StatInf_prog1

jjperez

Sunday, Septembre 27, 2015

Title: Simulation Exercise

0. Sinopsis

The objetive of this project is to show how the exponential distribution in R can be adjusted to a normal distribution according to the Central Limit Theorem.

To do that I will simulate 40 exponential distributions with a fix parameter lambda for the exponential distribution. All the simulated values will be storaged on a matrix in order to be used to calculate the **sample mean** and **sample standard deviation** for each simulation.

After that I will calculate the theorical values for the mean and the standard deviation. The mean of exponential distribution is 1/lambda and the standard deviation is also 1/lambda.

1. Before the simulation

Prior to generate the simulated data we should load some libraries.

```
library(lubridate, quietly = TRUE)
library(RColorBrewer, quietly = TRUE)
library(lattice, quietly = TRUE)
library(xtable, quietly = TRUE)
library(knitr, quietly = TRUE)
library(dplyr, quietly = TRUE)
```

```
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:lubridate':
##
##     intersect, setdiff, union
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

The data is generated using the rexp(n, lambda) command and it is storaged on a Matrix. To generate the data I will define default values for n and lambda. Once I have all the values storaged on a matrix I proceed to calculate the mean and standard deviation for each row. These values are going to be storaged on a vector which will be use later to print the histograms.

Find below a list of the main variables I have used.

- **sim**. Simulated data.
- **mns**. Vector of means for the simulated values.
- **sds**. Vector of standard deviations for the simulated values
- **theo_mean**. Theorical value for the population mean.
- **theo_sd**. Theorical value for the population standard deviation.
- **sample_mean**. Calculated value. It is the mean of the values in the mns vector. Is the estimator for the population mean.
- **sample_sd**. Calculated value. It is the standard deviation of the values in the sdvs vector. Is the estimator for the population standard deviation.

1. Data Simulation

First of all we initialice all the variables.

```
lambda <- 0.2
m <- 1000
n <- 40

sim <- NULL
mns <- NULL
sds <- NULL
theo_mean <- NULL
theo_sd <- NULL
sample_mean <- NULL
sample_sd <- NULL
```

Next step is generate the data:

```
# Generated data
for (i in 1 : m) sim = rbind(sim, rexp(n,lambda))
```

2. Analysis

2.1. Theorical values

According to the theory the mean and the standard devation for the exponential distribution is equal 1/lambda.

```
# Theorical values

theo_mean <- 1/lambda
theo_sd <- 1/lambda
```

So, according to that, the population mean should be **5**, with a standard deviation of **5**.

2.2. Calculated values

Next step is to calculate the expected values for the and the standard deviation from our data. We have a total of **40** different samples, and each of them has **1000** observations. Samples are storaged on the sim matrix in rows. The following code calculates the mean vector and the standard deviations vector from these rows.

```
#First we calculate the mean distribution

for (i in 1 : m) mns = c(mns,mean(sim[i,]))

# Now we calculate the standard deviations

for (i in 1 : m) sds = c(sds,sd(sim[i,]))

# Now calculate the estimated population values.

sample_mean <- mean(mns)
sample_sd <- mean(sds)
```

The estimated values for the population mean and the population standard deviation could be calculated from these vectors. The calculated value for the population mean, as the mean of all the sample mean, is **4.9891873**. And the calculated value for the population standard deviation, as the **mean of all the sample standard deviations**, is **4.9019794**.

2.3. Histograms and normal curves

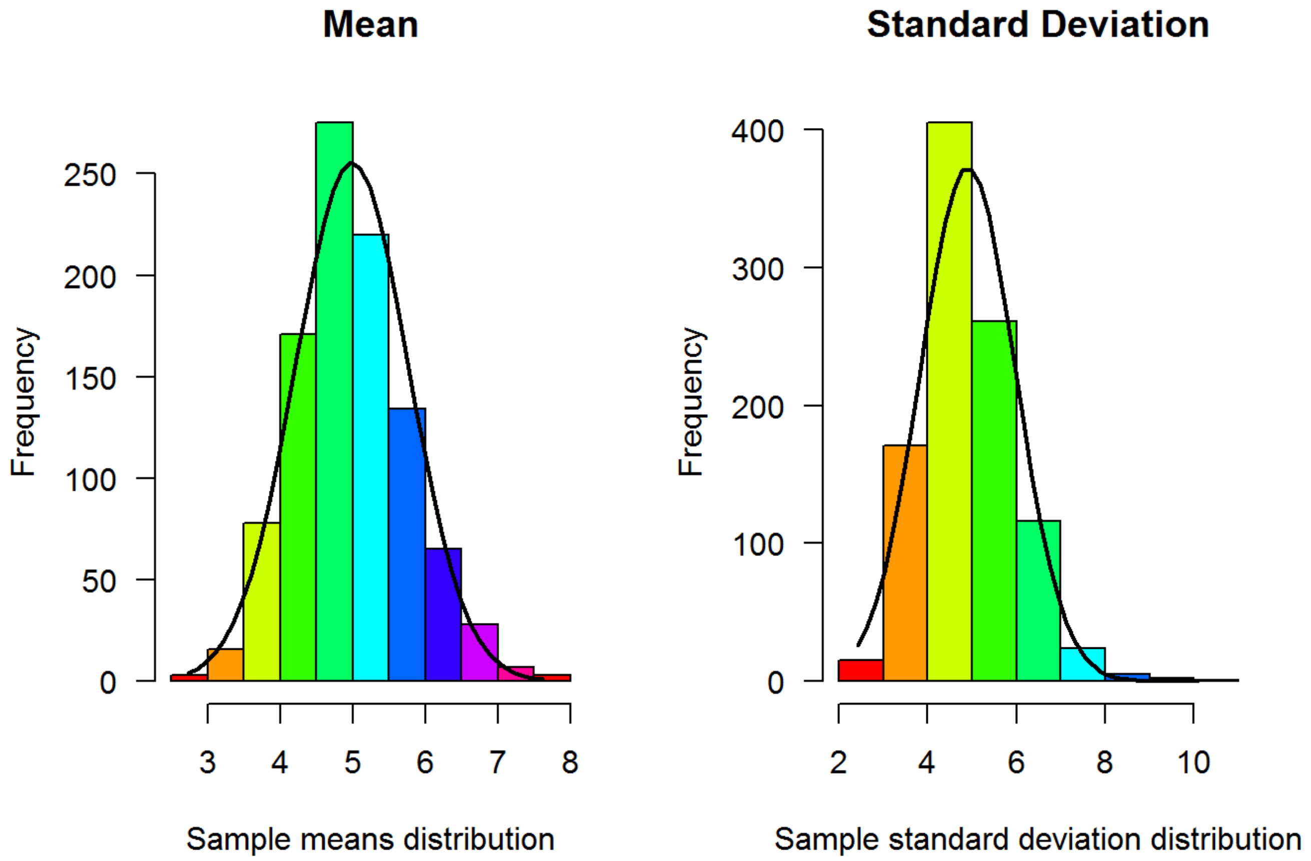
Once we have these vectors we can represent them on histograms. On the figure below you can see how the sample mean and the sample standard deviation distribution looks like. You will also see the normal distribution for each set of values on black.

```
# Now we plot the histograms on a combined panel.

par(mfrow = c(1,2))

h1<- hist(mns, breaks = 10, xlab = "Sample means distribution", main = "Mean", col
=rainbow(10), las=1)
xfit<-seq(min(mns),max(mns),length=40)
yfit<-dnorm(xfit,mean=mean(mns),sd=sd(mns))
yfit <- yfit*diff(h1$mids[1:2])*length(mns)
lines(xfit, yfit, col="black", lwd=2)

h2<- hist(sds, breaks = 10, xlab = "Sample standard deviation distribution", main
= "Standard Deviation", col=rainbow(10), las=1)
xfit<-seq(min(sds),max(sds),length=40)
yfit<-dnorm(xfit,mean=mean(sds),sd=sd(sds))
yfit <- yfit*diff(h2$mids[1:2])*length(sds)
lines(xfit, yfit, col="black", lwd=2)
```



As we can see, the equivalent normal distribution is centered close to the value expected for the population value.

3. Results

After this simulation we can easily see that the exponential distribution converge to a normal, as the Central Limit Theorem said. The average value for the sample mean, **4.9891873**, estimates really well the population mean, **5**. Also the mean of the standard deviations from the samples, **4.9019794**, converge to estimated the population standard deviation, **5**.

4. Extras

You will find the R script in my GitHub repostory. You can go there through that (link)[https://github.com/jjperez78/StatisticalInference.git (https://github.com/jjperez78/StatisticalInference.git)]