Namburu_Setu

Bootstrapping will be used to estimate confidence intervals for the variance of a non-normal distribution. Earthquake magnitude data will be used. Results will be compared to confidence intervals constructed using the traditional chi-square method that assumes normality.

The Percentile Bootstrap Method will be used with earthquake magnitude data. The earthquake magnitude data are right-skewed and are not derived from a normal distribution.

The following code ‘chunk’ defines the vector “mag” with the earthquake magnitudes.

```
mag <- c(0.70, 0.74, 0.64, 0.39, 0.70, 2.20, 1.98, 0.64, 1.22, 0.20, 1.64, 1.02,
         2.95, 0.90, 1.76, 1.00, 1.05, 0.10, 3.45, 1.56, 1.62, 1.83, 0.99, 1.56,
         0.40, 1.28, 0.83, 1.24, 0.54, 1.44, 0.92, 1.00, 0.79, 0.79, 1.54, 1.00,
         2.24, 2.50, 1.79, 1.25, 1.49, 0.84, 1.42, 1.00, 1.25, 1.42, 1.15, 0.93,
         0.40, 1.39)
```

There are two steps to this problem. First, compute the confidence interval for the population variance using the traditional chi-square method which assumes normality.

Step #1 (3 points)

The sample variance point estimate is calculated using the *var()* function. You will need to add code calculating a 95% confidence interval for the sample variance estimate, based on a chi-square distribution. This assumes normality.

```
###Sample variance of earthquake magnitute data
mvar <- var(mag)
mvar
```

```
## [1] 0.4401215
```

```
###Quantiles from Chi-square distribution
chi.values <- qchisq(c(0.025, 0.975), 49, lower.tail = TRUE)
chi.values
```

```
## [1] 31.55492 70.22241
```

```
###95% confidence interval for population variance
c(signif(49*mvar/chi.values[2],digits = 4),signif(49*mvar/chi.values[1],digits=4))
```

```
## [1] 0.3071 0.6834
```

```
##95% confidence interval for population standard deviation
result <- c(sqrt(signif(49*mvar/chi.values[2],digits = 4)),sqrt(signif(49*mvar/chi.values[1],d
igits=4)))
result
```

```
## [1] 0.5541660 0.8266801
```

```
##Verifying the above result
cat("95% confidence interval for population variance of earthquake magnitude data: ", result*r
esult)
```

```
## 95% confidence interval for population variance of earthquake magnitude data:  0.3071 0.683
4
```

```
#result*result
```

There is an extensive literature on bootstrapping. The methods shown here give an indication of the possibilities for estimating confidence intervals for a wide range of parameters. Some literature citations will be mentioned.

Next, use the Percentile Bootstrap Method for estimating a 95% confidence interval for the variance. This requires drawing 1000 random samples of size *n* = 50 with replacement from the earthquake data. The sample variance will be computed for each, and the 2.5% (0.025) and 97.5% (0.975) quantiles computed for these data using *quantile()*. Present a histogram of the 1000 sample variances along with the confidence interval. Compare to the traditional chi-square method.

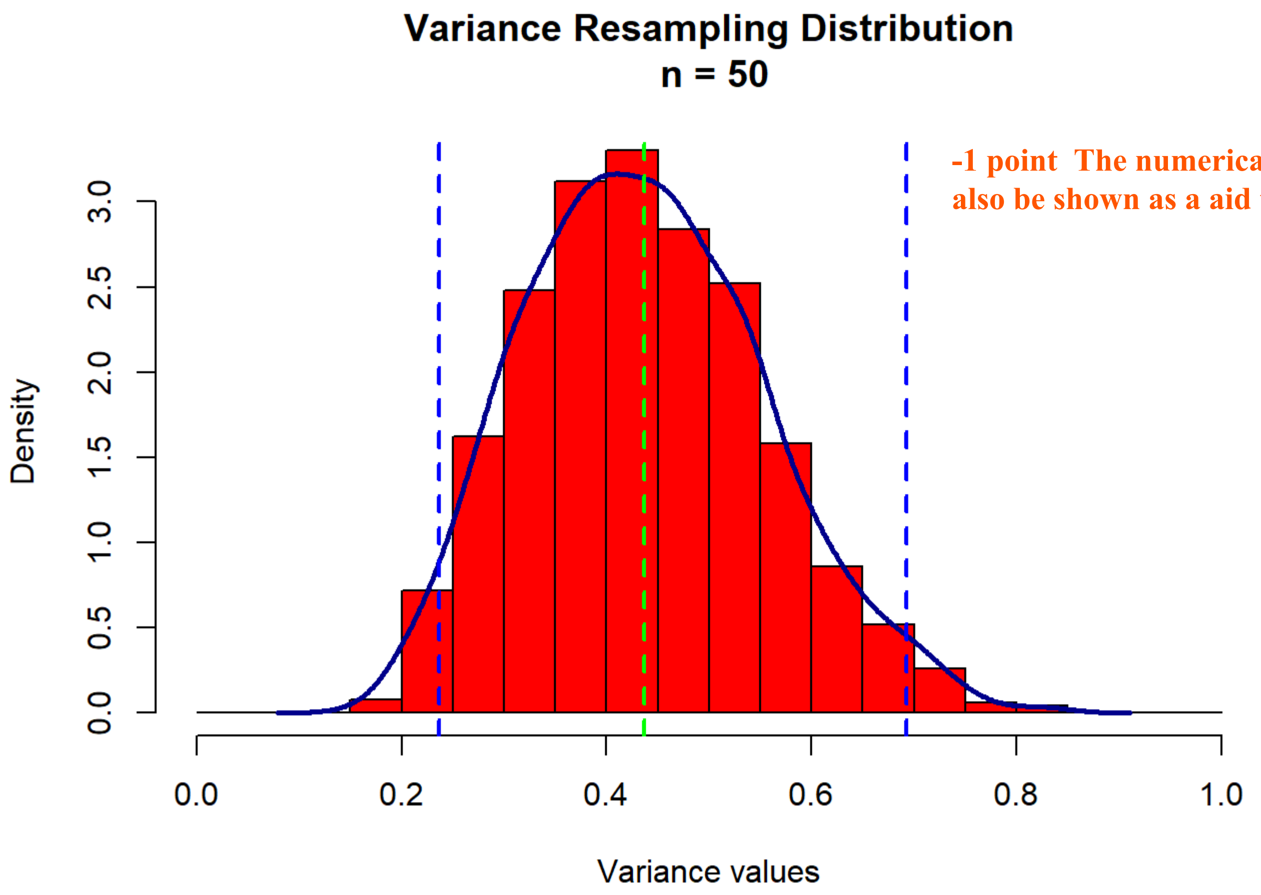
Step #2 (5 points)

1000 random samples will be drawn from “mag” and the sample variance calculated for each. The distribution of results will then be presented via histogram and quantiles determined. The quantiles provide a 95% bootstrap confidence interval. Please keep *set.seed(123)*. The *replicate()* function can be used to easily “replicate” the sampling and sample variance steps, with *n* = 1000.

```
set.seed(123)

n <- 1000
N <- 50

# v <-numeric(0)
# for (i in 1:N)
# {
#   w <- sample(mag, n, replace = TRUE)
#   v[i] <- var(w)
#
# }
# summary(v)

m.var<-replicate(n,var(sample(mag, N, replace = TRUE)))
summary(m.var)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1582  0.3516  0.4311  0.4371  0.5185  0.8310
```

```
#par(mfrow = c(1,2))
cells <- seq(0.0, 1.0, by = 0.05)
hist(m.var, main = "Variance Resampling Distribution\n n = 50", breaks = cells,
     freq  = FALSE, col = "red", xlab = "Variance values")
lines(density(m.var), lwd=2.5, col = "darkblue")
abline(v= quantile(m.var, probs = 0.025), col = "blue", lwd = 2, lty = 2)
abline(v= quantile(m.var, probs = 0.975), col = "blue", lwd = 2, lty = 2)
abline(v= mean(m.var), col = "green", lwd = 2, lty = 2)
```



-1 point The numerical values should also be shown as a aid to the reader.

```
#qqnorm(m.var, col = "red", pch = 16)
#qqline(m.var, col = "green", lwd = 2, lty = 2)
#par(mfrow = c(1,1))

print("95% confidence interval for population variance of earthquake magnitude data: ")
```

```
## [1] "95% confidence interval for population variance of earthquake magnitude data: "
```

```
round(c(quantile(m.var, probs = c(0.025)), quantile(m.var, probs = c(0.975))), digits = 4)
```

```
##   2.5%  97.5%
## 0.2365 0.6920
```

Compare the traditional (i.e. chi-square) and bootstrap results.

-1 point No commentary?

Step 3, additional (7 points)

Additional credit (7 points) will be given for using the “boot” package discussed in Kabacoff Section 12.6, pages 292-298. Examples are given there.

The “boot” package requires a function written to return the sample variance values for each individual resample drawn. Use the following function in *boot()* for the argument “statistic.”

This function is defined for you below.

```
f <- function(data, i){
  d <- data[i]
  return(var(d))
}
```

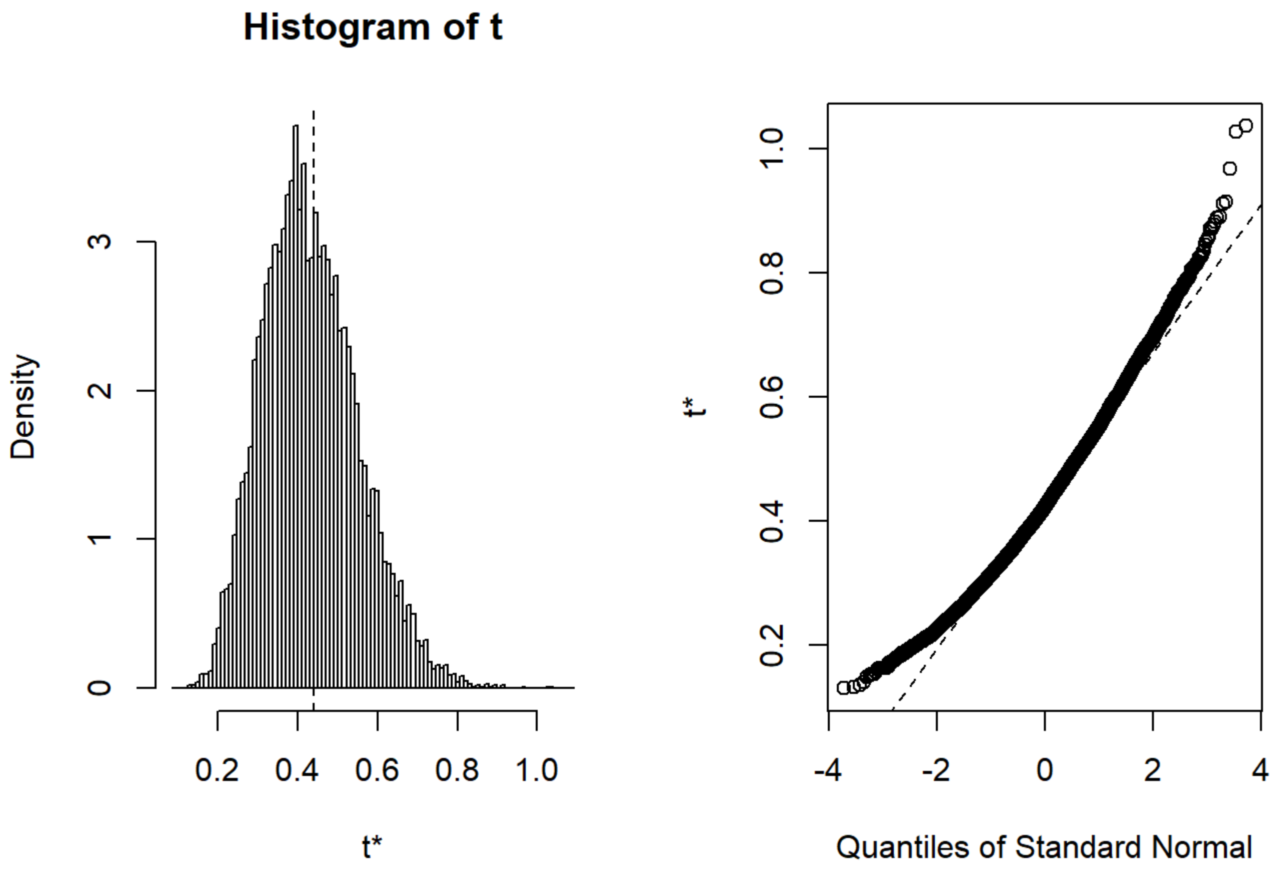
The user-defined function is passed to *boot()* with “mag” and the number of samples to be drawn with replacement. The confidence bounds follow. The “boot” package has a variety of options for determining confidence intervals. See *boot.ci()*, shown below, with the percentile option. The different computational options may produce slightly different results depending on the number of samples drawn during bootstrapping. For this problem, use 10,000 samples drawn with replacement. Again, please keep *set.seed(123)*.

I would encourage you to read the documentation pages for both; *?boot()* and *?boot.ci()* with the package loaded.

```
library(boot)  # install.packages("boot")
set.seed(123)


results <- boot(data=mag, statistic=f, R=10000)
round(quantile(results$t,c(0.025, 0.975)), digits=4)
```

```
##   2.5%  97.5%
## 0.2277 0.6891
```

```
plot(results)
```



Alternatively, we could use *boot.ci()*. To do this, we need to pass the object defined above by *boot()*, specifying “conf = 0.95” and “type =”perc“. *boot.ci()* calculates confidence intervals and stores them at”$percent" of the output.

```
# interval <- boot.ci(results, conf=0.95, type="perc")
# round(c(interval$perc[4],interval$perc[5]), digits=4)

boot.ci(results,conf=0.95,type="perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results, conf = 0.95, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.2276,  0.6892 )
## Calculations and Intervals on Original Scale
```

The “boot” package provides a considerable capability for bootstrapping. Other discussions are given at http://www.ats.ucla.edu/stat/r/faq/boot.htm and http://www.statmethods.net/advstats/bootstrapping.html.

13 points Note my comments above.