Central Limit Theorem: A Simulation Exploration

Sean Tully - 10/23/2015

Overview

In this paper, the r function 'rexp' will be used to synthesize a dataset of 1000 simulations of 40 observations each. This dataset will then be used to explore the principals of the Central Limit Theorem.

> The averages of independent and identically distributed (iid) variables will tend towards a normal distribution as the population of measurements grows.

Simulation Setup

This paper will be focused on the *mean* and *standard deviation* of 1000 samples of the rexp function with 40 observations and frequency (λ) 0.2 each. The theoretical mean and standard deviation of rexp are both $1/\lambda = 1/0.2 = 5$. Based on the CLT it is expected that the average of the means and standard deviations from 1000 simulations should follow a normal distribution centered at 5.

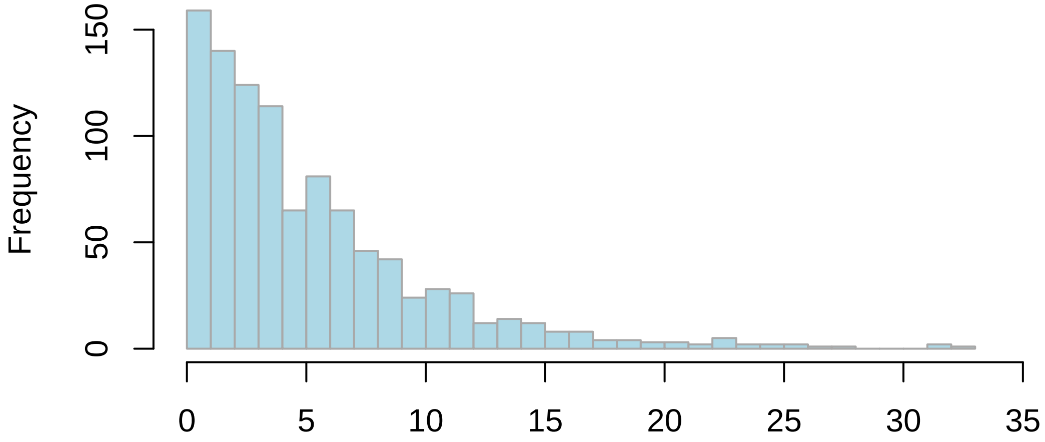
Data Simulation

Initialization of the required data sets needed for analysis requires setup of base variables (simulations, observations, frequency) and the setting of a system seed (0) to allow for replication of this analysis. Code 1

Two main sets of data are of interest in this analysis. Code 2

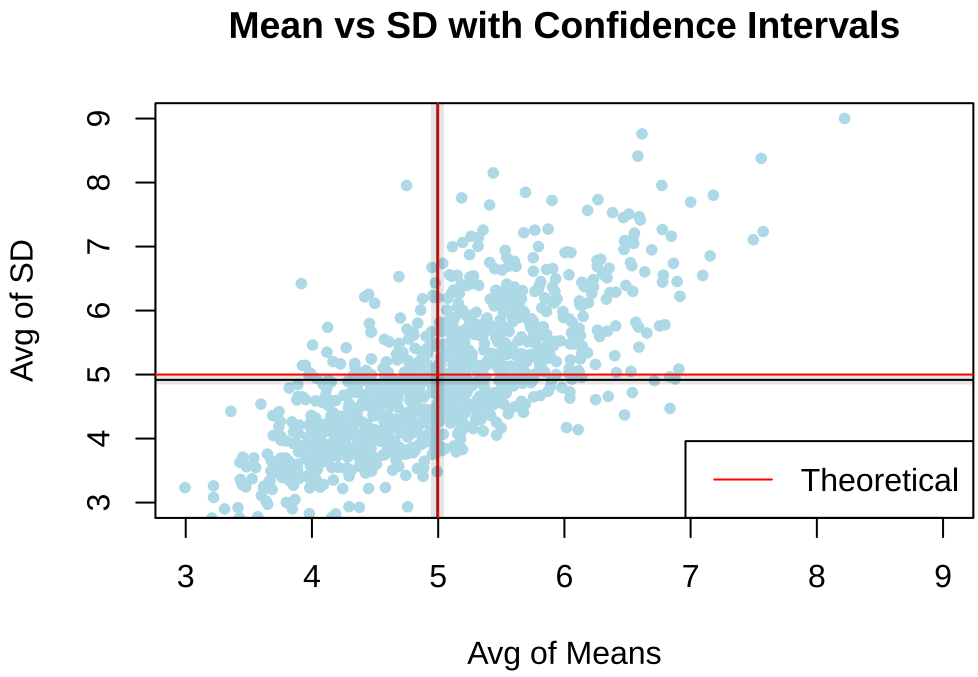
- A control group which is a simple representation of the exponential function with 1000 observations
- A dataset containing the mean and standard deviation for each of 1000 samples

Taking a look at the distribution of the control group, it's apparent that the function does not conform to a normal distribution. Code 3

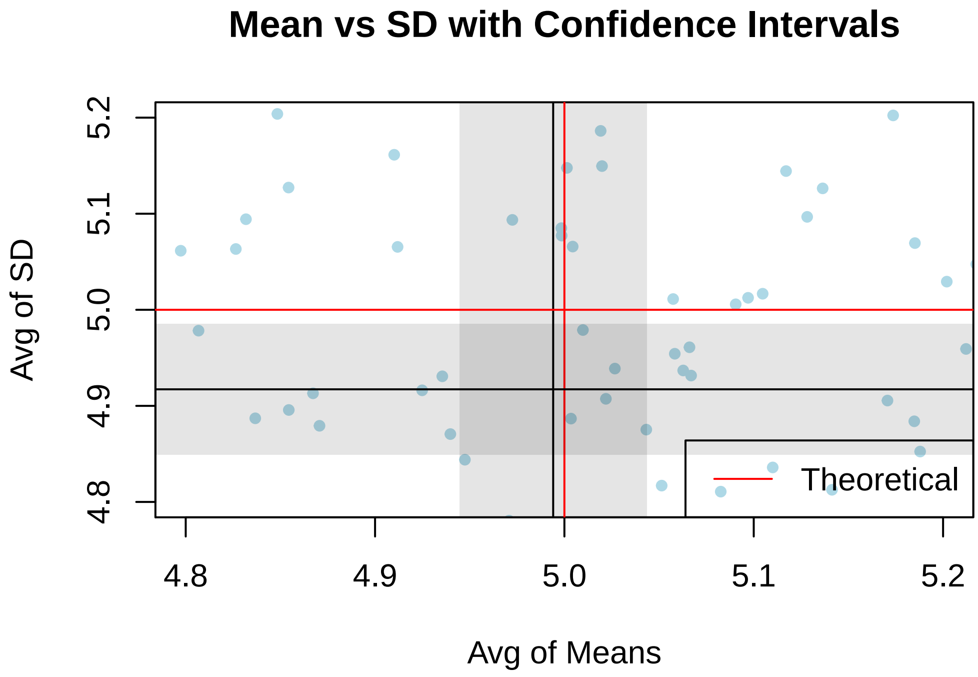


Sample vs. Theory

Plotting average means against average standard deviations the relative clustering can be compared to theoretical expectation concurrently. Code 4 Code 5

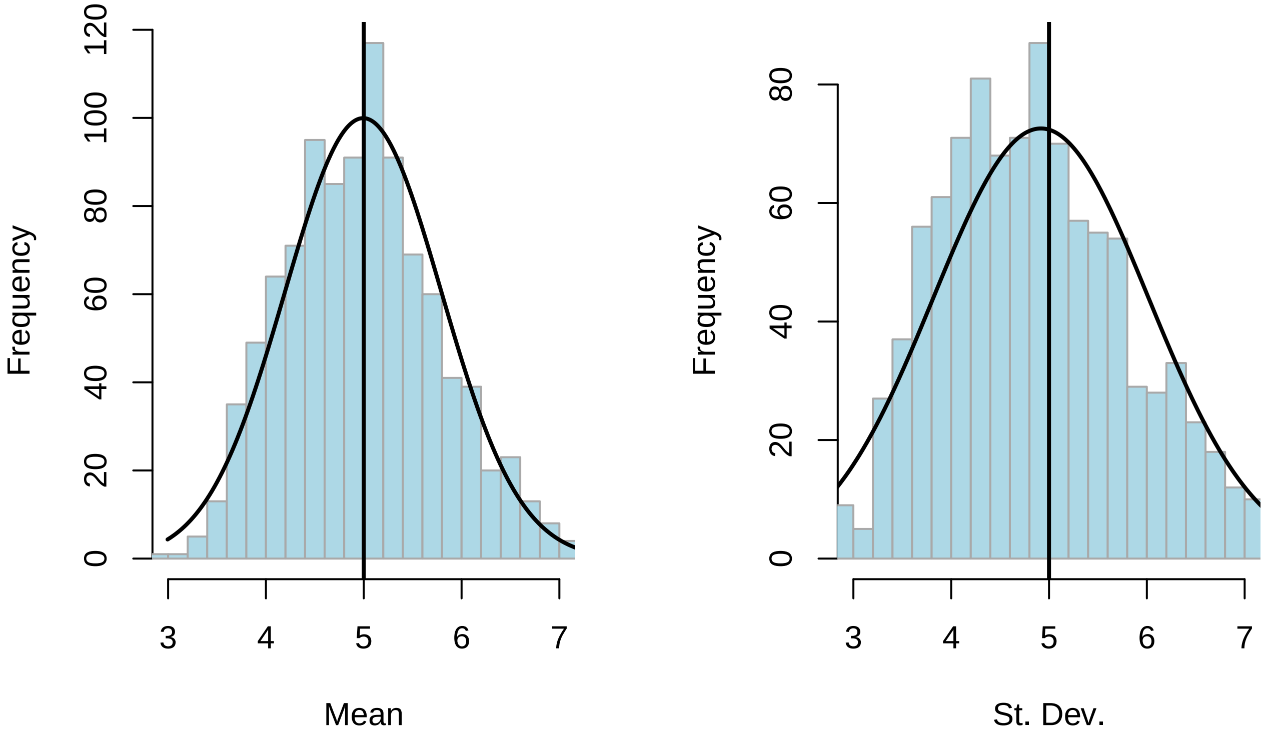


Zooming in to discern fine detail it can be seen that the theoretical mean falls within the sample's confidence interval (grayed region). However, this does not hold true for the theoretical standard deviation. This indicates that the individual sample observations may need to be increased to properly conform to a normal distribution. Code 6



Normalcy of iid Variables

Looking at the distribution of mean and SD calculations against their corresponding normals it can be seen that they both closely follow a normal distribution centered around 5. Code 7



Again numerically:

- The theoretical mean falls within the sample mean confidence intervals and as such the sample can be treated as a normal distribution
- The theoretical st dev falls outside of the sample confidence interval indicating that while approaching a normal distribution, the observation size was too small of a sample group in each observation and should be increased say from 40 to 200. Code 8

iid	Var Low	er Limit Upp	er Limit
mlims	Avg Mean	4.945	5.044
sdlims	Avg SD	4.849	4.985

Appendix:

c01 Variable Initialization:

```
set.seed(0)
sim_n <- 1000
n <- 40
lam <- 0.2
```

c02 Dataset Creation:

```
control <- rexp(sim_n,lam)
measures <- data.frame(obs=NULL,mean=NULL,stdev=NULL)

  for (i in 1:sim_n) {
    tmp <- rexp(n,lam)
    measures <- rbind(measures, data.frame(obs=i,mean=mean(tmp),stdev=sd(tmp)))
  }
```

c03 Visualization of rexp distributation:

```
par(mai=c(1,1.2,0.1,0.1))
h <- hist(control, breaks = 35
            , xlim = c(0,35)
            , col="light blue"
            , border = "dark gray"
            , xlab = NULL
            , main = NULL)
```

c04 Set confidence intervals:

```
mlims <- mean(measures$mean) + c(-1,1) * qnorm(0.975) *     sd(measures$mean)/sqrt(sim_n)
sdlims <- mean(measures$stdev) + c(-1,1) * qnorm(0.975) * sd(measures$stdev)/sqrt(sim_n)
lims <- cbind(data.frame(c("Avg Mean","Avg SD")),rbind(mlims,sdlims))
names(lims) <- c("iid Var","Lower Limit","Upper Limit")
```

c05 Plot mean vs. SD (normal):

```
plot(measures$mean,measures$stdev
                  ,col="light blue"
                  ,pch=20
                  ,xlim = c(3,9)
                  ,ylim = c(3,9)
                  , xlab="Avg of Means"
                  ,ylab="Avg of SD"
                  ,main="Mean vs SD with Confidence Intervals")
legend("bottomright",lty=c(1,1),col = c(rgb(1,0,0)),legend=(c("Theoretical")))
polygon(c(mlims[1],mlims[1],mlims[2],mlims[2],mlims[1])
        ,c(0,10,10,0,0)
        , col = rgb(0,0,0,0.1)
        , border = FALSE)
abline(v = mean(measures$mean),col = rgb(0,0,0))
abline(v = 5,col = rgb(1,0,0))
polygon(c(0,0,10,10,0)
        ,c(sdlims[1],sdlims[2],sdlims[2],sdlims[1],sdlims[1])
        , col = rgb(0,0,0,0.1)
        , border = FALSE)
abline(h = mean(measures$stdev),col = rgb(0,0,0))
abline(h = 5,col = rgb(1,0,0))
```

c06 Plot mean vs. SD (zoomed):

```
plot(measures$mean,measures$stdev
                  ,col="light blue"
                  ,pch=20
                  ,xlim = c(4.8,5.2)
                  ,ylim = c(4.8,5.2)
                  , xlab="Avg of Means"
                  ,ylab="Avg of SD"
                  ,main="Mean vs SD with Confidence Intervals")
legend("bottomright",lty=c(1,1),col = c(rgb(1,0,0)),legend=(c("Theoretical")))
polygon(c(mlims[1],mlims[1],mlims[2],mlims[2],mlims[1])
        ,c(0,10,10,0,0)
        , col = rgb(0,0,0,0.1)
        , border = FALSE)
abline(v = mean(measures$mean),col = rgb(0,0,0))
abline(v = 5,col = rgb(1,0,0))
polygon(c(0,0,10,10,0)
        ,c(sdlims[1],sdlims[2],sdlims[2],sdlims[1],sdlims[1])
        , col = rgb(0,0,0,0.1)
        , border = FALSE)
abline(h = mean(measures$stdev),col = rgb(0,0,0))
abline(h = 5,col = rgb(1,0,0))
```

c07 Plots with normal distribution overlays:

```
par(mai=c(1.0,1.2,0.1,0.1))
h <- hist(measures$mean, breaks = 35
                       , xlim=c(3,7)
                       , col="light blue"
                       , border = "dark gray"
                       , xlab="Mean"
                       , main=NULL)

  xfit<-seq(min(measures$mean),max(measures$mean),length=500)
  yfit<-dnorm(xfit,mean=mean(measures$mean),sd=sd(measures$mean))
  yfit <- yfit*diff(h$mids[1:2])*length(measures$mean)

    lines(xfit, yfit, col="black", lwd=2)
    abline(v = 5,col = "black", lwd = 2)

h <- hist(measures$stdev, breaks = 70
                        , xlim=c(3,7)
                        , col="light blue"
                        , border = "dark gray"
                        , xlab="St. Dev."
                        , main=NULL)

  xfit<-seq(min(measures$stdev),max(measures$stdev),length=500)
  yfit<-dnorm(xfit,mean=mean(measures$stdev),sd=sd(measures$stdev))
  yfit <- yfit*diff(h$mids[1:2])*length(measures$stdev)

    lines(xfit, yfit, col="black", lwd=2)
    abline(v = 5,col = "black", lwd = 2)
```

c08 Confidence interval table:

```
library(knitr)
kable(lims,digits=3)
```