Statistical Inference

Simulation Exercise

The exponential distribution can be simulated in R with `rexp(n, lambda)` where lambda is the rate parameter. The mean of exponential distribution is `1/lambda` and the standard deviation is also also `1/lambda`. Set `lambda = 0.2` for all of the simulations. In this simulation, you will investigate the distribution of averages of `40` exponentials. Note that you will need to do a thousand or so simulated averages of 40 exponentials.

First we will evaluate the first two points of the problem:

1. Show where the distribution is centered at and compare it to the theoretical center of the distribution.
2. Show how variable it is and compare it to the theoretical variance of the distribution.

In the appendix you can find `code chunk 1 and figure 1 (histogram of meanValue)` which solve this part. The interpretation of the code and histogram is the following: the distribution is centered at `mean(meanValues) = 4.986963` and the theoretical center of distribution is: `alpha = 1/lambda = 5`. The simulated variance of `set.seed(1000)` is `var(meanValue) = 0.6583551`, while the theoretical value is `sigma^2 / n = 1 / (lambda^2 * n) = alpha^2 / population = 0.625`. Where `alpha, population, meanValue` are variables from code chunk 1 and `sigma, n` are from the mathematical formula for variance.

3. Show that the distribution is approximately normal.

Using the code in the Appendix code chunk 2, we have generated the QQ histogram. And it shows that it is approximately normal.

4. Evaluate the coverage of the confidence interval for `1/lambda:  X /plusminus 1.96 S/sqrt(n)`.

Using the code in the Appendix code chunk 3, we have generated the histogram, that shows the coverage around `0.9 (yintercept=0.94)`.

Appendix

Code chunk 1 with figure 1

```
set.seed(1000)
lambda <- 0.2
simulations <- 1000
population <- 40
alpha <- 1/lambda
sdValue <- alpha/sqrt(population)

values <- rexp(simulations  * population, lambda)
meanValue <- mean(values)

myMatrix <- matrix(values, simulations, population)
meanValue <- rowMeans(myMatrix)


hist(meanValue, xlab = 'Density', ylab = 'Means',
      breaks=100, prob = TRUE, col = 'white')

abline(v=alpha, col='blue')

xLine <- seq(min(meanValue), max(meanValue), length = 1000)
yLine <- dnorm(xLine, mean=alpha, sd=sdValue)
zLine <- density(meanValue)
lines(xLine, yLine, col='red')
lines(zLine, col = 'green')

legend('topright',
    c('simulation', 'theoretical'),
    col=c('green', 'red'),
    lty=c(1,1))
```

Code chunk 2 with figure 2:

```
qqnorm(meanValue, col = 'red')
qqline(meanValue)
```

Code chunk 3 with figure 3:

```
library(ggplot2)
upper <- meanValue +  1.96 * (sdValue)
lower <- meanValue -  1.96 * (sdValue)
lowerLower <- min(lower)
upperUpper <- max(upper)

xVal <- seq(lowerLower, upperUpper, by=0.0001)

xLine <- sapply(xVal, function(x) {
      mean(lower < x & upper > x)
})
```

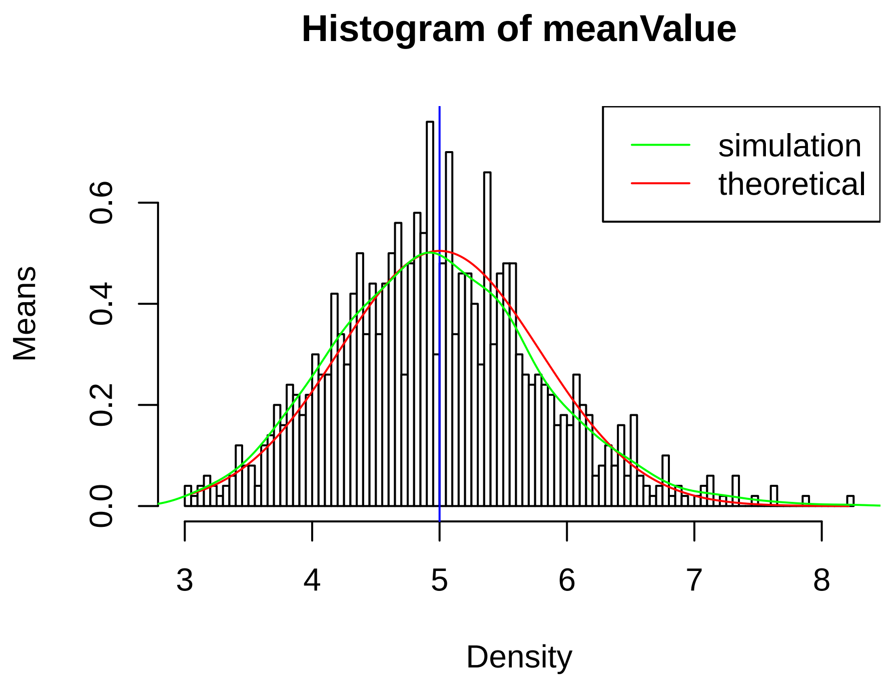

Figure 1: plot of chunk 1

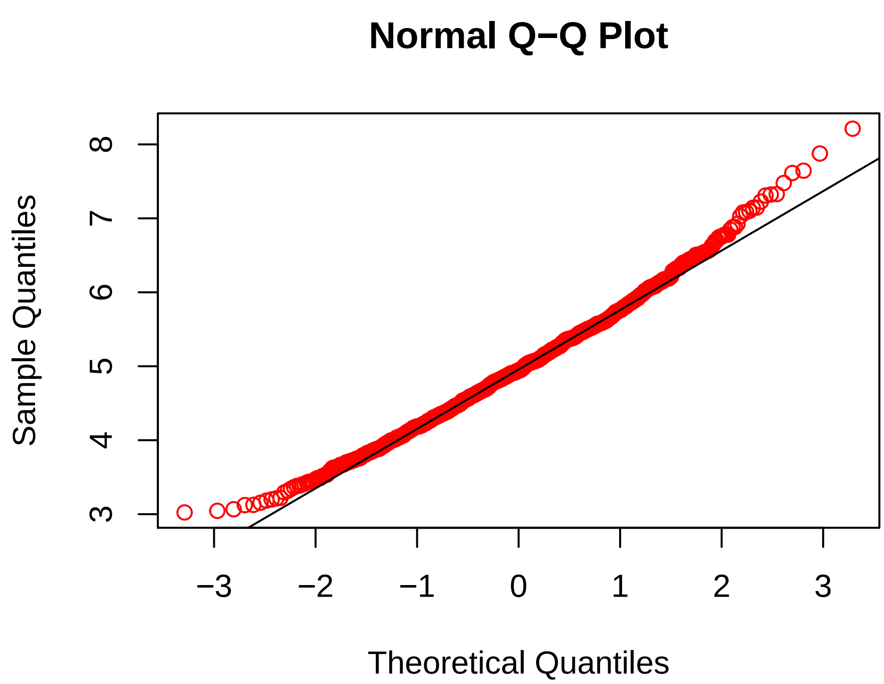

Figure 2: plot of chunk 2

```
## Code chunk continued 3

qplot(xVal, xLine) +
geom_hline(yintercept=0.94,col='red') +
xlab('Values') +
ylab('Coverage Line')
```

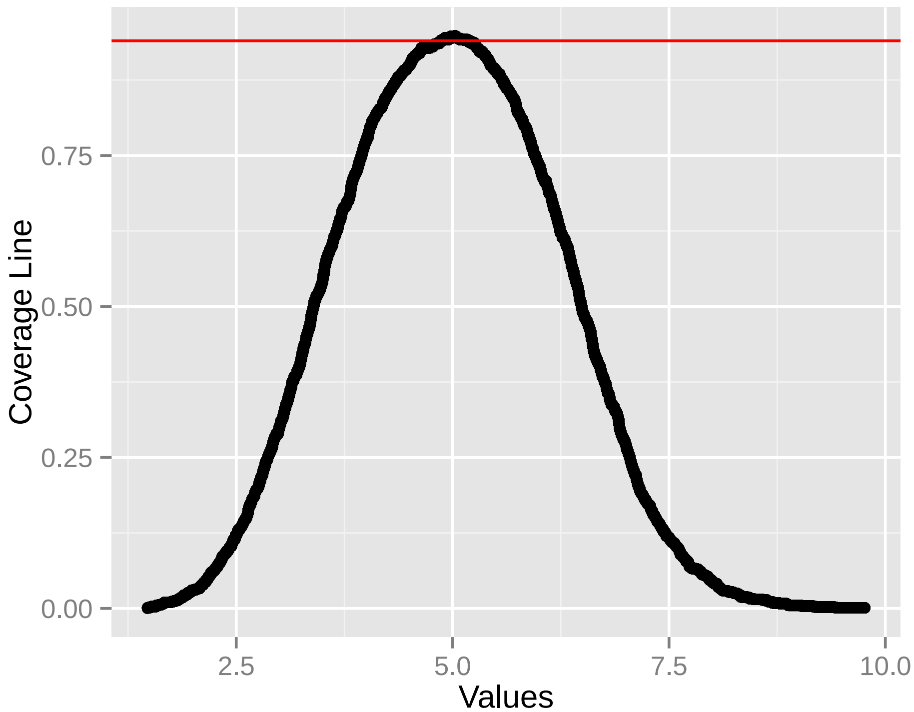

Figure 3: plot of chunk 3