Central Limit Theorem

Lucas Mendicino

12/1/2021

Central Limit Theorem

The Central Limit Theorem says that equally-weighted averages of samples from any distribution themselves are normally distributed.

Formally, consider the sample mean of i.i.d random variables $X_1, X_2 /dots $ such that $E[X_i] = \mu$ and $Var(X_i) = \sigma^2$:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n}(X_i)$$

The Central Limit Theorem states as $n \to \infty$:

$$\bar{X} \sim N(\mu, \frac{\sigma^2}{n})$$

It is often expressed as a way of obtaining the standard normal, Z. Thus, as $n \to \infty$:

$$Z = \frac{(\sum_{i=1}^{n} X_i) - n\mu}{\sigma\sqrt{n}}$$

Through some algebraic manipulation, we can show that if the sample mean of i.i.d random variables is normal, it follows that the sum of i.i.d random variables must also be normal Let $\bar{Y}$ be equal to the sum of i.i.d random variables:

$$\bar{Y} = \sum_{i=1}^{n}(X_i) = n\bar{X}$$

Then, as $n \to \infty$:

$$\bar{Y} \sim N(n\mu, n^2\frac{\sigma^2}{n})$$

since $\bar{X}$ is normal and n is a constant. Then

$$\bar{Y} \sim N(n\mu, n\sigma^2)$$

In summary, the Central Limit Theorem explains that both the average of i.i.d random variables and the sum of i.i.d random variables are normal. This is true regardless of what distribution the i.i.d variables came from. Note that a continuity correction is needed to get the best results for small n, say $n < 50$.

Let's simulate the Central Limit Theorem.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

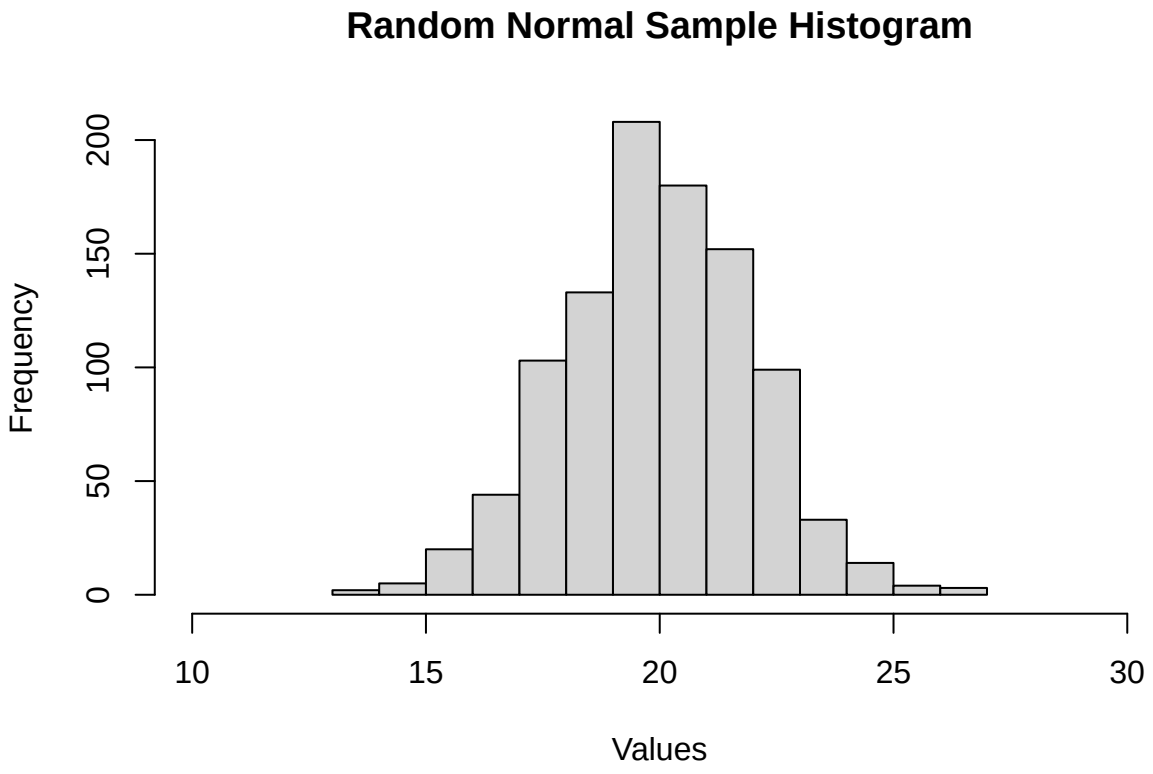
```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.0     v forcats 0.5.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

First, let's explore the case of the sample means from a normal distribution.

```
set.seed(42)
n_obs = 1000
rand_n_dist = rnorm(1000, 20, 2) # random normal distribution with mean = 5, standard deviation = 2

hist(rand_n_dist, main = "Random Normal Sample Histogram",
     xlab = "Values",
     xlim = c(10,30))
```




```
pop_mean = mean(rand_n_dist)
pop_sd = sqrt(sum((rand_n_dist-mean(rand_n_dist))^2)/n_obs)

pop_mean
```

```
## [1] 19.94835
```

```
pop_sd
```

```
## [1] 2.00404
```

Let's sample from the distribution

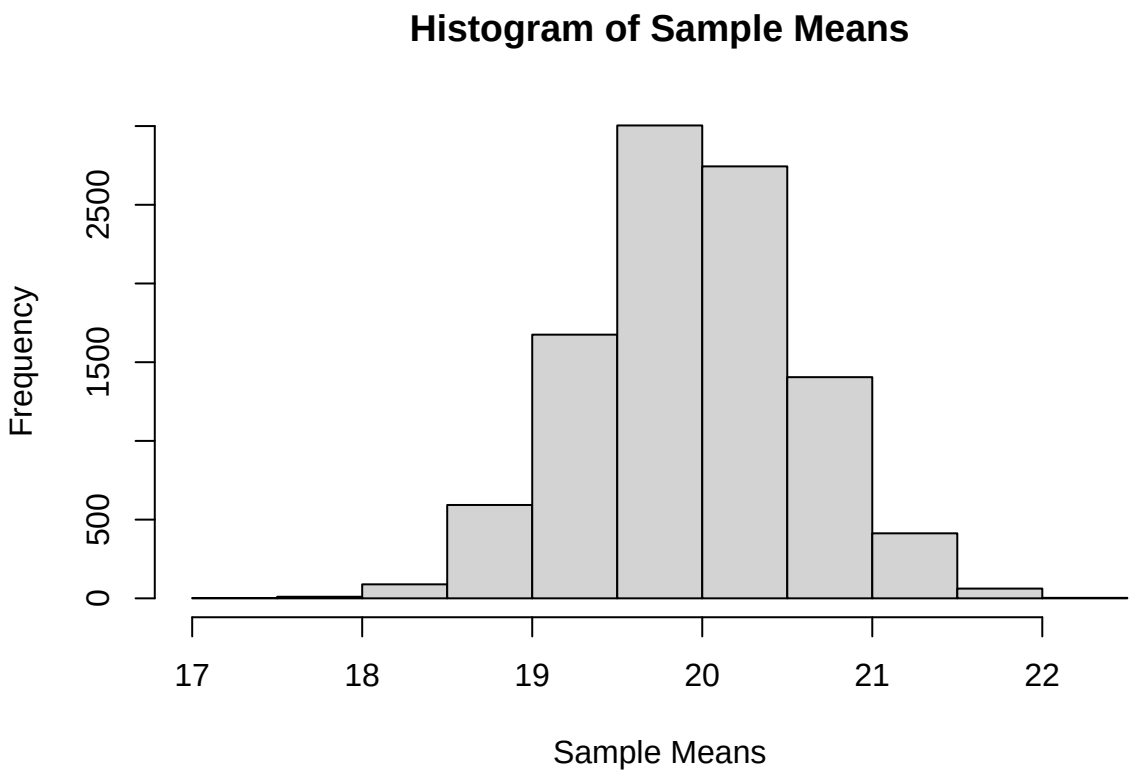
```r
sample_mean=vector()

nperm = 10000

n = 10

for (i in 1:nperm){
  temp = sample(rand_n_dist, n)
  sample_mean[i] = mean(temp)
  }

hist(sample_mean, main = "Histogram of Sample Means",
     xlab = "Sample Means")
```




```r
mean_sample_mean = mean(sample_mean)
sd_sample_mean = sqrt(sum((sample_mean-mean(sample_mean))^2)/length(sample_mean))



mean_sample_mean
```

```
## [1] 19.94413
```

```r
sd_sample_mean
```

```
## [1] 0.629135
```

```r
###

par(mfrow=c(2,1))
hist(rand_n_dist, xlim=c(10,30))
```

```
hist(sample_mean, xlim=c(10,30))
```