

Monte Carlo Simulation Exercise

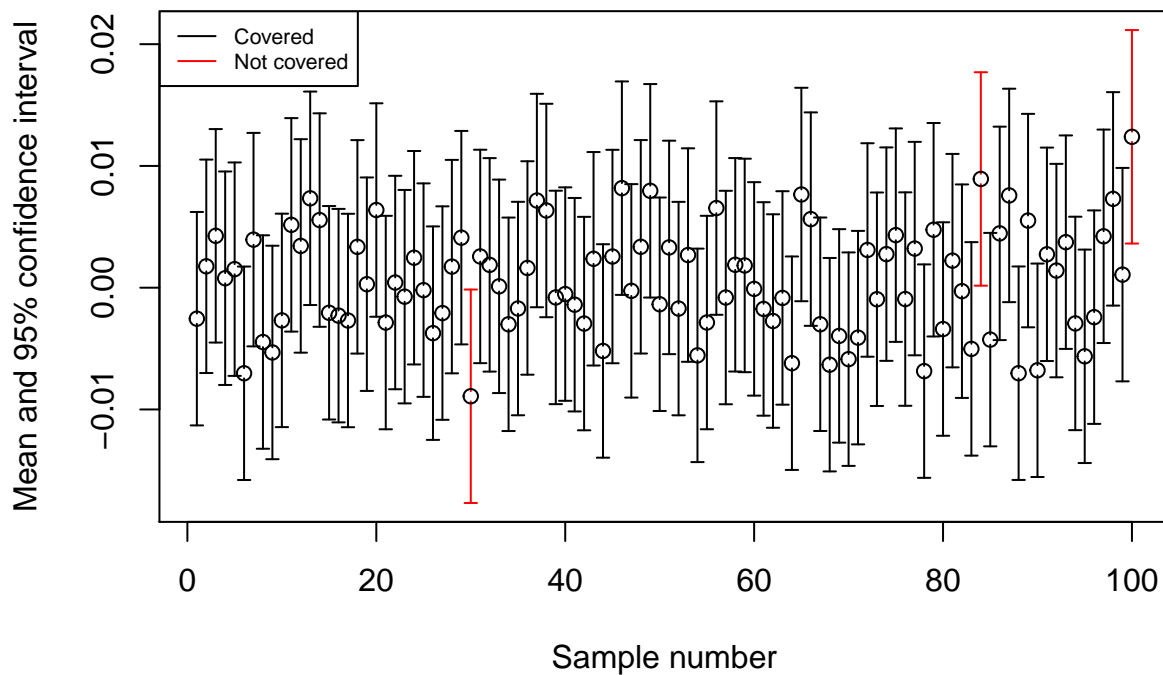
Jan-Willem Simons

9/28/2020

```
#####  
### Monte Carlo Simulation Exercise ###  
#####  
  
### Draw 100 samples of size n = 50000 from a standard normal distribution and store in  
### a list.  
## Set seed for reproducibility.  
set.seed(seed = 3791)  
## Random sampling.  
# Number of samples.  
samples <- 100  
# Draws per sample.  
n <- 50000  
# Drawing.  
norm.samps <- lapply(X = 1 : samples, FUN = function(x) rnorm(n = n, mean = 0, sd = 1))  
  
### Calculating some statistics for the mean.  
## Bias of mean.  
# Per sample.  
bias <- lapply(X = norm.samps, FUN = mean)  
# Over all samples.  
bias.vector <- unlist(x = bias)  
mean.bias <- mean(x = bias.vector)  
## Standard error of mean.  
# Per sample.  
se <- 1 / sqrt(x = n)  
# Over all samples.  
mean.se <- se  
## Lower bound of 95% confidence interval.  
# Per sample.  
lower <- bias.vector - (1.96 * mean.se)  
# Over all samples.  
lower.mean <- bias.vector - (1.96 * mean.se)  
## Upper bound of 95% confidence interval.  
# Per sample.  
upper <- bias.vector + (1.96 * mean.se)  
# Over all samples.  
upper.mean <- bias.vector + (1.96 * mean.se)  
  
### A plot which demonstrates that a 95% confidence interval has at least 95% coverage.  
## If necessary, install "plotrix" package.  
if(!require(package = plotrix)) install.packages(pkgs = "plotrix")
```

```
## Pull "plotrix" package from library.
library(package = plotrix)
## Define plot.
# Plotting.
plotrix::plotCI(x = 1:samples, y = bias.vector, ui = upper, li = lower,
               scol = as.factor(!(lower < 0 & 0 < upper)), gap = 0.01, sfrac = 0.005,
               xlab = "Sample number", ylab = "Mean and 95% confidence interval",
               main = "95% confidence interval coverage plot")
# Add legend.
legend(x = "topleft", legend = c("Covered", "Not covered"),
      col = c("black", "red"), lty = 1, cex = 0.65)
```

95% confidence interval coverage plot



```
## For this random seed the population value has not been covered three times.

### A table containing all simulated samples for which the resulting confidence interval
### does not contain the population value.
## Store id's for those samples which confidence interval does not cover the
## population mean.
uncov.ids <- which(!(lower < 0 & 0 < upper))
## Calculate statistics.
# Bias.
uncov.bias <- unlist(bias[uncov.ids])
# se.
uncov.se <- rep(1 / sqrt(x = n), 3)
# Lower confidence interval.
```

```

uncov.lower <- lower[uncov.ids]
# Upper confidence interval.
uncov.upper <- upper[uncov.ids]
## Table.
atable <- cbind(uncov.ids, uncov.bias, uncov.se, uncov.lower, uncov.upper)
colnames(atable) <- c("Sample ID", "Bias", "SE", "Lower CI", "Upper CI")
atable

```

```

##      Sample ID      Bias      SE      Lower CI      Upper CI
## [1,]        30 -0.008915443 0.004472136 -0.0176808294 -0.0001500565
## [2,]        84  0.008930485 0.004472136  0.0001650988  0.0176958718
## [3,]       100  0.012392158 0.004472136  0.0036267711  0.0211575441

```