already that the sample itself is quite skewed and



so bootstrapping from this single case is a poor choice. The mean from bootstrapping is -5.2888, which is also indicative of the sample size problem.

Figure 2

In the top-right plot of Figure 2, it shows the case where bootstrapping is done for $n = 50$ observations in a sample. Again, looking at the sample mean of approximately -1.5762 shows that there's an issue with the sample itself being properly representative of the entire population. Looking at the histogram, there's a skew towards the left. The bootstrap mean is -1.6022 and it also shows the same problems as the sample mean from the original sample size of 50.

At the bottom-left of Figure 2 is the case where bootstrapping was done for a sample size of $n = 100$. The sample mean this time is closer to 0, at approximately 0.6179. Looking at the histogram however, it still shows that there's a skew, but this time it's to the right. The bootstrap

mean is 0.6294 and it's evident in the histogram that there's an apparent skew in the bootstrap data.

Finally, at the bottom-right of Figure 2 is the case where bootstrapping was done with $n = 1{,}000$ observations. When taking a sample of this size for the Cauchy distribution, the original sample mean of approximately -0.0739 seems to indicate that the sample has reached a size large enough where it can begin to become representative of the entire population. There's no longer the case of observations from the tails dominating the distribution of the sample. The histogram itself looks far more symmetric and is centered around 0. The bootstrap mean of -0.0882 shows too that there's finally a sample size large enough to be representative of the true population.

Uniform example

Another situation is that of parameter estimation using bootstrap with the Uniform distribution. In this case, the goal is to estimate θ from a $Unif(0, \theta)$ distribution. First, the goal will be to show how θ can be estimated from a single sample.

Let $X \sim i.i.d. Unif(0, \theta)$, then $f_X(x; \theta) = \frac{1}{\theta}; 0 \leq x \leq \theta$. The support in this case contains θ so the normal method of obtaining the MLE will not work.

$$L(\theta) = f_X(x_1, \cdots, x_n; \theta) = \prod_{i=1}^{n} \frac{1}{\theta} 1_{\{x_i \leq \theta\}} = \frac{1}{\theta^n} \prod_{i=1}^{n} 1_{\{x_i \leq \theta\}}$$

where

$$1_{\{x_i \leq \theta\}} = \begin{cases} 1 & x_i \leq \theta \\ 0 & \text{otherwise} \end{cases}$$

When $\theta < x_{(n)}$, $f_X(x_1, \cdots, x_n; \theta) = 0$ (where $x_{(n)}$ represents the maximum or $n'th$ order statistic). Therefore, the only need is to look at cases with $\theta \geq x_{(n)}$. The j.p.d.f. can therefore be rewritten as follows:

$$L(\theta) = \frac{1}{\theta^n} 1_{\{x_i \leq \theta\}}$$

When θ decreases, $f_X(x_1, \cdots, x_n; \theta)$ increases. So, to maximize $f_X(x_1, \cdots, x_n; \theta)$ the minimum value for θ must be chosen where $\theta \in [x_{(n)}, \infty)$. $\therefore \hat{\theta} = x_{(n)}$.

Next, the problem of the necessity for extreme values (in this case, the maximum order statistic) from the $Unif(0, \theta)$ in order to obtain the MLE $\hat{\theta}$ will be highlighted. Below in Figure 3 is a grid of histograms of the bootstrap MLE, $\hat{\theta}_i^*$ for $i = 1, \cdots, B$ ($B = 10{,}000$), for different original sample sizes of $n = 10, 100, 1{,}000$, and $10{,}000$. (*NOTE:* The original maximum order statistic was calculated from an original sample of $Unif(0, \theta = 10)$, however, the value of θ in this case is arbitrary and is only used to show the contradiction in trying to bootstrap when there's an emphasis on extreme values.)

The issue is that in order to calculate $\hat{\theta}_i^*$, it's necessary to calculate $X_{(n)}^*$ from the $i'th$ bootstrap sample. However, with the bootstrap principle the largest value from the original sample will determine the largest possible value for all bootstrap samples. Therefore, the resulting $X_{(n)}^*$ from each of the bootstrap samples can never be larger than the original $X_{(n)}^*$. The resulting appearance of each of the histograms of $\hat{\theta}_i^*$ for $i = 1, \cdots, B$ is an exponential distribution with an extreme skew to the left and a spike on the right-hand side near the $X_{(n)}^*$ of the original sample. If, however, the value of $\hat{\theta}^*$ were calculated with the mean or the median, this type of situation wouldn't occur since they're far less dependent on extreme values.

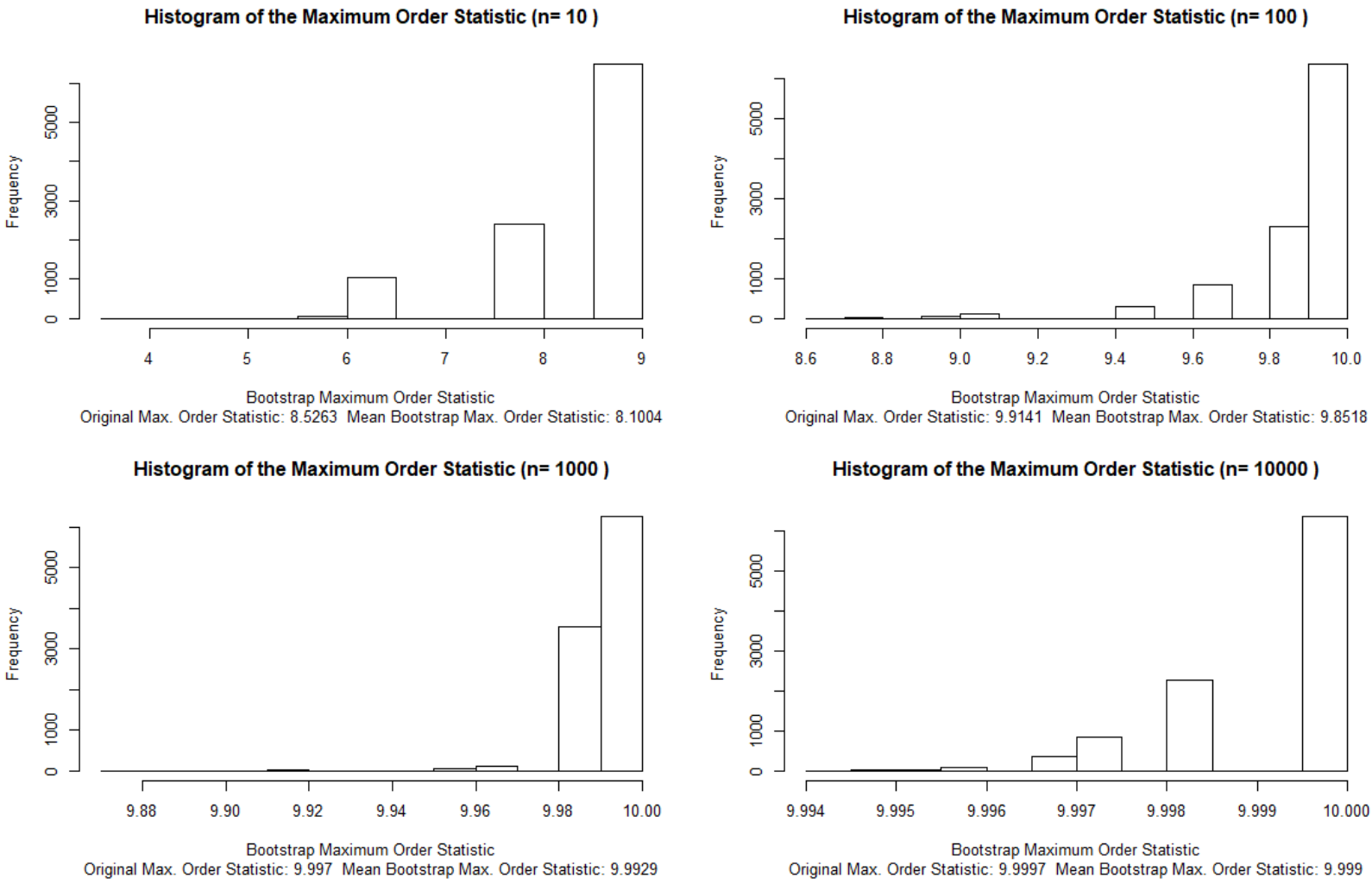


Figure 3

3. Let $\mathcal{S} = \{Y_1, \cdots, Y_n\}$ be a random sample from a population with mean μ, variance σ^2, and distribution P. Let $\hat{P}$ be the empirical distribution function. Let $\bar{Y}$ be the sample mean for $\mathcal{S}$. Let $\mathcal{S}^* = \{Y_1^*, \cdots, Y_n^*\}$ be a random sample taken with replacement from $\mathcal{S}$. Let $\bar{Y}^*$ be the sample mean for $\mathcal{S}^*$.

a. Show that

$$E_{\hat{P}}(\bar{Y}^*) = \bar{Y}$$

First define a functional of the population distribution, $\Theta(P)$, along with a parameter θ where

$$\theta = \Theta(P) = \int g(y)dP(y).$$

In this case, $\theta = \mu$, $\Theta(\cdot) = M(\cdot)$, and $g(y) = y$. Therefore, the above can be rewritten where the functional of the population is $M(P)$ and the parameter μ can be defined as,

$$\mu = M(P) = \int ydP(y).$$

Likewise, it follows that for the empirical distribution function, $\hat{P}$ that,

$$M(\hat{P}) = \int yd\hat{P}(y) = \bar{Y}.$$

Then the following holds true:

$$E_{\hat{P}}(\bar{Y}^*) = E_{\hat{P}}\left(\frac{\sum_{i=1}^{n} Y_i^*}{n}\right) = \frac{\sum_{i=1}^{n} E_{\hat{P}}(Y_i^*)}{n}$$

$$= \frac{n}{n} E_{\hat{P}}(Y_1^*) \because Y_1^*, \cdots, Y_n^* \sim i.i.d.\, \hat{P}$$

$$= E_{\hat{P}}(Y_1^*) = \int y_1^* \hat{p}(y)dy = \int y_1^* d\hat{P}(y)$$

$$= \int yd\hat{P}(y) \because y_1^* \text{ must be a value from the original sample } \mathcal{S}$$

$$= M(\hat{P}) = \bar{Y} \blacksquare$$

b. Show that

$$E_P(\bar{Y}^*) = \mu$$

Using the same rules that were mentioned in part (a), the following also holds true:

$$E_P(\bar{Y}^*) = E_P\left(\frac{\sum_{i=1}^{n} Y_i^*}{n}\right)$$

$$= \frac{n}{n} E_P(Y_1^*) \because Y_1^*, \cdots, Y_n^* \sim i.i.d.\, \hat{P}$$

$$= E_P(Y_1^*) = \int y_1^* p(y)dy = \int y_1^* dP(y)$$

$$= \int ydP(y) \because y_1^* \text{ must be a value from the original sample } \mathcal{S}$$

$$= M(P) = \mu \blacksquare$$

4. In this problem we will compare the results of a normal bootstrap sample to those of a balanced bootstrap sample.
    a. Draw a random sample of size $n = 100$ from the standard Normal distribution, $N(0,1)$, and calculate the sample mean, $\hat{\mu}$.

First $\hat{\mu}$ needs to be calculated.

$$X \sim \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}; \theta = (\mu = 0, \sigma^2 = 1)^\top$$

$$L(\theta) = f(x_1, \cdots, x_n; \mu, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x_i-\mu)^2}{2\sigma^2}\right\}$$

$$= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{\sum_{i=1}^{n}(x_i-\mu)^2}{2\sigma^2}\right\}$$

$$l(\theta) = \ln L(\theta) = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\sum_{i=1}^{n}(x_i-\mu)^2}{2\sigma^2}$$

$$\frac{\partial l(\theta)}{\partial \mu} = \frac{\sum_{i=1}^{n}(x_i-\mu)}{\sigma^2} \overset{set\ to}{=} 0 \Rightarrow \sum_{i=1}^{n} x_i = n\mu \Rightarrow \hat{\mu} = \frac{\sum_{i=1}^{n} x_i}{n} = \bar{x}$$

Using a seed of 999 in R, the sample mean is calculated to be $\boxed{\hat{\mu} = \bar{x} \approx -0.1072}$

    b. Use the standard bootstrap method to generate $B = 10$ bootstrap pseudodata sets. For each data set calculate the sample mean, $\hat{\mu}_j^*$. Then find the bootstrapped bias corrected estimator, $\hat{\mu}_{Bias}$, and the bootstrapped estimation of the variance, $\hat{V}(\hat{\mu})$.

$$\hat{\mu}_{Bias} = E^*[\hat{\mu}^* - \hat{\mu}] = E^*[\hat{\mu}^*] - \hat{\mu} = \bar{\mu}^* - \hat{\mu}$$

where

$$\bar{\mu}^* = \frac{\sum_{j=1}^{B} \hat{\mu}_j^*}{B}$$

The result using R is $-0.1249573 - (-0.1071936)$ $\boxed{\approx -0.01776 = \hat{\mu}_{Bias}}$

While $\mu_{Bias} = \hat{\mu} - [\bar{\mu}^* - \hat{\mu}] = 2\hat{\mu} - \bar{\mu}^* = 2 * (-0.1071936) - (-0.1249573) \approx -0.0894.$

The bootstrapped estimation of the variance, $\hat{V}(\hat{\mu})$ is calculated as follows:

$$\hat{V}(\hat{\mu}) \approx \hat{V}(\hat{\mu}^*) = \frac{1}{B-1}\sum_{j=1}^{B}\left(\hat{\mu}_j^* - \bar{\mu}^*\right)^2$$

The result using R is

$$\frac{1}{10-1}\sum_{j=1}^{B}\left(\hat{\mu}_j^* - (-0.1249573)\right)^2 \boxed{\approx 0.008374}$$

c. Repeat part (b) using the balanced bootstrap method.

In this case, the following occurred:

$$\hat{\mu}_{Balanced-Bias} = E^*[\hat{\mu}^* - \hat{\mu}] = E^*[\hat{\mu}^*] - \hat{\mu} = \bar{\mu}^* - \hat{\mu}$$

$$= (-0.1071936) - (-0.1071936) \boxed{= 0}$$

(with the exception that this time, $\hat{\mu}^*$ values are obtained through the balanced bootstrap method)

While $\mu_{Bias} = \hat{\mu} - [\bar{\mu}^* - \hat{\mu}] = 2\hat{\mu} - \bar{\mu}^* = 2 * (-0.1071936) - (-0.1071936) \approx -0.1072.$

Likewise, using $\hat{\mu}^*$ values obtained through the balanced bootstrap method, $\hat{V}(\hat{\mu})$ is calculated as follows:

$$\frac{1}{10-1}\sum_{j=1}^{B}\left(\hat{\mu}_j^* - (-0.1071936)\right)^2 \boxed{\approx 0.0087245}$$

d. Discuss your results.

Utilizing the balanced bootstrap method, the bias of the estimate $\hat{\mu}$ is reduced to 0, while using normal bootstrap the bias of the same estimate is approximately -0.01776. This result is expected, since when using balanced bootstrap, the values in the bootstrap samples, $\{\mathcal{X}_1^*, \cdots, \mathcal{X}_B^*\}$, occur with the same relative frequency as they do in the original sample $\mathcal{X}$. Therefore, the $\bar{\mu}^*$ (the average of the balanced bootstrap sample means) will have the same value as $\hat{\mu}$ (the original sample mean). This eliminates a source of potential Monte Carlo error.

In reducing the bias of the estimator, the variance increases from approximately 0.008374 to 0.0087245. The increase in variance is quite small compared to the reduction in bias. Therefore, it seems that in this case it's worth using the balanced bootstrap method to reduce the bias of the estimator $\hat{\mu}$.

Another interesting note is that the bias of the parameter μ (μ_{Bias}) itself also increases from approximately -0.0894 to -0.1072. Also, the notes mention that μ_{Bias} should be less than $\hat{\mu}_{Bias}$ which in both cases it is not. However, in the case of the balanced bootstrap it shouldn't be expected that μ_{Bias} would be smaller since $\hat{\mu}_{Balanced-Bias}$ is 0.

Appendix

```r
### Problem 1
# part (a)
stomach <- c(25, 42, 45, 46, 51, # Load data
  103, 124, 146, 340, 396, 412, 876, 1112)
breast <- c(24, 40, 719, 727, 791,
  1166, 1235, 1581, 1804, 3460, 3808)
log_stomach <- log(stomach)
log_breast <- log(breast)

par(mfrow=c(1,2))
boxplot(stomach)
boxplot(breast)
dev.off()

par(mfrow=c(1,2))
boxplot(log(stomach))
boxplot(log(breast))
dev.off()

mean(log_stomach) # 4.96792
mean(log_breast) # 6.558603

### Finds the bootstrap t C.I. on log scale
bootstrap_t <- function(df, B = 1e3, alpha = 0.05) {
  log_df <- log(df) # Take log scale
  n <- length(df) # Initialize variables
  T_F <- mean(log_df)
  data_matrix <- matrix(NA, nrow = B, ncol = length(log_df))
  theta_matrix <- matrix(NA, nrow = B)
  R_X_F_matrix <- matrix(NA, nrow = B)
  set.seed(999) # Set seed
  for (i in 1:B) { # Bootstrap sample
    data_matrix[i,] <- sample(log_df, size = length(log_df), replace = TRUE)
    theta_matrix[i] <- mean(data_matrix[i,])
    V_F_star <- (1 / (n - 1)) * sum((data_matrix[i,] - theta_matrix[i])^2)
    R_X_F_matrix[i] <- (theta_matrix[i] - T_F) / sqrt(V_F_star)
  }

  theta_bar <- mean(theta_matrix) # mean bootstrap theta
  # variance of bootstrap theta
  var_theta_hat_star <- sum((theta_matrix - theta_bar)^2) / (B - 1)

  # Find quantiles
  xi <- quantile(R_X_F_matrix, probs = c(alpha / 2, 1 - alpha / 2))
  names(xi) <- NULL
  bootstrap_t <- c(T_F - sqrt(var_theta_hat_star) * xi[2],
                   T_F - sqrt(var_theta_hat_star) * xi[1])
  return(bootstrap_t)
}
stomach_bootstrap_t_CI <- bootstrap_t(df = stomach, B = 1e4, alpha = 0.05) # 4.772891 5.195032
breast_bootstrap_t_CI <- bootstrap_t(df = breast, B = 1e4, alpha = 0.05) # 5.891237 6.803640

# part (b)
# H_0: no diff. in mean survival times (mu_log_breast = mu_log_stomach)
# H_1: there is a diff. in mean survival times (mu_log_breast =/= mu_log_stomach)
n1 <- length(log_stomach)
n2 <- length(log_breast)

t0 <- mean(log_stomach) - mean(log_breast)
```

```r
m <- choose((n1 + n2), n2)
m_combinations <- t(combn(c(log_stomach, log_breast), n2)) # m combinations of both data
stomach_breast <- c(log_stomach, log_breast) # Combined data
t_matrix <- matrix(NA, nrow = m) # Empty matrix for t_i

for (i in 1:m) { # Calculate t_i for all m combinations of the data
  ys <- m_combinations[i,]
  # Reference: https://stackoverflow.com/questions/5812478/how-i-can-select-rows-from-a-datafr
ame-that-do-not-match
  xs <- subset(stomach_breast, !(stomach_breast %in% ys))
  xbar_new <- mean(xs); ybar_new <- mean(ys)
  t_matrix[i] <- xbar_new - ybar_new
}
boxplot(t_matrix)
k <- which(t0 == sort(t_matrix, decreasing = TRUE))
k / m # 0.9925185
# rej. H_0 with sig. level. 99.25%

# part (c)
# percentile method
percentile_method <- function(df = breast, B = 1e3, alpha = 0.05, take_log = TRUE) {
  if (take_log == TRUE) {
    log_df <- log(df) # Take log scale
  } else {
    log_df <- df
  }
  data_matrix <- matrix(NA, nrow = B, ncol = length(log_df))
  theta_matrix <- matrix(NA, nrow = B)
  set.seed(888) # Set seed
  for (i in 1:B) { # Bootstrap sample
    data_matrix[i,] <- sample(log_df, size = length(log_df), replace = TRUE)
    theta_matrix[i] <- mean(data_matrix[i,])
  }

  percentile_CI <- quantile(theta_matrix, probs = c(alpha / 2, 1 - alpha / 2))
  names(percentile_CI) <- NULL

  return(percentile_CI)
}
breast_percentile_CI <- percentile_method(df = breast) # 5.617261 7.375876
exp(breast_percentile_CI) # 275.1347 1596.9896
exp(breast_bootstrap_t_CI) # 361.8524 901.1214
mean(breast) # 1395.909
mean(log_breast) # 6.558603
breast_percentile_CI_nonlog <- percentile_method(df = breast, # 778.1273 2145.3841
                                                 take_log = FALSE)

stomach_percentile_CI <- percentile_method(df = stomach) # 4.352572 5.631810
exp(stomach_percentile_CI) # 77.67798 279.16689
exp(stomach_bootstrap_t_CI) # 118.2606 180.3738
mean(stomach) # 286
mean(log_stomach) # 4.96792
stomach_percentile_CI_nonlog <- percentile_method(df = stomach, # 133.30 492.25
                                                  take_log = FALSE)
### Problem 2
# Cauchy distribution
xs <- seq(-4, 4, length.out = 1e4)
plot(xs, dnorm(x = xs, mean = 0, sd = 1), type = 'l',
     main = 'Normal and Cauchy', xlab = 'x', ylab = 'f(x)')
lines(xs, dcauchy(x = xs, location = 0, scale = 2), type = 'l', col = 'red')
legend("topleft", legend = c('Normal', 'Cauchy'), col = c('black', 'red'), lty = c(1,1))
```

```r
par(mfrow=c(2,2))
cauchy_bootstrap <- function(sample_size = 10, B = 1e4, alpha = 0, beta = 2) {
  set.seed(888)
  cauchy_sample <- rcauchy(n = sample_size, location = alpha, scale = beta) # Original sample
  data_matrix <- matrix(NA, nrow = B, ncol = length(cauchy_sample)) # Initialize variables
  theta_matrix <- matrix(NA, nrow = B)
  set.seed(888) # Set seed
  for (i in 1:B) { # Bootstrap sample
    data_matrix[i,] <- sample(cauchy_sample, size = length(cauchy_sample), replace = TRUE)
    theta_matrix[i] <- mean(data_matrix[i,])
  }
  hist(theta_matrix, main = paste('Histogram of the Sample Mean (n=',sample_size,')'),
       xlab = 'Bootstrap Sample Mean',
       sub = paste('Original Sample Mean:', round(mean(cauchy_sample), 4),
                   ' Bootstrap Sample Mean:', round(mean(theta_matrix), 4)))
}
cauchy_bootstrap(sample_size = 1e1)
cauchy_bootstrap(sample_size = 5e1)
cauchy_bootstrap(sample_size = 1e2)
cauchy_bootstrap(sample_size = 1e3)

# Uniform distribution
xs <- seq(0, 10, length.out = 1e4)
plot(xs, dunif(xs, 0, 1e4), type = 'l', ylim = c(0, 0.00014))
segments(x0 = c(0,10), y0 = 0, x1 = c(0,10), y1 = dunif(xs, 0, 1e4))
abline(h = 0)

theta <- 10
set.seed(999)
unif_sample <- runif(n = 1e1, min = 0, max = theta)
sample(unif_sample, size = length(unif_sample), replace = TRUE)
theta_hat <- max(unif_sample)

unif_bootstrap <- function(sample_size = 100, B = 1e4, theta_max = 10) {
  set.seed(999) # Set seed
  unif_sample <- runif(n = sample_size, min = 0, max = theta_max) # Original sample
  data_matrix <- matrix(NA, nrow = B, ncol = sample_size) # Initialize variables
  theta_matrix <- matrix(NA, nrow = B)
  set.seed(999) # Set seed
  for (i in 1:B) { # Bootstrap sample
    data_matrix[i,] <- sample(unif_sample, size = sample_size, replace = TRUE)
    theta_matrix[i] <- max(data_matrix[i,])
  }
  hist(theta_matrix, main = paste('Histogram of the Maximum Order Statistic (n=',sample_size,
')'),
       xlab = 'Bootstrap Maximum Order Statistic',
       sub = paste('Original Max. Order Statistic:', round(max(unif_sample), 4),
                   ' Mean Bootstrap Max. Order Statistic:', round(mean(theta_matrix), 4)))
}
par(mfrow = c(2,2))
unif_bootstrap(sample_size = 1e1, B = 1e4)
unif_bootstrap(sample_size = 1e2, B = 1e4)
unif_bootstrap(sample_size = 1e3, B = 1e4)
unif_bootstrap(sample_size = 1e4, B = 1e4)
dev.off()

### Problem 4
# part (a)
set.seed(999)
sample_size <- 1e2
```

```r
random_sample <- rnorm(n = sample_size, mean = 0, sd = 1)
sample_mean <- mean(random_sample)

# part (b)
B <- 1e1
set.seed(999) # Set seed
data_matrix <- matrix(NA, nrow = B, ncol = sample_size) # Initialize variables
theta_matrix <- matrix(NA, nrow = B)
set.seed(999) # Set seed
for (i in 1:B) { # Bootstrap sample
  data_matrix[i,] <- sample(random_sample, size = sample_size, replace = TRUE)
  theta_matrix[i] <- mean(data_matrix[i,])
}
hist(theta_matrix, main = paste('Histogram of the Sample Mean (n=',sample_size,')'),
     xlab = 'Bootstrap Sample Mean',
     sub = paste('Original Sample Mean:', round(sample_mean, 4),
                 ' Mean Bootstrap Sample Mean:', round(mean(theta_matrix), 4)))
mu_bar_star <- mean(theta_matrix)
mu_bar_star - sample_mean
2 * sample_mean - mu_bar_star

(1 / (B - 1)) * sum((theta_matrix - mu_bar_star)^2)

# part (c)
concat_B <- rep(random_sample, B) # create n*B vector
set.seed(999)
balanced_bootstrap <- sample(concat_B, # permute n*B vector
                             size = length(concat_B), replace = FALSE)
# Reference: https://stackoverflow.com/questions/3318333/split-a-vector-into-chunks-in-r
balanced_groups <- split(balanced_bootstrap, # split into B groups
                         rep_len(1:B, length(balanced_bootstrap)))
balanced_theta <- unlist(lapply(balanced_groups, mean)) # calculate theta for each
names(balanced_theta) <- NULL

hist(balanced_theta, main = paste('Histogram of the Balanced Sample Mean (n=',sample_size,')
'),
     xlab = 'Balanced Bootstrap Sample Mean',
     sub = paste('Original Sample Mean:', round(sample_mean, 4),
                 ' Mean Bootstrap Sample Mean:', round(mean(balanced_theta), 4)))
mu_bar_star <- mean(balanced_theta)
mu_bar_star - sample_mean
2 * sample_mean - mu_bar_star
(1 / (B - 1)) * sum((theta_matrix - mu_bar_star)^2)
```