Simulation - Course Project - Statistical Inference

attila.toth86

23 Sep 2015

This is the project for the statistical inference class. In this, I use simulation to explore inference and do some simple inferential data analysis. The project consists of two parts:

1. A simulation exercise.
2. Basic inferential data analysis.

This document aims to fulfill requirements of first task.

Simulation

In this project I will investigate the exponential distribution in R and compare it with the Central Limit Theorem.

- *Mean* of exponential distribution is *1/lambda*
- Its *standard deviation* is also *1/lambda*

The exponential distribution can be simulated in R with `rexp(n, lambda)` function, where `n` represents the number of observations and `lambda` is the so-called rate parameter. For further analyses, I create a major population of 1 million observation with exponential distribution.

For all simulation I use `lambda = 0.2`.

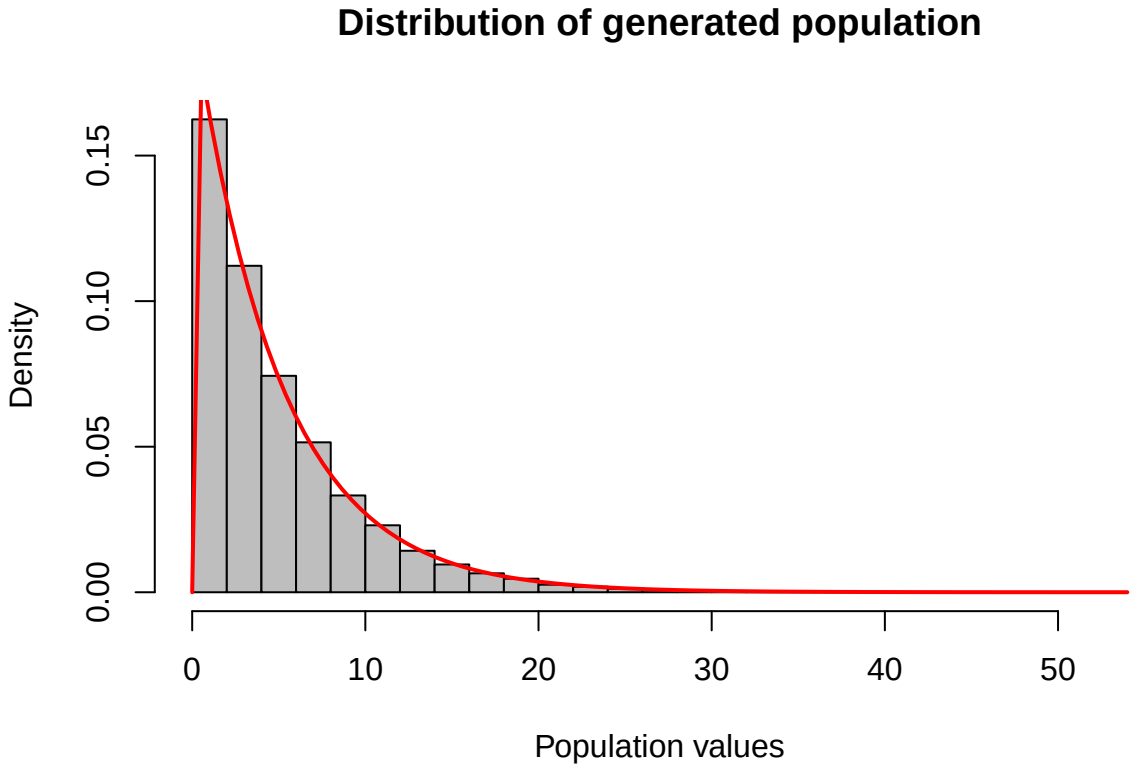
```
lambda <- 0.2
```

In order to ensure reproduciblilty, I set a seed.

```
set.seed(10000)
```

Then I create a random exponential population:

```
pop <- rexp(n = 10000, rate = lambda)
hist(pop,
     main="Distribution of generated population",
     freq=FALSE,
     xlab='Population values',
     breaks=30,
     col=8)
curve(dexp(x,lambda), add=TRUE, col=2, lwd=2)
```



Considering lambda equals to 0.2, literature states that

- **Theoretical mean of such distribution is 5,**
- **and theoretical standard deviation is also: 5** (In addition to that: **theoretical variance is then: 25**)

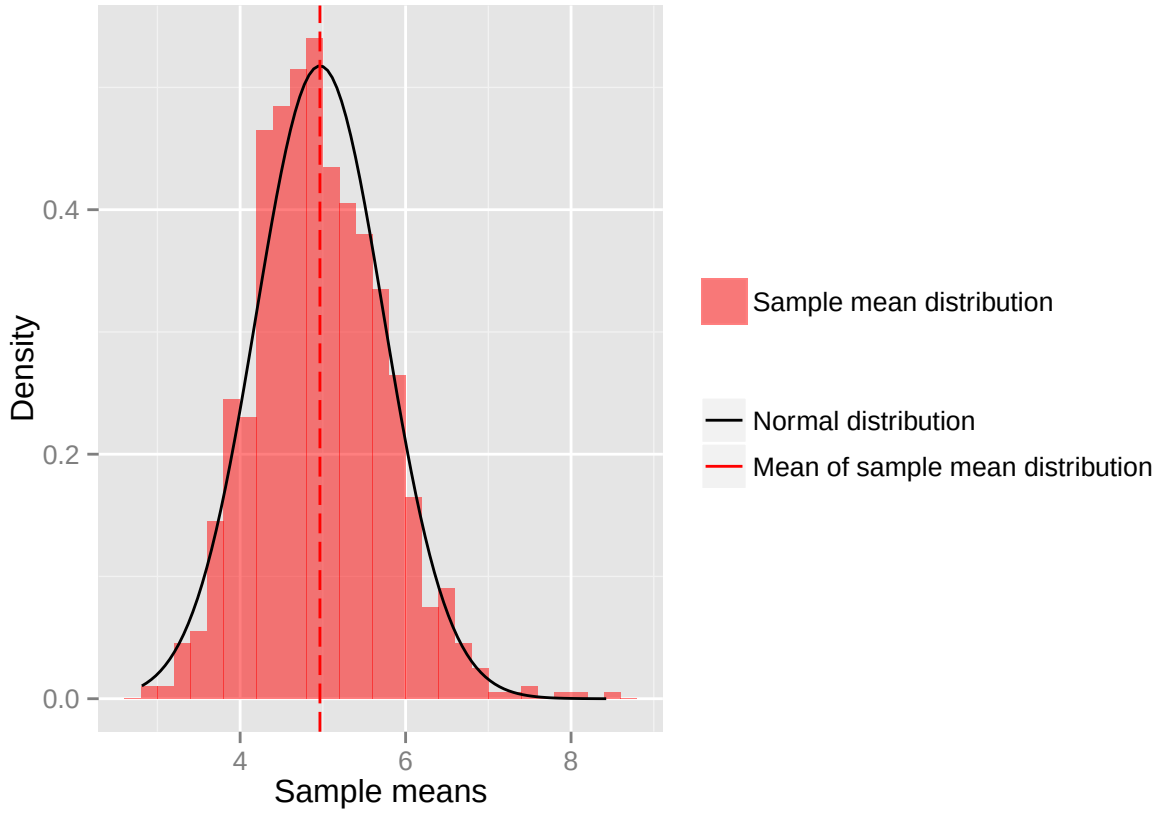
I investigate the distribution of averages of 40 exponentials by generating 1000 simulation. The code snippet below generates samples of 40 exponentials 1000 times, stores the samples into a 1000x40 data frame called `sample_matrix`. Then for each row (samples) I calculate the mean and store it into the 41th column.

```
no_of_exp <- 40
no_of_iteration <- 1000
sample_matrix <- data.frame(
                            matrix(
                                sample(pop,
                                       no_of_exp*no_of_iteration,
                                       replace = TRUE),
                                nrow = no_of_iteration
                                )
                            )
sample_matrix <- cbind(sample_matrix, means=rowMeans(sample_matrix))
```

The distribution of averages of the observed exponentials looks like this:

```
library(ggplot2)
ggplot() +
geom_histogram(data = sample_matrix,
               aes(x = means,
```

```
                    y = ..density..,
                    fill= "r"),
                 binwidth = 0.2,
                 alpha = 0.5) +
stat_function(data = sample_matrix,
              fun = dnorm,
              args = list(mean = mean(sample_matrix$means),
                          sd = sd(sample_matrix$means)),
              aes(colour = "b")
              ) +
geom_vline(aes(xintercept=mean(sample_matrix$means),
               colour = "r"),
           linetype="longdash") +
ylab("Density") +
xlab("Sample means") +
scale_colour_manual(name="",
                    values=c("r" = "red",
                             "b"="black"),
                    labels=c("b"="Normal distribution",
                             "r"="Mean of sample mean distribution")) +
scale_fill_manual(name="",
                  values=c("r" = "red",
                           "b"="blue"),
                  labels=c("b"="blue values",
                           "r"="Sample mean distribution"))
```



Inspecting the shape of the sample mean distribution, we can state it looks almost normally distributed (black curve vs histogram) so the new distribution I assume to be approximately normal.

Comparing this new distribution to the generated & theoretical population:

Population	Mean	Standard Deviation	Variance	Standard Error
Theoretical	**5**	5	25	
Simulated population	4.98	4.95	24.49	
Average of 40 randomly selected set	**4.96**	**0.77**	0.59	**0.79**

So, the sample mean estimates the original population's mean quite well. As it was expected. Since the Law of Large Numbers states that averages of iid samples converge to the population means that they are estimating. Also, Central Limit Theorem states that averages are approximately normal and their distribution is centered at the population mean with standard deviation equal to the standard error of the mean (meaning: much narrower variability), that is in our case: `Population Standard Deviation / Square Root of Sample Size` = `(1/lambda)/(sqrt(40))` = 0.7905694. This is what I expected as standard deviation from the new distribution, and is quite close to the value I actually calculated from simulated distribution (0.77).