Part 1: Simulation Exercise

Junyoung Kim

Feb. 25, 2018

1. Overview

The **central limit theorem** proves that regardless of the shape of a population, the distribution of the mean of its sample means approximates the Gaussian(normal) distribution as the number of samplings gets larger. This article simulates the central limit theorem by taking a large number of sample means out of the exponential distribution population. It randomly draws 1000 samplings of 40 exponentials and compare its distribution with that of the sample means, and tries to show normality of the distribution of the mean of the sample means using Quantile-Quantile Plots.

2. Simulations

This simulation**(Appendix 1)** generates 1000 samplings of 40 exponentials using a random generation function. I set the the parameter of R function **'rexp()'** as λ **= 0.2** so that the distribution of following samples results right-skewedly. Simulation properties are listed in **Table 1**.

Table 1: Simulation Properties

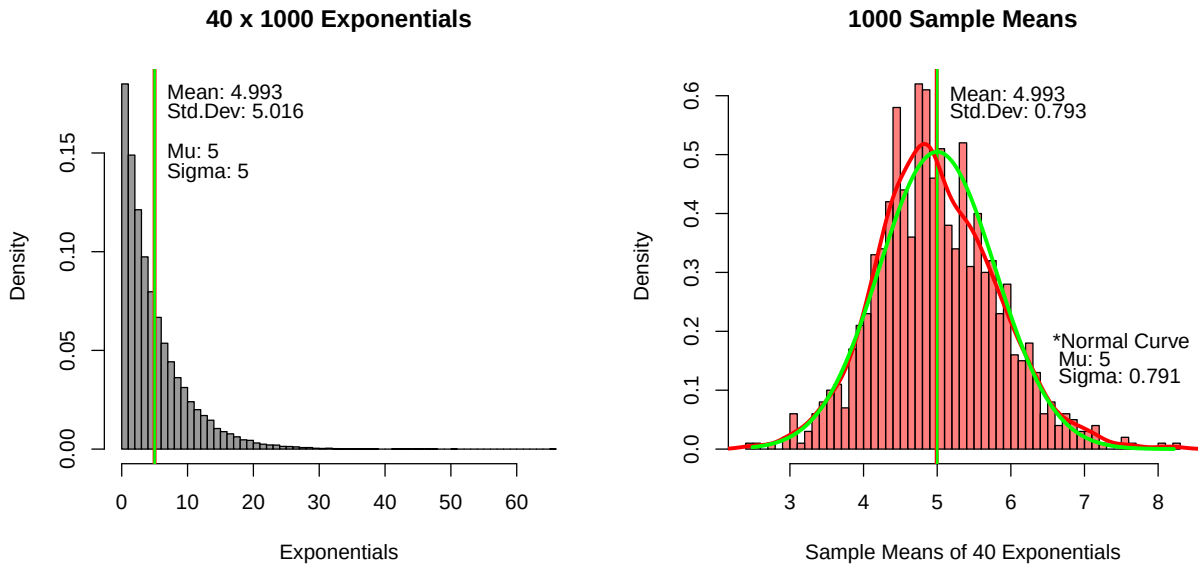
Type	Lambda	n	Number of Samplings
Exponential	0.2	40	1000

3. Estimate versus Parameters

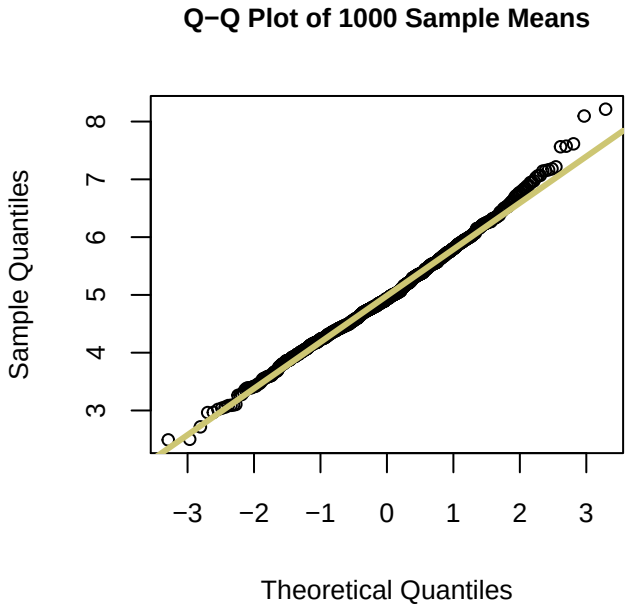
Table 2: Parameter-Estimate Comparison (1000 samplings, n=40)

	Mean	Std.Dev	Mean of Sample Means	Std.Err
Parameter	5.000	5.000	5.000	0.791
Estimate	4.993	5.016	4.993	0.793

Table 2 briefly summarizes major statistics of the samples generated, and compares them with assumed parametric values. As we can see, both mean and variance of the samples and of theory are very close. Since the sample standard deviation $s = 5.016$ is an unbiased estimator of population standard deviation σ, the standard deviation of sampling means $\frac{s}{\sqrt{n}} = 0.793$ is also an unbiased estimator of population standard error $\frac{\sigma}{\sqrt{n}} = 0.791$. According to the central limit theorem, a distribution of sampling means tends toward a normal distribution, and the sampling count of 1000 used in the simulation above is large enough. Therefore, we can expect the distribution of sampling means out of the simulation approximates toward a normal distribution with parameters of $\mu = 5$, $\frac{\sigma}{\sqrt{n}} = 0.791$. **(see Appendix 2 for codes.)**



The left plot is a histogram of the samples and the right one is that of sampling means. The distribution of samples results right-skewedly, and the sampling distribution shows a normal-like distribution. Although the density curve of sampling distribution(red) is not perfectly bell-curved, it is still quite similar with that of the population(green) which is normally distributed, also with a close mean and a standard error. **(see Appendix 3 for codes.)**



To examine normality of the distribution of the sampling means, a Quantile-Quantile Plot test is performed. The result in the Q-Q plot above, where the points of actual quantiles align close to the straight line of theoretical quantile, supports the normality of the sampling means. **(see Appendix 4 for codes.)**

Appendix

1. Simulation Codes

```
# Table 1: Property overview
tab1 <- as.table(cbind("Exponential", 0.2, 40, 1000))
dimnames(tab1) <- list("", c("Type", "Lambda", "n", "Number of Samplings"))
knitr::kable(tab1, digits = 3, align = 'c',
             caption = "Simulation Properties")


# Set basic properties
set.seed(9) # set seed number for reproducibility
lambda <- 0.2 # set lambda 0.2
n <- 40
mean.param <- 1 / lambda # calculate mean of exponential distribution
sd.param <- 1 / lambda # calculate std.dev of exponential distribution
se.param <- sd.param / sqrt(n) # theoretical std.dev. of
                                          # 40 sample means with C.L.T.

# Simulation
samples <- vector()
for (i in 1:1000){
        samples <- c(samples, rexp(n, lambda))
        } # get 40 * 1000 exponentials
sample.matrix <- matrix(samples, n, 1000) # transform into matrix(40, 1000)
mean.1000 <- apply(sample.matrix, 2, mean) # 1000 sample means
```

2. Comparison Codes

```
## parameter-estimate comparison
tab2 <- as.table(rbind(c(mean.param, sd.param,
                         mean.param, se.param),
                       c(mean(samples), sd(samples),
                         mean(mean.1000), sd(samples)/sqrt(n))))
dimnames(tab2) <-list(c("Parameter", "Estimate"),
                      c("Mean", "Std.Dev", "Mean of Sample Means", "Std.Err"))

## output
knitr::kable(tab2, digits = 3, align = 'c',
             caption = "Parameter-Estimate Comparison (1000 samplings, n=40)")
```

3. Plot Codes

```
# Create density plots
par(mfrow = c(1,2))

## plot 40 * 1000 samples
hist(samples, breaks = 50, col = "grey60", xlab = "Exponentials",
     main = "40 x 1000 Exponentials", prob = TRUE)
abline(v = mean(samples), lwd=3, lty=1, col="red")
```

```r
abline(v = mean.param, lwd=2, lty=1, col="green")
text(5, 0.18, pos=4, labels=paste("Mean:", round(mean(samples),3)))
text(5, 0.17, pos=4, labels=paste("Std.Dev:", round(sd(samples),3)))
text(5, 0.15, pos=4, labels=paste("Mu:", mean.param))
text(5, 0.14, pos=4, labels=paste("Sigma:", sd.param))

## plot 1000 sample means
hist(mean.1000, breaks = 50, col = rgb(1,0,0,0.5),
     xlab = "Sample Means of 40 Exponentials",
     main = "1000 Sample Means", prob = TRUE)
abline(v = mean(mean.1000), lwd=3, lty=1, col="red")
abline(v = mean.param, lwd=2, lty=1, col="green")
lines(density(mean.1000), col="red", lwd=3)
text(5, 0.6, pos=4, labels=paste("Mean:", round(mean(mean.1000),3)))
text(5, 0.57, pos=4, labels=paste("Std.Dev:", round(sd(samples)/sqrt(n),3)))

### add a parametric normal distribution line
x <- seq(min(mean.1000), max(mean.1000), length = 1000)
density <- dnorm(x, mean = mean.param, sd = se.param)
lines(x, density, lwd=3, col = "green")
text(6.4, 0.18, pos=4, labels="*Normal Curve")
text(6.4, 0.15, pos=4, labels=paste(" Mu:", mean.param))
text(6.4, 0.12, pos=4, labels=paste(" Sigma:", round(se.param,3)))
```

4. Normality Test Codes

```r
## QQplot
qqnorm(mean.1000, main="Q-Q Plot of 1000 Sample Means", cex=0.8,
       cex.main=0.8, cex.lab=0.8, cex.axis=0.8)
qqline(mean.1000, lwd=3, col="khaki3")
```