Demonstration of Central Limit Theorem on Exponential Distribution

Anthony Liu

Overview

One implication of the Central Limit Theorem is that we expect means of reasonably large samples from the original distribution (with mean μ and variance σ^2) to belong to a normal distribution (with mean μ and variance $\frac{\sigma^2}{n}$). In this report we demonstrate this by using the R programming language (R 3.6.1) to simulate 1000 random samples (each of size 40) from the exponential distribution and compare properties of the simulated distribution with properties predicted by theory.

Simulations

In the following code chunk we first define the parameters of the simulation: set the rate parameter of the exponential distribution to 0.2 ($\lambda = 0.2$), the sample size of each simulation to 40 ($n = 40$) and the number of sample simulations to 1000 ($k = 1000$). We then simulate $n \times k = 40000$ draws from the exponential distribution with rate parameter 0.2 and arrange it within a matrix of 1000 rows and 40 columns, where each row corresponds to a single simulation of size 40. Finally, we obtain a sample of 1000 means by taking the mean of each row of the matrix.

```
lambda <- 0.2
n <- 40
k <- 1000
exp_sim <- matrix(rexp(n * k, lambda), k, n)
exp_means <- apply(exp_sim, 1, mean)
```

Graph preparation

The following code chunk is to prepare the data from the simulation in such a form for easy plotting using the ggplot2 R package. It loads the relevant libraries, creates a dataframe version of the 1000 simulated means, and creates a separate dataframe containing values and labels for sample and theoretical means and standard deviations.

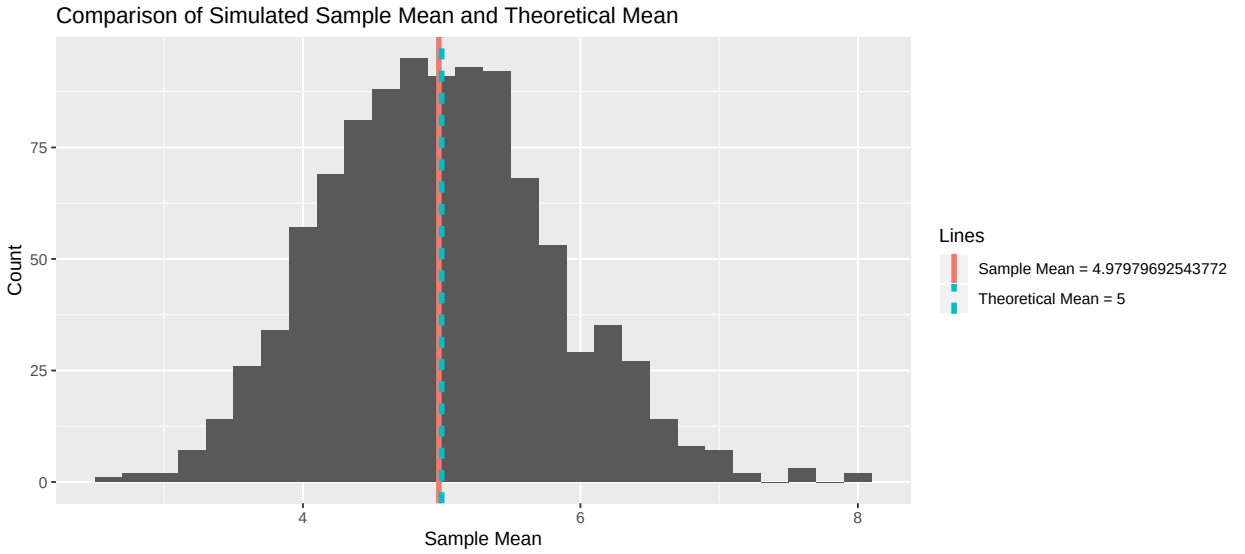
```
library(ggplot2)
library(gridExtra)
df_means <- data.frame(exp_means)
df_stats <- data.frame(means = c(mean(exp_means), 1/lambda),
    sds = c(sd(exp_means), 1/lambda/sqrt(n)), Lines = c(paste("Sample Mean =",
        mean(exp_means)), paste("Theoretical Mean =", 1/lambda)),
    Arrows = c(paste("Sample Standard Deviation =", sd(exp_means)),
        paste("Theoretical Standard Deviation =", 1/lambda/sqrt(n))),
    y = c(50, 75))
```

Sample Mean versus Theoretical Mean

The following plot displays the simulated distribution of sample means and compares the simulated sample mean against the theoretical mean. Here the "Sample Mean", somewhat confusingly named, refers to the mean of the 1000 sample means previously calculated. The "Theoretical Mean" is the expected value of the distribution of means, and is identical to the expected value of the original exponential distribution given by

$1/\lambda$. As can be seen, the simulated sample mean approximates the theoretical mean very well as the Central Limit Theorem would suggest.

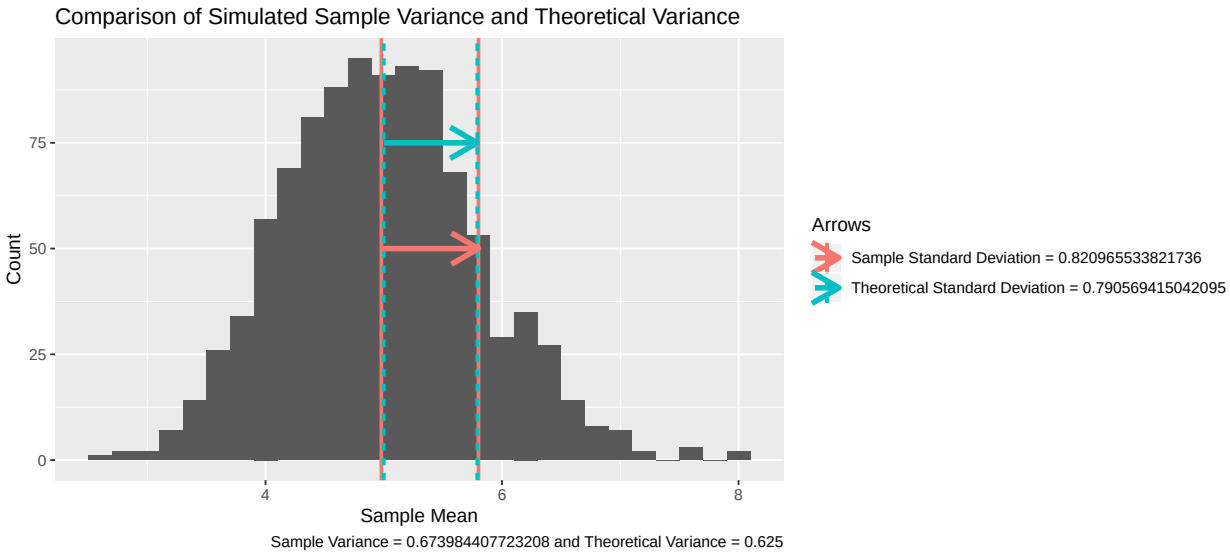
```
ggplot(df_means, aes(exp_means)) + geom_histogram(binwidth = 0.2) +
    geom_vline(aes(xintercept = means, colour = Lines, linetype = Lines),
        size = 1.5, show.legend = TRUE, data = df_stats) + labs(x = "Sample Mean",
    y = "Count", title = "Comparison of Simulated Sample Mean and Theoretical Mean")
```



Sample Variance versus Theoretical Variance

The following plot displays the simulated distribution of sample means and compares the simulated sample variance against the theoretical variance. The "Sample Standard Deviation" here refers to the standard deviation of the 1000 sample means previously calculated, in other terms it is the standard error of the mean of the original exponential distribution simulation. The "Theoretical Standard Deviation" is the theoretical standard error of the mean given by $\frac{\sigma}{\sqrt{n}} = \frac{1}{\lambda\sqrt{n}}$. Here we plot the Sample and Theoretical Standard Deviations as opposed to variances since it is more sensible to display the metric which has the same units as the x axis of the plot. The sample and theoretical variances are noted in a caption below the plot. As can be seen, the simulated sample variance approximates the theoretical variance very well, again as the Central Limit Theorem would suggest. Differences between the two variances are due to random error and would be increasingly negligible with greater sample size or number of simulations.

```
ggplot(df_means, aes(exp_means)) + geom_histogram(binwidth = 0.2) +
    geom_vline(aes(xintercept = means, colour = Arrows, linetype = Arrows),
        size = 1, data = df_stats) + geom_vline(aes(xintercept = means +
    sds, colour = Arrows, linetype = Arrows), size = 1, data = df_stats) +
    geom_segment(mapping = aes(x = means, y = y, xend = means +
        sds, yend = y, colour = Arrows), arrow = arrow(), size = 1.5,
        show.legend = TRUE, data = df_stats) + labs(x = "Sample Mean",
    y = "Count", title = "Comparison of Simulated Sample Variance and Theoretical Variance",
    caption = paste("Sample Variance =", df_stats$sds[1]^2, "and Theoretical Variance =",
        df_stats$sds[2]^2))
```



Distribution

Here we plot a comparison between 1000 random draws from the original exponential distribution (with $\lambda = 0.2$) against the 1000 simulated sample means obtained from this same distribution. Overlayed on top of each histogram is a normal curve for reference. The normal curve fitted on the left uses the same mean and standard deviation as the original exponential distribution (i.e. $\mu = \sigma = \frac{1}{\lambda}$). The normal curve fitted on the right also uses the mean $\mu = \frac{1}{\lambda}$ but in accordance with theory uses standard deviation (standard error of mean) $SE(\bar{x}) = \frac{\sigma}{\sqrt{n}} = \frac{1}{\lambda\sqrt{n}}$ where $n = 40$. The plot here clearly demonstrates the distribution of sample means approximating a normal curve, once again an illustration of the Central Limit Theorem.

```
p1 <- ggplot(data.frame(x = rexp(k, lambda)), aes(x)) + geom_histogram(binwidth = 1) +
    stat_function(fun = function(x) k * 1 * dnorm(x, mean = 1/lambda,
        sd = 1/lambda), colour = "red", size = 1.5) + ylim(0,
    200) + labs(x = "Value", y = "Count", title = "Exponential Distribution")
p2 <- ggplot(df_means, aes(exp_means)) + geom_histogram(binwidth = 0.2) +
    stat_function(fun = function(x) k * 0.2 * dnorm(x, mean = 1/lambda,
        sd = 1/lambda/sqrt(n)), colour = "red", size = 1.5) +
    ylim(0, 200) + labs(x = "Sample Mean", y = "Count", title = "Distribution of Sample Means")
grid.arrange(p1, p2, nrow = 1)
```