# Simulating exponential distributions and comparing the distribution of averages with the Central Limit Theorem

## Author: “Yosuke Ishizaka”

## Overview:

In this project I investigate the exponential distribution using R. I simulate the distribution of averages of 40 exponentials by running thousand simulations and compare the result with the Central Limit Theorem.

## Simulations:

The exponential distribution in R is expressed in the form rexp(n, lambda) where n is the number of exponentials and lambda is the rate parameter. In this project I set n = 40 exponentials and lambda = 0.2 for all thousand simulations. Here’s a code for the distribution of averages of 40 exponentials. (Appendix: A)

## Sample Mean versus Theoretical Mean:

Using the parameters given I can use function mean to calculate the sample mean. Here’s a code. (Appendix: B)

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

Now I compare sample mean to theoretical mean of the distribution. Since Lambda is 0.2, I can calculate mean by the formula 1/lambda. (Appendix: C)

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

Sample mean 4.99 is close to the theoretical mean 5.

## Sample Variance versus Theoretical Variance:

I can use function var to calculate variance of the averages of exponentials. (Appendix: D)

```
## [1] 0.6111165
```

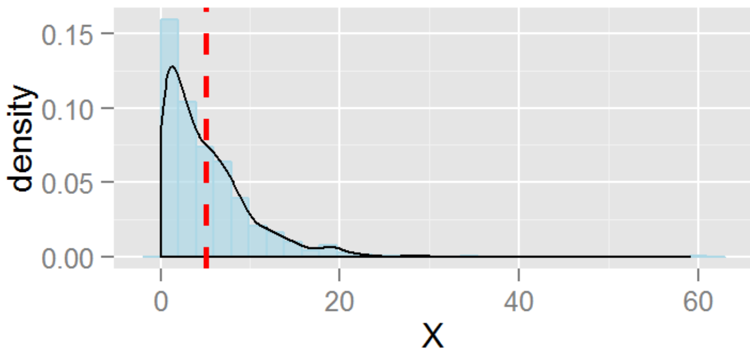
Now I compare sample variance to theoretical variance of the distribution. To find theoretical variance, I use formula S^2/n. S^2 is a sample variance of the exponential distribution, 5^2 = 25. n is the number of exponentials, 40. (Appendix: E)

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

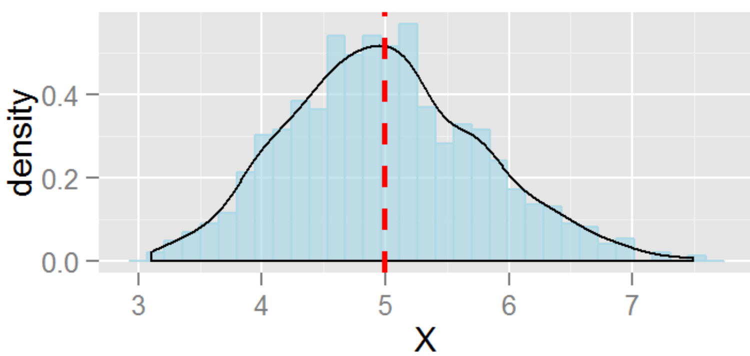
Sample variance 0.611 is close to the theoretical variance 0.625.

## Distribution:

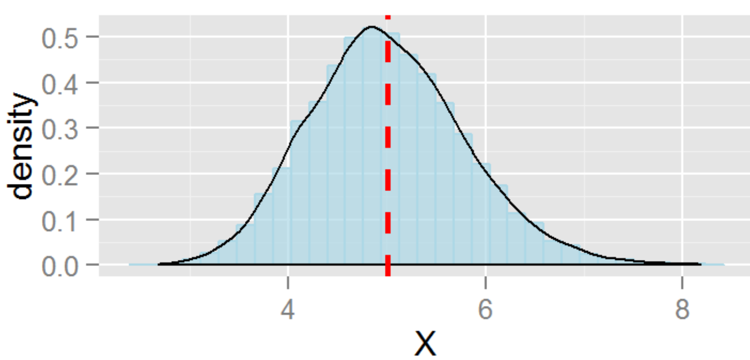
First I show the distribution of a large collection of random exponentials. I use 1000 random exponentials. Plotting a histogram and a density curve shows how the distribution is shaped. For visual guide, a red vertical line indicates a sample mean of the distribution. (Appendix: F)




As you can see, the distribution of a large collection of exponentials is not normal. Next, I show the distribution of 1000 averages of 40 exponentials. (Appendix: G)

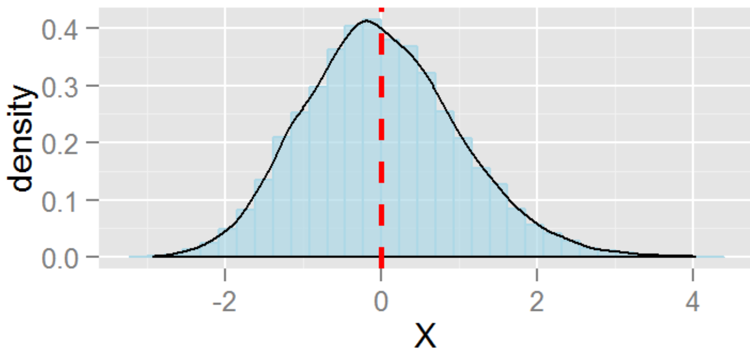



Here's what 10000 averages of 40 exponentials looks like. (Appendix: H)




Distribution is tighter around the mean.

Now I use the formula z = (mean - mu)/standard error to check for normality. Standard error is s/sqrt(n). Here's distribution for 10000 averages of 40 exponentials. (Appendix: I)




Showing the sample mean and standard deviation. (Appendix: J)

```
## [1] 0.003009022
```

```
## [1] 0.9954444
```

## Conclusion:

My conclusion is that as n gets larger, the distribution of averages of exponentials becomes that of a standard normal. For n = 1000, mean is -0.0126 and standard deviation is 0.9889. For n = 10000, mean is 0.0030 and standard deviation is 0.9954. Sample mean is approximately equal to 0 and sample standard deviation is approximately equal to 1. Therefore, the distribution is approximately normal.

## Appendix:

R codes used in this report are shown for reproducibility.

A.

```
set.seed(1)
means <- NULL
lambda <- 0.2
n <- 40
for(i in 1:1000)
        means = c(means, mean(rexp(n, lambda)))
```

B.

```
mean(means)
```

C.

```
meant <- 1/lambda
print(meant)
```

D.

```
var(means)
```

E.

```
s <- 1/lambda
vart <- s^2/n
print(vart)
```

F.

```
library(ggplot2)
meanl <- rexp(1000, lambda)
dataset1 <- data.frame(X = meanl)
g <- ggplot(dataset1, aes(x = X))
g + geom_histogram(aes(y = ..density..), color = "light blue", fill = "light blue", alpha
= .7, bindwidth = .5) + geom_density() + geom_vline(aes(xintercept = mean(meanl)), linety
pe = "dashed", size = 1, color = "red")
```

G.

```
dataset2 <- data.frame(X = means)
g <- ggplot(dataset2, aes(x = X))
g + geom_histogram(aes(y = ..density..), color = "light blue", fill = "light blue", alpha
= .7, bindwidth = .5) + geom_density() + geom_vline(aes(xintercept = mean(means)), linety
pe = "dashed", size = 1, color = "red")
```

H.

```
meanx <- NULL
for(i in 1:10000)
        meanx = c(meanx, mean(rexp(n, lambda)))
dataset3 <- data.frame(X = meanx)
g <- ggplot(dataset3, aes(x = X))
g + geom_histogram(aes(y = ..density..), color = "light blue", fill = "light blue", alpha
= .7, bindwidth = .5) + geom_density() + geom_vline(aes(xintercept = mean(meanx)), linety
pe = "dashed", size = 1, color = "red")
```

I.

```
z <- (meanx-meant)/(s/sqrt(n))
dataset4 <- data.frame(X = z)
g <- ggplot(dataset4, aes(x = X))
g + geom_histogram(aes(y = ..density..), color = "light blue", fill = "light blue", alpha
= .7, bindwidth = .5) + geom_density() + geom_vline(aes(xintercept = mean(z)), linetype =
"dashed", size = 1, color = "red")
```

J.

```
mean(z)
sqrt(var(z))
```