Statistical Inference - Course Project - Part 1 - Simulation Exercise

Ioannis Moros

01/07/2017

R Setup

```
require(ggplot2)
```

Introduction

Consider the Exp(lambda = 0.2) distribution. We will create 1000 samples of size 40 each from this distribution and calulate the mean value of each one (X'_i for i=1,2,…1000). We will then use these 1000 simulated sample means to investigate key parameters of the distribution of the sample mean and compare to the parameters of the theoretical distribution. We will also show that the sample means are approximately normally distributed.

Questions 1 & 3.

Show the sample mean and compare it to the theoretical mean of the distribution. Show that the sample means are approximately normally distributed.

First, simulate the data:

```
# Set the parameters of the exponential distribution
my_lambda = 0.2
n = 40
nsim = 1000

# Set a seed for the sake of reproducibility
set.seed(365)

# The known population parameters of the exponential distribution
exp_mu = 1 / my_lambda
exp_var = 1 / my_lambda^2

# simulate the sample means
means = NULL

for(i in 1:nsim) {
        means = c(means, mean(rexp(n, my_lambda)))
}
```

Now calculate the means and investigate the difference:

```
# The average of sample means
myexp_mean <- mean(means)
myexp_mean
```

```
## [1] 4.996394
```

```
# The theoretical population mean
exp_mu
```

```
## [1] 5
```

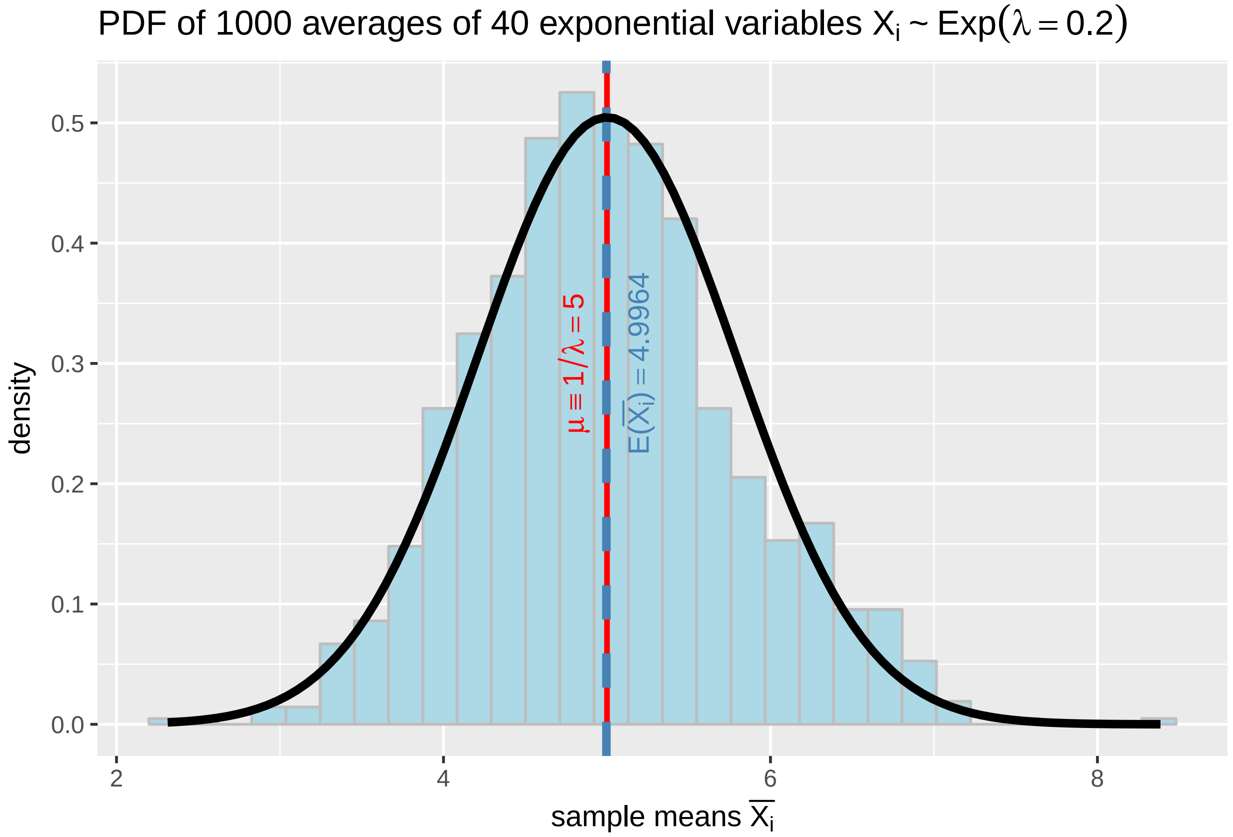
```
# And the difference between theoretical and simulated average of sample means
abs(exp_mu - myexp_mean)
```

```
## [1] 0.003606055
```

Now we create the PDF of the simulated sample means and overlay the PDF of Normal(mu, sigmaˆ2 / n) on top, to demonstrate the covergence:

```
# math expressions to use in the ggplot annotations
myexpr1 <- substitute(mu %==% 1/lambda == m, list(m = exp_mu))
myexpr2 <- substitute(paste("E(", bar(X[i]), ")") == m, list(
        m = format(myexp_mean, digits = 5)))

# create the pdf of sample means
# The pdf of N(mu, sigma^2 / n) is overlayed on top
g <- ggplot(data = NULL, aes(means))
g + geom_histogram(aes(y = ..density..), col = "gray", fill = "lightblue") +
        geom_vline(xintercept = 1/my_lambda, col = "red",
                   linetype = "solid", size = 1) +
        annotate("text", x = (1/my_lambda) - 0.2, y = 0.3,
                 label = deparse(myexpr1), col = "red", angle = 90,
                 parse = TRUE) +
        geom_vline(xintercept = mean(means), col = "steelblue",
                   linetype = "dashed", size = 1.5) +
        annotate("text", x = mean(means) + 0.2, y = 0.3,
                 label = deparse(myexpr2), col = "steelblue", angle = 90,
                 parse = TRUE) +
        stat_function(fun = dnorm, size = 1.5, args = list(mean = exp_mu, sd = sqrt(exp_var/n))) +
        ggtitle(substitute(paste("PDF of ", a, " averages of ", b,
                                 " exponential variables ", X[i] %~%
                                         Exp(lambda == j)), list(
                                                 a = nsim,
                                                 b = n,
                                                 j = my_lambda))) +
        xlab(expression(paste("sample means ", bar(X[i]))))
```



The average of sample means converges to the theoretical mean of the exponential distribution. We expect the convergence to get better as the number of simulations increases (although we got "lucky" with our seed). The distribution of sample means follows approximately the normal distribution N(mu, sigma^2 / n), where mu and sigma^2 the theoretical mean and variance respectively of the exponential distribution that was used to simulate the data (lambda = 0.2) and n = 40 the sample size.

2. Investigate the difference between sample variance and theoretical population variance

```
# observed variance of sample means
var(means)
```

```
## [1] 0.6535221
```

```
# theoretical variance of sample means
exp_var / n
```

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

```
# difference
abs((exp_var / n) - var(means))
```

```
## [1] 0.02852212
```

The observed variance converges to the theoretical population variance. Again, we expect the convergence to become better as we increase the number of simulations.