# Statistical Inference on Coursera: Final Project Part 1

Navid Lambert-Shirzad

December 5, 2016

## Statistical Inference Course Project 1: Simulations

### Overview

Reproduced from the instructions posted on the Coursera page: In this project I will investigate the exponential distribution in R and compare it with the Central Limit Theorem. The exponential distribution can be simulated in R with `rexp(n, lambda)` where lambda is the rate parameter. The mean of exponential distribution is `1/lambda` and the standard deviation is also `1/lambda`. For all of the simulations in this report `lambda = 0.2`. I will investigate the distribution of averages of 40 exponentials over a thousand simulations.

I will illustrate the properties of the distribution of the mean of 40 exponentials via simulation and associated explanatory text. I am expected to:

-Show the sample mean and compare it to the theoretical mean of the distribution.

-Show how variable the sample is (via variance) and compare it to the theoretical variance of the distribution.

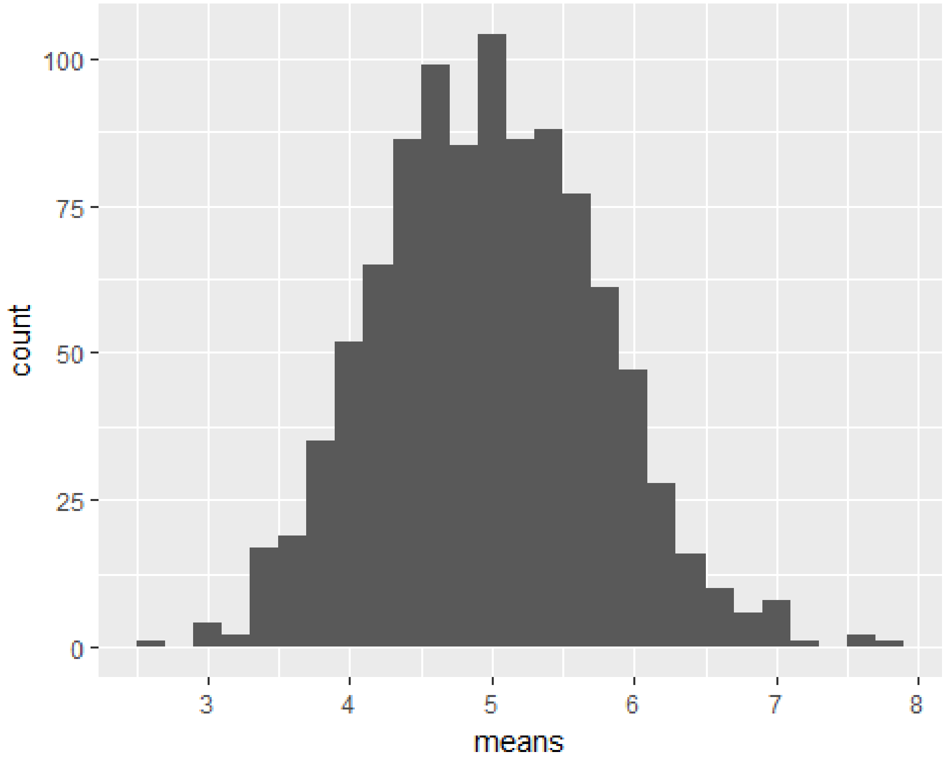
-Show that the distribution is approximately normal.

### Simulations

```r
# load neccesary libraries
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.2.5

library(knitr)

## Warning: package 'knitr' was built under R version 3.2.5

# set constants
lambda <- 0.2 # lambda for rexp
n <- 40 # number of exponetials
numSim <- 1000 # number of simulations

# set the seed to create reproducability
set.seed(810684111) #This used to be my student ID :)

# run the test resulting in n x numSim matrix
expDist <- matrix(data=rexp(n * numSim, lambda), nrow=numSim) #exponential Distribution
meanExpDist <- data.frame(means=apply(expDist, 1, mean))
```

Now that the simulation is complete we can look at the means:

```
# plot the means
ggplot(data = meanExpDist, aes(x = means)) +
  geom_histogram(binwidth=0.2) +
  scale_x_continuous(breaks=round(seq(min(meanExpDist$means), max(meanExpDist$means),
by=1)))
```



## 1. Sample Mean versus Theoretical Mean

The expected mean `mu` of a exponential distribution of rate `lambda` is `mu= 1\lambda`.

```
mu <- 1/lambda
print(paste("The expected value of means is ", mu))

## [1] "The expected value of means is  5"
```

The average sample mean of 1000 simulations of 40 randomly sampled exponential distributions can be easily calculated.

```
sampleMean <- mean(meanExpDist$means)
print(paste("The sampled value of means is", round(sampleMean,3)))

## [1] "The sampled value of means is 4.984"
```

The expected mean and the sample mean are very close.

## 2. Sample Variance versus Theoretical Variance

The expected standard deviation `sigma` of an exponential distribution of rate `lambda` is `sigma = (1/lambda)/sqrt(n)`.

```
sd <- 1/lambda/sqrt(n)
print(paste("The expected value of standard deviation is", round(sd,3)))
```

```
## [1] "The expected value of standard deviation is 0.791"
```

The expected variance var of standard deviation sigma is var = \sigma^2.

```
var <- sd^2
print(paste("The expected value of variance is", round(var,3)))

## [1] "The expected value of variance is 0.625"
```

Let var_sampled be the variance of the average sample mean of 1000 simulations of 40 randomly sampled exponential distribution, and sd_sampled the corresponding standard deviation.

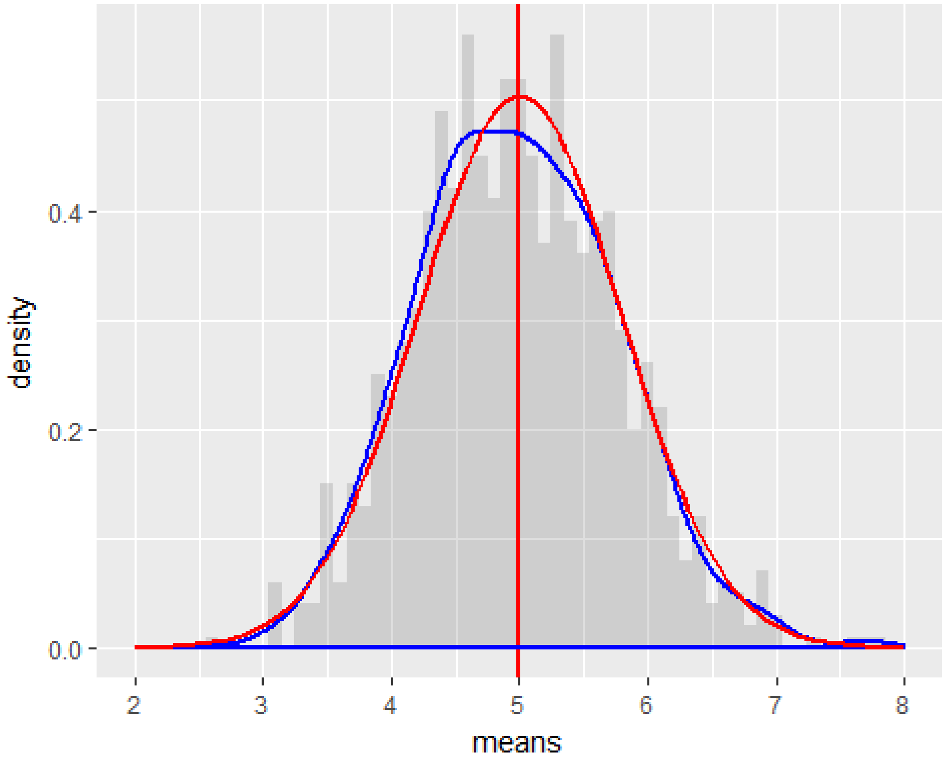
```
sd_sampled <- sd(meanExpDist$means)
print(paste("The sampled value of standard deviation is", round(sd_sampled,3)))

## [1] "The sampled value of standard deviation is 0.777"

var_sampled <- var(meanExpDist$means)
print(paste("The sampled value of variance is", round(var_sampled,3)))

## [1] "The sampled value of variance is 0.604"
```

As you can see both of the standard deviations and variances are very close.

## 3. Distribution of the Data

Comparing the population means & standard deviation with a normal distribution of the expected values. Added lines for the calculated and expected means

```
# plot the means
ggplot(data = meanExpDist, aes(x = means)) +
  geom_histogram(binwidth=0.1, aes(y=..density..), alpha=0.2) +
  geom_vline(xintercept = mu, size=1, colour="red") +
  geom_density(colour="blue", size=1) +
  scale_x_continuous(breaks=seq(mu-3,mu+3,1), limits=c(mu-3,mu+3))  +
  stat_function(fun = dnorm, args = list(mean = mu , sd = sd), colour = "red", size=1)
```



As you can see from the graph, the calculated distribution of means of random sampled exponantial distributions, overlaps quite nice with the normal distribution with the expected values based on the given lamba.