Statistical Inference Part 1

Rithika

23/10/2021

Here we want to run a simulation consisting in generating 40 variables from an exponential function with given parameters and repeating this for 1000 times. Then we compute the mean for each simulation(1000 means in total).

First of all we load the required packages and set the relevant parameters for the simulations.

```
#Load packages
library(dplyr, warn.conflicts = F)
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
library(ggplot2)

#Exponential function parameters
lambda <- 0.2
n <- 40
num.of.sim <- 1000

#set the seed
set.seed(119983)

#Create a 1000x40 matrix containing the results of the simulation
sim.distrib <- matrix(data=rexp(n * num.of.sim, lambda), nrow=num.of.sim)
```

###Sample mean vs theoretical mean

we compute the means and store the results in a dataframe which is what the dplyr and ggplot2 packages take as input and it's also the typical datastructure in R. So we create a dataframe, *sim_mns*.

Here we want to compare the theoretical mean for an exponetial distribution, given by mu = 1/lambda = 5, to the mean of our simulated distribution.

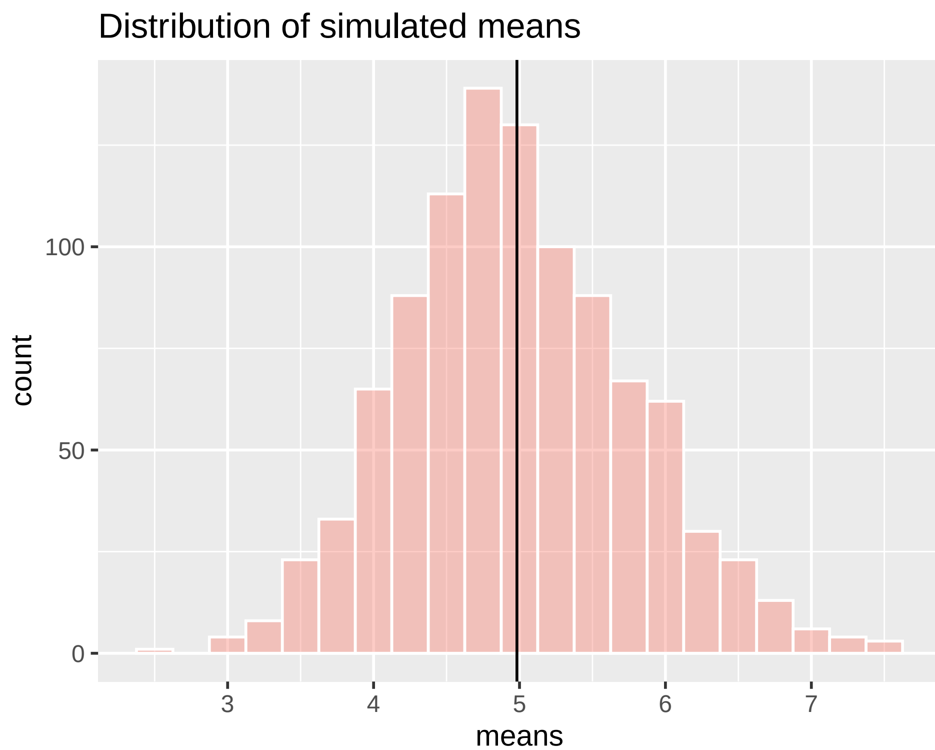
```
#compute the mean for each of the 1000 simulations(rows)
sim_mns <- data.frame(means=apply(sim.distrib, 1, mean))

#Convert dataframe to tbl_df object for more convenient printing
sim_mns <- tbl_df(sim_mns)
```

```
## Warning: `tbl_df()` was deprecated in dplyr 1.0.0.
## Please use `tibble::as_tibble()` instead.
```

```
#compute the mean of the simulated means
(mean_sim <- sim_mns %>% summarize(simulated.mean = mean(means)) %>% unlist())
```

```
## simulated.mean
##       4.982365
```

```
#Plot sample means distribution with the calculated...bla bla...
sim_mns  %>%
  ggplot(aes(x = means) ) + geom_histogram(alpha=0.4, binwidth= .25, fill = "salmon", col = "white") +
  geom_vline(xintercept = mean_sim, color="black", size = 0.5) +
  ggtitle("Distribution of simulated means")
```



From the plot we can see that the distribution of the means is centered around the mean of our simulated distribution, that is **4.982365** (the black vertical line) which is very close to the theoretical mean 1/lambda = **5**

###Sample Variance versus Theoretical Variance

```
#Compute the variance of the sample means
sd.samp <- sim_mns  %>% select(means)  %>% unlist() %>%  sd()
(var.samp <- sd.samp ^ 2)
```

```
## [1] 0.628253
```

```
#Theoretical variance of the exponential distribution
(((1/lambda))/sqrt(40))^2
```

```
## [1] 0.625
```

As we can see they're very close, 0.628253 and 0.625, respectively.

###Normality of the Distribution

From the Central limit theorem we know that the distribution of averages of normalized variables becomes that of a standard normal distribution as the sample size increases.

Here we want to normalize our sample means. To do this we need to transform each mean in our simulated datset according to the following formula:

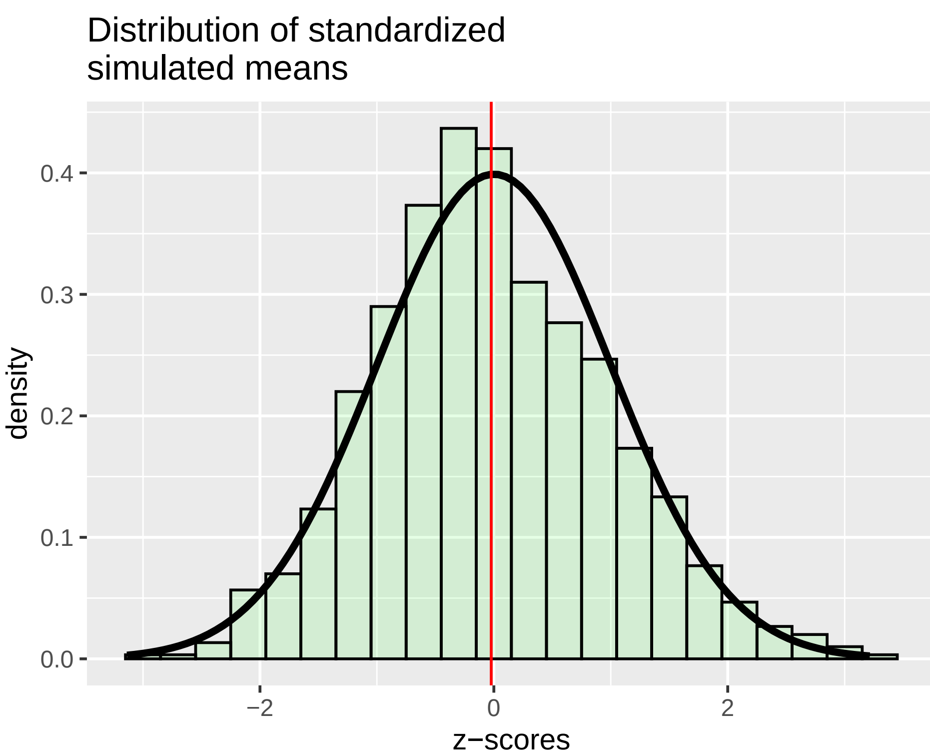
z-score = (xbar - 1/lambda) / (1/lambda / sqrt(n))

Based on the CLT the result should be a normal distribution centered at(more or less) zero. To see if this is the case we create a plot comparing the density of our transformed sample means distribution with the density of the standard normal distribution.

```
#Compute mean of our normalized means
(z_mean <- sim_mns %>%  mutate(z_score = (means - 1/lambda) / (1/lambda / sqrt(n)))  %>% select(z_score
   summarise(z_mean = mean(z_score)) %>% unlist())
```

```
##      z_mean
## -0.02230626
```

```
#create Z scores from the sample means
sim_mns %>%  mutate(z_score = (means - 1/lambda) / (1/lambda / sqrt(n))) %>%
  ggplot(aes(x = z_score)) +
   geom_histogram(alpha=0.1,  binwidth = 0.3, fill="green", color="black", aes(y = ..density..)) +
   stat_function(fun = dnorm, size = 1.3) +
   geom_vline(xintercept = z_mean, color="red", size = 0.5) +
   ggtitle("Distribution of standardized \nsimulated means") +
   xlab("z-scores")
```



From the plot we see that the normalized distribution of sample means is approximately the same as the standard normal distribution as we can see comparing it to the density function, the black bell-shaped curve. Also, the mean is -0.02230626, very close to zero. This is consistent with what is stated in the Central Limit Theorem.