Statistical Inference (Simulation Exercise)

EHarris

6/27/2017

Packages for Course 6 Project

```
## Loading required package: plyr
```

```
## Loading required package: dplyr
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
## Loading required package: data.table
```

```
## -------------------------------------------------------------------------
```

```
## data.table + dplyr code now lives in dtplyr.
## Please library(dtplyr)!
```

```
## -------------------------------------------------------------------------
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## Loading required package: dtplyr
```

```
## Loading required package: lubridate
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday,
##     week, yday, year
```

```
## The following object is masked from 'package:plyr':
##
##     here
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
## Loading required package: ggplot2
```

```
## Loading required package: scales
```

```
## Loading required package: reshape2
```

```
##
## Attaching package: 'reshape2'
```

```
## The following objects are masked from 'package:data.table':
##
##     dcast, melt
```

```
## Loading required package: knitr
```

```
## Loading required package: R.cache
```

```
## R.cache v0.12.0 (2015-11-12) successfully loaded. See ?R.cache for help.
```

```
## Loading required package: stringr
```

```
## Loading required package: gtools
```

```
## Loading required package: quantreg
```

```
## Loading required package: SparseM
```

```
##
## Attaching package: 'SparseM'
```

```
## The following object is masked from 'package:base':
##
##     backsolve
```

Simulation Exercise Overview

This report will illustrate how well sample populations can provide a reliable picture or understanding of an entire population. To do this, we will compare the sample means and sample variances for different sample sizes to the theoretical mean and variance. We will be looking at an *Exponential Distribution* to illustrate. Some facts about this distribution:
1. Lambda = 0.2 2. Mean of exponential distribution is 1/lambda = 5 [1 / 0.2]
3. Standard deviation of exponential distribution is 1/lambda or 5
4. Variance of exponential distribution is 25, standard deviation squared

If we look at very large population, maybe 10,000 random exponential observations, we might expect the mean and variance for this sample population to be close to the theoretical values previously noted. Let's create a sample of 10,000 random exponential observations using the function rexp(n, lambda).

Large Sample Mean & Variance (10,000 random exponential observations)

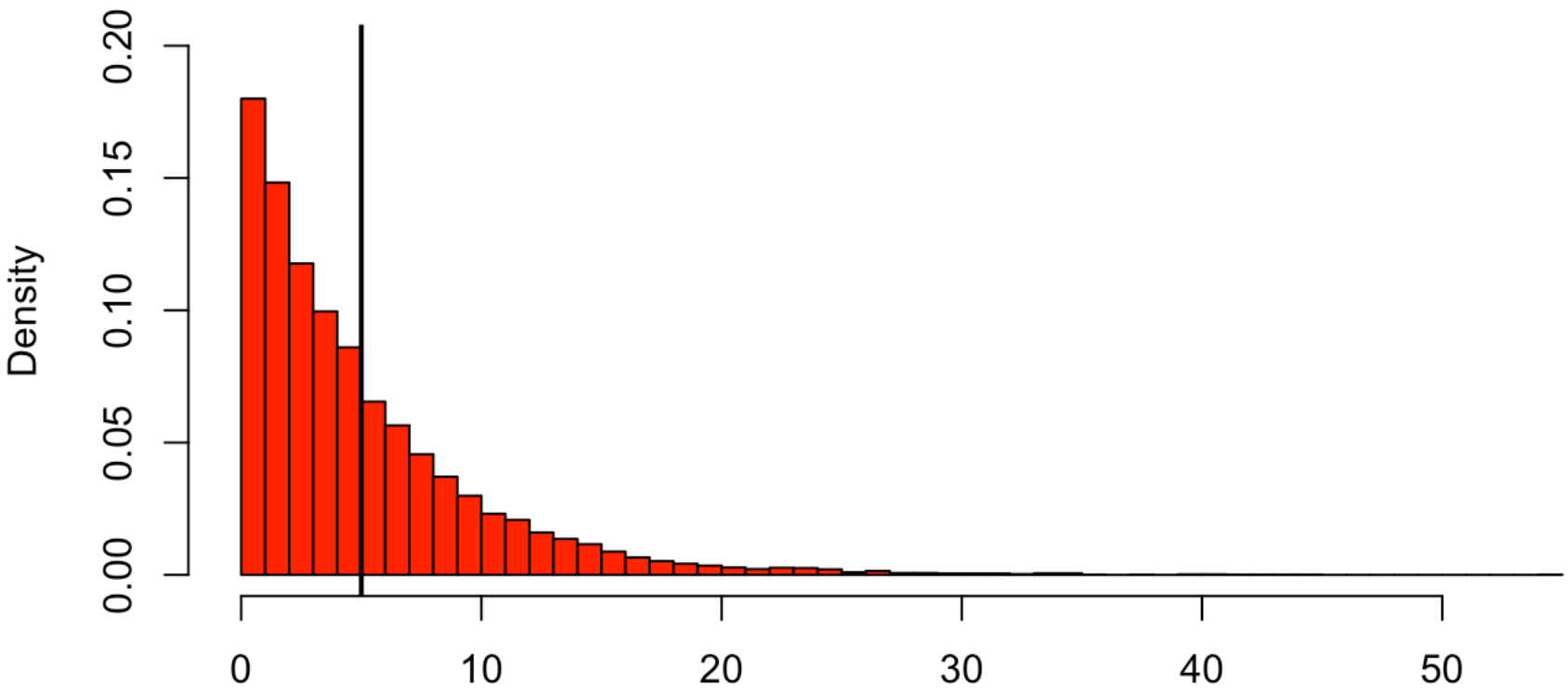
```
set.seed(100)
n.lrg <- 10000                    ## Large Sample, number of random exponential observ
ations
lrg.sample <- data.frame(rating = c(rexp(n.lrg, lambda)))
sample.mean <- round(mean(lrg.sample$rating), 3)   ## Population Mean
lrg.sample <- mutate(lrg.sample, diff.square = (rating - sample.mean)^2)
sample.var <- round(sum(lrg.sample$diff.square)/(n.lrg-1) ,3)   ## For sample varianc
e divide by n - 1
```

Here is a comparison sample versu theoretical:
* Mean: Sample 5.036 versus a Theoretical of 5
* Variance: Sample 26.006 versus a Theoretical of 25

Pretty close. Although, you might have expected the sample variance to be a little closer to the theoretical variance.

Let's look at a density distribution for this sample of 10,000 random exponential values.

```
par(mfcol = c(1,1), mar = c(7,4,6,1))
hist(lrg.sample$rating, prob = TRUE, breaks = 40, col = "red", xlab = "", ylim = c(0,
.2),
     main = "Distribution of Random Exponentials (10k)", cex.main = 1.0)
abline(v = 5, col = "black", lwd = 2)
```



Simulations

Instead of a single large sample, we will look at the average mean and average variance based on a varying number of simulations on sample of 40 Random Exponential values. Below is a brief outline of steps to be followed for our simulation. Our expectation is that, as the number of samples within a simulation increases, the average mean and variance will draw nearer to the theoretical values.

1. Simuluations will include 1 to 1,000 random sample exponenetial (variable name = ‘nsamp’)
2. Evaluate exponential distribution, using rexp(n, lambda), where
+ n = 40 [random sample of 40 exponentials]
+ lambda = 0.2
3. Calculate the mean and variance for each sample of 40 random exponentials within a simulation
4. Calculate an avearge mean and variance of all samples within a simulation
5. Plot / compare the average mean and variance against the theoritical values

Sample Mean Simulations

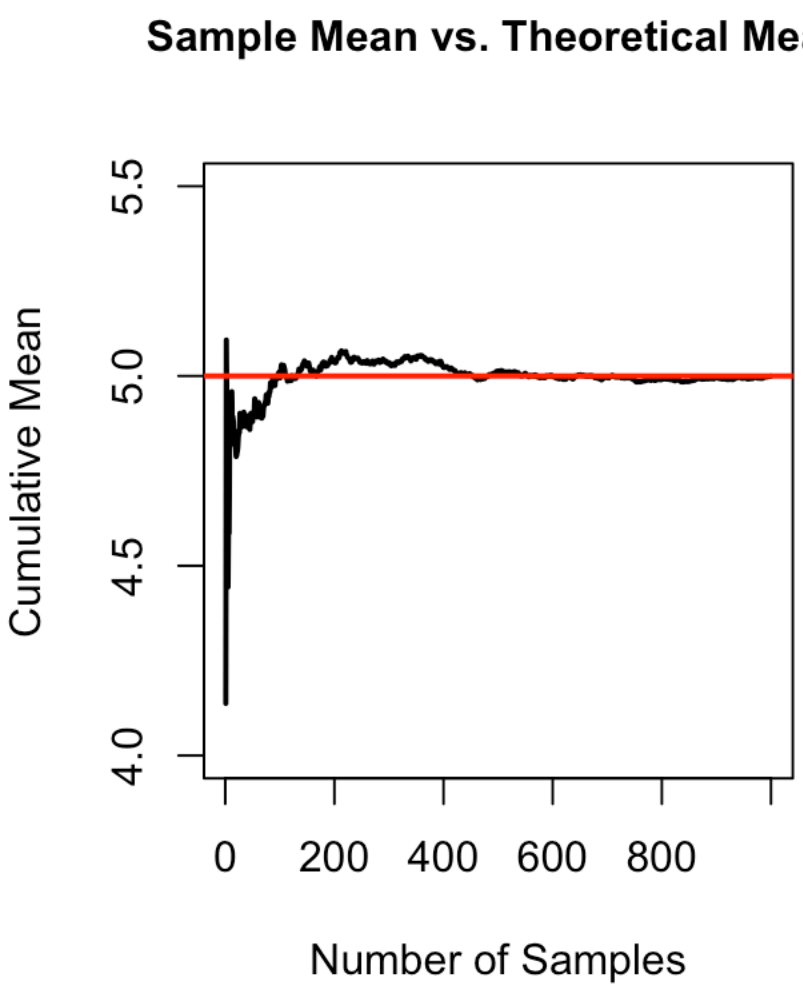
```
set.seed(100)
nsamp <- 1000
n <- 40              ## Sample size (number of random exponentials in each sample)
msim = NULL
for (i in 1: nsamp) msim <- c(msim, mean(rexp(n,lambda)))
```

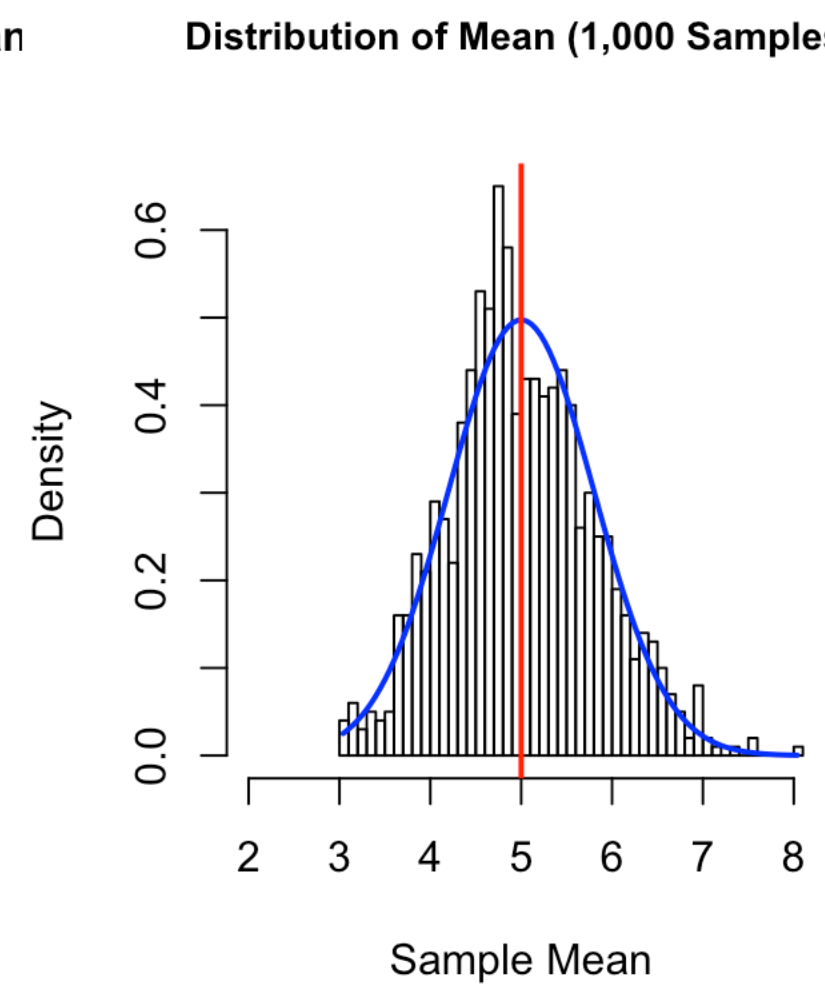
Plot 1: Sample Mean versus Theoretical Mean

Here we plot the average mean for each simulation. The x-axis reflects the number of samples within the simulation, reflecting between 1 and 1,000 samples. The y-axis reflects the calculated average mean within each simulation. A line is included on the y-axis representing the theoretical mean, 1/lambda which equals 5 [1/0.2].

```
sim.mean <- cumsum(msim) / (1:nsamp)
par(mfcol = c(1,2), oma = c(1,1,1,1), mar = c(6,4,5,1))
plot(sim.mean, type = "l", lwd = 2, col = "black", ylim = c(4,5.5),
     xlab = "Number of Samples",
     ylab = "Cumulative Mean",
     main = "Sample Mean vs. Theoretical Mean", cex.main = 1.0)
abline(h = 1/lambda, lwd = 2, col = "red")          ## Line for Theoretical Mean

hist(msim, prob = TRUE, breaks = 40,
     xlab = "Sample Mean", xlim = c(2,8),
     main = "Distribution of Mean (1,000 Samples)", cex.main = 0.9)
lines(seq(min(msim), max(msim), length = 100),
      dnorm(seq(min(msim), max(msim), length = 100), mean = mean(msim), sd = sd(msim)
),
      col = "blue", lwd = 2)
abline(v = 1/lambda, lwd = 2, col = "red")
```





This plot would seem to confirm that, as the number of simulations or samples increases, the Sample Mean closely approximates the Theoretical Mean. By the time we use 600 simulations, the Sample Mean and Theoretical Mean are almost identical. The other thing we wanted to check is whether the distribution of the sample mean would represent a normal distribution. Here we will illustrate a histogram representing the mean values for each of the 1,000 samples. Since the Theoretical Mean is 5 (1/lambda), we should anticipate that the greatest number, count, of means would be around 5.

The histogram does have the general shape of a normal distribution. It is worth noting, however, that the distribution is narrower than what you might expect. Based on a mu = 1/lambda and sd = 1/lambda, the theoritical distribution for 1 standard deviation, 90% confidence, would be 0 [mu - 1 sd] to 10 [mu + 1 sd]. The distribution for the sample means is between 3 and 8. This, actually, is consistent with theory as well. Essentially, as the population size increases, the closer it gets to the thing it is trying to estimate.

Sample Variance Simulations

Now let look at Sample Varance. Similar to the simulation for sample mean we are running simulations and calculating the average variance for simulations with 1 to 1,000 random samples of 40 exponential values.

To calculate sample variance we use the following formula: $1/(n-1)\Sigma(x_i - \mu)^2$.

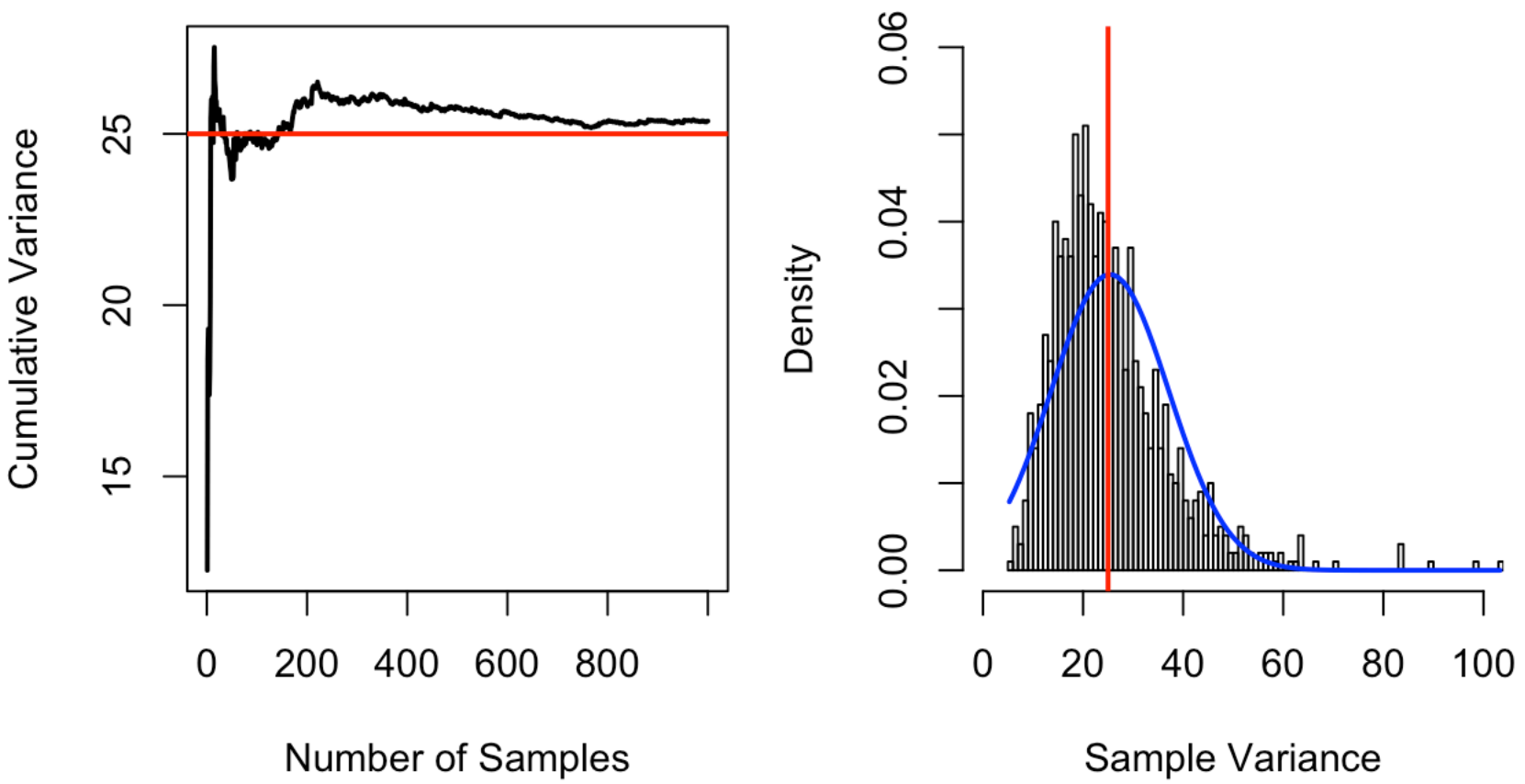
```
set.seed(100)
vsim = NULL
for (i in 1: nsamp) {
      x <- rexp(n,lambda)
      mean.x <- mean(x)
      vsim <- c(vsim, sum(c((x - mean.x)^2))/(n-1))
}
```

Plot 2: Sample Variance versus Theoretical Variance

In the first simulation, we will calculate the mean for a single sample of 40 random exponentials. The mean for this simulation may be considerably different from the theoretical mean. However, in the final simulation, we will have 1,000 samples of 40 random exponentials. It is anticipated the mean across each of the 1,000 samples will be more consistent with the theoretical mean.

```
sim.var <- cumsum(vsim) / (1:nsamp)
par(mfcol = c(1,2), oma = c(1,1,1,1), mar = c(6,4,5,1))
plot(sim.var, type = "l", lwd = 2, col = "black",
     xlab = "Number of Samples",
     ylab = "Cumulative Variance",
     main = "Sample Variance vs. Theoretical Variance", cex.main = 0.8)
abline(h = (1/lambda)^2, lwd = 2, col = "red")      ## Line for Theoretical Variance

hist(vsim, prob = TRUE, breaks = 100,
             xlab = "Sample Variance", xlim = c(0,100), ylim = c(0.00, 0.06),
             main = "Distribution of Variance(1,000 Samples)", cex.main = 0.8)
lines(seq(min(vsim), max(vsim), length = 100),
      dnorm(seq(min(vsim), max(vsim), length = 100), mean = mean(vsim), sd = sd(vsim)
),
      col = "blue", lwd = 2)
abline(v = (1/lambda)^2, lwd = 2, col = "red")
```



The graphs of variance are consistent with Mean. In the first graph, we see that as the number of sample variances increases, the average variance becomes closer to the theoretical mean. The second graph illustrating the distribution of the sample variances (1,000 Samples) may not represent a normal distribution as closely as the mean. Although, there remains some similarity to a normal. A portion of this distorted appearance may be attributed to the size/number of bins.