Statistical Inference Course Project: Simulation Exercise

Tushar Kataria

Contents

Overview

The first part of the course project focuses on a simulation exercise: investigating the exponential distribution in R and comparing it to the Central Limit Theorem (CLT). For this, the exponential distribution is modeled using the function `rexp(n, lambda)` where the two arguments are the number of observations, `n`, and the rate parameter, `lambda`. The exponential distribution describes the times between events happening at a constant rate λ with an expected value (mean) of $\mu = \frac{1}{\lambda}$ and variance of $\sigma^2 = \frac{1}{\lambda^2}$. Therefore, the distribution's standard deviation (σ) is equal to the mean.

This exercise investigates the distribution of averages of 40 exponentials, where the rate parameter is set to $\lambda = 0.2$, and a total of 1000 simulations are carried out. To make this comparison with the CLT, this report shows the sample mean and compares it to the theoretical mean of the distribution, shows how variable the sample is (through its variance) and also compares it with the theoretical variance of the distribution, and shows that the distribution is approximately normal.

Simulations

Sample Exponential Distribution

The data for the simulation is generated by drawing `ndist` independent and identically distributed (i.i.d.) samples from an exponential distribution and repeating this `nsim` times to create a `nsim \times ndist` matrix. In this case, `ndist = 40` and `nsim = 1000` for a matrix of dimensions 1000×40 such that each row represents an individual simulation of drawing `ndist` (40) random samples from an exponential distribution. Following this, the sampling mean of each row is calculated using the `apply()` function to generate a distribution of 1000 averages of 40 random exponentials.

```
set.seed(8888)

lambda <- 0.2 # rate parameter
ndist <- 40 # number of exponentials
nsim <- 1000 # number of simulations

sim_data <- matrix(rexp(nsim * ndist, rate = lambda), nsim, ndist)
sim_means <- apply(sim_data, 1, mean) # means of each row
```

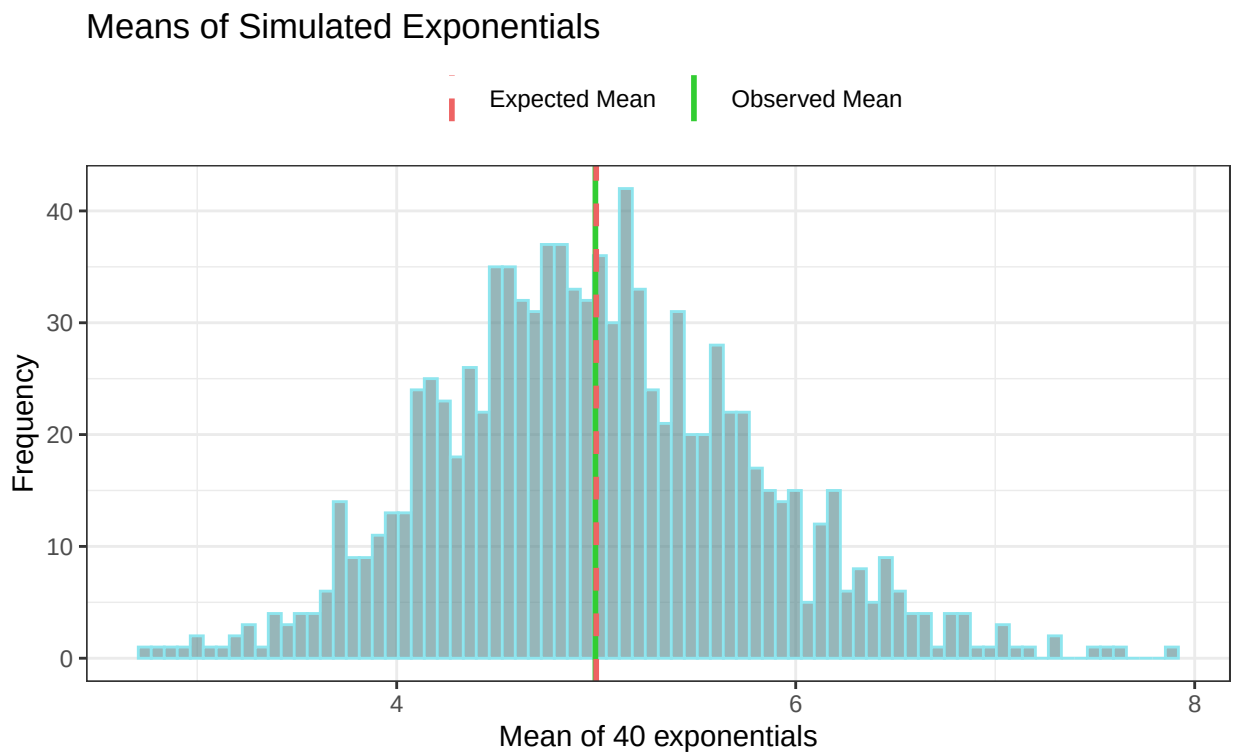
Now, the sample mean, standard deviation and variance of this simulated data can be calculated.

```
sample_mean <- mean(sim_means)
sample_sd <- sd(sim_means)
sample_var <- var(sim_means)
```

Theoretical Exponential Distribution

With the rate parameter and number of distributions known, the theoretical mean, standard deviation and variance are also calculated.

```
theory_mean <- 1 / lambda
theory_sd <- (1 / lambda) * (1 / sqrt(ndist))
theory_var <- theory_sd ^ 2
```



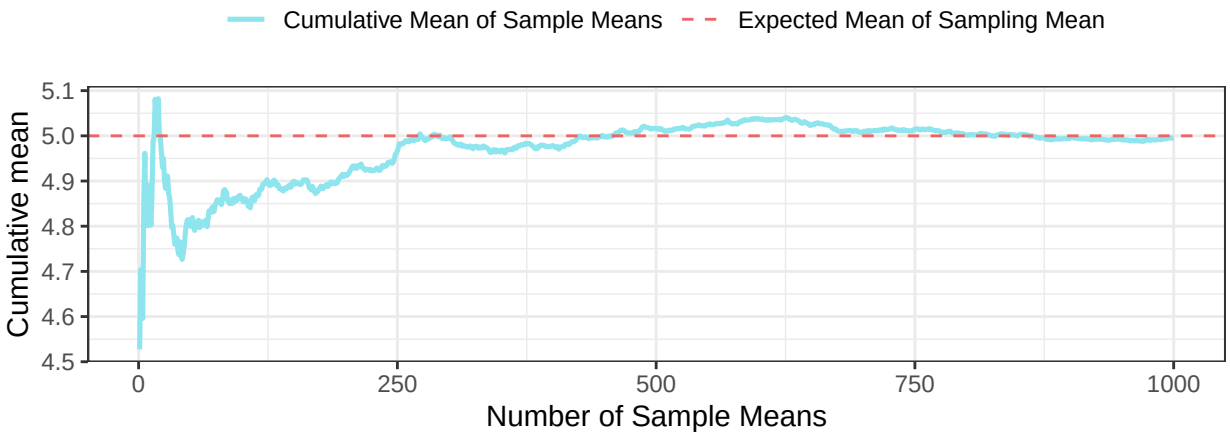
Above, the histogram shows the distribution of 1000 averages of 40 random exponentials; this follows an approximately normal distribution (which is investigated further in this report). Additionally, the expected (theoretical) and observed (sample) means are shown as an initial comparison between the two.

Comparison betwen Sample and Theoretical Statistics

```
##                    Sample_stats Theoretical_stats  Difference
## Mean                  4.9959232         5.0000000 0.004076841
## Standard deviation    0.7862758         0.7905694 0.004293618
## Variance              0.6182296         0.6250000 0.006770371
```

Sample Mean versus Theoretical Mean

The sample mean is a random variable that should be centred around the theoretical mean of $\mu = \frac{1}{\lambda}$. From the table above, the sample mean of 5.035 is very close to the theoretical mean of $\mu = \frac{1}{0.2} = 5$ (a difference of 0.0041) and, according to the Law of Large Numbers (LLN), this sample mean tends to the theoretical mean as the number of samples increases, converging to the theoretical value as the sample size approaches infinity. This is evidenced through the cumulative mean as a function of the number of sample means below.

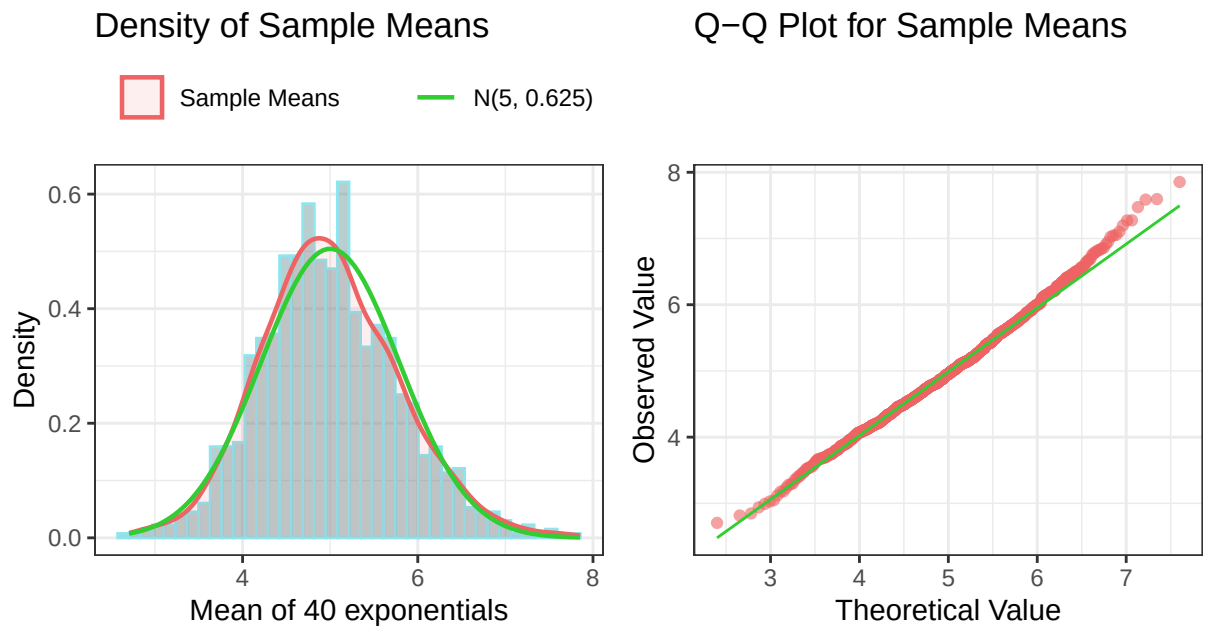


Sample Variance versus Theoretical Variance

The variance of the distribution of averages is close to the theoretical squared standard error of the mean: the sample variance is 0.618 while the theoretical variance is $\frac{\sigma^2}{n} = \frac{1}{\lambda^2} \times \frac{1}{n} = \frac{5^2}{40} = 0.625$, so there is a difference of 0.0068. The variance is calculated as the variance of the distribution from which the sample was taken, divided by the sample size, n.

Distribution

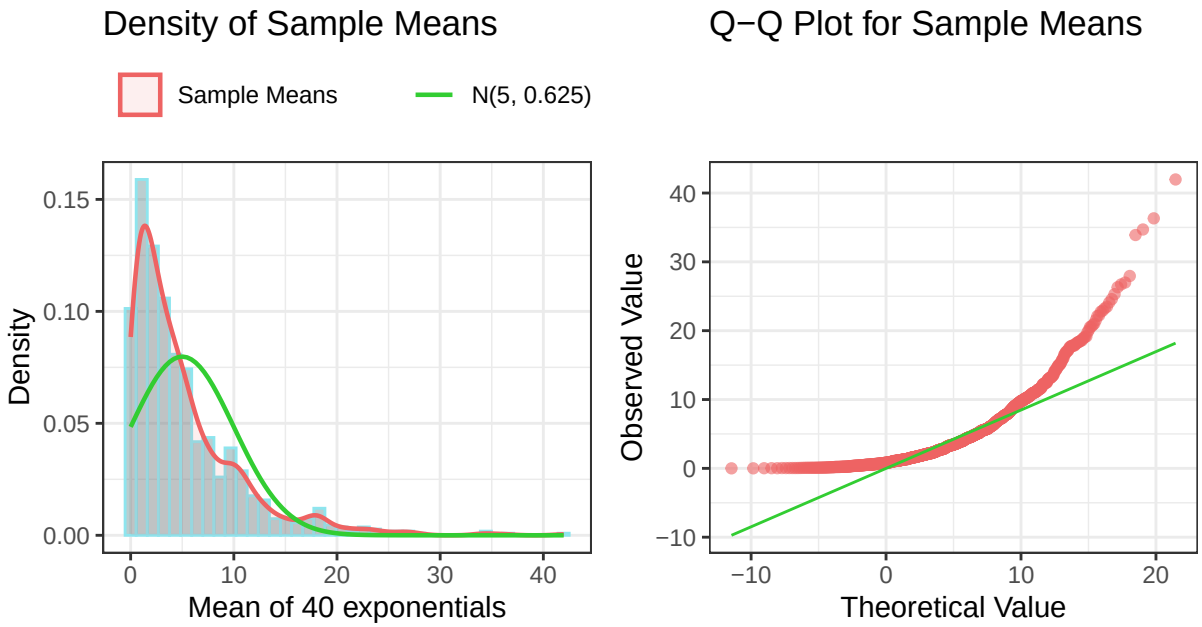
Finally, the distribution of the 1000 sample means is shown below as well as an overlay of the normal distribution with the appropriate mean and standard variation, μ and σ such that $N(\mu, \sigma^2) = N(\frac{1}{\lambda}, \frac{1}{\lambda^2} \times \frac{1}{n}) = N(5, 0.625)$. This density function is the theoretical sampling distribution and its similarity with the sample distribution suggests that the sample distribution of sampled means is approximately normal.



The Quantile-Quantile (Q-Q) plot also suggests that this distribution of averages of exponentials is approximately normal by comparing the quantiles of this distribution and a normal distribution - a point in this plot corresponds to a quantile from one distribution plotted against the same quantile from the other distribution.

The points are largely linear and lie on the line $y = x$ which indicates that the simulation data follow a roughly normal distribution.

In contrast, this distribution of averages of 40 exponentials is very different to a similarly-sized collection of random exponentials as seen below. Here, another simulation is run but now with a sample size of 1 instead of the average of 40 draws.



Now, the simulation results follow the underlying exponential distribution so the normal curve does not fit as well as in the plot of the distribution of averages of 40 exponentials. This is emphasised with the Q-Q plot where the points are strongly nonlinear and, so are not normally distributed.

Conclusion

The differences between the respective histograms and Q-Q plots for the distribution of averages of 40 exponentials and the distribution of random exponentials highlight how the simulated sampling mean obeys the CLT and asymptotically converges to the theoretical normal distribution as the number of i.i.d. random variables, n, approached infinity. However, for this to hold, the random variables must be independent and identically distributed.

Appendix

Code for Figure 1

```
g <- ggplot(data.frame(x = sim_means), aes(x)) +
  geom_histogram(bins = 80, color = "cadetblue2", fill = "cadetblue4",
                 alpha = 0.6) +
  geom_vline(aes(xintercept = sample_mean, linetype = "Observed Mean"),
             linewidth = 1,
             color = "limegreen") +
  geom_vline(aes(xintercept = theory_mean, linetype = "Expected Mean"),
             linewidth = 1,
             color = "indianred2") +
  scale_linetype_manual(values = c("dashed", "solid")) +
  labs(title = "Means of Simulated Exponentials",
       x = "Mean of 40 exponentials",
       y = "Frequency",
       color = "",
       linetype = "") +
  theme_bw() +
  theme(legend.position = "top")

g
```

Code for Table 1

```
Sample_stats <- c(sample_mean, sample_sd, sample_var)
Theoretical_stats <- c(theory_mean, theory_sd, theory_var)
Difference <- c(
  abs(theory_mean - sample_mean),
  abs(theory_sd - sample_sd),
  abs(theory_var - sample_var))

data.frame(Sample_stats, Theoretical_stats, Difference,
           row.names = c("Mean", "Standard deviation", "Variance"))
```

Code for Figure 2

```
mean_cumsum <- cumsum(sim_means) / (1:nsim)

g <- ggplot(data.frame(x = 1:nsim, y = mean_cumsum), aes(x = x, y = y)) +
  geom_line(
    aes(color = "Cumulative Mean of Sample Means"),
    linewidth = 0.9) +
  geom_hline(
    aes(yintercept = theory_mean, color = "Expected Mean of Sampling Mean"),
    linetype = "dashed") +
  scale_color_manual(values = c("cadetblue2", "indianred2")) +
  guides(color = guide_legend(
    override.aes = list(linetype = c("solid", "dashed"))
  )) +
  labs(x = "Number of Sample Means",
       y = "Cumulative mean",
```

```
       color = "") +
  theme_bw() +
  theme(legend.position = "top")
g
```

Code for Figure 3

```
g1 <- ggplot(data.frame(x = sim_means), aes(x)) +
  geom_histogram(bins = 40, color = "cadetblue2", fill = "cadetblue4", alpha = 0.4,
                 aes(y = after_stat(density))) +
  geom_density(aes(fill = "Sample Means"),
               color = "indianred2", alpha = 0.1, lwd = 0.8) +
  scale_fill_manual(values = "indianred2") +
  stat_function(aes(color = "N(5, 0.625)"),
                fun = dnorm,
                args = list("mean" = theory_mean, "sd" = theory_sd),
                lwd = 0.8,
                geom = "line") +
  scale_color_manual(values = "limegreen") +
  labs(title = "Density of Sample Means",
       x = "Mean of 40 exponentials",
       y = "Density",
       color = "",
       fill = "",) +
  theme_bw() +
  theme(legend.position = "top")

g2 <- ggplot(data.frame(x = sim_means), aes(sample = x)) +
  geom_qq(distribution = qnorm,
          dparams = list("mean" = theory_mean, "sd" = theory_sd),
          color = "indianred2",
          alpha = 0.6) +
  geom_qq_line(distribution = qnorm,
               dparams = list("mean" = theory_mean, "sd" = theory_sd),
               color = "limegreen") +
  labs(title = "Q-Q Plot for Sample Means",
       x = "Theoretical Value",
       y = "Observed Value") +
  theme_bw()
g1 + g2
```

Code for Figure 4

```
ndist2 <- 1

sim_data2 <- matrix(rexp(nsim * ndist2, rate = lambda), nsim, ndist2)

theory_sd2 <- (1 / lambda) * (1 / sqrt(ndist2))
theory_var2 <- theory_sd2 ^ 2

g1 <- ggplot(data.frame(x = sim_data2), aes(x)) +
  geom_histogram(bins = 40, color = "cadetblue2", fill = "cadetblue4", alpha = 0.4,
                 aes(y = after_stat(density))) +
```

```
  geom_density(aes(fill = "Sample Means"),
               color = "indianred2", alpha = 0.1, lwd = 0.8) +
  scale_fill_manual(values = "indianred2") +
  stat_function(aes(color = "N(5, 0.625)"),
                fun = dnorm,
                args = list("mean" = theory_mean, "sd" = theory_sd2),
                lwd = 0.8,
                geom = "line") +
  scale_color_manual(values = "limegreen") +
  labs(title = "Density of Sample Means",
       x = "Mean of 40 exponentials",
       y = "Density",
       color = "",
       fill = "",) +
  theme_bw() +
  theme(legend.position = "top")

g2 <- ggplot(data.frame(x = sim_data2), aes(sample = x)) +
  geom_qq(distribution = qnorm,
          dparams = list("mean" = theory_mean, "sd" = theory_sd2),
          color = "indianred2",
          alpha = 0.6) +
  geom_qq_line(distribution = qnorm,
               dparams = list("mean" = theory_mean, "sd" = theory_sd2),
               color = "limegreen") +
  labs(title = "Q-Q Plot for Sample Means",
       x = "Theoretical Value",
       y = "Observed Value") +
  theme_bw()

g1 + g2
```