Johns Hopkins Coursera: Statistical Inference Peer Reviewed Project, Part 1

Part 1: Simulation Exercise

Overview

In this project, we want to use simulation to make a discovery about the exponential distribution. An exponential distribution of 40 observances was simulated 1000 times. The simulated distribution had a mean of 5 and a standard deviation of 5. We then compare the sample mean to the theoretical mean, the sample variance and standard deviation to the theoretical variance and standard deviation, and finally show that the distribution of the mean is approximately normal through visual aids.

Simulation, Means, and Variances

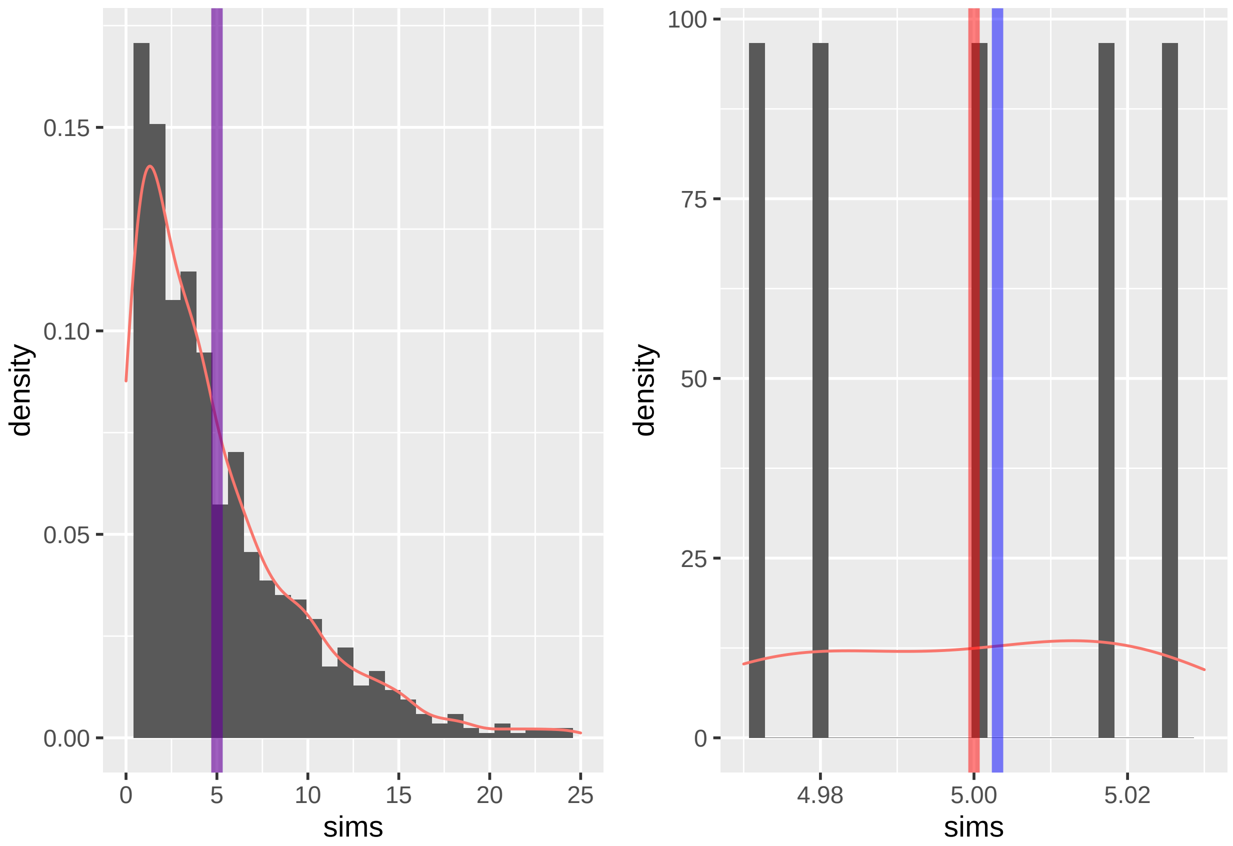
```r
library(ggplot2)
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.1.1
```

```r
#Set seed for reproducibility.
set.seed(1234)

#Generate a dataframe of exponential values
sims <- data.frame("sims" = rexp(1000, rate = .2))

#The first plot containing the entire distribution
plt1 <- ggplot(data = sims, mapping = aes(x = sims, y = ..density..)) +
        geom_histogram() +
        geom_density(mapping = aes(colour = "red")) +
        geom_vline(xintercept = mean(sims$sims), color = "blue", alpha = .5, size = 2)+
        geom_vline(xintercept = 1/.2, color = "red", alpha = .25, size = 2) +
        xlim(0,25) +
        theme(legend.position = "none")

#The second plot, zooming into where the differences are.
plt2 <- ggplot(data = sims, mapping = aes(x = sims, y = ..density..)) +
        geom_histogram() +
        geom_density(mapping = aes(colour = "red")) +
        geom_vline(xintercept = mean(sims$sims), color = "blue", alpha = .5, size = 2)+
        geom_vline(xintercept = 1/.2, color = "red", alpha = .5, size = 2) +
        xlim(4.97,5.03) +
        theme(legend.position = "none")

#Plotting
plot_grid(plt1, plt2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
## Warning: Removed 8 rows containing non-finite values (stat_density).
## Warning: Removed 2 rows containing missing values (geom_bar).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 995 rows containing non-finite values (stat_bin).
## Warning: Removed 995 rows containing non-finite values (stat_density).
## Warning: Removed 2 rows containing missing values (geom_bar).
```



On the left is the entire distribution, and on the right is the distribution zoomed in to see the lines diverge.

```
print(c("Means",1/.2, mean(sims$sims)))
```

```
## [1] "Means"             "5"                  "5.0030668729407"
```

```
print(c("Standard Deviations",1/.2, sd(sims$sims)))
```

```
## [1] "Standard Deviations" "5"                    "5.05671826956297"
```

$\mu - \bar{x} = 0.003067$, % difference from $\mu = 0.0006134$ %

$\sigma - s = .05672$, % difference from $\sigma = 0.011344$ %

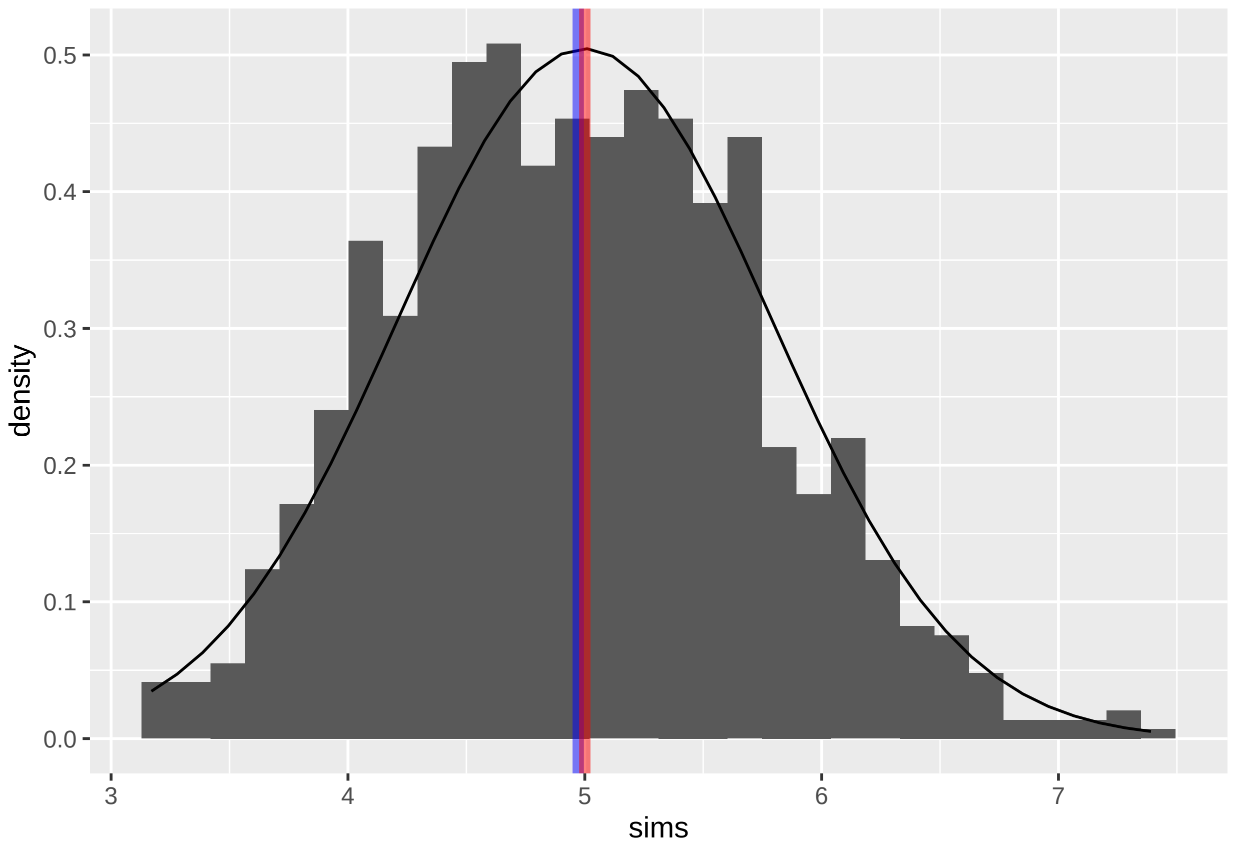
At n = 1000, we are over 98% accurate when measuring mean and variance. This is due to the Central Limit Theorem, which states that as n increases, our accuracy in approximating a distribution also increases.

Distribution

Now, let's generate 1000 sample distributions, each with 40 observations, and take the mean of each distribution, and plot them.

```
#Initializing the list
means = NULL

#Declaring size externally for cleaner code
size = 40

#Generating the distributions 1000 times
for (i in 1:1000){
        means <- c(means, mean(rexp(size, rate = .2)))
}

#Writing the list to a dataframe
sims <- data.frame("sims" = means)

#Plotting the distribution of the means
ggplot(data = sims) +
        geom_histogram(mapping = aes(x = sims, y = ..density..)) +
        stat_function(fun = dnorm, n = 40, args = list(mean = 1/.2, sd = 1/.2/sqrt(40))) +
        geom_vline(xintercept = mean(sims$sims), color = "blue", alpha = .5, size = 2)+
        geom_vline(xintercept = 1/.2, color = "red", alpha = .5, size = 2) +
        theme(legend.position = "none")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
print(c("Mean", mean(sims$sims)))
```

```
## [1] "Mean"              "4.97231912836865"
```

```
print(c("Standard Deviation", sd(sims$sims)))
```

```
## [1] "Standard Deviation" "0.76022251264432"
```

The means appear to follow a normal distribution, but how can we check?

We know a few things:

$\mu = 1/\lambda$

$\sigma = \sqrt{(}1/\lambda^2)$

Given: $\lambda = .2$

We can check against what we already know about normal distributions

1. $\mu = 1/\lambda = 5$, $\bar{X} = 4.972$
2. $\sigma = .79057$, $s = .7602$

Discussion

- Means: Our simulated mean was almost perfectly in line with the population mean, due to the central limit theorem
- Variance: As n increases, the variance in our sample mean distribution will decrease.