Statiscal Inference Final Project

Maria Cristina Alameda Salas

25/4/2023

Part 1: Simulation Exercise

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:

1. Show the 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.

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.

Simulating

First, we will generate our data. The following code starts defying some constants used to simulate our 1000 simulations with 40 exponentials for each of them. After that, our simulations are generated and it is calculated the mean of each simulation.

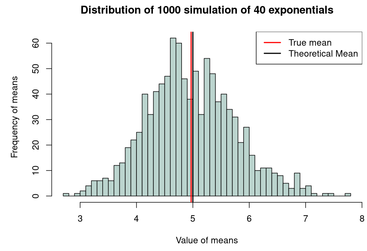
```
lambda <- 0.2
mean <- 1/lambda
std <- 1/lambda
n_simulations <- 1000
n_exponentials <- 40

set.seed(50)

# Generate 1000 simulations of 40 exponentials
distSims <- matrix(rexp(n_exponentials*n_simulations, lambda), nrow=n_simulations, ncol=n_exponentials)

# Calculate the mean for each simulation (row of the matrix)
# MARGIN=1 indicate that apply calculates the mean through each row
distMeans <- apply(X=distSims, MARGIN=1, FUN=mean)

# Calculate the std for each simulation (row of the matrix)
# MARGIN=1 indicate that apply calculates the std through each row
distVars <- apply(X=distSims, MARGIN=1, FUN=var)
```

Comparing Sample Mean and Theorical Mean

The central limit theorem (CLT) states that the distribution of sample means approximates a normal distribution as the sample size gets larger, regardless of the population's distribution. One of its properties is the the true mean is equal to the inverse of lambda. With our rate=lambda=0.2 paramenter, our true mean should be 5.

To `show prove this, we will plot an histogram showing the distribution of our 1000 means of 40 exponentials and the theorical mean = 5.

```
hist(x = distMeans,
     main = "Distribution of 1000 simulation of 40 exponentials",
     freq = TRUE,
     breaks = 50,
     xlab = "Value of means",
     ylab = "Frequency of means",
     col = "#bbd4ce",
     border = NULL)

abline(v = 1/lambda, lty = 1, lwd = 2, col = "black")

abline(v = mean(distMeans), lty = 1, lwd = 2, col = "red")

legend("topright", lty = 1, lwd =2, col=c("red", "black"), legend=c("True mean", "Theoretical Mean"))
```



As we can see, our two means are very similar, around 5.

Comparing Variance and Theorical Variance

```
trueVariance <- mean(distVars)
theoricalVariance <- (1/lambda)^2

sprintf("True Variance %f", trueVariance)
```

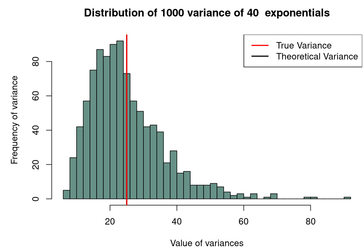
```
## [1] "True Variance 24.986592"
```

```
sprintf("Theorical Variance %f", theoricalVariance)
```

```
## [1] "Theorical Variance 25.000000"
```

The previous code calculate the mean of variance of each row (simulation) of exponents. This value is similar to the theorical variance as CLT says.

```
hist(distVars, breaks = 50, main = "Distribution of 1000 variance of 40  exponentials", xlab = "Value of variance
s", ylab = "Frequency of variance", col = "#679186")

abline(v = theoricalVariance, lty = 1, lwd = 2, col = "black")

abline(v = trueVariance, lty = 1, lwd = 2, col = "red")

legend("topright", lty = 1, lwd =2, col=c("red", "black"), legend=c("True Variance", "Theoretical Variance"))
```



Normal distribution of averages

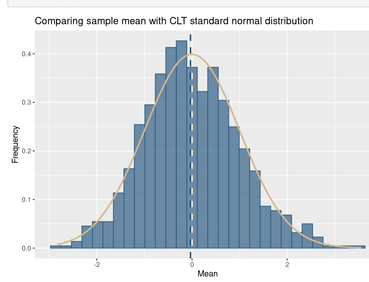
For this task, we need to convert our distribution of average to a 0 mean distribution and compare it to standard normal distribution.

```
standardError <- mean(distMeans)/sqrt(n_exponentials)

# normalization function
standNormalFunc <- function(b){
        round((b - mean(distMeans))/standardError,2)
}

#normalized data of averages
distNormalMeans <- unlist(lapply(distMeans, FUN=standNormalFunc))


library(tibble)
df <- enframe(distNormalMeans)
```

```
# load "tidyverse" plotting package
library(ggplot2)


histogram <- ggplot() +
        geom_histogram(aes(x = df$value, y = ..density..),
                       bins = 30,
                       color = "#315F85",
                       fill="#315F85",
                       alpha = 0.7
        ) +
        labs(
                title = 'Comparing sample mean with CLT standard normal distribution',
                x = 'Mean',
                y = 'Frequency'
        ) +
        geom_vline(
                xintercept = mean(distMeans) - (1/lambda),
                color = "#264e70",
                size = 1,
                linetype="dashed"
        ) +
        geom_vline(
                xintercept = 0,
                color = "#fdebd3",
                size = 1,
                linetype="dashed"
        ) +
        stat_function(
                aes(x = df$value),
                size = 1,
                fun = dnorm,
                color = "#D6B78B",
                args = list(
                        mean = 0,
                        sd = 1
                )
        )
```

```
## Warning: `mapping` is not used by stat_function()
```

```
plot(histogram)
```



The two previous graphs looks quite similar. Consequently, it can be said that our data (blue bars) follows a normal distribution (yellow line).