Learnings from Statistical Inference

By Andrew Doyle

Overview: In this report we will analyze the Central Limit Theorem (CLT) through sampling the exponential distribution. We compare results for drawing a single large sample as well as repeatedly drawing small samples. We find that…

The exponential distribution:

For the purposes of this report the probability density function of exponential distribution is defined as follows:

$$p(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

Drawing one large sample:

In Figure 1 we show 1,000 values pulled from an exponential distribution with λ=0.2. We see, as expected, that the histogram appears to be bounded by an exponentially decreasing function. We show the population mean with a dashed vertical line, because this distribution is not symmetric, so it is difficult to see directly. It's difficult to visualize the variance this way, so we haven't attempted to do so.

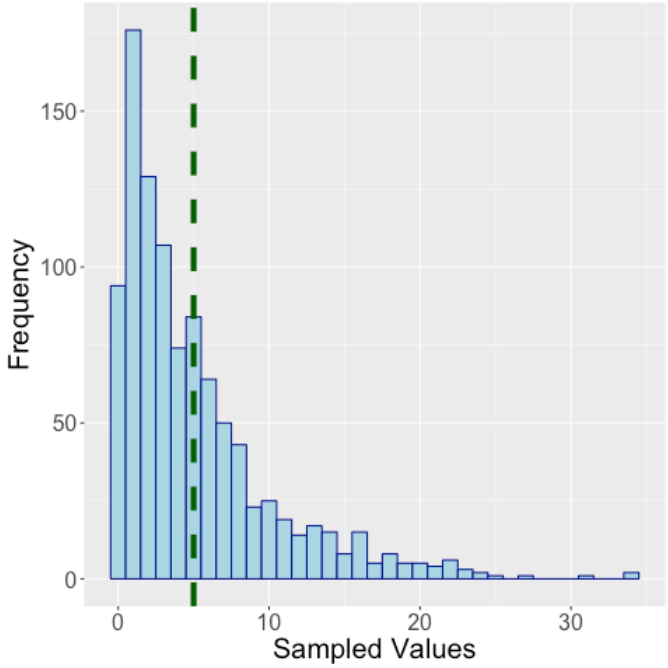

Figure 1-Values for 1,000 points pulled from an exponential distribution with λ=0.2. The dashed green vertical line indicates the population mean

As we show in the following table the sample mean reflects the population value well, though the sample variance is still notably different the population value. Part of this is likely due to the fact that the variance is a squared function of the mean, so it may be more fair to compare the mean and the standard deviation. However, even then we'd see a larger discrepancy than for the means (5.14 sample STD, 5.00 population STD).

	Calculated (n=1000)	Distribution
Mean	5.09	5
Variance	26.44	25

Repeatedly drawing smaller samples:

Sample Mean

In Figure 2 we show our distribution of sample means from drawing a far smaller set (n=40) from an exponential distribution with λ=0.2 a total of one thousand times. The values in the histogram here represent the mean of each sample.

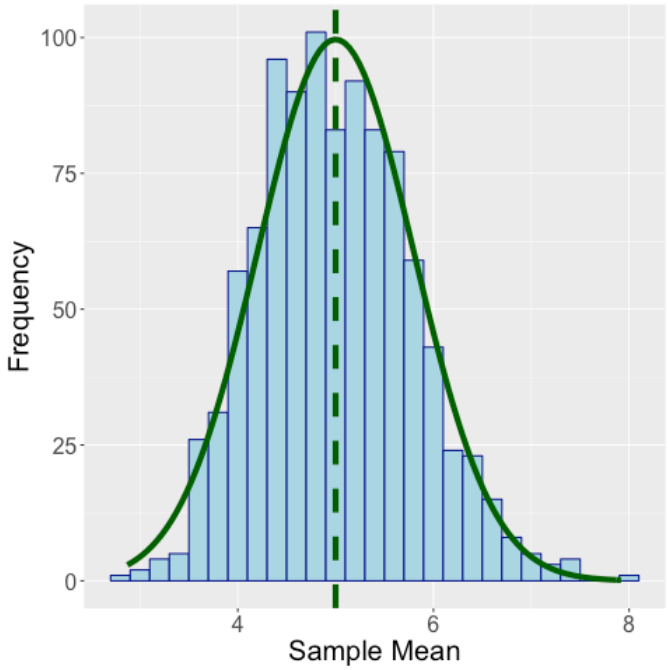

Figure 2- Means from 1,000 iterations of samples of 40 points from the exponential distribution with λ=0.2. Histogram values represent the mean of each sampled set, the dashed green vertical line indicates the distribution mean, and the solid green curve indicates the predicted normal distribution via the CLT.

We know that the mean of the exponential distribution as defined here is 1/ λ, and we show that value as a dashed vertical green line in the plot. Encouragingly we see that the distribution of the sample means is approximately symmetric and peaks around this point, in congruence with what we would expect from the Central Limit Theorem. The solid green line indicates the predicted normal distribution for the sample means, and we see that it represents an excellent fit to the simulated results. The mean of sample means is virtually indistinguishable from the distribution mean (often accurate to two decimal places, though the code is stochastic).

Sample Variance

In Figure 3 we show our distribution of sample variances from drawing a far smaller set (n=40) from an exponential distribution with λ=0.2 a total of one thousand times. The values in the histogram here represent the variance of each sample.

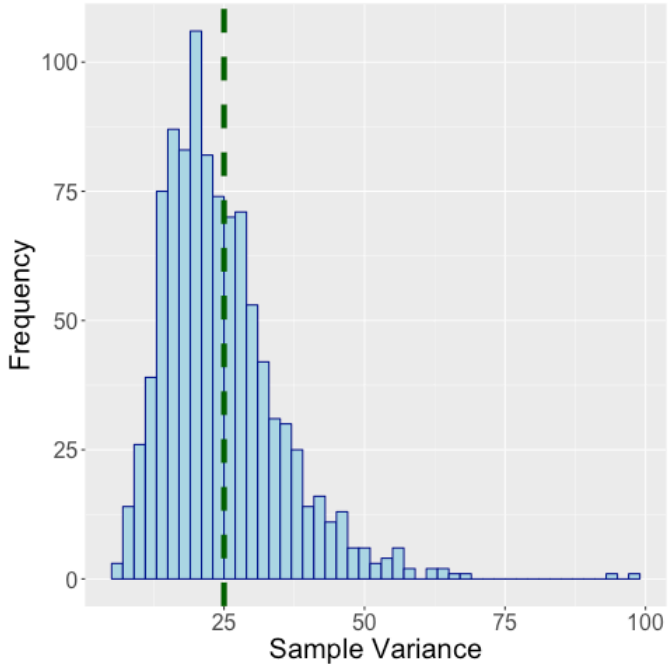

Figure 3-Variances from 1,000 iterations of samples of 40 points from the exponential distribution with λ=0.2. Histogram values represent the mean of each sampled set, the dashed green vertical line indicates the distribution variance.

The distribution variance is $1/\lambda^2$ and we see our sample variance predictions peak around that point, as shown by the dashed green vertical line. However, this distribution is *not* normal. This makes some sense when we remember that the variance may never take on a negative value, but in some cases it can be quite large. The exact form of this "curve" bounded by this histogram is outside the scope of the current discussion, though it bears some resemblance to a χ^2 distribution (and would approximate it if we'd sampled from a normal random variables).

Summary

The Central Limit Theorem (CLT) indicates that the mean of a sample of independent random variables should be normally distributed, even if the distribution the samples are pulled from is not itself normally distributed. We verified this by taking the mean of 1,000 samples of 40 points each, but we pulled those points from an exponential distribution. In Figure 2 we see the results, and that there is an exceptional fit to the predicted normal distribution. However, In Figure 3 we see that these results do not extend to the sample variance.

	Calculated (Samples of 40; n=1000)	**Distribution**
Mean	5.02	5.00
Variance	25.43	25.00

The Code:

One sample of one thousand points (Figure 1)

```
# Setting the stage
rm(list=ls())
library(ggplot2)
library(scales)
setwd('~/Dropbox/Coursera/JHU_DataScience/Statistical_Inference/Project')

# To pull n data from an exponential distribution, lambda*exp(-lambda*x) for x>0
# p <- rexp(n,lambda)
# The distribution mean is 1/lambda
# The distribution standard deviation is 1/lambda

# The  tuning parameter for the exponential distribution
lambda <- 0.20
ilambda <- 1/lambda
# The number of points we'll pull from a distribution for each average
n <- 1
# The number of times we're running the simulation
m <- 1000

# Run the simulations
p <- rexp(m*n,lambda)
# Organize the values for easy access
q <- matrix(p,nrow=m,ncol=n)

# Save our results to a data frame
df <- data.frame(q)
colnames(df) <- c('vals')

# Show the expected distribution of means via CLT
custom_normal <- function(x, mean, sd, m, n, bw){m * bw * dnorm(x=x, mean=mean, sd=sd/sqrt(n-1)) }

# Save output to a file
png("sample_large_distribution.png", width=480, height=480)

# Create an output histogram for the distribution of the means
bw <- 1
g <- ggplot(df, aes(x=vals)) +
  geom_histogram(color="darkblue", fill="lightblue", binwidth=bw) +
  geom_vline(aes(xintercept=ilambda),color="darkgreen", linetype="dashed", size=2) +
  labs(x="Sampled Values", y="Frequency") +
  theme(axis.text=element_text(size=16), axis.title=element_text(size=20))

print(g)
dev.off()
```

The Mean: 1,000 iterations of 40 sampled points (Figure 2)

```
# Setting the stage
rm(list=ls())
library(ggplot2)
library(scales)
setwd('~/Dropbox/Coursera/JHU_DataScience/Statistical_Inference/Project')

# The  tuning parameter for the exponential distribution
lambda <- 0.20
ilambda <- 1/lambda
# The number of points we'll pull from a distribution for each average
n <- 40
# The number of times we're running the simulation
m <- 1000

# Run the simulations
p <- rexp(m*n,lambda)
# Organize the values for easy access
q <- matrix(p,nrow=m,ncol=n)

# Calculate the means and standard deviations
q_means <- apply(q,1,mean)
q_std <- sqrt(apply(q,1,var))

# Save our results to a data frame
df <- data.frame(q_means,q_std)
colnames(df) <- c('mean','std')

# Show the expected distribution of means via CLT
custom_normal <- function(x, mean, sd, m, n, bw){m * bw * dnorm(x=x, mean=mean, sd=sd/sqrt(n-1)) }

# Save output to a file
png("sample_mean_distribution.png", width=480, height=480)
# Create an output histogram for the distribution of the means
bw <- 0.2
g <- ggplot(df, aes(x=mean)) +
  geom_histogram(color="darkblue", fill="lightblue", binwidth=bw) +
  geom_vline(aes(xintercept=ilambda),color="darkgreen", linetype="dashed", size=2) +
  stat_function(fun=custom_normal,
            args=c(mean=ilambda, sd=ilambda, m=m, n=n, bw=bw),
            size=2,
            color="darkgreen") +
  labs(x="Sample Mean", y="Frequency") +
  theme(axis.text=element_text(size=16), axis.title=element_text(size=20))

print(g)
dev.off()
```

The Variance: 1,000 iterations of 40 sampled points (Figure 3)

```
# Setting the stage
rm(list=ls())
library(ggplot2)
library(scales)
setwd('~/Dropbox/Coursera/JHU_DataScience/Statistical_Inference/Project')

# To pull n data from an exponential distribution, lambda*exp(-lambda*x) for x>0
# p <- rexp(n,lambda)
# The distribution mean is 1/lambda
# The distribution standard deviation is 1/lambda

# The  tuning parameter for the exponential distribution
lambda <- 0.2
ilambda <- 1/lambda
# The number of points we'll pull from a distribution for each average
n <- 40
# The number of times we're running the simulation
m <- 1000

# Run the simulations
p <- rexp(m*n,lambda)
# Organize the values for easy access
q <- matrix(p,nrow=m,ncol=n)

# Calculate the means and standard deviations
q_means <- apply(q,1,mean)
q_var <- apply(q,1,var)

# Save our results to a data frame
df <- data.frame(q_means,q_var)
colnames(df) <- c('mean','var')

# Save output to a file
png("sample_var_distribution.png", width=480, height=480)

# Create an output histogram for the distribution of the means
bw <- 2
ddf <- n-1
g <- ggplot(df, aes(x=var)) +
  geom_histogram(color="darkblue", fill="lightblue", binwidth=bw) +
  geom_vline(aes(xintercept=ilambda^2),color="darkgreen", linetype="dashed", size=2) +
  labs(x="Sample Variance", y="Frequency") +
  theme(axis.text=element_text(size=16), axis.title=element_text(size=20))

print(g)
dev.off()
```