102B_Daren_Sathasivam_hw01

Daren Sathasivam

2024-04-07

```
source("bootsample.R")
source("bootstats.R")
```

Problem 1:

Consider the correlation coefficient r between two random variables X and Y. Recall that it is a measure of *linear association* between X and Y and takes values in the interal $[-1, 1]$ i.e. $(X, Y)\epsilon[-1, 1]$. (a.) Write code to simulate data from a *bivariate normal distribution* with mean vector $\mu = [0, 0]$ and correlation matrix:

- $R = \begin{bmatrix} 1 & r \\ r & 1 \end{bmatrix}$
- *Hint*: Use the **mvnorm** function in the R library **MASS**

```
library(MASS)
r <- 0.27
Sigma <- matrix(c(1, r, r, 1), 2, 2)
mu <- c(0, 0)
mvrnorm(n = 5, mu = mu, Sigma = Sigma)
```

```
##             [,1]        [,2]
## [1,] -1.70677239 -1.37204386
## [2,] -1.21510238 -0.18840715
## [3,]  0.05403796 -0.21877902
## [4,] -0.41758138 -0.02488981
## [5,] -2.06609116  0.25617604
```

(b.) Generate the simulated data from a bivariate multivariate normal distribution with mean vector $\mu = [0, 0]$ and correlation matrix R for the following cases: - 1. Sample size $n\epsilon\{20, 50, 100, 200\}$ and correlation coefficient r = 0

```
set.seed(1)
mu <- c(0, 0)
r <- 0
sample_size <- c(20, 50, 100, 200)
for (n in sample_size) {
  Sigma <- matrix(c(1, r, r, 1), 2, 2)
  data <- mvrnorm(n = n, mu = mu, Sigma = Sigma)
  corr_coef <- cor(data[, 1], data[, 2])
  cat("Sample size: ", n, "Correlation coefficient: ", r, "\n")
  print(tail(data, 5)) # Print tail to show that the sample size is correct
}
```

```
## Sample size:  20 Correlation coefficient:  0
##             [,1]        [,2]
```

```
## [16,]  0.4149946 -0.04493361
## [17,]  0.3942900 -0.01619026
## [18,]  0.0593134  0.94383621
## [19,] -1.1000254  0.82122120
## [20,] -0.7631757  0.59390132
## Sample size:  50 Correlation coefficient:  0
##             [,1]       [,2]
## [46,] 1.53644982  0.3329504
## [47,] 0.30097613  1.0630998
## [48,] 0.52827990 -0.3041839
## [49,] 0.65209478  0.3700188
## [50,] 0.05689678  0.2670988
## Sample size:  100 Correlation coefficient:  0
##               [,1]         [,2]
##  [96,] -0.732750042 -0.098178744
##  [97,] -0.946585640  0.560820729
##  [98,] -0.004398704 -1.186458639
##  [99,]  0.352322306  1.096777044
## [100,]  0.529695509 -0.005344028
## Sample size:  200 Correlation coefficient:  0
##              [,1]       [,2]
## [196,]  0.4513135 -0.7695923
## [197,] -0.9250848  0.3033610
## [198,]  0.1986208  1.2817374
## [199,] -1.1948510  0.6022228
## [200,] -0.4955447 -0.3070223
```

- 2. Sample size $n\epsilon\{20, 50, 100, 200\}$ and correlation coefficient r = 0.5

```r
r <- 0.5
for (n in sample_size) {
  Sigma <- matrix(c(1, r, r, 1), 2, 2)
  data <- mvrnorm(n = n, mu = mu, Sigma = Sigma)
  corr_coef <- cor(data[, 1], data[, 2])
  cat("Sample size: ", n, "Correlation coefficient: ", r, "\n")
  print(tail(data, 5)) # Print tail to show that the sample size is correct
}
```

```
## Sample size:  20 Correlation coefficient:  0.5
##             [,1]        [,2]
## [16,] -1.2719136 -2.25281593
## [17,]  1.2069627  2.29411297
## [18,]  0.8015351  0.94086211
## [19,]  0.9007858  0.51451375
## [20,] -1.1368234 -0.01323801
## Sample size:  50 Correlation coefficient:  0.5
##              [,1]       [,2]
## [46,]  0.05096352  0.2369987
## [47,] -1.18771394 -1.0112961
## [48,]  1.57679680  2.4926450
## [49,] -1.38292092 -1.0627442
## [50,]  0.16865456 -0.1980327
## Sample size:  100 Correlation coefficient:  0.5
##               [,1]        [,2]
##  [96,] -0.35302396 -1.31156749
```

```
##  [97,]  1.13140782 -0.47290244
##  [98,]  1.36020607 -0.48540335
##  [99,]  1.47761340  2.03335059
## [100,]  0.08602824  0.02590905
## Sample size:  200 Correlation coefficient:  0.5
##               [,1]       [,2]
## [196,]  1.35262193  0.7716506
## [197,] -0.60249396  1.1342734
## [198,]  0.62702588  0.4543506
## [199,]  0.52211951 -0.4158730
## [200,]  0.04631888 -0.2382748
```

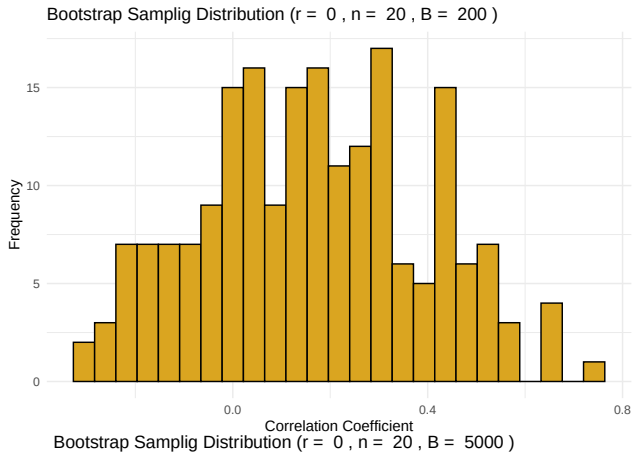
- 3. Sample size $n\epsilon\{20, 50, 100, 200\}$ and correlation coefficient r = 0.85

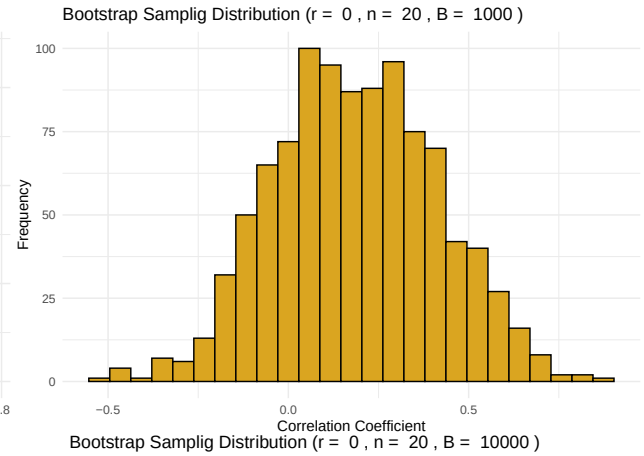
```
r <- 0.85
for (n in sample_size) {
  Sigma <- matrix(c(1, r, r, 1), 2, 2)
  data <- mvrnorm(n = n, mu = mu, Sigma = Sigma)
  corr_coef <- cor(data[, 1], data[, 2])
  cat("Sample size: ", n, "Correlation coefficient: ", r, "\n")
  print(tail(data, 5)) # Print tail to show that the sample size is correct
}
```

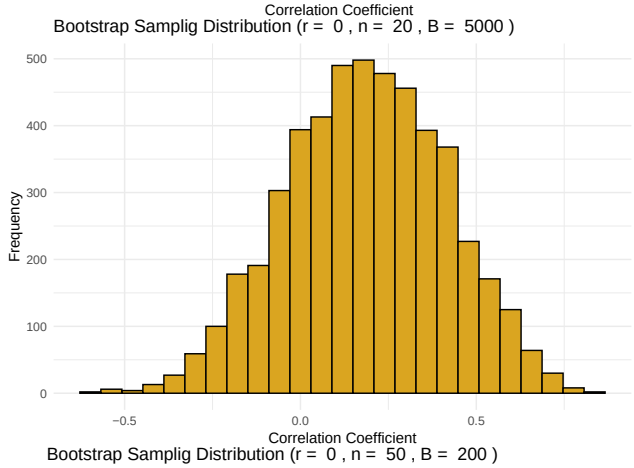
```
## Sample size:  20 Correlation coefficient:  0.85
##             [,1]        [,2]
## [16,]  0.1744441  0.04561933
## [17,] -0.3256948 -0.42884122
## [18,]  1.3286673  1.84970648
## [19,] -1.1539714 -1.23832874
## [20,] -1.3848680 -0.36925645
## Sample size:  50 Correlation coefficient:  0.85
##             [,1]       [,2]
## [46,] -2.2059461 -1.4858264
## [47,]  2.0649055  1.8193582
## [48,]  0.6398970  0.1972471
## [49,]  1.0336456  0.9863944
## [50,] -0.7343688  0.1340039
## Sample size:  100 Correlation coefficient:  0.85
##              [,1]       [,2]
##  [96,] -1.0520322 -0.3077153
##  [97,] -1.9477941 -1.8856304
##  [98,]  1.7795069  1.8749786
##  [99,] -0.7126503 -0.7419428
## [100,] -0.2471815 -0.4798310
## Sample size:  200 Correlation coefficient:  0.85
##              [,1]       [,2]
## [196,]  1.6701694  0.8136419
## [197,] -0.1922027 -1.0102107
## [198,] -0.5412265 -1.1421519
## [199,]  0.1458475  0.1299051
## [200,]  0.7754154  1.9659916
```

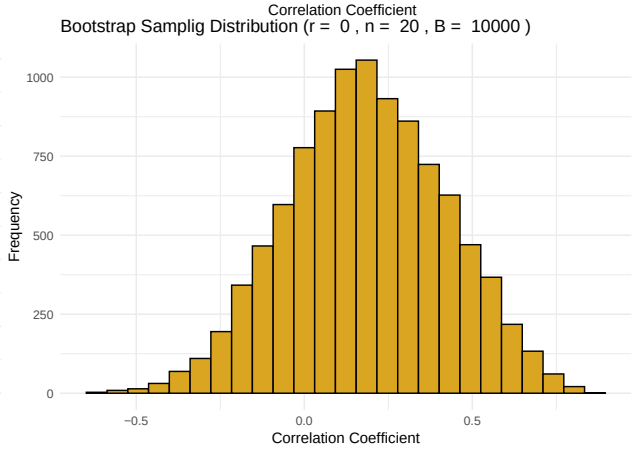
(c.) Obtain the *bootstrap sampling distribution* of the sample correlation coefficient $\hat{r}$ for the three cases in part (b.), for the following number of bootstrap replicates $B\epsilon\{200, 1000, 5000, 10000\}$.

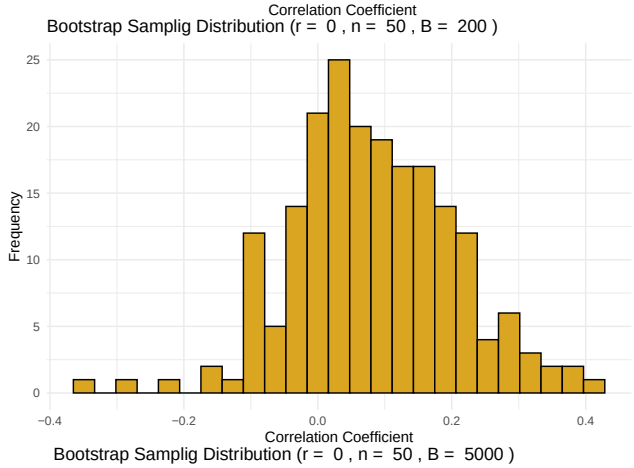
```r
# Initializers
library(ggplot2)
mu <- c(0, 0)
Sigma <- matrix(c(1, r, r, 1), 2, 2)
cor_coef <- c(0, 0.5, 0.85)
sample_size <- c(20, 50, 100, 200)
B_values <- c(200, 1000, 5000, 10000)

# Correlation coefficient function
cor_fun <- function(data) {
  cor(data[, 1], data[, 2])
}
# Bootstrap sampling function - Referenced "bootsample.R"
my_bootsampling <- function(data, B) {
  n <- nrow(data)
  bootsamples <- replicate(B, {
    resample <- sample(1:n, n, replace = TRUE)
    sample_data <- data[resample, ]
    cor_fun(sample_data)
  })
  bootsamples
}

# Iterate through the different correlation coefficients and sample sizes
for (r in cor_coef) {
  Sigma <- matrix(c(1, r, r, 1), 2, 2)
  for (n in sample_size) {
    data <- mvrnorm(n = n, mu = mu, Sigma = Sigma)
    # Iterate through bootstrap replicates
    for (B in B_values) {
      bootstrap_samples <- my_bootsampling(data, B)
      mean_sample <- mean(bootstrap_samples)
      sd_sample <- sd(bootstrap_samples)
      p <- ggplot() +
        geom_histogram(aes(x = bootstrap_samples), bins = 25, fill = "goldenrod", color = "black") +
        # stat_function(fun = dnorm, args = list(mean = mean_sample, sd = sd_sample), color = "blue") +
        labs(title = paste("Bootstrap Samplig Distribution (r = ", r, ", n = ", n, ", B = ", B, ")"), x
        theme_minimal()
      print(p)
    }
  }
}
```

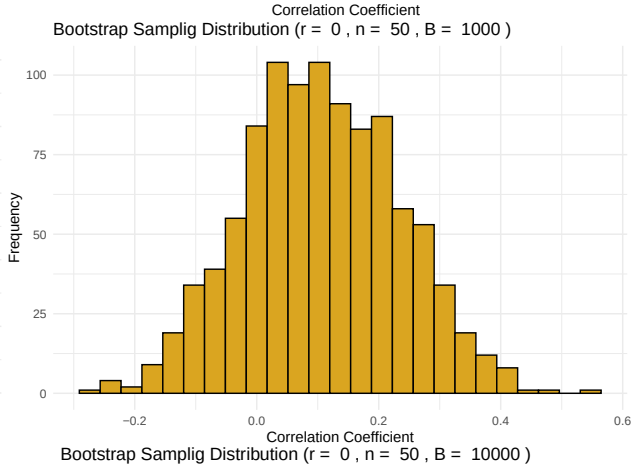

Bootstrap Samplig Distribution (r = 0 , n = 20 , B = 200)
Frequency
Correlation Coefficient

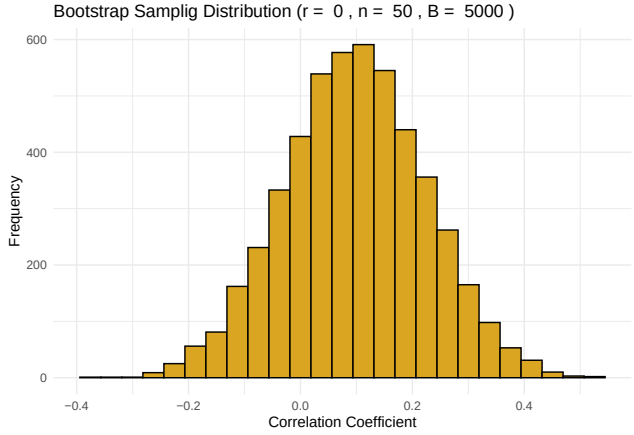

Bootstrap Samplig Distribution (r = 0 , n = 20 , B = 1000)
Frequency
Correlation Coefficient

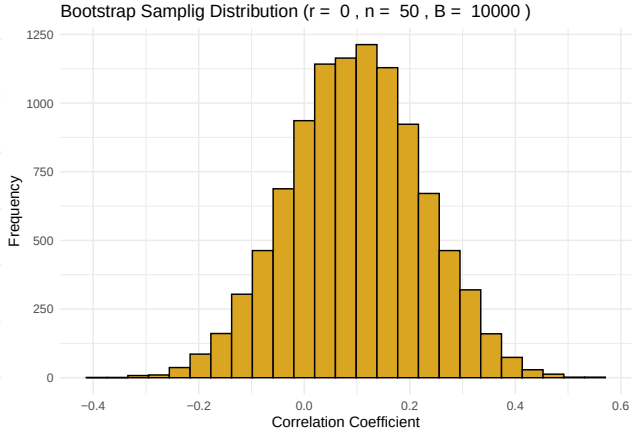

Bootstrap Samplig Distribution (r = 0 , n = 20 , B = 5000)
Frequency
Correlation Coefficient

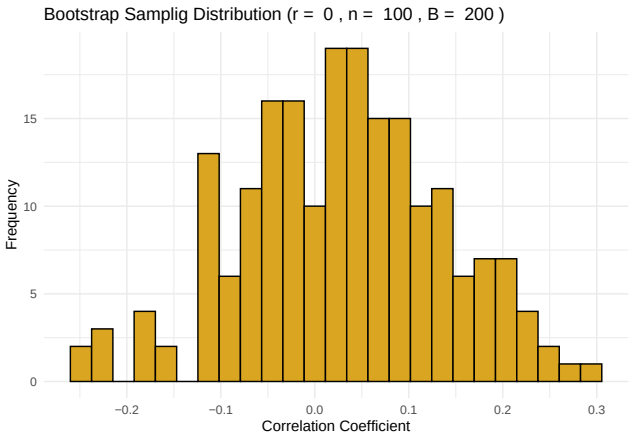

Bootstrap Samplig Distribution (r = 0 , n = 20 , B = 10000)
Frequency
Correlation Coefficient

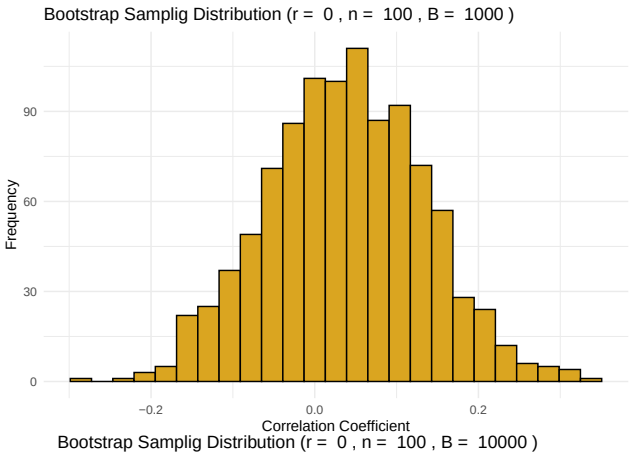

Bootstrap Samplig Distribution (r = 0 , n = 50 , B = 200)
Frequency
Correlation Coefficient

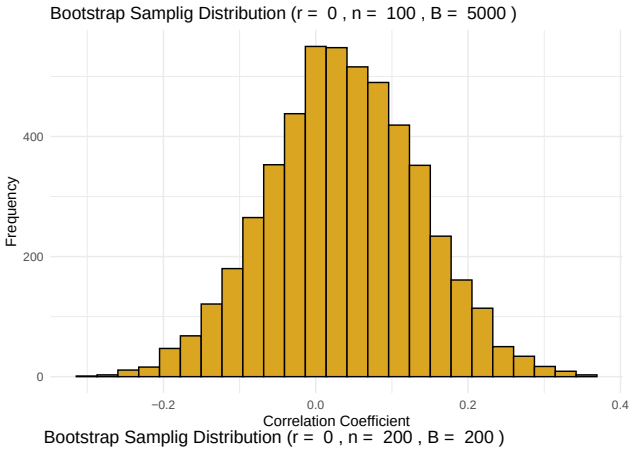

Bootstrap Samplig Distribution (r = 0 , n = 50 , B = 1000)
Frequency
Correlation Coefficient

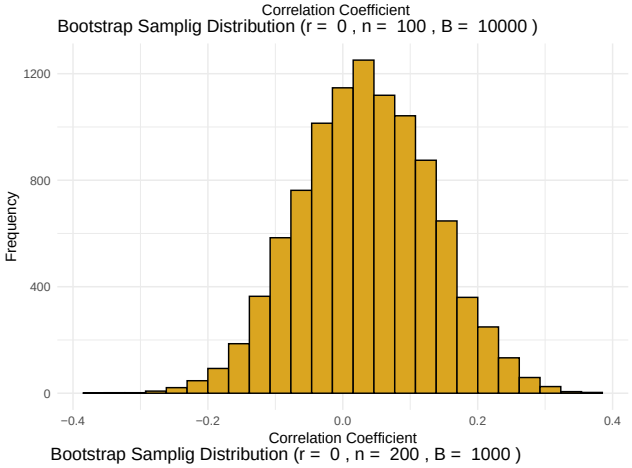

Bootstrap Samplig Distribution (r = 0 , n = 50 , B = 5000)
Frequency
Correlation Coefficient

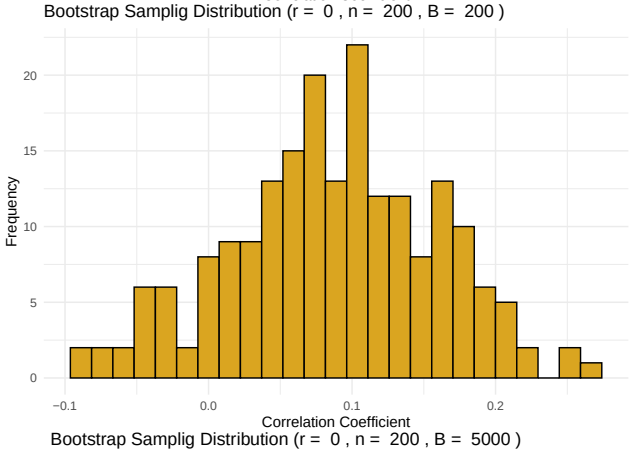

Bootstrap Samplig Distribution (r = 0 , n = 50 , B = 10000)
Frequency
Correlation Coefficient

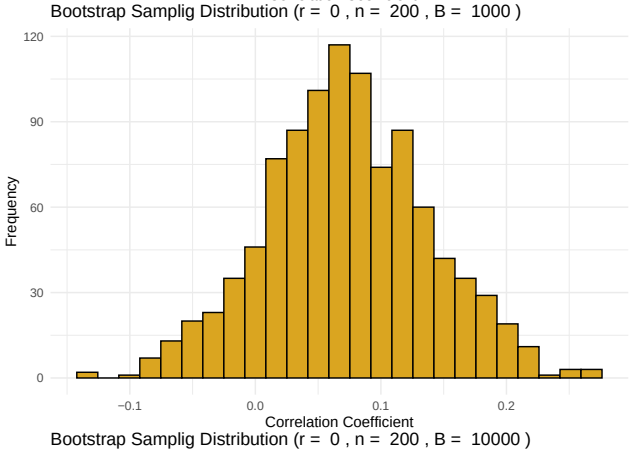

Bootstrap Samplig Distribution (r = 0 , n = 100 , B = 200)
Frequency
Correlation Coefficient

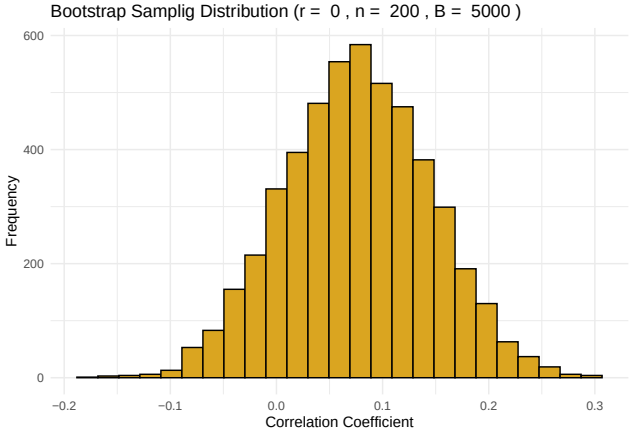

Bootstrap Samplig Distribution (r = 0 , n = 100 , B = 1000)
Frequency
Correlation Coefficient

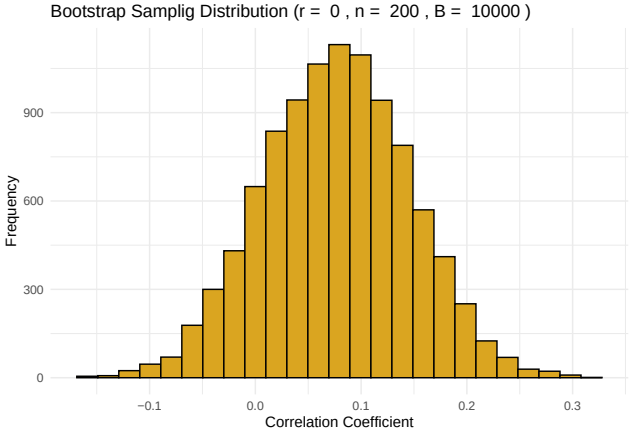

Bootstrap Samplig Distribution (r = 0 , n = 100 , B = 5000)
Frequency
Correlation Coefficient

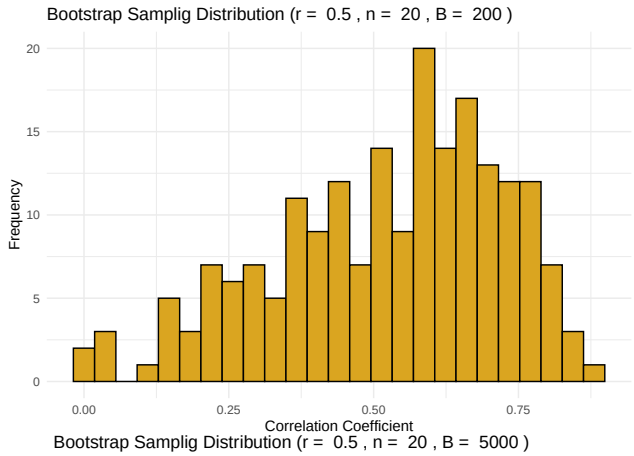

Bootstrap Samplig Distribution (r = 0 , n = 100 , B = 10000)
Frequency
Correlation Coefficient

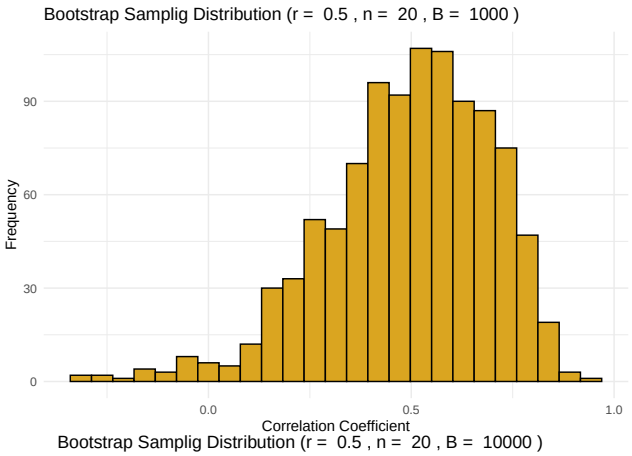

Bootstrap Samplig Distribution (r = 0 , n = 200 , B = 200)
Frequency
Correlation Coefficient

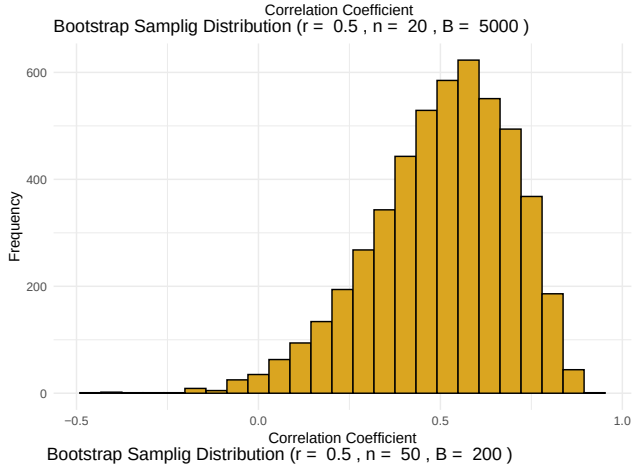

Bootstrap Samplig Distribution (r = 0 , n = 200 , B = 1000)
Frequency
Correlation Coefficient

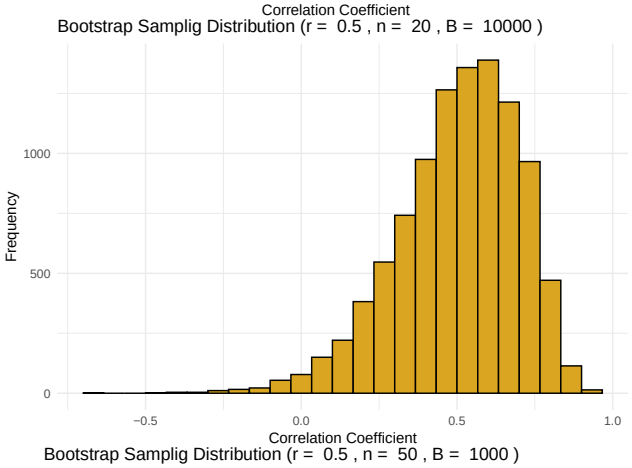

Bootstrap Samplig Distribution (r = 0 , n = 200 , B = 5000)
Frequency
Correlation Coefficient

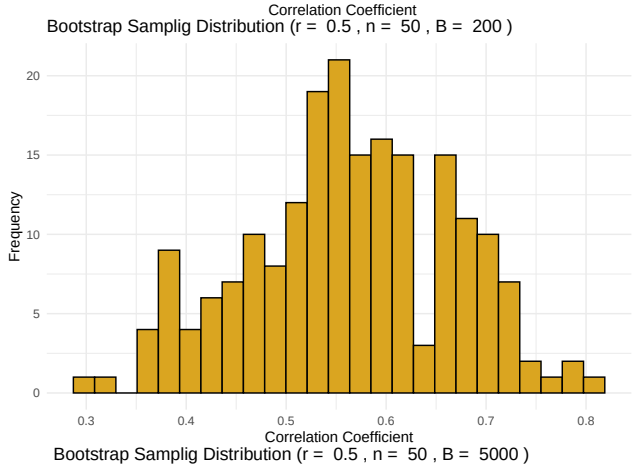

Bootstrap Samplig Distribution (r = 0 , n = 200 , B = 10000)
Frequency
Correlation Coefficient

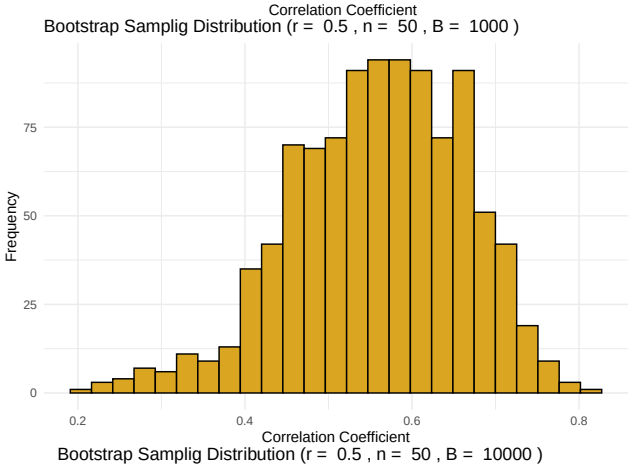

Bootstrap Samplig Distribution (r = 0.5 , n = 20 , B = 200)
Frequency
Correlation Coefficient

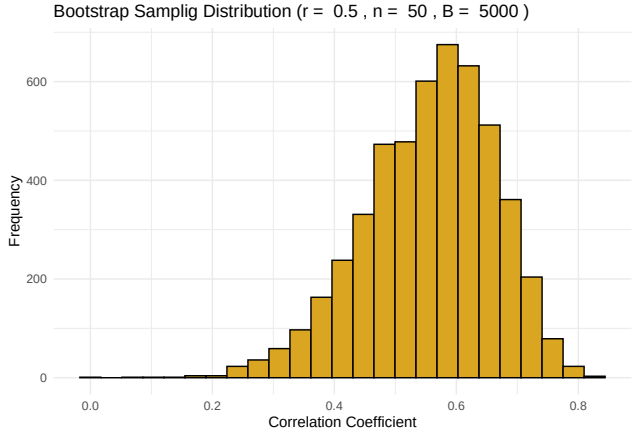

Bootstrap Samplig Distribution (r = 0.5 , n = 20 , B = 1000)
Frequency
Correlation Coefficient

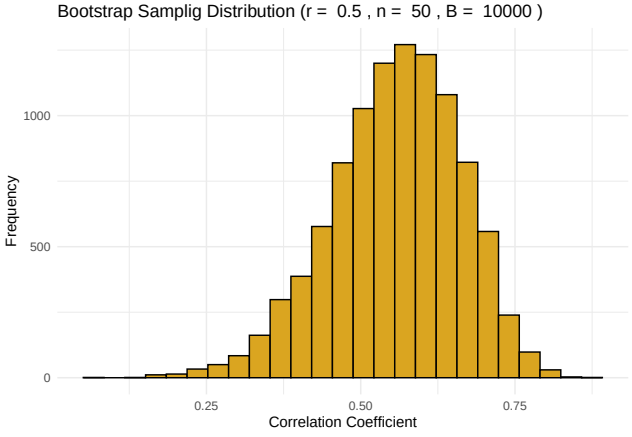

Bootstrap Samplig Distribution (r = 0.5 , n = 20 , B = 5000)
Frequency
Correlation Coefficient

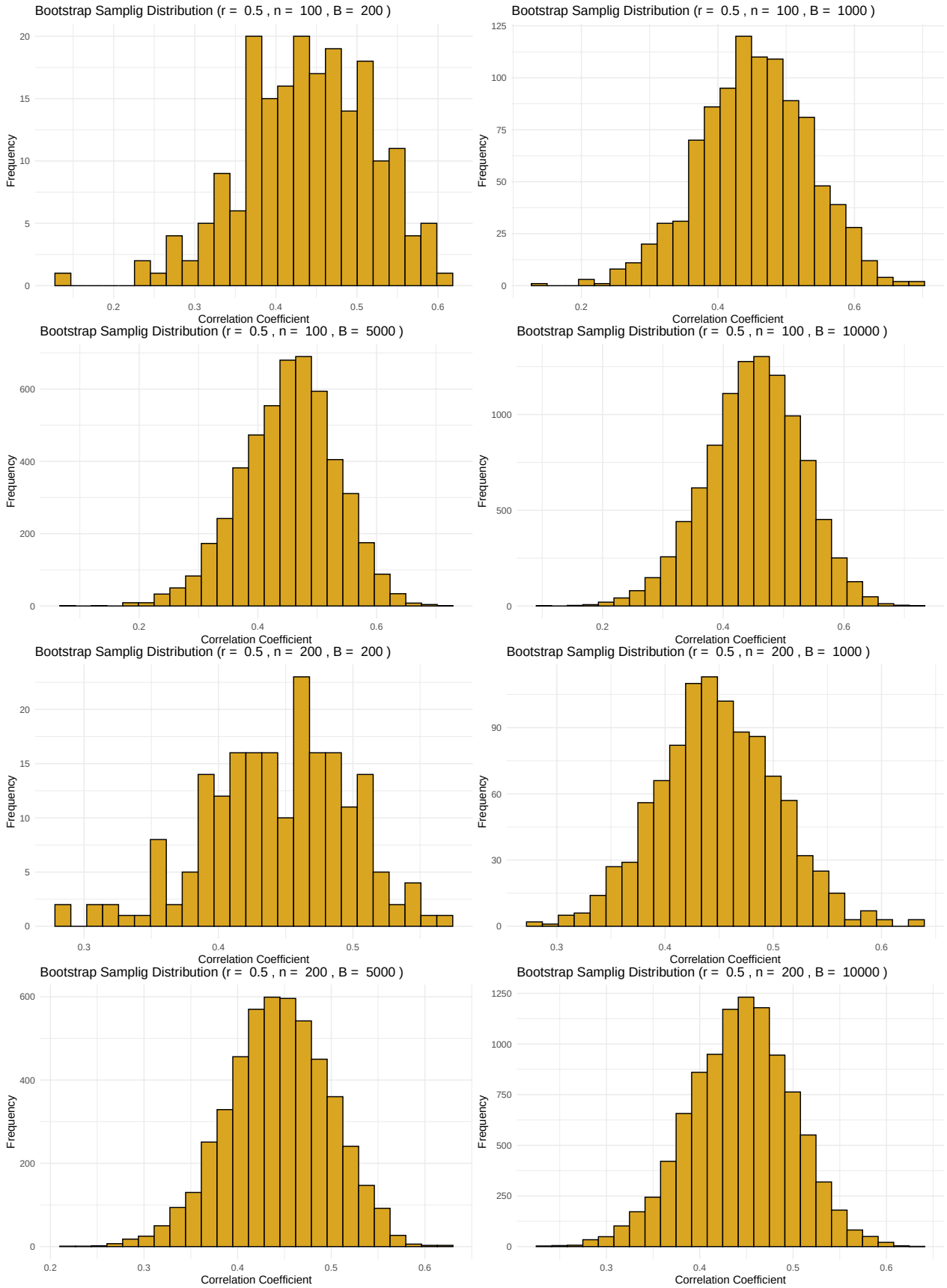

Bootstrap Samplig Distribution (r = 0.5 , n = 20 , B = 10000)
Frequency
Correlation Coefficient

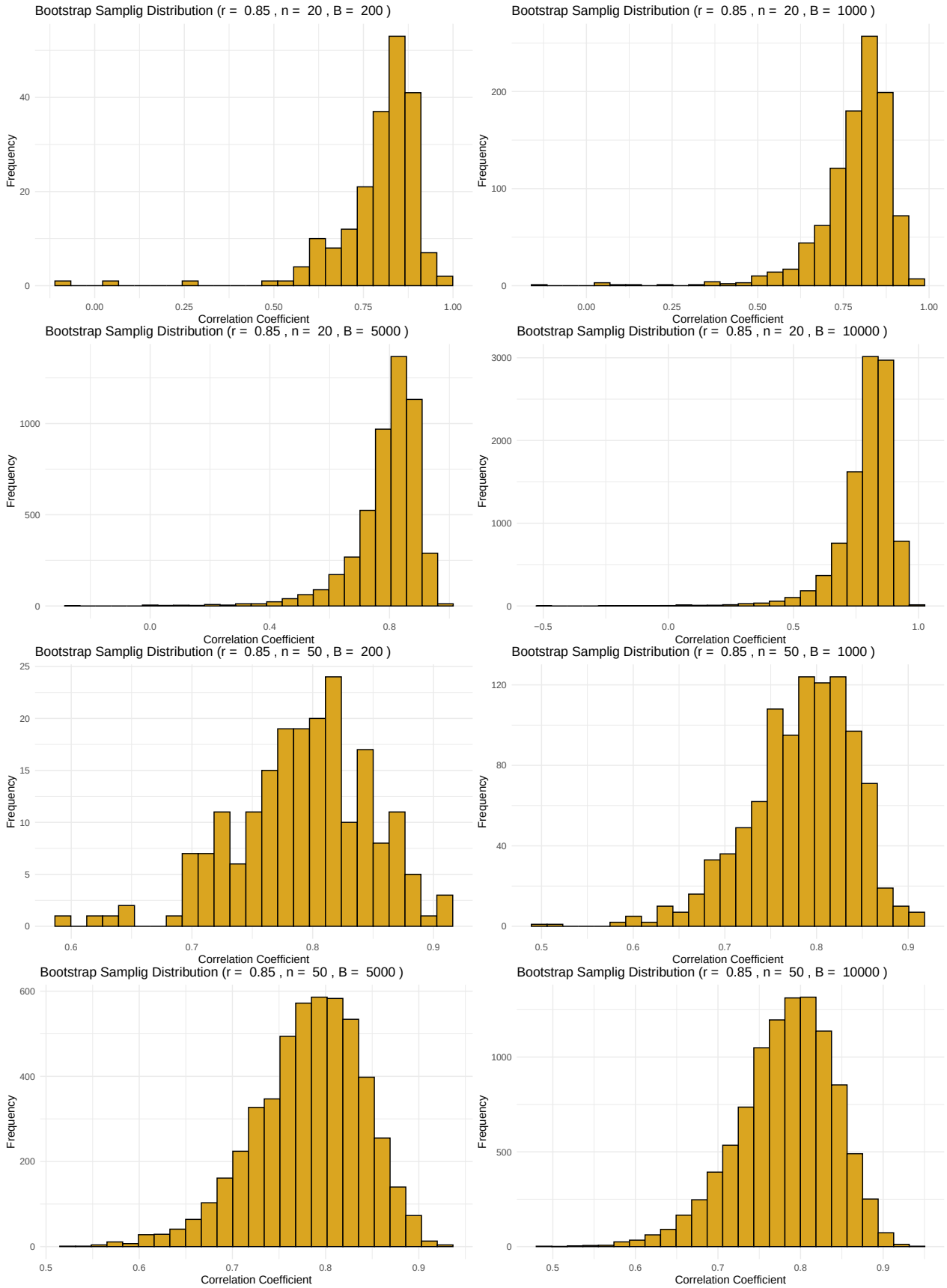

Bootstrap Samplig Distribution (r = 0.5 , n = 50 , B = 200)
Frequency
Correlation Coefficient

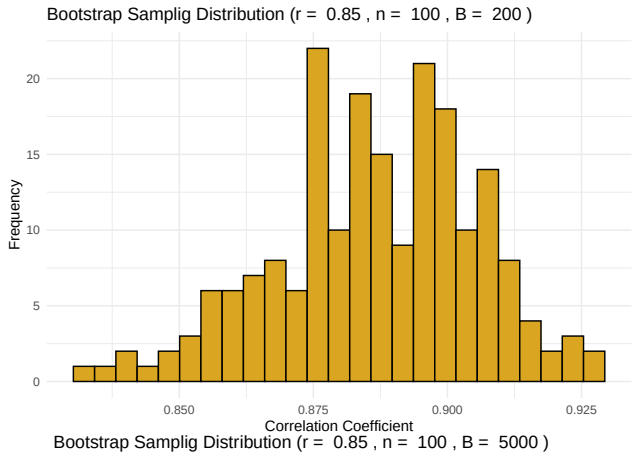

Bootstrap Samplig Distribution (r = 0.5 , n = 50 , B = 1000)
Frequency
Correlation Coefficient

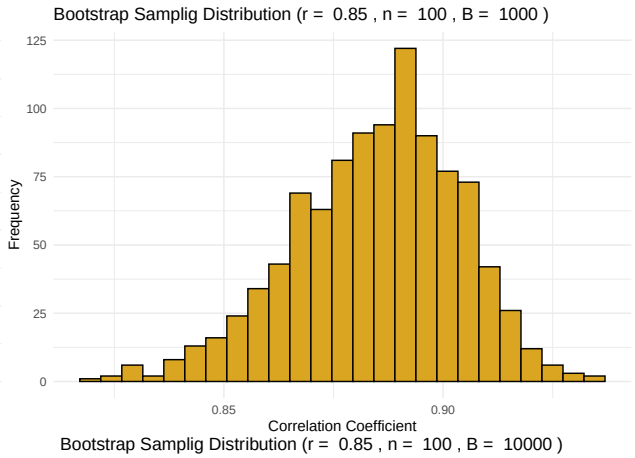

Bootstrap Samplig Distribution (r = 0.5 , n = 50 , B = 5000)
Frequency
Correlation Coefficient

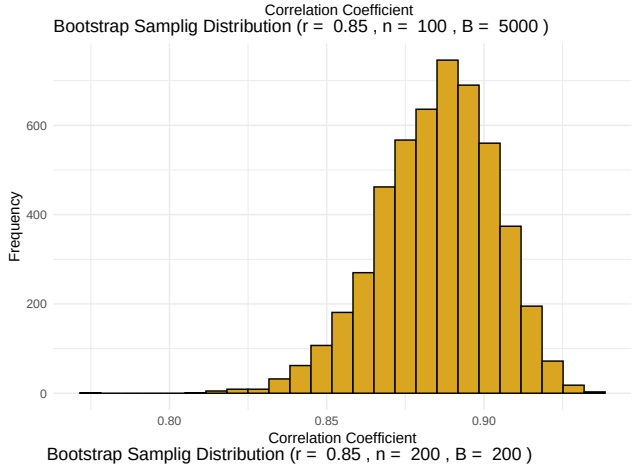

Bootstrap Samplig Distribution (r = 0.5 , n = 50 , B = 10000)
Frequency
Correlation Coefficient

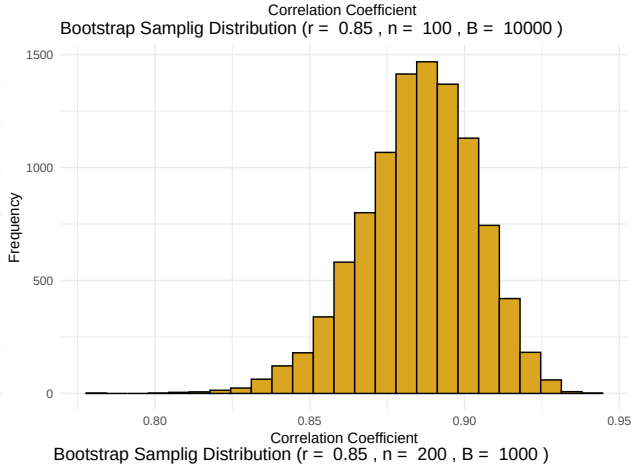

Bootstrap Samplig Distribution (r = 0.5 , n = 100 , B = 200)
Bootstrap Samplig Distribution (r = 0.5 , n = 100 , B = 1000)
Bootstrap Samplig Distribution (r = 0.5 , n = 100 , B = 5000)
Bootstrap Samplig Distribution (r = 0.5 , n = 100 , B = 10000)
Bootstrap Samplig Distribution (r = 0.5 , n = 200 , B = 200)
Bootstrap Samplig Distribution (r = 0.5 , n = 200 , B = 1000)
Bootstrap Samplig Distribution (r = 0.5 , n = 200 , B = 5000)
Bootstrap Samplig Distribution (r = 0.5 , n = 200 , B = 10000)
Frequency
Correlation Coefficient

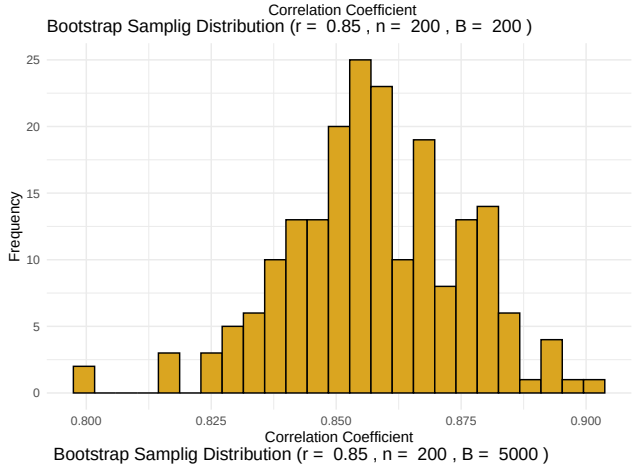

Bootstrap Samplig Distribution (r = 0.85 , n = 20 , B = 200)
Bootstrap Samplig Distribution (r = 0.85 , n = 20 , B = 1000)
Bootstrap Samplig Distribution (r = 0.85 , n = 20 , B = 5000)
Bootstrap Samplig Distribution (r = 0.85 , n = 20 , B = 10000)
Bootstrap Samplig Distribution (r = 0.85 , n = 50 , B = 200)
Bootstrap Samplig Distribution (r = 0.85 , n = 50 , B = 1000)
Bootstrap Samplig Distribution (r = 0.85 , n = 50 , B = 5000)
Bootstrap Samplig Distribution (r = 0.85 , n = 50 , B = 10000)
Frequency
Correlation Coefficient

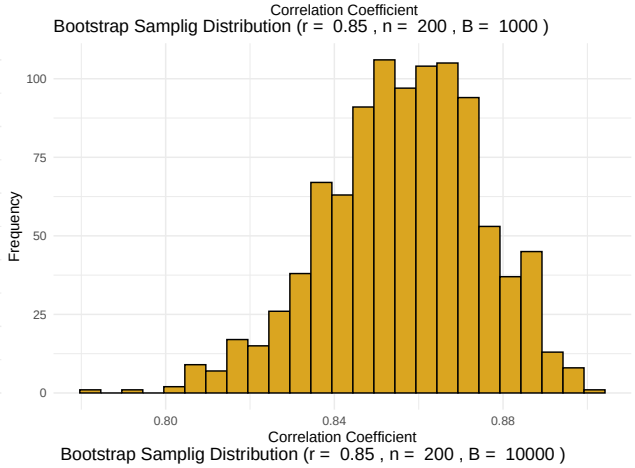

Bootstrap Samplig Distribution (r = 0.85 , n = 100 , B = 200)
Frequency
Correlation Coefficient

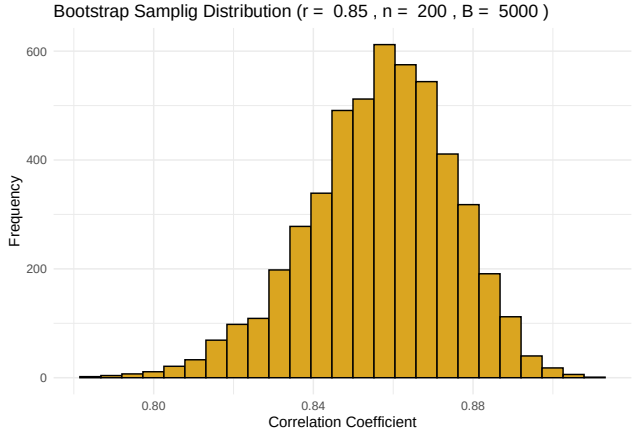

Bootstrap Samplig Distribution (r = 0.85 , n = 100 , B = 1000)
Frequency
Correlation Coefficient

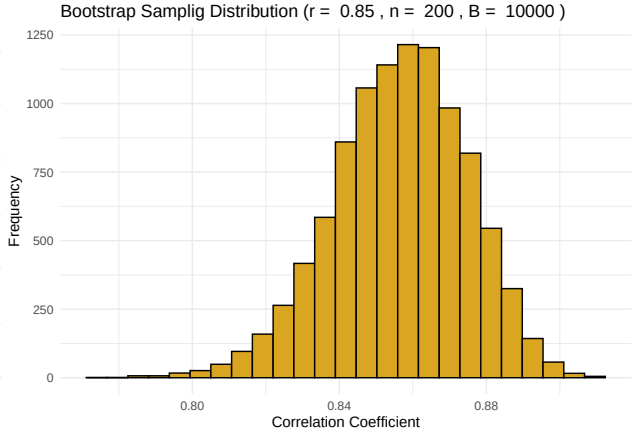

Bootstrap Samplig Distribution (r = 0.85 , n = 100 , B = 5000)
Frequency
Correlation Coefficient


Bootstrap Samplig Distribution (r = 0.85 , n = 100 , B = 10000)
Frequency
Correlation Coefficient


Bootstrap Samplig Distribution (r = 0.85 , n = 200 , B = 200)
Frequency
Correlation Coefficient


Bootstrap Samplig Distribution (r = 0.85 , n = 200 , B = 1000)
Frequency
Correlation Coefficient


Bootstrap Samplig Distribution (r = 0.85 , n = 200 , B = 5000)
Frequency
Correlation Coefficient


Bootstrap Samplig Distribution (r = 0.85 , n = 200 , B = 10000)
Frequency
Correlation Coefficient

- Comment on the results; in particular how the bootstrap sampling distribution behaves as a function of the sample size n, the number of bootstrap replicates B and the value of the correlation coefficient r.
  - For sample size n, larger sample sizes lead to tighter bootstrap distributions around the true correlation coefficient.
  - For the number of bootstrap replicates B, increasing the number of bootstrap replicates leads to a smoother and more accurate representation of the sampling distribution. Thus the bootstrap distribution approximates the true sampling distribution of $\hat{r}$ better.
  - For the correlation coefficient r, as r increases, the bootstrap sampling distribution shifts towards the true correlation coefficient value and the variablity decreases.

Problem 2:

```
# Load/Initializers
library(MASS)
library(ggplot2)
data(cats)
summary(cats)
```

```
##  Sex         Bwt             Hwt
##  F:47   Min.   :2.000   Min.   : 6.30
##  M:97   1st Qu.:2.300   1st Qu.: 8.95
##         Median :2.700   Median :10.10
##         Mean   :2.724   Mean   :10.63
##         3rd Qu.:3.025   3rd Qu.:12.12
##         Max.   :3.900   Max.   :20.50
```

```
# BWT in kg and HWT in g of 47F and 97M cats

# F/M Body Weight Data
female_bwt <- cats$Bwt[cats$Sex == "F"] # 47 obs
male_bwt <- cats$Bwt[cats$Sex == "M"] # 97 obs
# F/M Heart Weight Data
female_hwt <- cats$Hwt[cats$Sex == "F"] # 47 obs
male_hwt <- cats$Hwt[cats$Sex == "M"] # 97 obs
```

Part(a):

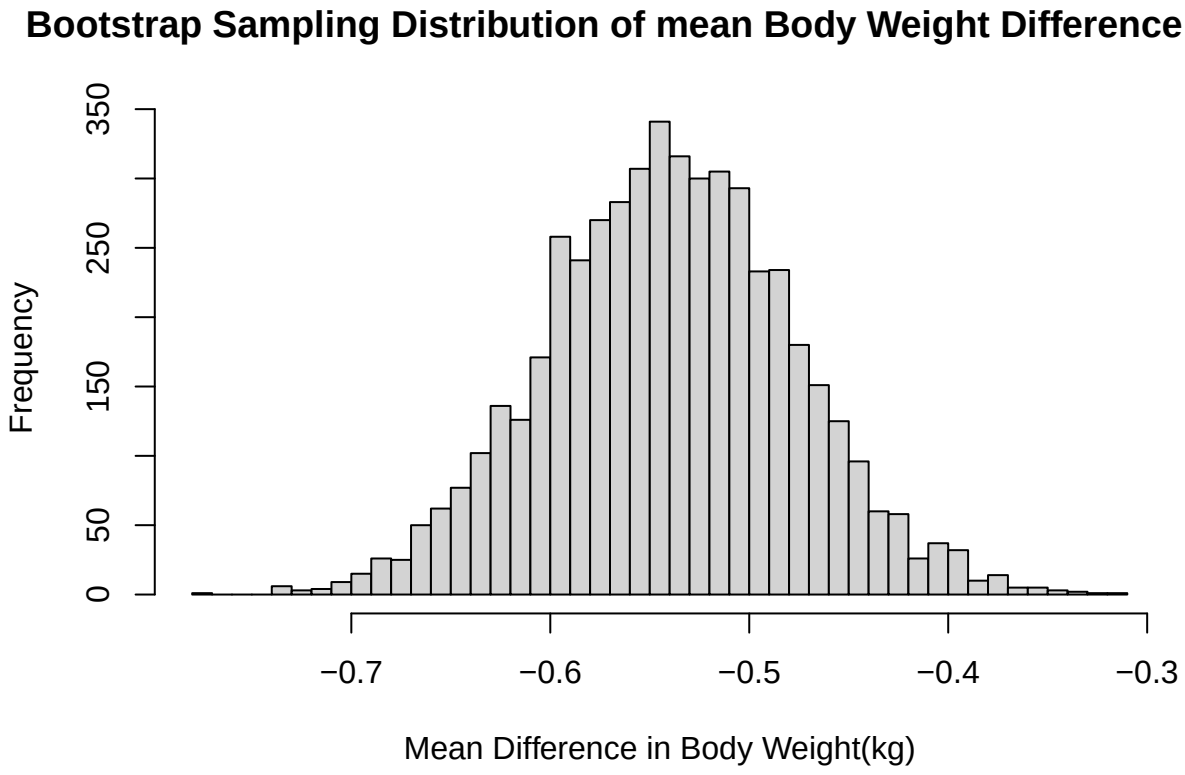
- Obtain the bootstrap sampling of the difference of sample means for body weight between female and male cats

```
# Number of bootstrap replicates
B <- 5000
# Function to perform bootstrap sampling and calculate mean differences
bootstrap_mean_diff <- function(f_data, m_data, B) {
  mean_diffs <- rep(NA, B)
  # Iterate through number of bootstrap replicates
  for (i in seq_along(mean_diffs)) {
    f_sample <- sample(f_data, replace = TRUE)
    m_sample <- sample(m_data, replace = TRUE)
    mean_diffs[i] <- mean(f_sample) - mean(m_sample)
  }
  mean_diffs
}
# Perform bootstrap sampling
```

```r
bwt_diffs <- bootstrap_mean_diff(female_bwt, male_bwt, B)
summary(bwt_diffs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -0.7735 -0.5825 -0.5404 -0.5403 -0.4991 -0.3160
```

```r
# Plot bootstrap sampling distribution
hist(bwt_diffs, breaks = 40, main = "Bootstrap Sampling Distribution of mean Body Weight Difference (F-
     xlab = "Mean Difference in Body Weight(kg)",
     ylab = "Frequency")
```

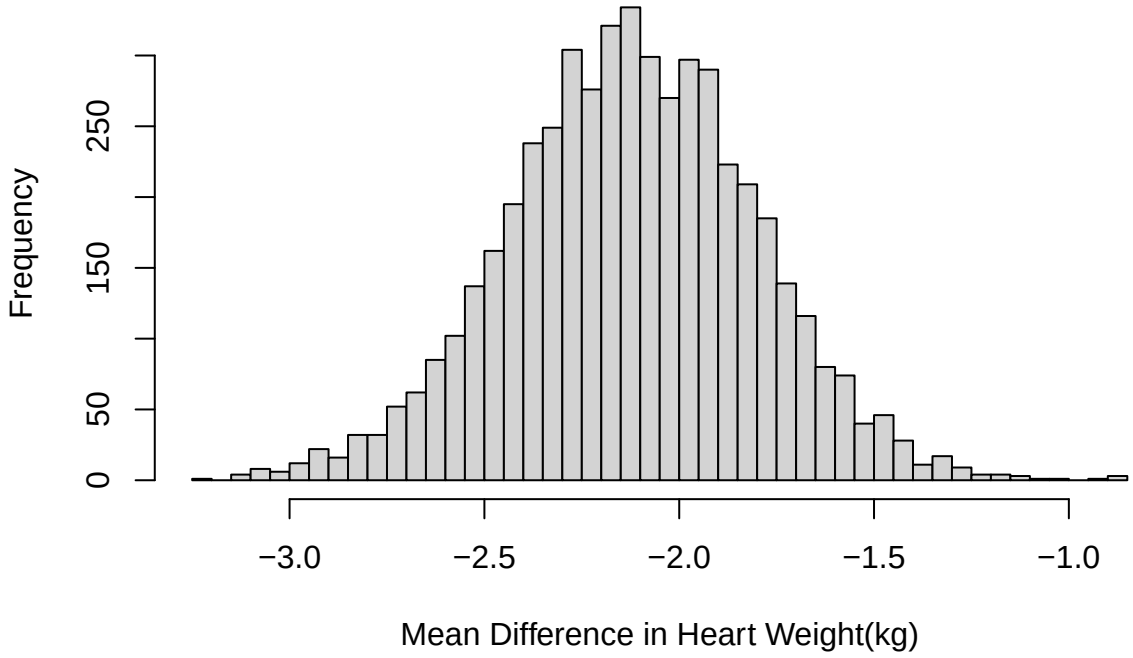


- Obtain the bootstrap sampling distribution of the difference of sample means for heart weight between female and male cats

```r
# Number of bootstrap replicates
B <- 5000
# Perform bootstrap sampling
hwt_diffs <- bootstrap_mean_diff(female_hwt, male_hwt, B)
summary(hwt_diffs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.2356 -2.3351 -2.1223 -2.1227 -1.9119 -0.8582
```

```r
# Plot bootstrap sampling distribution
hist(hwt_diffs, breaks = 40, main = "Bootstrap Sampling Distribution of mean Heart Weight Difference (F
     xlab = "Mean Difference in Heart Weight(kg)",
     ylab = "Frequency")
```



Explain how many bootstrap replicates you decided to use and comment on the results:
- For part a, I decided to use 5000 bootstrap replicates to estimate the sampling distribution of mean body/heart weight between female and male cats. The resulting histogram provides an approximation of how the mean difference in body/heart weight might vary across random sampled from the population. It can be observed in both histograms that the values are centered around a negative value, indicating the larger mean of male weights in comparison to female weights.

Part(b):

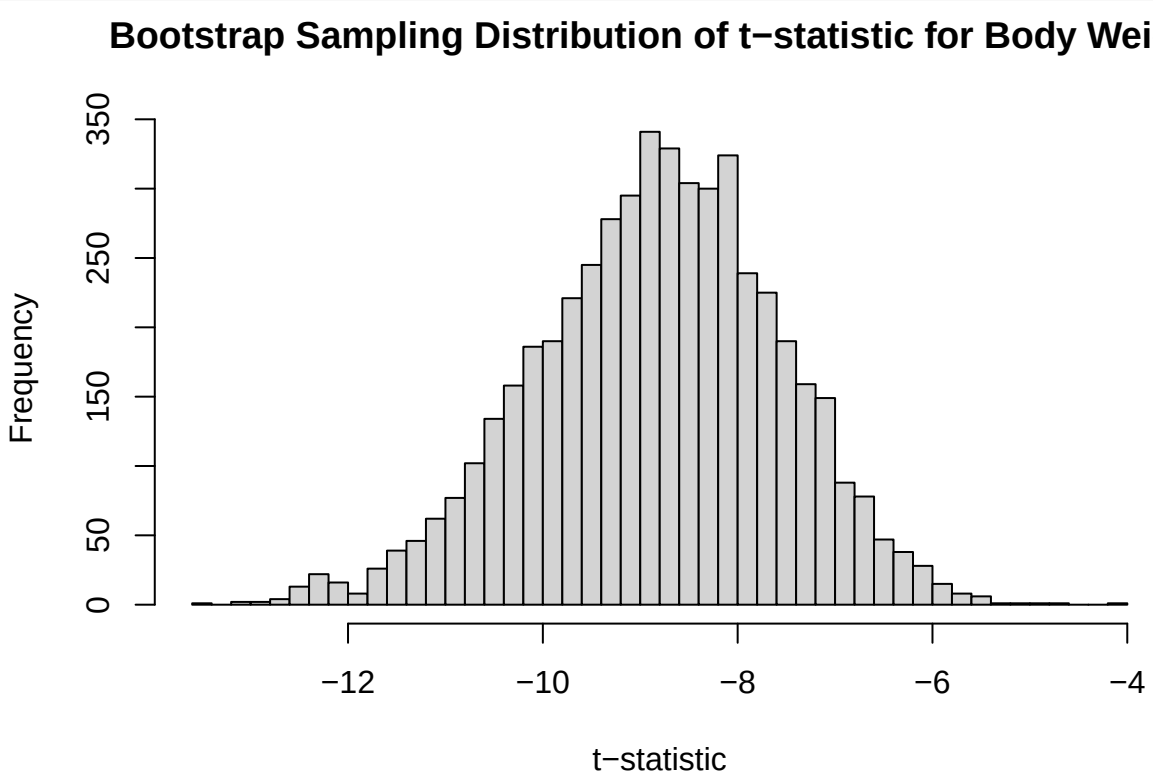
- Obtain the bootstrap sampling distribution of the t-statistic wen testing for mean differences for body weight between female and male cats

```
# Number of bootstrap replicates
B <- 5000
# Function to calculate t-statistic
calc_t_stat <- function (f_data, m_data) {
  # Values to calculate t-statistic
  len_f <- length(f_data)
  len_m <- length(m_data)
  mean_f <- mean(f_data)
  mean_m <- mean(m_data)
  var_f <- var(f_data)
  var_m <- var(m_data)

  mean_diff <- mean_f - mean_m
  sqrt_var <- sqrt(var_f / len_f + var_m / len_m)

  # t_statistic <- (mean(f_data) - mean(m_data)) / (sqrt(var(f_data)/length(f_data) + var(m_data)/lengt
  t_statistic <- mean_diff / sqrt_var
  return(t_statistic)
}
```

```r
# Function to perform bootstrap sampling and calculate t-statistics
bs_t_stat <- function(f_data, m_data, B) {
  t_stat <- rep(NA, B)
  for (i in seq_along(t_stat)) {
    f_sample <- sample(f_data, replace = TRUE)
    m_sample <- sample(m_data, replace = TRUE)
    t_stat[i] <- calc_t_stat(f_sample, m_sample)
  }
  t_stat
}
# Perform bootstrap sampling
bwt_t_stat <- bs_t_stat(female_bwt, male_bwt, B)
summary(bwt_t_stat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -13.465  -9.647  -8.781  -8.831  -7.977  -4.049
```

```r
# Plot bootstrap sampling distribution
hist(bwt_t_stat, breaks = 40, main = "Bootstrap Sampling Distribution of t-statistic for Body Weight",
     xlab = "t-statistic",
     ylab = "Frequency")
```



-

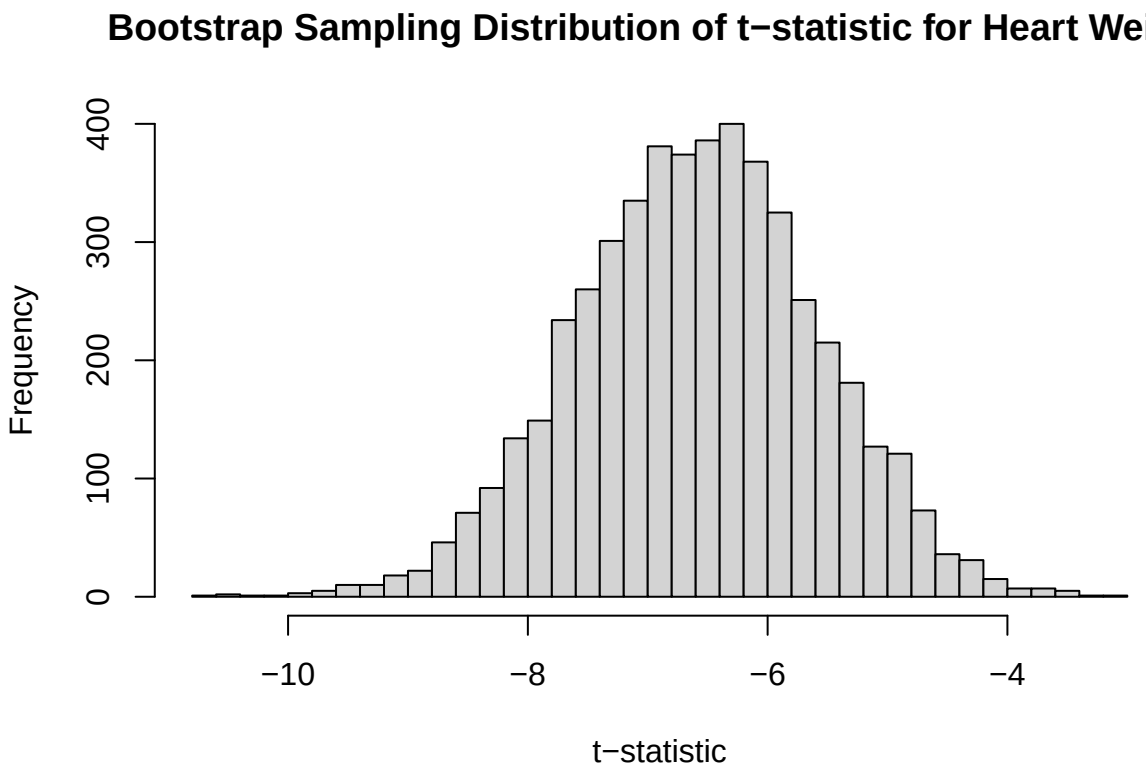
Obtain the bootstrap sampling distribution of the t-statistic when testing for mean differences for ehart weight between female and male cats

```r
# Number of bootstrap replicates
B <- 5000
# Perform bootstrap sampling
hwt_t_stat <- bs_t_stat(female_hwt, male_hwt, B)
summary(hwt_t_stat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
## -10.723  -7.260  -6.573  -6.591  -5.919  -3.068
```

```r
# Plot bootstrap sampling distribution
hist(hwt_t_stat, breaks = 40, main = "Bootstrap Sampling Distribution of t-statistic for Heart Weight",
     xlab = "t-statistic",
     ylab = "Frequency")
```



Exaplain how many bootstrap replicates you decided to use and comment on the results:
- For part b, I chose to use the same amount of bootstrap replicates(B = 5000) to estimate the sampling distribution of the t-statistic of the body/heart weight of cats between genders.The resulting histogram provides information about hw the t-statistic may vary across different random samples from the population. We can observe that in both of the sampling distributions, the t-statistic values tends to be negative.

Part(c):

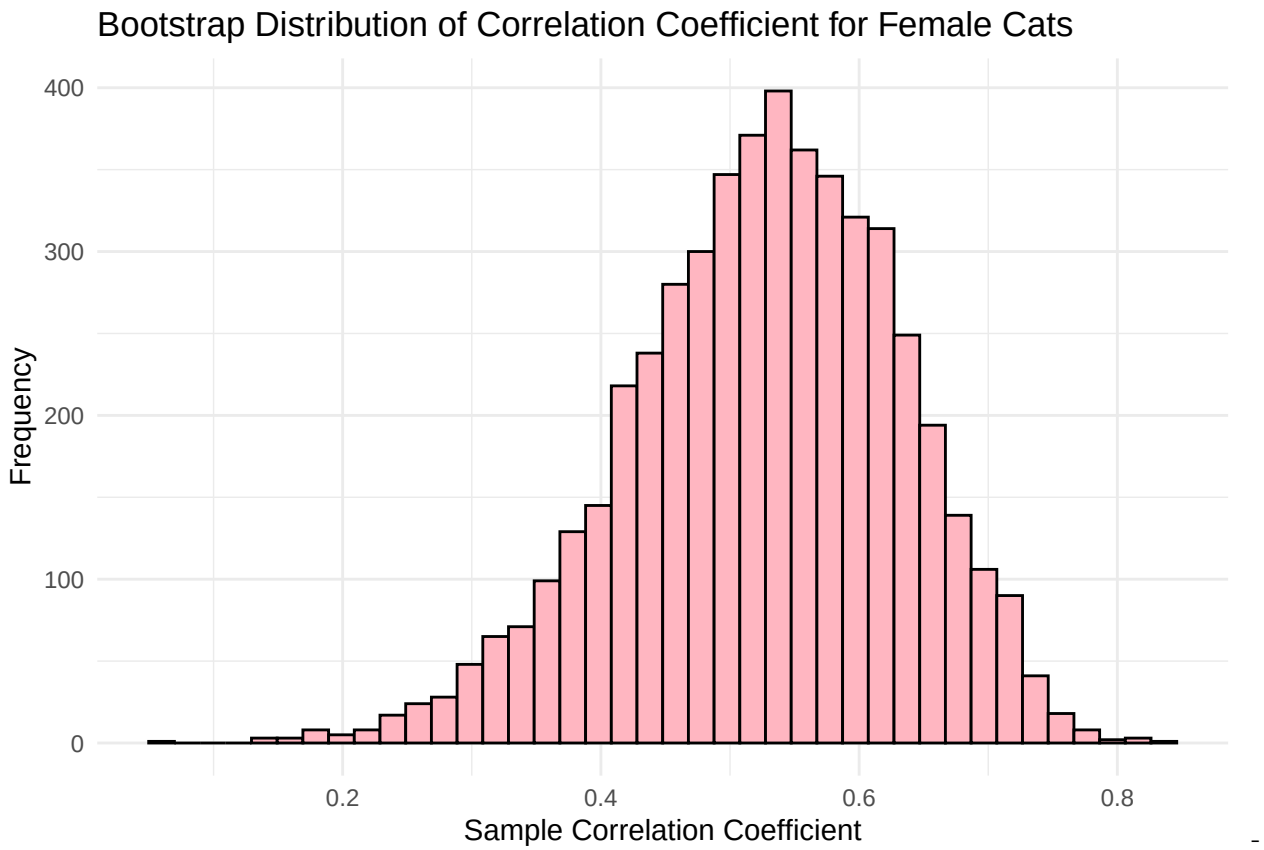
Using your code from Problem 1(c): - Obtain the bootstrap sampling distribution of the sample correlation coefficient between body weight and heart weight for female cats

```r
# Function to calculate the correlation coefficient
cor_fun <- function(data, idx) {
  sample_data <- data[idx, ]
  return(cor(sample_data$Bwt, sample_data$Hwt))
}
# Function for bootstrap sampling and calculating cor coef
bs_cor <- function(data, B) {
  n <- nrow(data)
  bs_samples <- rep(NA, B)
  for (i in seq_along(bs_samples)) {
    idx <- sample(1:n, n, replace = TRUE)
    bs_samples[i] <- cor_fun(data, idx)
  }
  bs_samples
```

```
}
# Number of bootstrap replications
B <- 5000
# Init
female_data <- cats[cats$Sex == "F", ] # nrow -> 47 obs
male_data <- cats[cats$Sex == "M", ] # nrow -> 97 obs

# Perform bootstrap sampling for correlation coefficient
cor_samples_female <- bs_cor(female_data, B)
summary(cor_samples_female)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.05118 0.45692 0.53219 0.52586 0.60130 0.82748
```

```
# Plot
plot_female <- ggplot(data.frame(Correlation = cor_samples_female), aes(x = Correlation)) +
  geom_histogram(bins = 40, fill = "lightpink", color = "black") +
  labs(title = "Bootstrap Distribution of Correlation Coefficient for Female Cats",
       x = "Sample Correlation Coefficient",
       y = "Frequency") +
  theme_minimal()
print(plot_female)
```

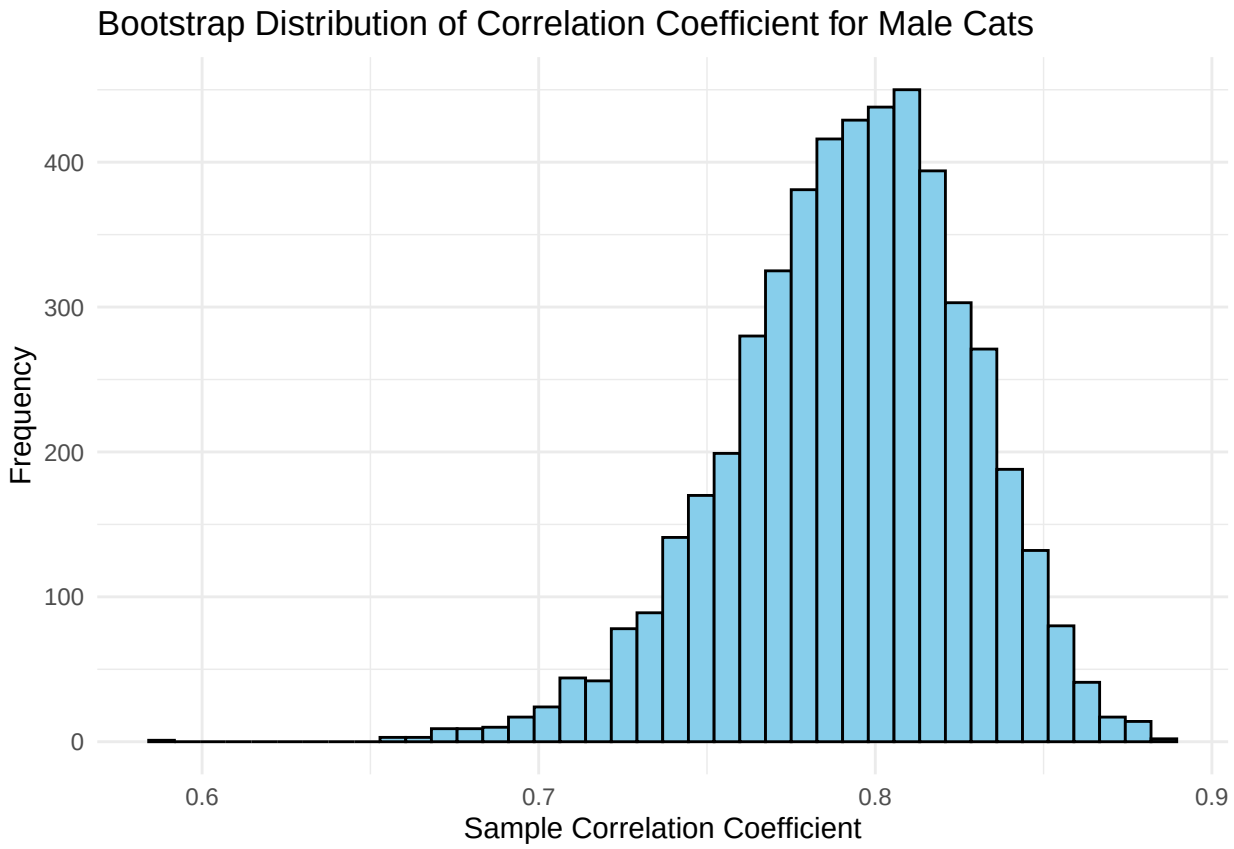


Obtain the bootstrap sampling distribution of the sample correlation coefficient between body weight and heart weight for male cats

```
# Perform bootstrap sampling for correlation coefficient
cor_samples_male <- bs_cor(male_data, B)
summary(cor_samples_male)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.5869  0.7706  0.7951  0.7923  0.8169  0.8847
```

```r
# Plot
plot_male <- ggplot(data.frame(Correlation = cor_samples_male), aes(x = Correlation)) +
  geom_histogram(bins = 40, fill = "skyblue", color = "black") +
  labs(title = "Bootstrap Distribution of Correlation Coefficient for Male Cats",
       x = "Sample Correlation Coefficient",
       y = "Frequency") +
  theme_minimal()
print(plot_male)
```



Explain how many bootstrap replicates you decided to use and comment on the results.

- For part 2c, I used 5000 bootstrap replicates to estimate the sampling distribution of the sample correlation coefficient between body weight and heart weight for both female and male cats. The resulting histograms for female cats reveal the variability and central tendecy to hover around 0.6. Whereas the male cats central tendency hover around 0.8, indicating a stronger correlation of body weight and heart weight relative to female cats.