# Peer Assessment on Statistical Inference and the CLT - Data Science Specialization

***Michael Karp***

***1/22/2015***

# Inference

In this project you 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. Set lambda = 0.2 for all of the simulations. You will investigate the distribution of averages of 40 exponentials. Note that you will need to do a thousand simulations.

```
# set seed to make research reproducible
set.seed(1000)
# setting the theo_mean = 1/lambda which is the theoretical mean and standard deviation for
an exponential distribution
lambda <- 0.2
# setting n = 40 as set by the spec
n <- 40

# Will illustrate the central limit theorem through running different amounts of simulation
s
# By running 1, 10, 100, and 1000 simulations we will observe that as n (the number of
# observations) increases the difference between the theoretical mean and standard deviatio
n
# with the mean of the sample mean and standard deviations will converge to 0

# 1 Show the sample mean and compare it to the theoretical mean of the distribution.
# theoretical mean = 1/lambda = 1/.2 = 5
theo_mean <- 1/lambda
theo_mean
```

```
[1] 5
```

```
# mean 1 samples variable
mns1 = mean(rexp(n, lambda))
# mean of sample means after running 1 simulation of the exponential rexp(40, .2)
sample_mean1 <- mean(mns1)
sample_mean1
```

[1] 4.514222

```
# mean 10 samples variable
mns10 = NULL
# loop 10 times to get 10 simulations
for (i in 1 : 10) mns10 = c(mns10, mean(rexp(n, lambda)))
# the distribution of averages with a small number of simulations is still quite skewed

# mean of sample means after running 10 simulations of the exponential rexp(40, .2)
sample_mean10 <- mean(mns10)
sample_mean10
```

[1] 4.743706

```
# mean 100 samples variable
mns100 = NULL
# loop 100 times to get 100 simulations
for (i in 1 : 100) mns100 = c(mns100, mean(rexp(n, lambda)))

# mean of sample means after running 100 simulations of the exponential rexp(40, .2)
sample_mean100 <- mean(mns100)
sample_mean100
```

[1] 5.002545

```
# mean 1000 samples variable
mns1000 = NULL
# loop 1000 times to get 1000 simulations
for (i in 1 : 1000) mns1000 = c(mns1000, mean(rexp(n, lambda)))

# mean of sample means after running 1000 simulations of the exponential rexp(40, .2)
sample_mean1000 <- mean(mns1000)
sample_mean1000
```

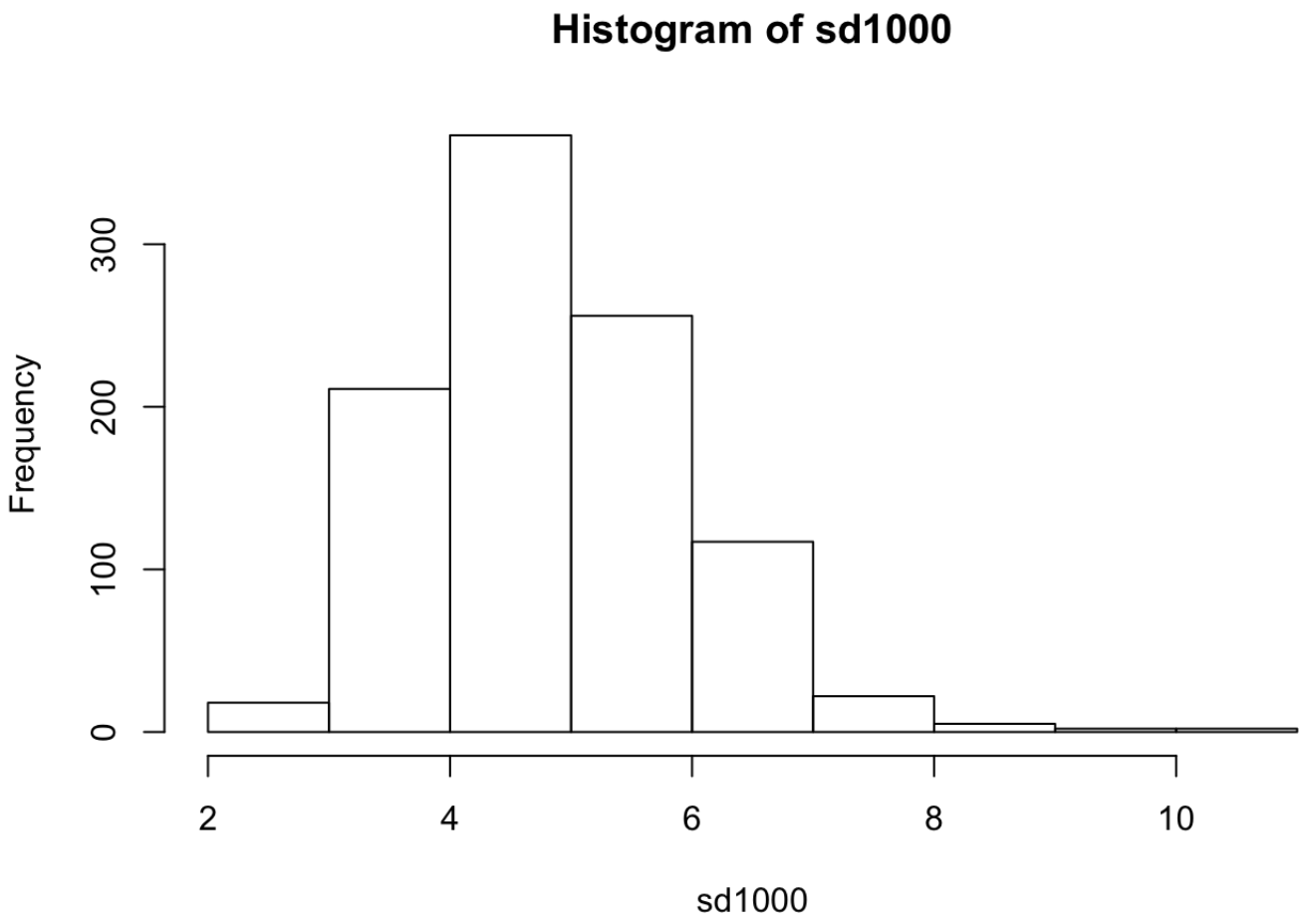
[1] 4.980874

```
# 2. Show how variable the sample is (via variance) and compare it to the theoretical varia
nce of the distribution.
# theoretical standard deviation = 1/lambda = 1/.2 = 5
theo_sd <- 1/lambda
theo_sd
```

[1] 5

```
# sd10 samples variable

# mean 1000 samples variable
sd1000 = NULL
# loop 1000 times to get 1000 simulations
for (i in 1 : 1000) sd1000 = c(sd1000,sd(rexp(n, lambda)))
# almost normally distributed
hist(sd1000)
```



```
# mean of sample means after running 1000 simulations of the exponential rexp(40, .2)
sample_sd1000 <- mean(sd1000)
sample_sd1000
```

[1] 4.856607

```
# 3. Show that the distribution is approximately normal.
# histogram of the sample means is approximately normal
# here's the histogram of 1000 random normal distributed variables for comparrison
hist(rnorm(1000))
```


```
sd(mns1000)
```

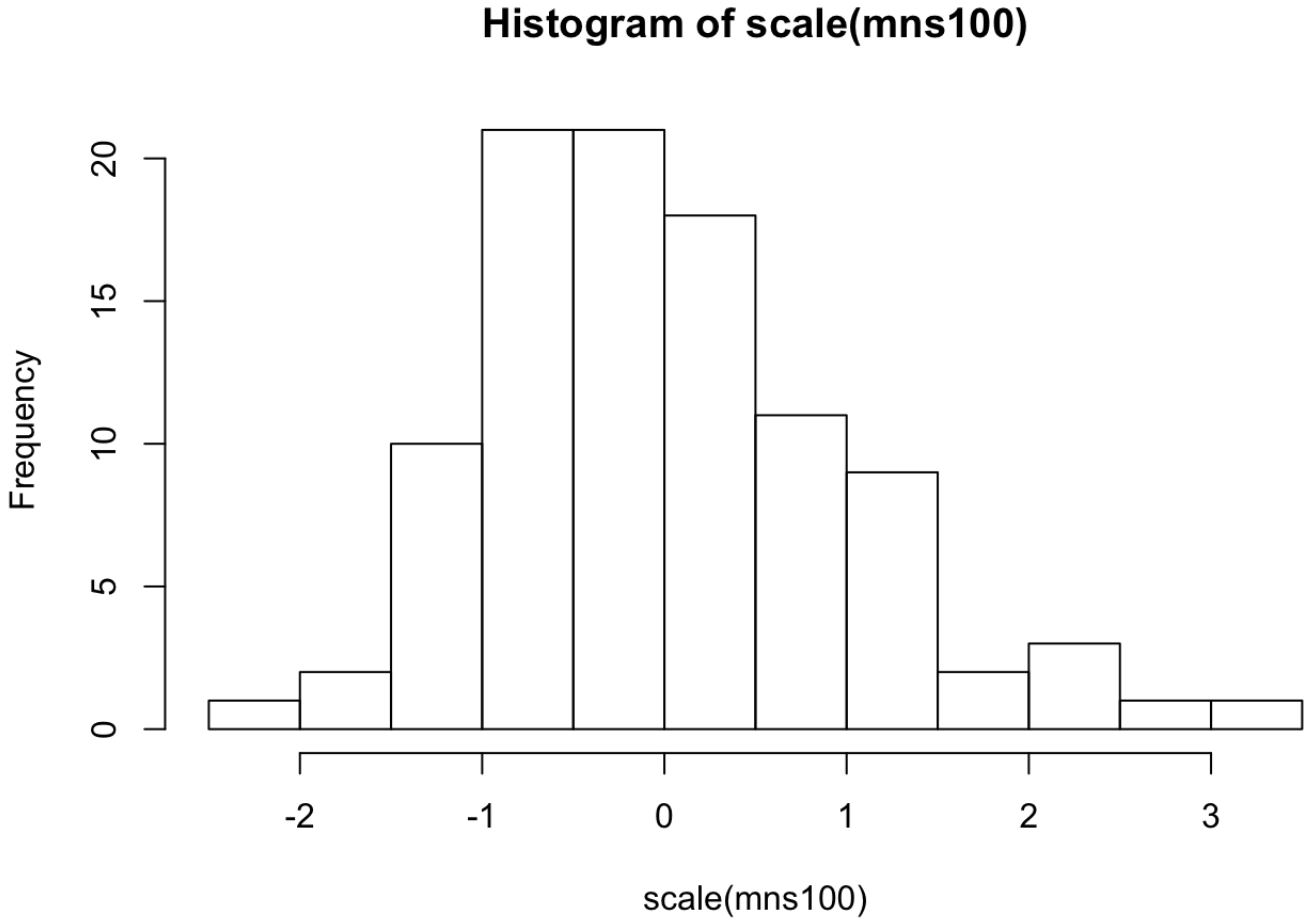
[1] 0.792266

```
# 0.7934196 - approcahing 1 as we increase n
mean(mns1000) - theo_mean
```

[1] -0.01912555

```
# the difference between the true mean and the theoretical mean of the distribution approac
hes 0 as n
# increases

# plotting a histogram of scaled mns100 data which was the 100 sample means from running th
e rexp reveals
# this data is not very normally distributed
hist(scale(mns100))
```



```
# plotting a histogram of scaled mns1000 data which was the 1000 sample means from running the rexp shows that
# it has roughly a mean of 0 and sd of 1 along with a nice bell curve
hist(scale(mns1000))
```