Comparing Exponential Distribution to the Central Limit Theorem

Shaun Collett

February 17, 2015

To demonstrate the Central Limit Theorem (CLT) below, we've used the exponential distribution and run 1000 simulations of sets of 40 samples per simulation. You'll see in the graphs that sample vs. theoretical distribution, mean and variance all work to support the CLT.

Simulations

To get started, let's setup our variables for 1000 simulations, via sets of 40 samples, then run them to create a data frame. We'll store the theoretical mean/sd and the sample mean/sd in variables for later use.

```
n <- 40
simulations <- 1000
lambda <- 0.2
mu <- 1/lambda
s <- 1/lambda
mns <- NULL
for (i in 1 : simulations) mns = c(mns, mean(rexp(n, lambda)))
df <- data.frame(x = mns, size = n)

meanx <- mean(df$x)
sdx <- sd(df$x)
```

Sample Mean vs. Theoretical Mean

We've used the function rexp(n, lambda) to create our normal distribution. rexp produces a theoretical mean of 1/lambda. Let's see how that holds versus our sample after simulation:

```
rbind(
    paste("Sample mean:", round(meanx, 2)),
    paste("Theoretical mean:", round(mu, 2))
)
```

```
##      [,1]
## [1,] "Sample mean: 5.02"
## [2,] "Theoretical mean: 5"
```

Very close!!

Sample Variance versus Theoretical Variance

Similar to the mean, rexp also produces a theoretical standard deviation of 1/lambda. To convert the theoretical standard deviation to the sample variance, we need to square the standard deviation, then divide to our sample size `n`.

Let's see how the theoretical variance holds vs. our sample variance:

```r
rbind(
    paste("Sample variance:", round(var(df$x), 2)),
    paste("Theoretical variance:", round(s^2/n, 2))
)
```

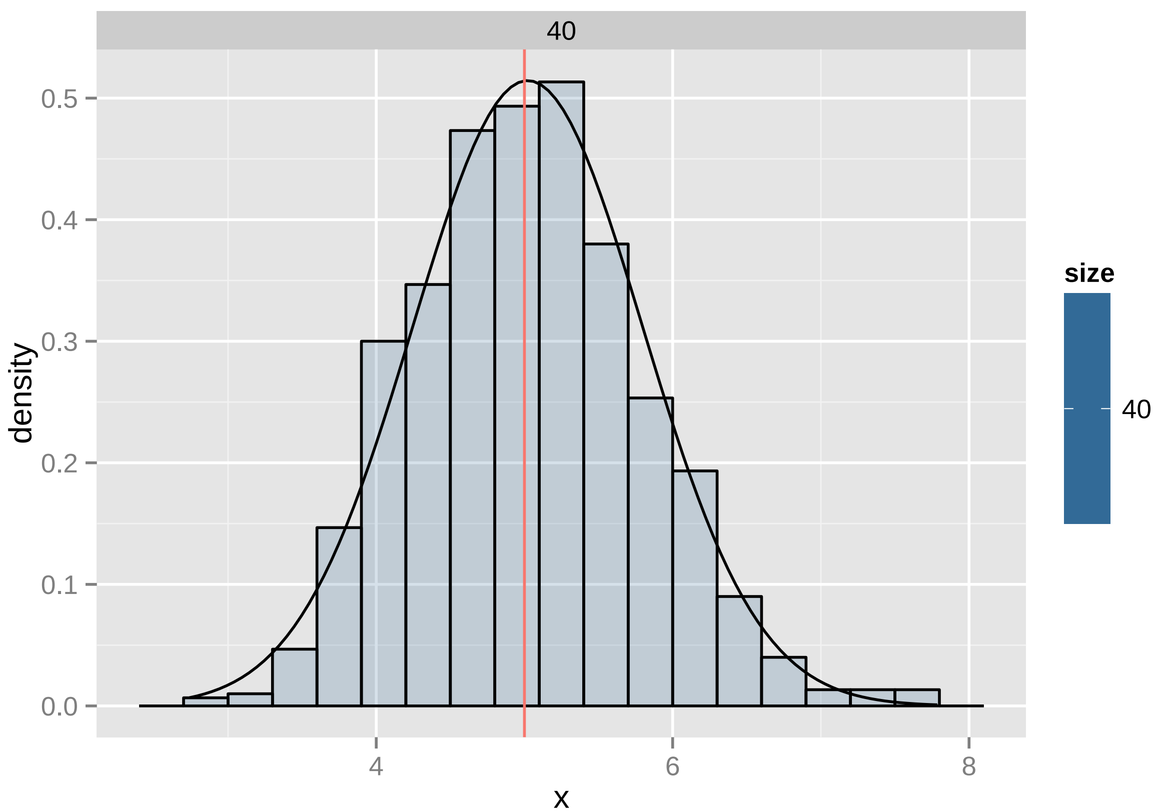
```
##      [,1]
## [1,] "Sample variance: 0.6"
## [2,] "Theoretical variance: 0.62"
```

Close again!!

Our Normal Distribution

Below we show a histogram of the distributions from our simulations. As you can visually see, it's a normal distribution.

```r
library(ggplot2)
g <- ggplot(df, aes(x = x, fill = size)) + geom_histogram(alpha = .20,
            binwidth=.3, colour = "black", aes(y = ..density..))
g <- g + stat_function(fun = dnorm, args = with(df, c(mean = meanx, sd = sdx)))
g + facet_grid(. ~ size) + geom_vline(aes(xintercept = mu, color="red"))
```



To prove this is a normal distribution, we'll calculate how many of the simulations fall within 1, 2 and 3 standard deviations of the mean (approximately 68%, 95% and 99% respectively if this distribution is in fact normal).

```r
rbind(
    paste("1 standard deviation:",
          round(pnorm(meanx + sdx, mean=meanx, sd=sdx) -
                    pnorm(meanx - sdx, mean=meanx, sd=sdx), 2)),
    paste("2 standard deviations:",
          round(pnorm(meanx + 2*sdx, mean=meanx, sd=sdx) -
                    pnorm(meanx - 2*sdx, mean=meanx, sd=sdx), 2)),
    paste("3 standard deviations:",
          round(pnorm(meanx + 3*sdx, mean=meanx, sd=sdx) -
                    pnorm(meanx - 3*sdx, mean=meanx, sd=sdx), 2))
)
```

```
##      [,1]
## [1,] "1 standard deviation: 0.68"
## [2,] "2 standard deviations: 0.95"
## [3,] "3 standard deviations: 1"
```

Wow, pretty much spot on!! That looks like a normal distribution to me.