

Statistical Inference: exponential function simulation

Daria Stepanyan

Synopsys

This report is dedicated to the investigation of the exponential function distribution. For this simulation is run and its results are compared against Central Limit Theorem predictions.

Simulation

For the purpose of this investigation we run a 1000 simulations of 40 exponentials with $\lambda = .2$, taking their mean

```
lambda = 0.2
n = 40
nsim = 1000
mns = NULL
set.seed(2019)
for (i in 1 : nsim) {
  mns = c(mns, mean(rexp(n, lambda)))
}
```

Mean and variance

Using the fact that mean function is linear, we can calculate sample mean:

```
mean(mns)
```

```
## [1] 5.038755
```

which is quite close to a theoretized value of $1/\lambda$, or 5

We can also calculate variance values

```
setNames(data.frame(matrix(data = c(1/(lambda^2)/n, var(mns)),
  ncol = 2, nrow = 1)),
  c("theoretical variance", "sample variance"))
```

```
##   theoretical variance sample variance
## 1                0.625      0.6253269
```

Both mean and variance are quite close to theoretized ones and it will be shown on the plot below.

Distribution

CLT tells us that assuming the variables are iid the distribution of means of that variable will approximate normal distribution - even if original values are not normally distributed - and its mean will estimate a population mean. Let's see it on the plot.

First we create a simulated means distribution plot

```
bw = .1
library(ggplot2)
library(gridExtra)

## plot a histogram of simulation means
plotmean <- ggplot(mapping = aes(mns)) +
```

```

    geom_histogram(alpha=0.3, fill='white', colour='black', binwidth = bw)
## add a theoretical mean to a plot (dashed red line)
plotmean <- plotmean + geom_vline(xintercept = 1/lambda,
    linetype="dashed", color = "red")
## add the sample mean
plotmean <- plotmean + geom_vline(xintercept = mean(mns),
    linetype="dashed", color = "blue")
## add titles
plotmean <- plotmean + labs(title = "Distribution of sample means of exponential function (1000 simulat.
    x="sample means")

## add a normal distribution curve scaled to the sample mean distribution
tmean = 1/lambda
tsd = sqrt(1/(lambda^2)/n)

smean = mean(mns)
ssd = sd(mns)

plotnorm <- plotmean +
    stat_function (fun = function(x, mean, sd, n){
        n * dnorm(x = x, mean = mean, sd = sd) * bw
    },
    args = c(mean = tmean, sd = tsd, n = nsim), color = "red") +
    stat_function (fun = function(x, mean, sd, n){
        n * dnorm(x = x, mean = mean, sd = sd) * bw
    },
    args = c(mean = smean, sd = ssd, n = nsim), color = "blue")

```

For truly seeing CLT in action we compare the distribution we got for plain exponentials obtained in 1000 simulations

```

randsam <- rexp(nsim, lambda)

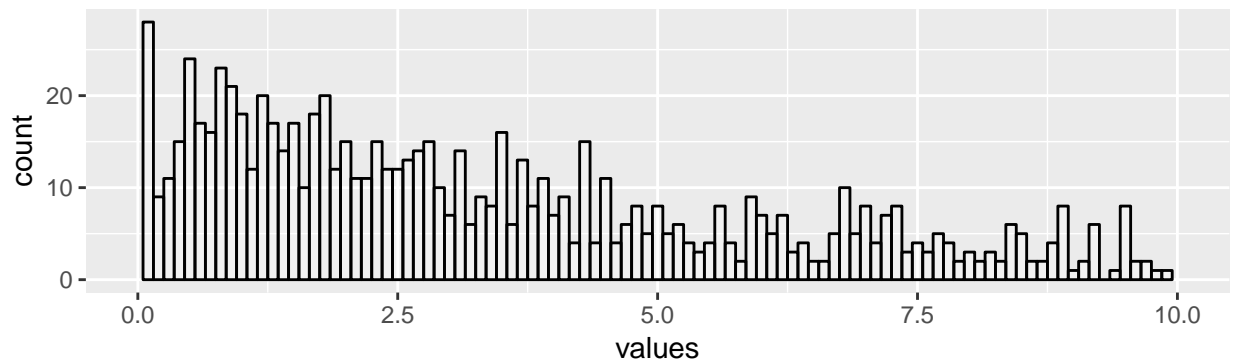
plotrand <- ggplot(mapping = aes(randsam)) +
    geom_histogram (alpha=0.3, fill='white',
        colour='black', binwidth = bw) +
    labs(title = "Distribution of exponential function values (1000 simulations)", x="values")

## adding the scale for better readability
plotrand <- plotrand + xlim(0,10)
plotnorm <- plotnorm + xlim(0,10)

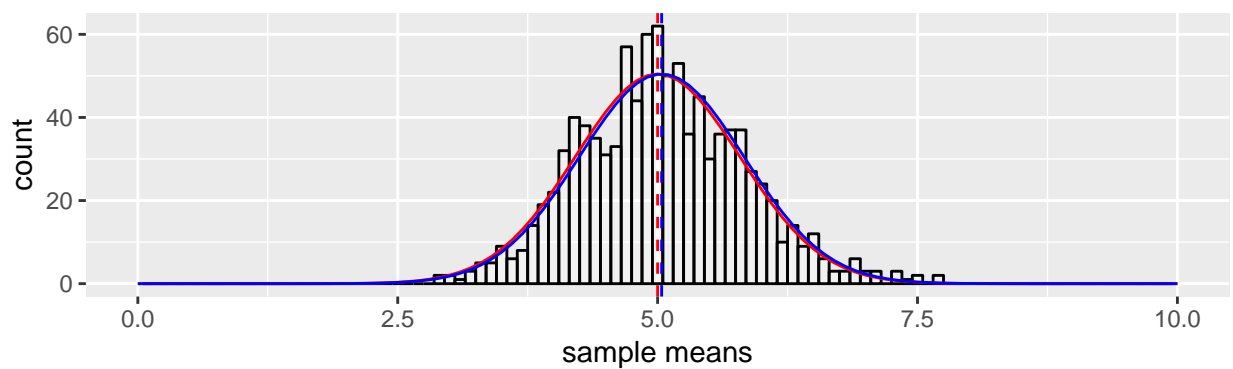
## plot 2 distributions
grid.arrange(plotrand, plotnorm, nrow = 2)

```

Distribution of exponential function values (1000 simulations)



Distribution of sample means of exponential function (1000 simulations)



Red bell curve represent theoretical values for the mean and variance of the distribution. The blue one uses the sample mean and variance for our initial simulation. Sample data appears to be slightly skewed to the right.

We can see that even though the underlying data is not normally distributed, the distribution of sample means approximate the classic bell curve of a normal distribution.