Statistical Inference Course Project, Part 1: Simulation Exercise

Marcus Vinicius Monteiro de Souza

June 17, 2017

Overview

In this project we're going to investigate the exponential distribution in R and compare it with the Central Limit Theorem, by doing the following:

1. Show a sample mean and compare it to the theoretical mean of the distribution.
2. Show how variable the sample is (via variance) and compare it to the theoretical variance of the distribution.
3. Show that the distribution is approximately normal.

Reproducibility

Here we install and load the R packages we used, and specify the seed to initialize the pseudorandom number generator.

```
InstallAndLoadRequiredPackages <- function() {
  # Load the required packages.
  if (!require('pacman')) {
    install.packages('pacman')
  }
  pacman::p_load(cowplot, grid, gridExtra, gtable, tidyverse)
}

InstallAndLoadRequiredPackages()
```

```
## Loading required package: pacman
```

```
set.seed(12345) # Everybody's in the car, so come on / Let's ride to the
```

Simulations

Here we use R's rexp function to generate 40 numbers from the exponential distribution, take their mean and store it in a tibble. We use .2 as the rate parameter. We repeat the process 1000 times.

```
ExponentialDistributionsMeans <- function(number.of.simulations,
                                          number.of.exponentials.per.simulation,
                                          rate) {
  replicate(number.of.simulations,
            mean(rexp(number.of.exponentials.per.simulation, rate))) %>%
    as_tibble() %>%
    rename(mean = value)
}

kNumberOfSimulations <- 1000
kNumberOfExponentialsPerSimulation <- 40
```

```r
kRate <- .2

exponential.distributions.means <- ExponentialDistributionsMeans(
  kNumberOfSimulations,
  kNumberOfExponentialsPerSimulation,
  kRate)
```

Sample Mean versus Theoretical Mean

Here we plot the simulations means' distribution, and vertical lines showing the it's mean and the theoretical mean(1 / rate). We can see that their values are very close to each other.

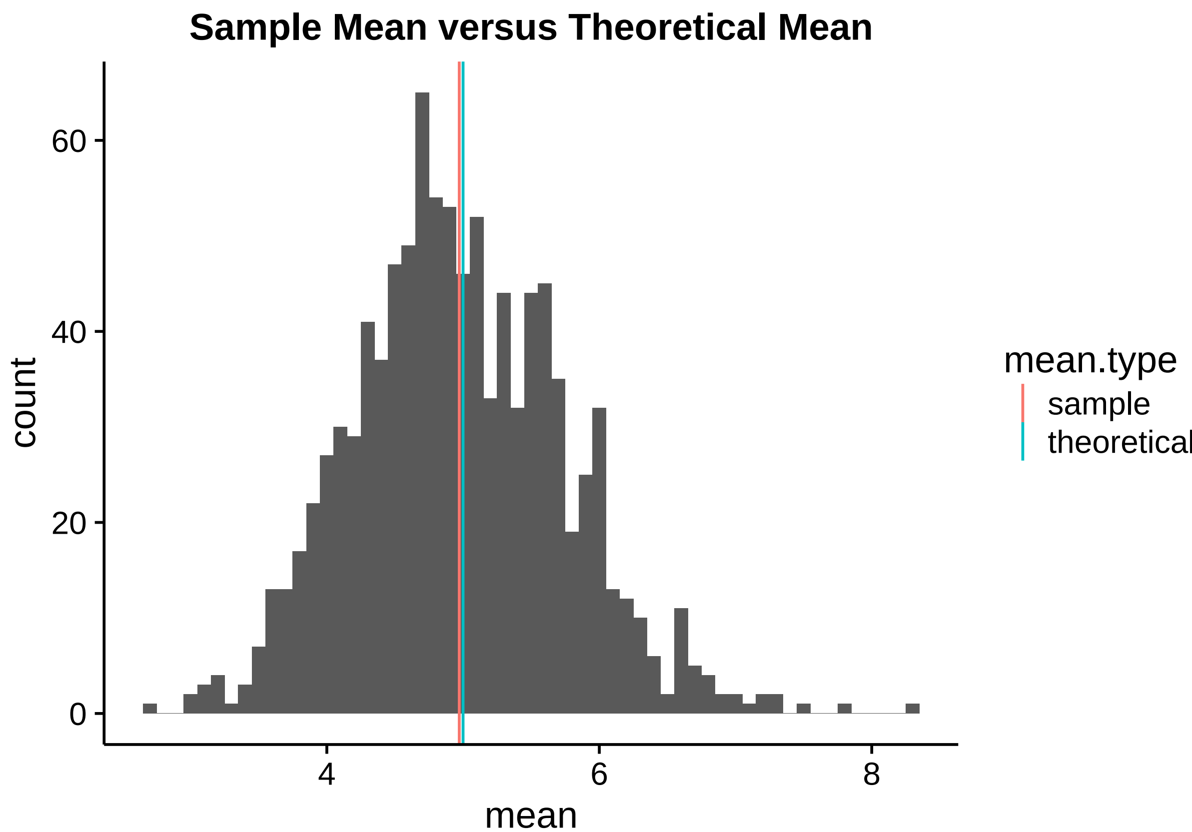
```r
SampleMeanVersusTheoreticalMean <- function() {
  sample.mean <- mean(exponential.distributions.means$mean)

  theoretical.mean <- 1 / kRate

  sample.mean.vs.theoretical.mean.table <- list(
    mean.type = as.factor(c('sample', 'theoretical')),
    mean = c(sample.mean, theoretical.mean)) %>%
    as.tibble()

  ggplot(exponential.distributions.means, aes(mean)) +
    ggtitle('Sample Mean versus Theoretical Mean') +
    geom_histogram(binwidth = .1) +
    geom_vline(
      data = sample.mean.vs.theoretical.mean.table,
      aes(xintercept = mean, color = mean.type))
}

SampleMeanVersusTheoreticalMean()
```



Sample Variance versus Theoretical Variance

The sample variance is also very close to the theoretical variance, as we can see in the table below.

```
SampleVarianceVersusTheoreticalVariance <- function() {
  sample.variance <- var(exponential.distributions.means$mean)

  sample.standard.deviation <- sd(exponential.distributions.means$mean)

  theoretical.variance <- 1 / (kRate ^ 2) / kNumberOfExponentialsPerSimulation

  theoretical.standard.deviation <- sqrt(theoretical.variance)

  sample.variance.vs.theoretical.variance.table <- list(
    mean.type = as.factor(c('sample', 'theoretical')),
    variance = c(sample.variance, theoretical.variance),
    standard.deviation = c(sample.standard.deviation,
                           theoretical.standard.deviation)) %>%
    as.tibble()

  t1 <- tableGrob(sample.variance.vs.theoretical.variance.table)
  t1.title <- textGrob('Sample Variance versus Theoretical Variance')
  padding <- unit(5,"mm")
  t <- gtable_add_rows(t1, heights = grobHeight(t1.title) + padding, pos = 0)
  t <- gtable_add_grob(t, t1.title,  1, 1, 1, ncol(t))
```

```
  grid.newpage()
  grid.draw(t)
}

SampleVarianceVersusTheoreticalVariance()
```

Sample Variance versus Theoretical Variance

	mean.type	variance	standard.deviation
1	sample	0.5954369	0.7716456
2	theoretical	0.6250000	0.7905694

Distribution

Here we show that the sample distribution is approximately normal by plotting a histogram of the density of sample's values and then overalying both the density line and a normal distribution built from the sample's mean and standard deviation.

```
SampleDistributionVersusNormalDistribution <- function() {
  sample.mean <- mean(exponential.distributions.means$mean)

  sample.standard.deviation <- sd(exponential.distributions.means$mean)

  ggplot(exponential.distributions.means, aes(x = mean)) +
    ggtitle('Sample Distribution Versus Normal Distribution') +
    geom_histogram(aes(y = ..density.., fill = ..count..), binwidth = .1) +
    geom_density(color = '#FF4136', size = 1) +
    stat_function(fun = dnorm, args = list(
      mean = sample.mean, sd = sample.standard.deviation),
      color = '#FF851B',
      size = 1)
```

```
}

SampleDistributionVersusNormalDistribution()
```