Statistical Inference Assignment 1

DC

29/05/2019

```r
library(ggplot2)
library(gridExtra)
```

Overview

This report will be the fist of 2 assignments in the statistical inference class.

The exact given criteria is as follows:

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.*

Simulations

In order to complete this assignment, we're going to need to set some parameter for our data generation. The assignment instructions make these crystal clear. To summarise, we will use the following criteria for our simulation:

Lambda = **0.2**
Exponentials = **40**
Number of Simulations = **1000**
Simulated Distribution = **Exponential**

Let's set the above criteria into variables.

```r
# Set seed for reproducability
set.seed(123)

# number of simulations
trials <- 1000

# lambda
lambda <- 0.2

# exponentials
n <- 40
```

Now that we have some parameter, let's take a look at what the theoretical distribution should look like.

Mean $= \mu = \frac{1}{\lambda}$
Standard Deviation $= \sigma = \frac{1}{\lambda}$
Sample Standard Deviation $= \sqrt{\frac{\sigma^2}{n}}$
Sample Variance $= V = \frac{\sigma^2}{n}$

therefore:
theoretical sample mean $= \frac{1}{0.2} = 5$
theoretical standard deviation $= \frac{1}{0.2} = 5$
theoretical sample standard deviation $= \sqrt{\frac{25}{40}} = 0.7905$
theoretical sample variance $= \frac{25}{40} = 6.25$

Let's assign these theoretical results to variables for future use:

```r
theoretical_mean <- 1 / lambda
theoretical_standard_dev <- sqrt((1/lambda) ^ 2 / n)
theoretical_variance <- 0.625
```

With the above all set, it's time to generate some data.
For this, we will use R's *rexp* function with our given criteria.
The following code will perform **1000** loops, and with each, it will generate the **mean** of **40** randomly generated exponentials from seed **123**, and then add them to a list of results.
The output variable, *results*, will contain 1000 mean results.

```r
# initiate results variable
results <- vector()

# perform 1000 mean iterations and collect results
for(i in 1:trials){

    results[i] <- mean(rexp(n, lambda))

}
```

Sample Mean & Variance vs Theoretical

Now that we have some data, let's compute it's mean, standard deviation and variance then assign them to variables.
Once we have done this, we will structure a table for comparison to their theoretical equivalents.

```r
sample_mean <- mean(results)
sample_standard_dev <- sd(results)
sample_variance <- var(results)
```

Statistic	Sample	Theoretical
Mean	5.0119113	5
Standard Deviation	0.7749147	0.7905694
Variance	0.6004928	0.625

from the above table, we have completed tasks **1** and **2** from the assignment.
We can see rather clearly that our sample mean and variances are within 5% of their theoretical counter parts. Additionally, this also re-enforces the *law of large numbers* which states that given enough samples, the sample mean will be inline with it's expected value.

Let’s move on to task 3.

Distribution

The easiest, and arguably fastest way to convey distribution information is by using visuals, but first, let’s quickly go over what we will display, and what we should expect to see.

The most straight forward way to display our simulated density will be with a density plot.
Due to the *Central Limit Theorem* which states that given a large enough size, the mean of our idd variables will form a normal distribution, even if the iid variables are themselves not (which is the case in our exponential simulation)

To compare our plotted density against what we expect, we will also generate some data to form a normal distribution. If all goes as planned, we will see that the two distributions look extremely similar.

Below I will perform the steps needed to convert our gathered data into plot, and to generate a normal distribution using **1000** random normals.
I will also create an additional new dataframe, containing the normalised (t-score) values of our sample distribution using the formula $X = \frac{X - \overline{x}}{s}$
This will give us the ability to make an overlapping comparison to our expected normal distribution.

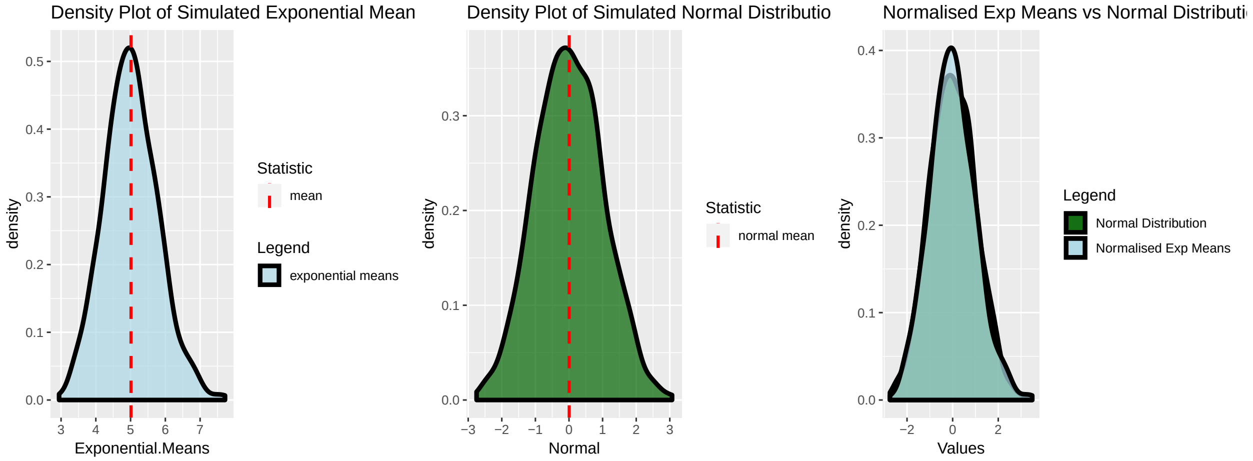
```r
# Convert results to dataframe
results <- data.frame(`Exponential Means` = results)

results_density_plot <- ggplot(results, aes(x =
                              `Exponential.Means`, fill = "exponential means")) +
  geom_density(lwd = 1.5, alpha = 0.7) +
  ggtitle("Density Plot of Simulated Exponential Means Distribution") +
  geom_vline(aes(xintercept = mean(results$`Exponential.Means`), colour = "mean"),
             lty = 2, lwd = 1) +
  scale_colour_manual(name = "Statistic", breaks = c("mean"), values = "red") +
  scale_fill_manual(name = "Legend", breaks = c("exponential means"),
                    values = "lightblue")

# Normal distibution
normal_dist <- data.frame("Normal" = rnorm(1000))

normal_density_plot <- ggplot(normal_dist, aes(x = Normal, fill = "normal")) +
  geom_density(lwd = 1.5, alpha = 0.7) +
  ggtitle("Density Plot of Simulated Normal Distribution") +
  geom_vline(aes(xintercept = mean(normal_dist$Normal), colour = "normal mean"),
             lty = 2, lwd = 1) +
  scale_colour_manual(name = "Statistic", breaks = c("normal mean"), values = "red") +
  scale_fill_manual(name = "Legend", breaks = c("results"), values = "darkgreen")

# Convert exponential means distribution to normal values
norm_results <- data.frame(
  "Normalised.Sample.Means" = (results$Exponential.Means - sample_mean) /
    sample_standard_dev)

comparison_density_plot <- ggplot() +
  geom_density(data = normal_dist, aes(x = Normal, fill = "Normal Distribution"),
               lwd = 1.5, alpha = 0.7) +
  geom_density(data = norm_results, aes(x = Normalised.Sample.Means,
                                        fill = "Normalised Exp Means"),
               lwd = 1.5, alpha = 0.7) +
  scale_fill_manual(name = "Legend", breaks = c("Normal Distribution",
                                                "Normalised Exp Means"),
                    values = c("darkgreen", "lightblue")) +
  xlab("Values") + ggtitle("Normalised Exp Means vs Normal Distribution")
```

The left side of the below plot is our generated distribution of 1000 exponential means.
We can see from this that the distribution has a normal shape and is centered around it's mean.

The middle figure is the density of our distribution of 1000 normals.
As a quick visual check, we can eyeball this and compare it's shape and center to the previous plot.

Finally, we arrive at our third plot. Which is the T-scores of our exponential means overlaid on top of the normal distribution. As we can see, this is a strong correlation.

```
grid.arrange(results_density_plot, normal_density_plot, comparison_density_plot, ncol = 3)
```

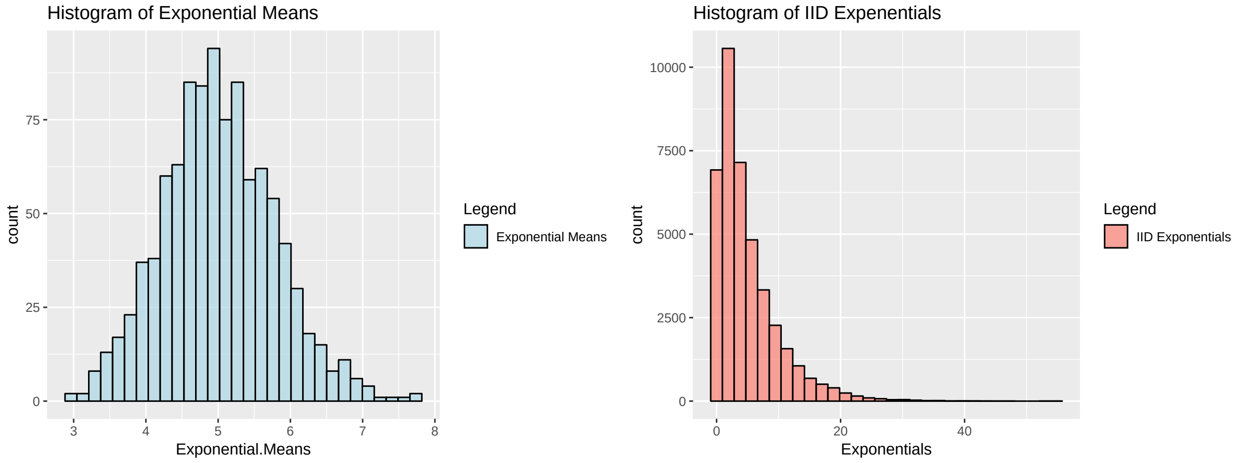


For the final step of this analysis, we will compare how our distribution of exponential means shapes up vs a large distribution of iid exponentials.
Again, let's simulate some data using the criteria for the original simulation and compare that to our exponential mean distribution.
This time we will set the number of iid's generated to 1000 * 40 so we're generating the same amount of individual exponentials.
To switch things up, let's also plot and compare our results using histograms.

```
iid_exponentials <- data.frame("Exponentials" = rexp(trials * n, lambda))
```

As we'd expect, the *Central Limit Theory* doesn't apply to the latest simulation as we're no longer working with sample averages. Instead, the non-mean IID distribution is strongly right-skewed as we'd have again expected.

```
mean_hist <- ggplot(data = results, aes(x = Exponential.Means)) +
  geom_histogram(aes(fill = "Exponential Means"), alpha = 0.7,
                 colour = "black", bins = 30) +
  scale_fill_manual(name = "Legend", breaks = c("Exponential Means"),
                    values = "lightblue") +
  ggtitle("Histogram of Exponential Means")

exp_hist <- ggplot(iid_exponentials, aes(x = Exponentials)) +
  geom_histogram(aes(fill = "IID Exponentials"), alpha = 0.7,
                 colour = "black", bins = 30) +
  scale_fill_manual(name = "Legend", breaks = c("IID Exponentials"),
                    values = "salmon") +
  ggtitle("Histogram of IID Expenentials")

grid.arrange(mean_hist, exp_hist, ncol = 2)
```


Histogram of Exponential Means
count
Exponential.Means
Legend
Exponential Means
Histogram of IID Expenentials
count
Exponentials
Legend
IID Exponentials