Statistical Inference Course

first peer graded project

Albert Fradera Sola

2020-04-27

Contents

Simulation exercise

Overview

In this project you will investigate the exponential distribution in R and compare it with the Central Limit Theorem. The exponential distribution can be simulated in R with rexp(n, lambda) where lambda is the rate parameter. The mean of exponential distribution is 1/lambda and the standard deviation is also 1/lambda. Set lambda = 0.2 for all of the simulations. You will investigate the distribution of averages of 40 exponentials. Note that you will need to do a thousand simulations.

Illustrate via simulation and associated explanatory text the properties of the distribution of the mean of 40 exponentials. You should:

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

In point 3, focus on the difference between the distribution of a large collection of random exponentials and the distribution of a large collection of averages of 40 exponentials.

0: Run the simulations

First step we store our fixed variables on R objects:

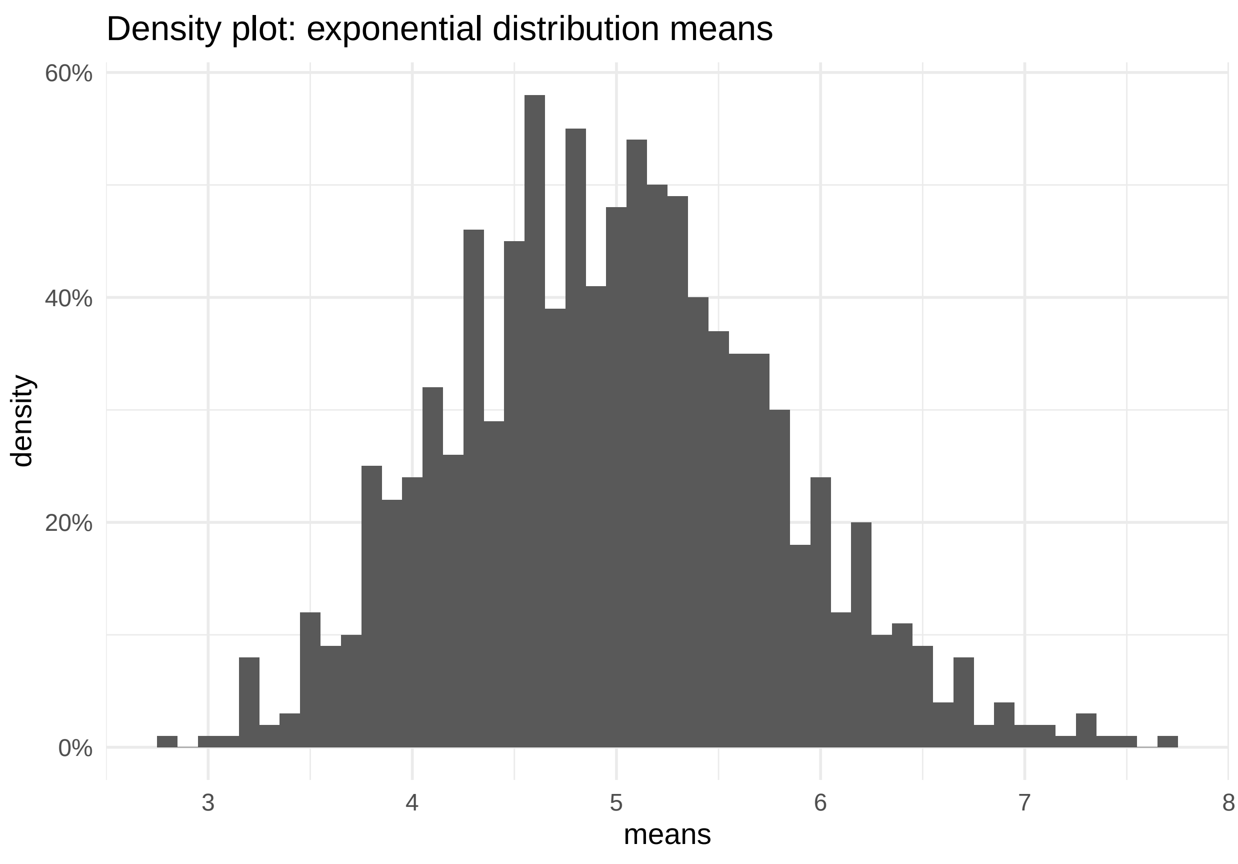
```
set.seed(666)  # seed to create reproducibility
lambda <- 0.2  # lambda parameter on the exponential distrubution
n_exp  <- 40   # Number of exponential distributions
n_sim  <- 1000 # Number of simulations
```

Once we have the variables needed for our exponetial distribution, we run the 1000 simulations, calculate the mean, and store both on a matrix:

```
# Compute 1000 simulations and store them on a matrix
exponentialDistributions <- matrix(data=rexp(n_exp * n_sim, lambda), nrow=n_sim)
# Compute the mean of each row (a.ka. each simulation mean)
exponentialDistributionMeans <- data.frame(means=apply(exponentialDistributions, 1, mean))
```

We can observe the mean distribution on a density plot:

```
# Density plot
plot <- ggplot(data = exponentialDistributionMeans, mapping = aes(x = means))+
        geom_histogram(binwidth = 0.1,
                 aes(y = ..density..))+
        ggtitle("Density plot: exponential distribution means")+
        scale_y_continuous(labels = percent_format()) +
        theme_minimal()
print(plot)
```



1: Exploring the mean

We want to compare the the theoretical mean of our distribution with the one we observe after a 1000 simulations. The mean is defined by:

$$\mu = \frac{1}{\lambda}$$

Thus, we can compute the theoritical mean using our fixed variables and compare it to the one obtained from the simulations:

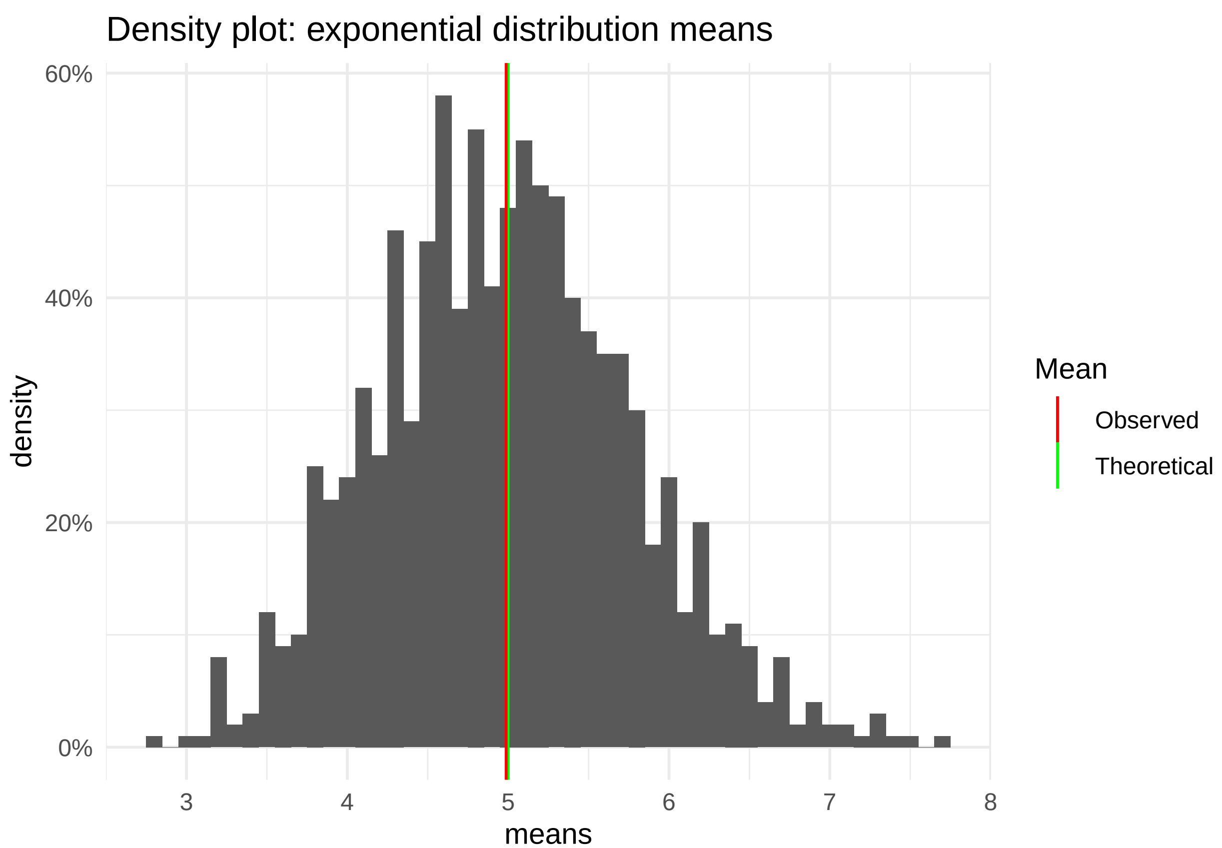
```
t_mean <- 1/lambda                                   #Theoretical mean
t_mean
```

```
## [1] 5
```

```
o_mean <- mean(exponentialDistributionMeans$means) #Observed mean
o_mean
```

```
## [1] 4.987818
```

```
# Plot both means on the density plot:
plot_means <- plot +
              geom_vline(aes(xintercept = t_mean,
                             color = "Theoretical"))+
              geom_vline(aes(xintercept = o_mean,
                             color = "Observed"))+
              scale_color_manual(name = "Mean",
                                 values = c("Theoretical" = "green",
                                            "Observed" = "red"))
print(plot_means)
```



We can see that the observerd mean is very cose to the theoretical mean

2: Exploring the variance

We want to compare the the theoretical variance of our distribution with the one we observe after a 1000 simulations. The variance is defined by:

$$Var = \sigma^2$$

Thus we need σ, which is defined by:

$$\sigma = \frac{\mu}{\sqrt{n}}$$

Since μ is equal to $\frac{1}{\lambda}$ we can comput σ with the following expression:

$$\sigma = \frac{\frac{1}{\lambda}}{\sqrt{n}}$$

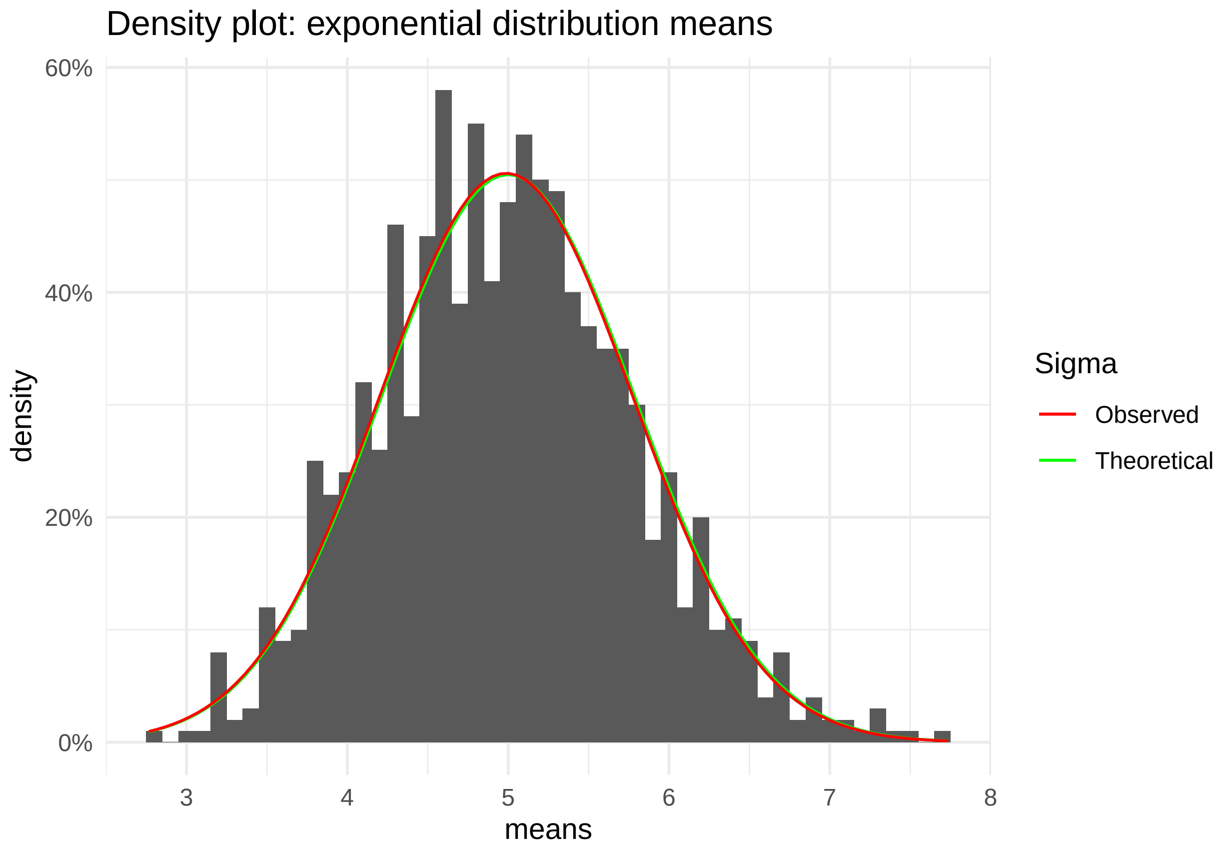
```
t_var <- (t_mean/sqrt(n_exp))^2                    #Theoretical variance
t_var
```

```
## [1] 0.625
```

```
o_var <- var(exponentialDistributionMeans$means) #Observed variance
o_var
```

```
## [1] 0.6221844
```

```
#We plot the shape of a normal distribution with theoretical and observed sigma
plot_variance <- plot +
                    stat_function(mapping = aes(color = "Theoretical"),
                                  fun = dnorm,
                                  args = list(mean = t_mean,
                                              sd = t_mean/sqrt(n_exp)))+
                    stat_function(mapping = aes(color = "Observed"),
                                  fun = dnorm,
                                  args = list(mean = o_mean,
                                              sd = o_mean/sqrt(n_exp)))+
                    scale_color_manual(name = "Sigma",
                                       values = c("Theoretical" = "green",
                                                  "Observed" = "red"))
print(plot_variance)
```

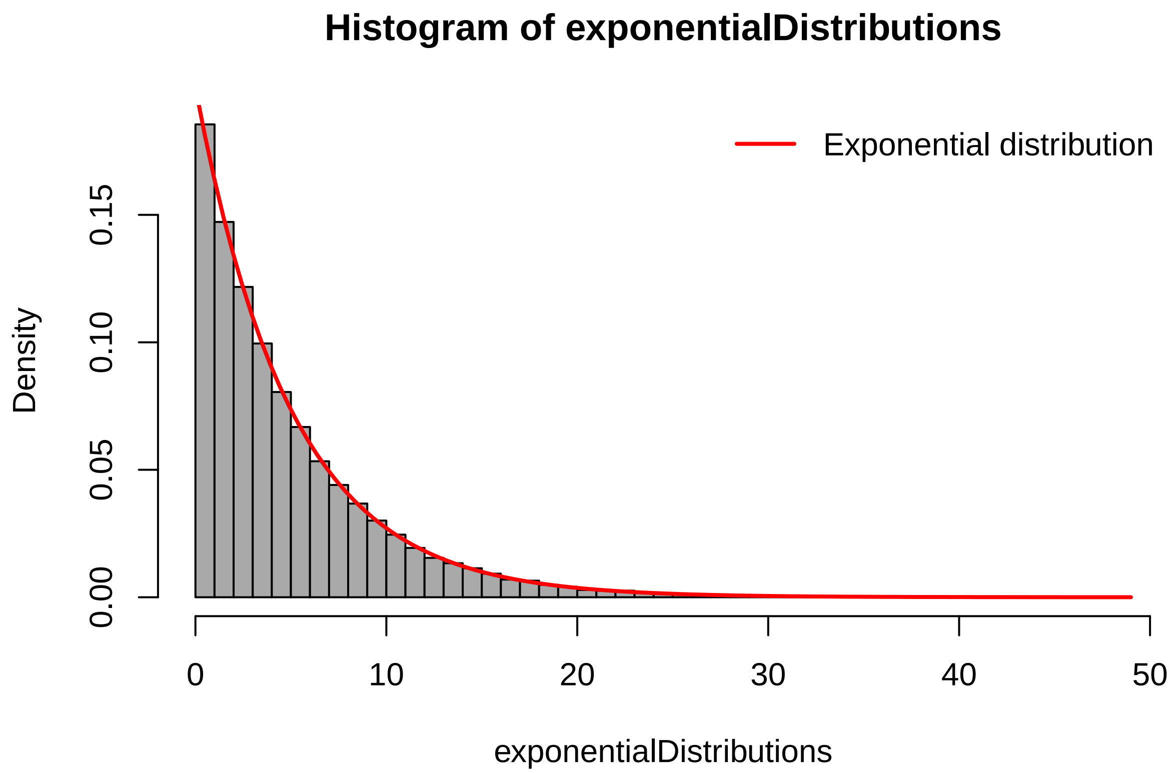


Again we can see that the expected and observed variance are very close and the shape of the normal distribution has minimal changes

3: Exploring the distribution

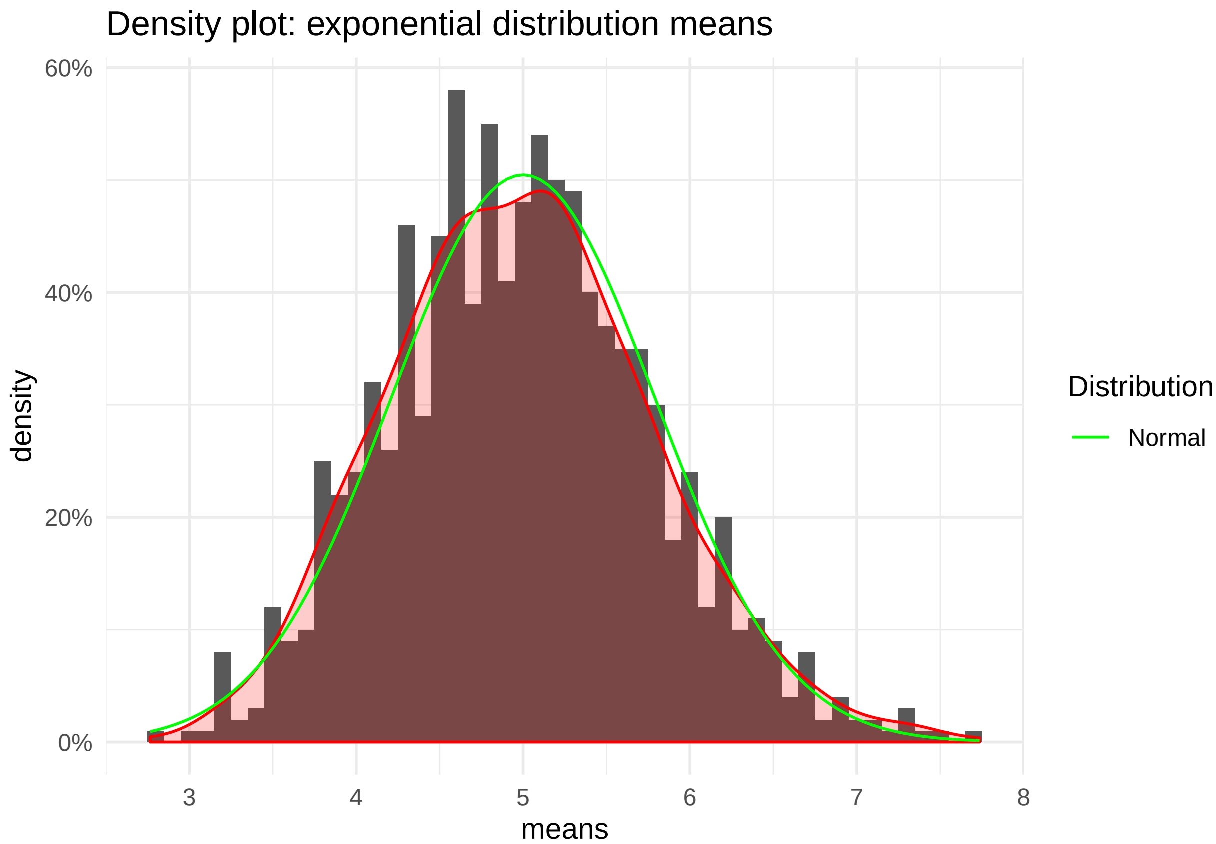
First we explore the distribution of the exponetials:

```
hist(exponentialDistributions,
     col="darkgray",
     breaks = 50.,
     freq = FALSE)
curve(dexp(x, rate=lambda, log=FALSE),col = 2, lwd = 2, add = TRUE)
legend("topright", c("Exponential distribution"), col = 2, lwd = 2,  bty='n')
```



We can see that they match an exponential distrubiton. On the other hand, we can plot the distrubtion of the averages:

```
plot_distribution <- plot +
                     geom_density(color = "red",
                                  alpha=.2,
                                  fill="red")+
                     stat_function(mapping = aes(color = "Normal"),
                                  fun = dnorm,
                                  args = list(mean = t_mean,
                                              sd = t_mean/sqrt(n_exp)))+
                     scale_color_manual(name = "Distribution",
                                        values = c("Normal" = "green"))
print(plot_distribution)
```



We can observe how our sample distribution (in red) is matching a function describing a normal ditribution with our therotical μ and our theoretical σ (in green)

Session info:

```
print(sessionInfo(), locale = F)
```

```
## R version 3.5.1 (2018-07-02)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 18363)
##
## Matrix products: default
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] scales_1.1.0  ggplot2_3.2.1
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.3       knitr_1.27       magrittr_1.5     tidyselect_0.2.5
##  [5] munsell_0.5.0    colorspace_1.4-1 R6_2.4.1         rlang_0.4.3
##  [9] stringr_1.4.0    dplyr_0.8.3      tools_3.5.1      grid_3.5.1
## [13] gtable_0.3.0     xfun_0.12        withr_2.1.2      htmltools_0.4.0
## [17] assertthat_0.2.1 yaml_2.2.0       lazyeval_0.2.2   digest_0.6.23
```

```
## [21] tibble_2.1.3     lifecycle_0.1.0  crayon_1.3.4     farver_2.0.3
## [25] purrr_0.3.3      glue_1.3.1       evaluate_0.14    rmarkdown_2.1
## [29] labeling_0.3     stringi_1.4.5    compiler_3.5.1   pillar_1.4.3
## [33] pkgconfig_2.0.3
```