

The Exponential Distribution - An Investigation Using R

Amol Suraj Mishra

Abstract

This report aims at investigating the exponential distribution in R and comparing it with the Central Limit Theorem. The investigation consists of one thousand simulations, each one of them comprised of the averages of 40 exponentials. This investigation utilizes R because it is the most comprehensive statistical analysis package available. It incorporates all of the standard statistical tests, models, and analyses, as well as providing a comprehensive language for managing and manipulating data.

The central limit theorem

The central limit theorem (CLT) states that, given certain conditions, the arithmetic mean of a sufficiently large number of iterates of independent random variables, each with a well-defined expected value and well-defined variance, will be approximately normally distributed, regardless of the underlying distribution. That is, suppose that a sample is obtained containing a large number of observations, each observation being randomly generated in a way that does not depend on the values of the other observations, and that the arithmetic average of the observed values is computed. If this procedure is performed many times, the central limit theorem says that the computed values of the average will be distributed according to the normal distribution (commonly known as a “bell curve”).

The Exponential Distribution

The exponential distribution describes the arrival time of a randomly recurring independent event sequence. If μ is the mean waiting time for the next event recurrence, its probability density function is $f(x) = (1/\mu) e^{-x/\mu}$ when $x \geq 0$; 0 when $x < 0$. If we call λ (rate) $= 1/\mu$, then the exponential probability density function (dexp) becomes $f(x) = \lambda e^{-\lambda x}$. The following is the plot of the exponential probability density function.

The exponential distribution can be simulated in R with `rexp(n, lambda)` where λ is the rate parameter. The mean of exponential distribution is $1/\lambda$ and the standard deviation is also $1/\lambda$. The investigation consists of one thousand simulations, each one of them comprised of the averages of 40 exponentials. The value of λ used in the `rexp` function is fixed in 0.2 for all of the simulations.

This experiment parameters are the following.

```
sims <- 1000
lambda <- 0.2
n <- 40
```

Objectives

1. Show the sample mean and compare it to the theoretical mean of the distribution.
2. Show how variable the sample is (via variance) and compare it to the theoretical variance of the distribution.
3. Show that the distribution is approximately normal.

Simulation

In order to keep this report reproducible, we must set a random seed - a number on which the random number stream depends on - so the reader can obtain exactly the same numbers in his/her R environment when following the steps presented in this report. In our case, the seed is set to the integer number 1.

```
set.seed(1)
```

We start by generating 1000 random samples of size 40, with rate (lambda) equals 0.2. Then, we store the calculated mean of these several samples into the means vector.

```
exp_sim <- matrix(rexp(sims * n, lambda), sims, n)
means <- rowMeans(exp_sim)
```

1. Sample Mean versus Theoretical Mean

```
sample_mean <- mean(means)
theoretical_mean <- 1 / lambda
sample_mean
```

```
## [1] 4.990025
```

```
theoretical_mean
```

```
## [1] 5
```

As we can see, they are almost equal...

2. Sample variance versus Theoretical Variance

```
sample_variance <- var(means)
theoretical_variance <- (((1 / lambda)^2) / 40)
sample_variance
```

```
## [1] 0.6177072
```

```
theoretical_variance
```

```
## [1] 0.625
```

As we can see, they are almost equal

3. Distribution

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```
g <- ggplot( data.frame(means), aes(x = means))  
g <- g + geom_histogram(aes(y=..density..), colour = "black", fill = "white", binwidth = lambda)  
g <- g + geom_density(colour = "skyblue", fill="skyblue", alpha =.3)  
g <- g + geom_vline( aes(xintercept = sample_mean), linetype = "dashed", size = 2, colour = "red")  
g <- g + scale_x_continuous(breaks = c(1: 10))  
g <- g + ggtitle("Means of 1000 Random Samples of Size 40"); g
```

Means of 1000 Random Samples of Size 40

