Bootstrap Homework Assignment

Raven Ico

Example (Boos & Stefanski Ex. 11.3)

The following data are from an experiment on food consumption of female rats treated with zinc-calcium EDTA are taken from Browne and Brownie (1986). The ordered sample values are:

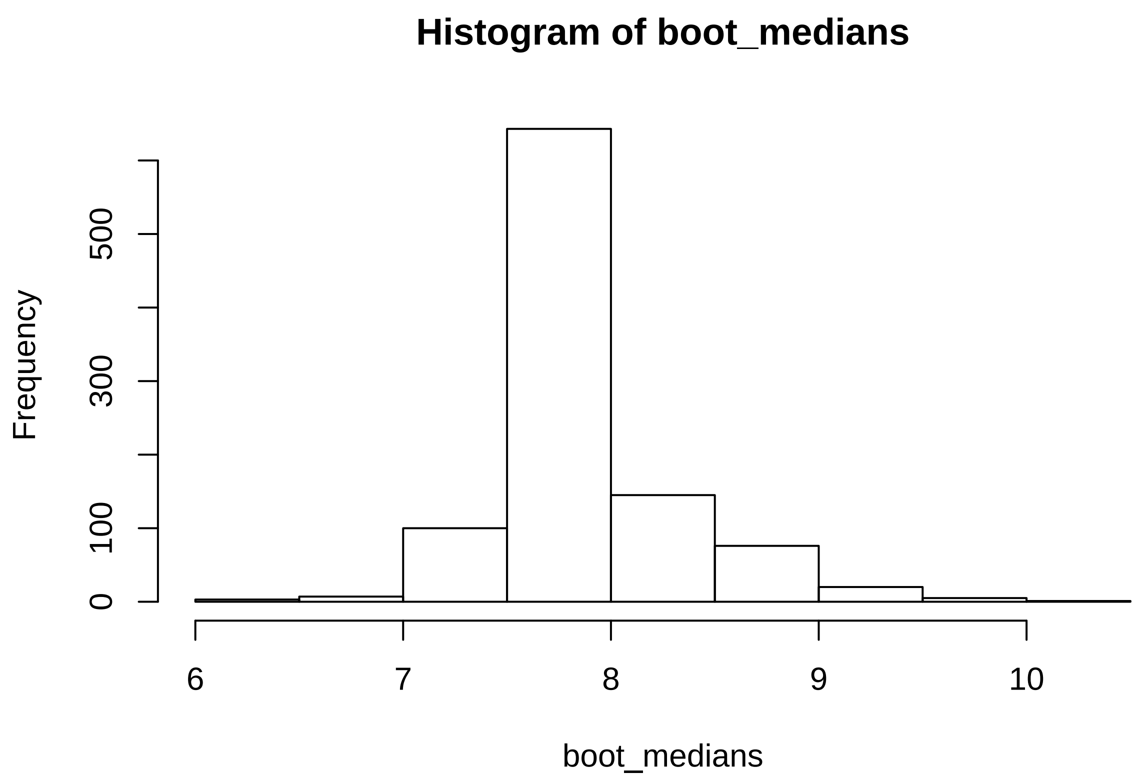
```
rats = c(5.35, 5.37, 5.53, 5.95, 6.20, 7.12, 7.22,
         7.62, 7.63, 7.63, 7.67, 7.97, 8.43, 8.68,
         9.20, 9.63, 11.32, 11.52, 15.27, 15.90)
```

Suppose we are interested in estimating a 95% confidence interval for the median of food consumption in this group of rats. We can calculate this using the bootstrap as follows:

```
#************************************************#
# Setup for the procedure
#************************************************#
# set a seed for reproducibility
set.seed(42)
# set the number of bootstrap iterations
B = 1000
# create a vector to store the statistics of interest
boot_medians = rep(NA, B)

#************************************************#
# Main loop to do the calculations
#************************************************#
for(i in 1:B)
{
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 1: resample data with replacement
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  this_rats = sample(x = rats, size = length(rats), replace = TRUE)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 2: calculate the statistic and store the results
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # In this step our estimator is the median, but we can do a more
  # complicated procedure than just using the median function
  boot_medians[i] = median(this_rats)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 3: Repeat B times
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
}

# plot the results
hist(boot_medians)
```




```r
#**********************************************************************#
# Calculate a 95% confidence interval from the vector of statistics
#**********************************************************************#
quantile(boot_medians, probs = c(0.025, 0.975))
```

```
##  2.5% 97.5%
##  7.17  9.03
```

Our 95% confidence interval for the median is (7.17, 8.94).

Problem 1

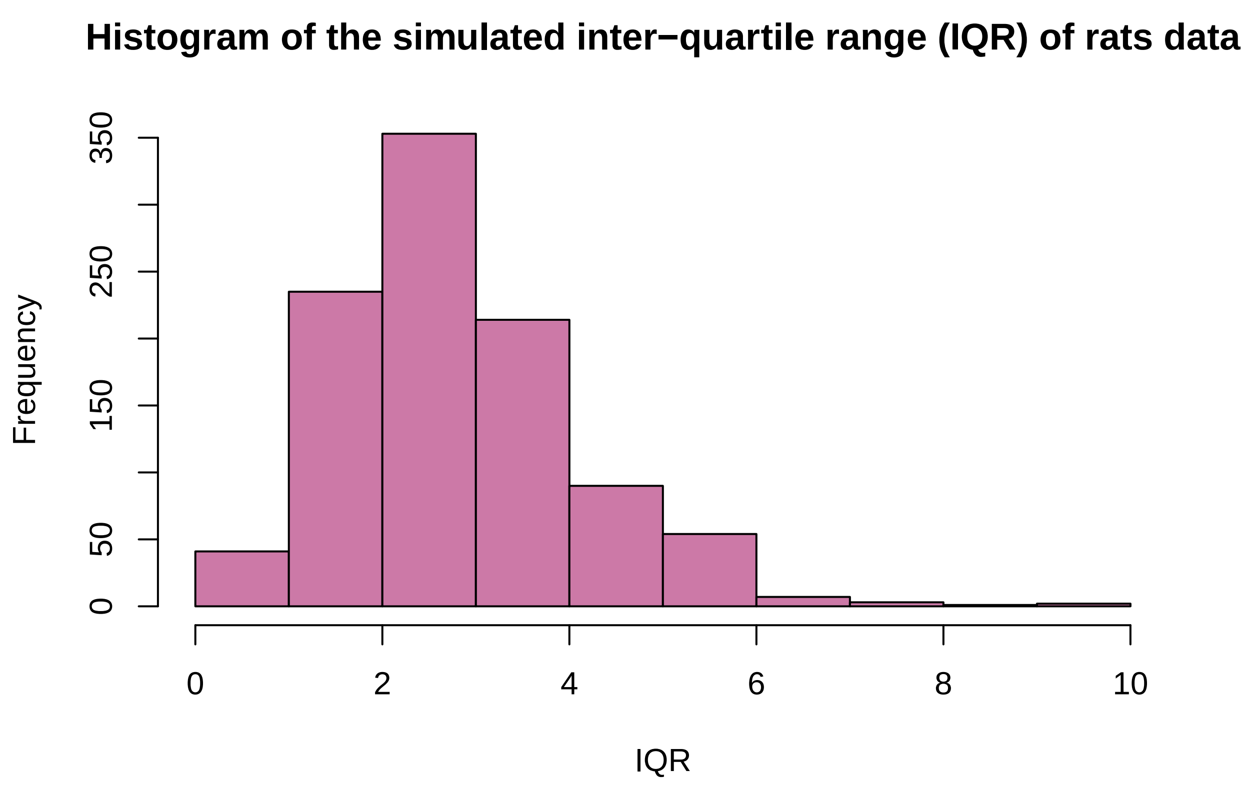
Modify the code above to calculate a 95% confidence interval for the inter-quartile range of the data (75th percentile - 25th percentile).

```r
#***************************************************#
# Setup for the procedure
#***************************************************#
# set a seed for reproducibility
set.seed(42)
# set the number of bootstrap iterations
B = 1000
# create a vector to store the statistics of interest
boot_IQR = rep(NA, B)

#***************************************************#
# Main loop to do the calculations
```

```r
#**************************************************#
for(i in 1:B)
{
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 1: resample data with replacement
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  this_rats = sample(x = rats, size = length(rats), replace = TRUE)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 2: calculate the statistic (inter-quartile range) and store the results
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  boot_IQR[i] = IQR(this_rats)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 3: Repeat B times
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
}

# plot the results
hist(boot_IQR, col="#CC79A7",main="Histogram of the simulated inter-quartile range (IQR) of rats data",
```




```r
#**********************************************************************#
# Calculate a 95% confidence interval from the vector of statistics
#**********************************************************************#
tibble("Lower CI(2.5th quantile)"= quantile(boot_medians, probs = 0.025),"Lower CI(97.5th quantile)"= q
```

```
## # A tibble: 1 x 2
##   `Lower CI(2.5th quantile)` `Lower CI(97.5th quantile)`
```

```
##                        <dbl>                        <dbl>
## 1                       7.17                         9.03
```

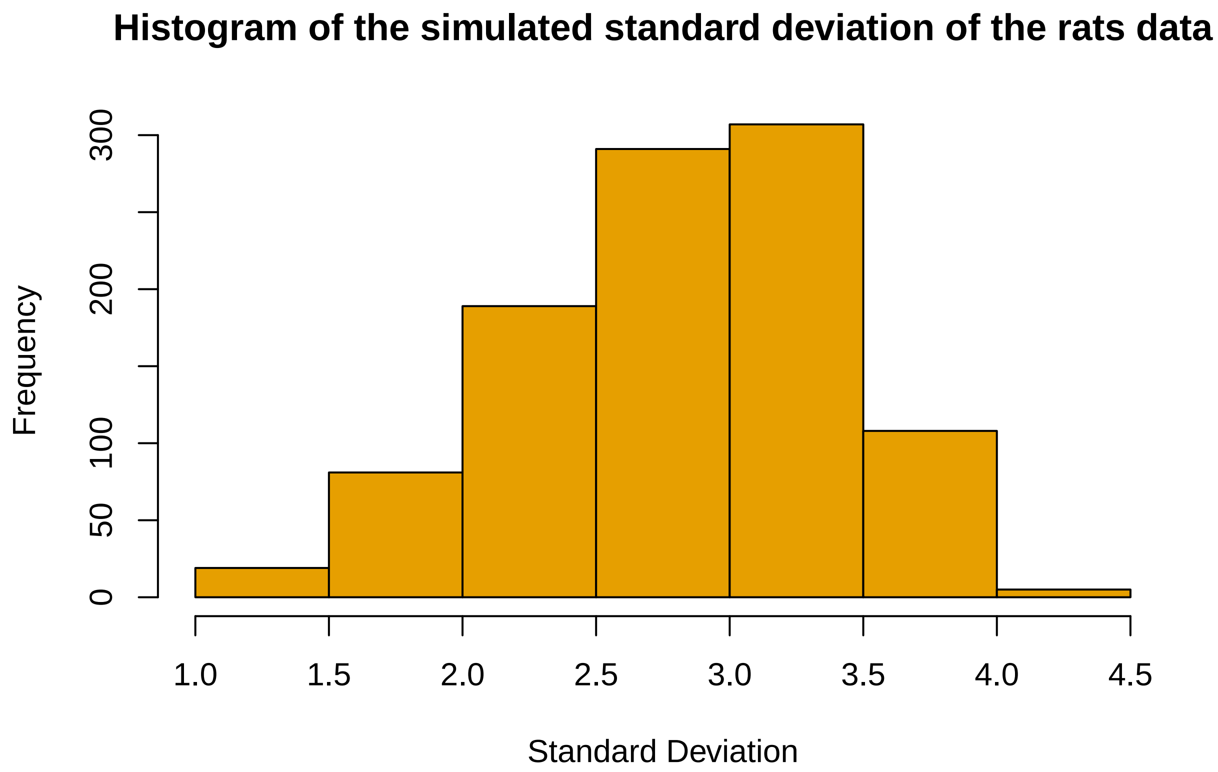
Problem 2

Use the bootstrap to calculated a 90% confidence interval for the standard deviation of the rats data

```r
#**************************************************#
# Setup for the procedure
#**************************************************#
# set a seed for reproducibility
set.seed(42)
# set the number of bootstrap iterations
B = 1000
# create a vector to store the statistics of interest
boot_sd = rep(NA, B)

#**************************************************#
# Main loop to do the calculations
#**************************************************#
for(i in 1:B)
{
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 1: resample data with replacement
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  this_rats = sample(x = rats, size = length(rats), replace = TRUE)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 2: calculate the statistic (standard deviation) and store the results
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  boot_sd[i] = sd(this_rats)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 3: Repeat B times
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
}

# plot the results
hist(boot_sd,col = "#E69F00",main="Histogram of the simulated standard deviation of the rats data",xlab=
```




```r
#**********************************************************************#
# Calculate a 90% confidence interval from the vector of statistics
#**********************************************************************#
tibble("Lower CI(5th quantile)"= quantile(boot_sd, probs = 0.05),"Lower CI(95th quantile)"= quantile(boo
```

```
## # A tibble: 1 x 2
##   `Lower CI(5th quantile)` `Lower CI(95th quantile)`
##                      <dbl>                     <dbl>
## 1                     1.74                      3.71
```

Problem 3

Suppose our interest lies in calculating the difference in the median between two populations, x1 and x2. Using the bootstrap calculate a 95% confidence interval for this difference using the data below. (Hint: In Step 1 you should resample x1 and x2 separately)

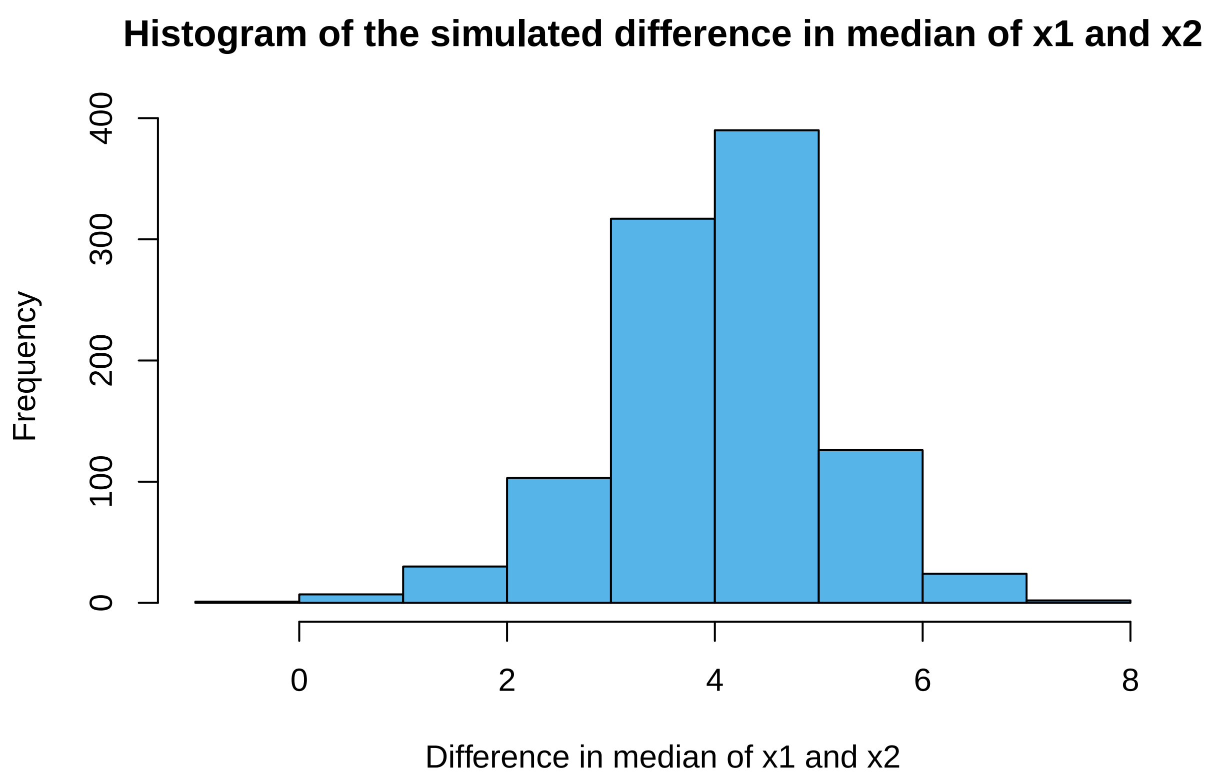
```r
set.seed(42)
n1 = 60; n2 = 40;
x1 = rgamma(n1, 2, 0.2)
x2 = rgamma(n2, 3, 0.5)

#****************************************************#
# Setup for the procedure
#****************************************************#
# set a seed for reproducibility
set.seed(42)
```

```r
# set the number of bootstrap iterations
B = 1000
# create a vector to store the statistics of interest
boot_median_diff = rep(NA, B)

#**************************************************#
# Main loop to do the calculations
#**************************************************#
for(i in 1:B)
{
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 1: resample data with replacement
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  resample_x1 = sample(x = x1, size = n1, replace = TRUE)
  resample_x2 = sample(x = x2, size = n2, replace = TRUE)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 2: calculate the statistic(difference in median) and store the results
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  boot_median_diff[i] = median(resample_x1) - median(resample_x2)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 3: Repeat B times
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
}

# plot the results
hist(boot_median_diff,col="#56B4E9",main="Histogram of the simulated difference in median of x1 and x2"
```



```r
#**********************************************************************#
# Calculate a 95% confidence interval from the vector of statistics
#**********************************************************************#
tibble("Lower CI(2.5th quantile)"= quantile(boot_median_diff, probs = 0.025),"Lower CI(97.5th quantile)
```

```
## # A tibble: 1 x 2
##   `Lower CI(2.5th quantile)` `Lower CI(97.5th quantile)`
##                        <dbl>                       <dbl>
## 1                       1.69                        6.00
```

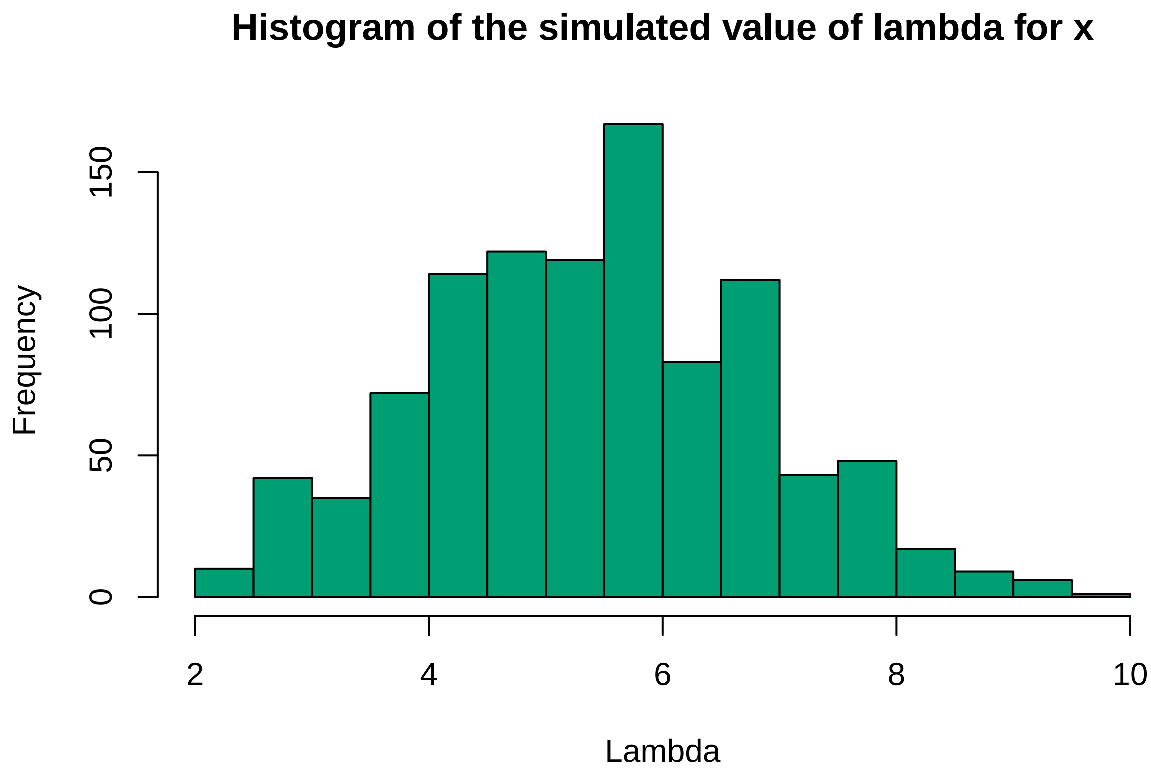
Problem 4

We can also create confidence intervals for maximum likelihood estimators. Like in the MLE section, lets calculate a 95% CI for λ when we assume the data is from a Poisson distribution, this time using the bootstrap.

```r
# use this as the data
x = c(3, 10, 2, 4, 8)
#***************************************************#
# Setup for the procedure
#***************************************************#
# set a seed for reproducibility
set.seed(42)
# set the number of bootstrap iterations
B = 1000
# create a vector to store the statistics of interest
boot_lambda = rep(NA, B)

#***************************************************#
# Main loop to do the calculations
#***************************************************#
for(i in 1:B)
{
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 1: resample data with replacement
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  resample = sample(x = x, size = length(x), replace = TRUE)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 2: calculate the statistic and store the results
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # In this step our estimator is the median, but we can do a more
  # complicated procedure than just using the median function
  boot_lambda[i] = mean(resample)
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
  # Step 3: Repeat B times
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~#
}

# plot the results
hist(boot_lambda, col="#009E73", main="Histogram of the simulated value of lambda for x", xlab="Lambda")
```




```
#**************************************************************************#
# Calculate a 95% confidence interval from the vector of statistics
#**************************************************************************#
tibble("Lower CI(2.5th quantile)"= quantile(boot_lambda, probs = 0.025),"Lower CI(97.5th quantile)"= qua
```

```
## # A tibble: 1 x 2
##   `Lower CI(2.5th quantile)` `Lower CI(97.5th quantile)`
##                        <dbl>                       <dbl>
## 1                        2.8                         8.4
```

Compare this result with the CI calculated using aymptotic MLE theory

```
n=length(x)

lower <- mean(x) - 2*sqrt(var(x)/n) #Theoretical lower and upper bounds for a 95% CI
upper <- mean(x) + 2*sqrt(var(x)/n)

tibble("Lower CI(Asymptotic MLE Theory)"= mean(x) - 2*sqrt(var(x)/n),
       "Upper CI(Asymptotic MLE Theory)"= mean(x) + 2*sqrt(var(x)/n))
```

```
## # A tibble: 1 x 2
##   `Lower CI(Asymptotic MLE Theory)` `Upper CI(Asymptotic MLE Theory)`
##                               <dbl>                             <dbl>
## 1                              2.33                              8.47
```

The obtained confidence intervals using the asymptotic MLE theory and bootstrap method are fairly close with one another.