Statistical Inference Project Report

Part 1 - Data Simulation and Exploration

Introduction

This is a report submitted to fulfill the requirements of the Statistical Inference course offered by Johns Hopkins University on Coursera. It covers some basic simulation and exploratory data analysis techniques.

Data Simulation and Exploration

For this exercise, data are simulated using the exponential distribution, and the mean value of 40 simulations are calculated. This is repeated 1000 times. The mean and variance of the simulated data are calculated, and these are compared with the mean ond variance values obtained through calculations.

Before starting, the necessary libraries must be loaded into R.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.1
```

```
library(datasets)
```

Simulation

The R code below is used to generate the simulated data to be used according the the exponential distribution specified. In this case, lambda = 0.2 and the number of samples (n) = 40.

```
set.seed(1986)
exponentials <- 40
lambda <- 0.2
sims <- 1000    # number of simulations to run

# create a blank data frame for the simulated data
simulated_Data <- data.frame(ncol=2, nrow=sims)
names(simulated_Data) = c("simulation number", "simulated mean")

# using a for loop, populate the new data frame with the simulated
# exponential distribution data
for(i in 1:sims) {
  simulated_Data[i,1] <- i
  simulated_Data[i,2] <- mean(rexp(exponentials, lambda))
}
```

Comparison of the Mean Values

The variance for the simulated data is calculated using the built-in "var" function, while the theoretical variance is calculated from fiirst principles. In this case, the mean is calculated b y 1/lambda, where lambda = 0.2.

```
# calculate and display the mean value of the simulated data
sim_Mean <- mean(simulated_Data[,2])
sim_Mean
```

```
## [1] 4.95159
```

```r
# calculate and display the mean of the theoretical data according to the
# equation 1/lambda
test_Mean <- 1/lambda
test_Mean
```

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

```r
# calculate and display the differnce in the simulated and theoretical mean values
diff_Mean <- test_Mean - sim_Mean
diff_Mean
```

```
## [1] 0.04841026
```

The theoretical and simulation values are very close, which indicates that the two are very good approximations of each other.

Comparison of the Variance Values

The variance for the simulated data is calculated using the built-in "var" function, while the theoretical variance is calculated from first principles. Note that, since the standard deviation is the square root of the variance, the variance can be calculated by squaring the variance and dividing it by the number of samples taken. In this particular case, the variance is therefore calculated by (1/lambda)2/n, where n = 40 and lambda = 0.2.

```r
# calculate and display the varaince value of the simulated data
sim_Variance <- var(simulated_Data[,2])
sim_Variance
```

```
## [1] 0.5986068
```

```r
# calculate and display the variance of the theoretical data according to the equation
# standard deviation)^2/number of samples
test_Variance <- ((1/lambda)^2)/exponentials
test_Variance
```

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

```r
# calculate and display the differnce in the simulated and theoretical variance values
diff_Variance <- test_Variance - sim_Variance
diff_Variance
```

```
## [1] 0.0263932
```

The theoretical and simulation values are very close, which indicates that the two are very good approximations of each other.

Plotting the Distributions

A plot of the density distribution of the data shows that it is approximately normally distributed. The addition of the distribution lines for the simulated data (shown in blue) and the theorectical data (shown in red) show that the curves drawn using their mean and standard deviation values also show a normal distribution. This confirms the Central Limit Theorem, which states that the distribution of the sample means will approach a normal distribution.

```r
sim_plot <- ggplot(data = simulated_Data, aes(simulated_Data$`simulated mean`)) +
            geom_histogram(aes(y = ..density..),bins = 50, col = "black", fill = "grey",
                           alpha = 0.5) +
            labs(title = "Histogram of Simulated Mean Values", x = "Simulated Mean Value",
                 y = "Density") +
```

```
            stat_function(fun = dnorm, args = list(mean = test_Mean,
                          sd = sqrt(test_Variance)),size = 1, col = "red") +
            stat_function(fun = dnorm, args = list(mean = sim_Mean,
                          sd = sqrt(sim_Variance)), size = 1, col = "blue")
plot(sim_plot)
```