Simulating Exponential Distributions in R

Ryan Barley

Saturday, December 20, 2014

```
## Install and load required libraries
if (!(require(ggplot2, quietly=T))) {
  install.packages('ggplot2')
}
```

```
## Warning: package 'ggplot2' was built under R version 3.1.2
```

```
library(ggplot2)
```

```
## Set the random seed for simulations
set.seed(1234)
```

The exponential distribution is defined as $f(x) = \lambda * e^{-\lambda x}$ when $x >= 0$, and 0 otherwise. $(\lambda > 0)$ is defined as the rate parameter of the distribution. The mean of the exponential distribution is $(E[x] = \frac{1}{\lambda})$ and the variance is $(var[x] = \frac{1}{\lambda^2})$.

First, let's explore the exponential distribution with 40 exponentials with a mean and standard deviation of $\frac{1}{\lambda} = \frac{1}{0.2} = 5$

```
x <- rexp(40, .2)
x
```

```
##  [1] 12.50879302  1.23379442  0.03290978  8.71373045  1.93591292
##  [6]  0.44974836  4.12040757  1.01308950  4.19020160  3.80215150
## [11]  9.40038339  7.98052709  8.29331192 15.26229050  8.75340067
## [16]  0.15862764  4.38480287  0.07306870  9.17532014  2.59670635
## [21]  9.98139331  3.64193228  1.91770809  4.70222204  1.21819198
## [26]  5.06113942  0.01997473  1.77094527  3.72143748  4.65946794
## [31]  2.17271743  0.04545912  8.05143016  0.03933488  4.07102607
## [36]  0.14506626 21.62281546  4.62763105  0.08547566 17.12640271
```

```
mean(x)
```

```
## [1] 4.969024
```

```
hist(x)
```

Histogram of x
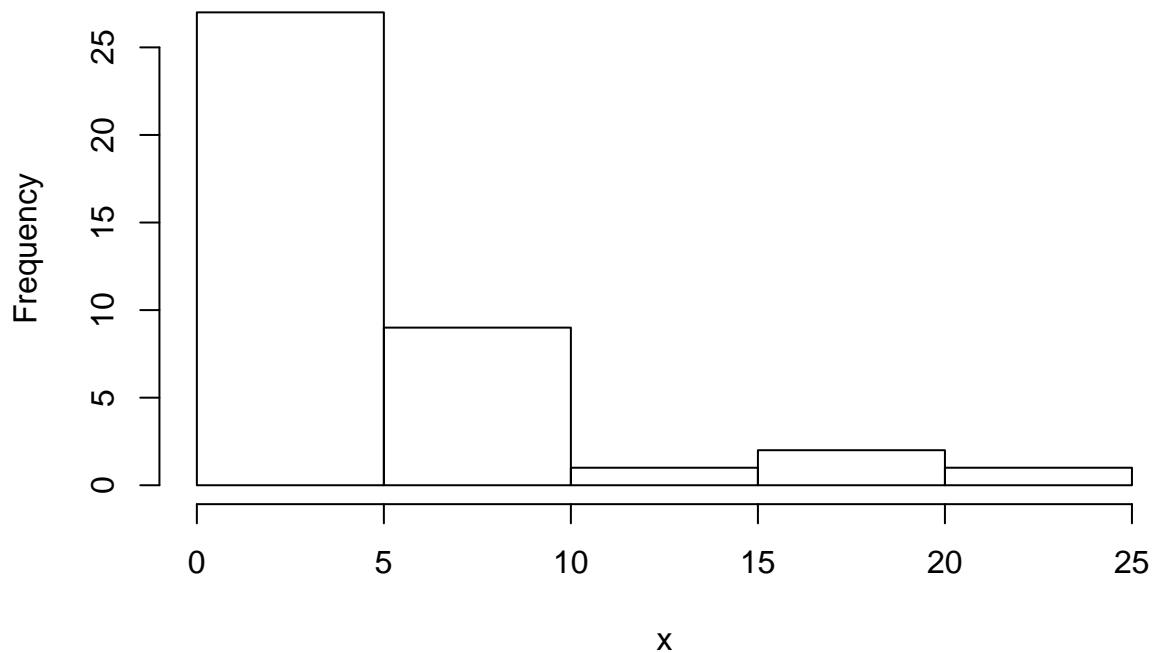
What we see from the histogram of our sample is a typical exponential curve. It is not normal, and it is skewed-right. We also see that the mean is not the predicted 5, but is instead 4.9690237.

Now we will up the ante, and simulate 40 random exponentials 1000 times. We will take the mean of each sample and save it to a vector called ev (for expectation value).

```
lambda <- 0.2
n <- 40
index <- rep.int(1:1000, 1)
ev <- rep(0, 1000)
for (i in index){
  ev[i] <- mean(rexp(n, lambda))
}

df <- data.frame(mean = ev, n = index)
```

```
g <- ggplot(df, aes(x = ev))
g + geom_histogram(aes(y = ..density..), colour = "black",
                   fill = "white", binwidth = 0.5) +
    xlab("Mean") +
    ggtitle("Distribution Curve for 1000 Randomly Sampled Exponential Means \n (n = 40, lambda = 0.2)")
    stat_function(fun = dnorm, aes(color = "red"), args = list(mean = mean(ev),
                                                   sd = sd(ev))) +
    stat_function(fun = dnorm, aes(color = "blue"),
                  args = list(mean = (1/.2),
                              sd = ((1/lambda) / sqrt(n)))) +
```

```
    scale_colour_manual("", values = c("red", "blue"),
                           labels = c("Actual Curve", "Expected Curve"))
```

tribution Curve for 1000 Randomly Sampled Exponential Means
(n = 40, lambda = 0.2)
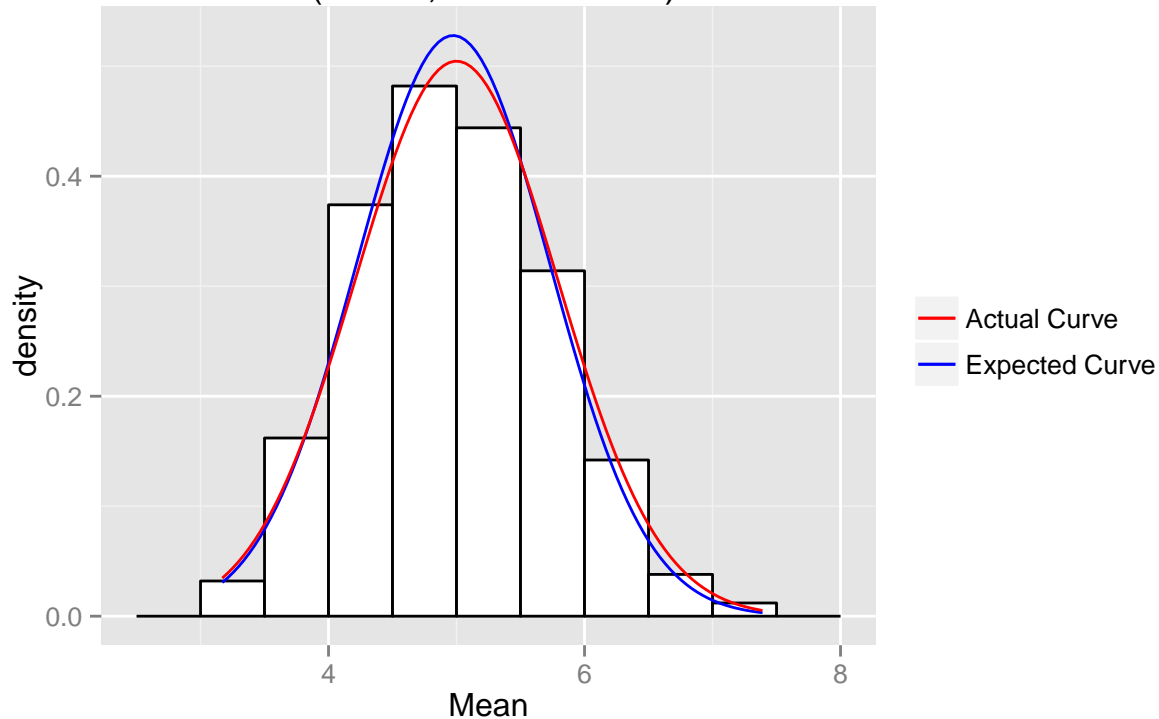
Just by looking at the histogram, we can tell that the distribution has begun to normalize. We can also see that the highest values of the distribution are clustering around 5, the mean we were aiming for. In fact, the mean of the distribution is 4.9736414, which is very close to approaching 5. The red and blue curves on the plot indicate where the theoretical distribution lies versus where the actual distribution lies.

```
center.theory <- 1/(lambda)
center.actual <- mean(ev)
center.error <- ((center.theory - center.actual) / center.theory)*100
variance.theory <- ((1/lambda) / sqrt(n))^2
variance.actual <- var(ev)
variance.error <- ((variance.theory - variance.actual)/variance.theory)*100
```

```
##                  Center  Variance
## Actual        4.9736414 0.5710181
## Theoretical   5.0000000 0.6250000
## Error         0.5271712 8.6371055
```

The mean is very close to being exactly in line with the theoretical predictions; however, the variance is still a small ways away, but with acceptable parameters for the sake of this exercise.