Practical Analysis of the CLM and Hypothesis Testing

Kate Prendergast

Part 1

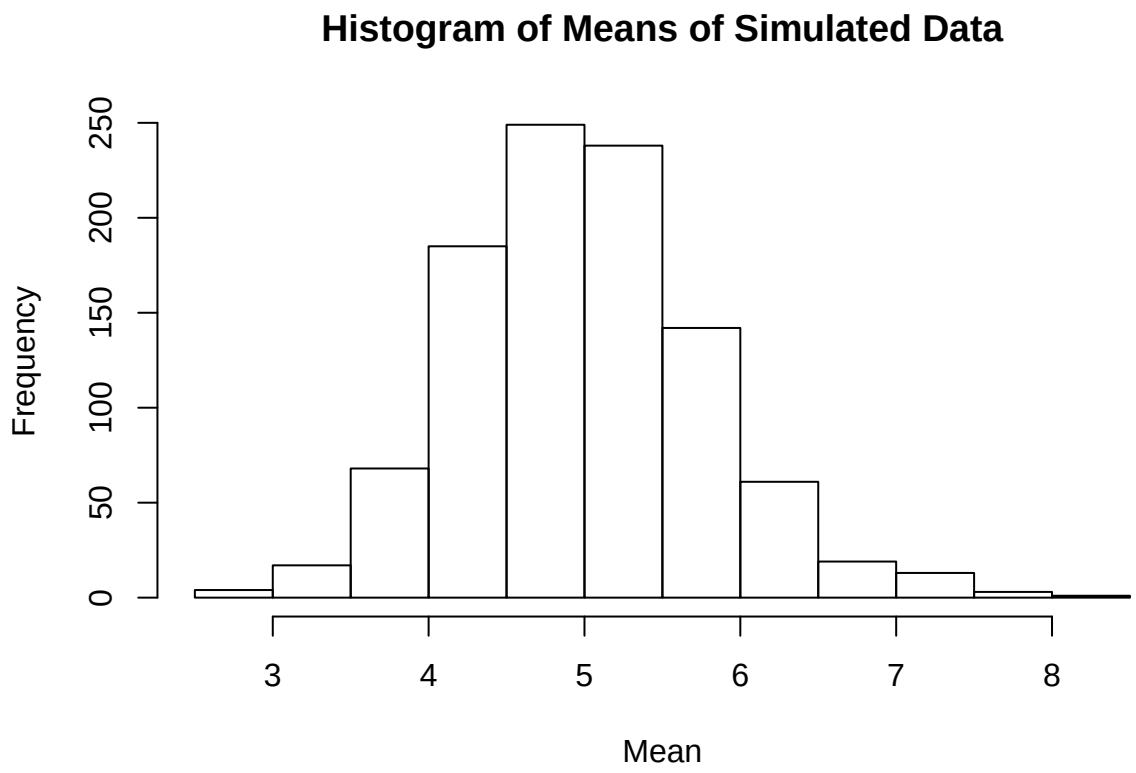
Overview

In Part 1 of this report, we will examine the distribution of means of 40 random exponentials, using the exponential distribution (rexp in R). This will be based on running 1000 simulations, where each simulation creates 40 random exponentials and takes their mean. We will then analyze the mean and variance of our simulated data (as compared to the theoretical values) and look at the distribution.

Simulations

We first create our 1000 simulations, taking the mean of 40 exponentials in each. For this analysis, we will use a lambda of 0.2 (the second argument to the rexp function).

```
## set the seed for the sake of reproducibility of this report
set.seed(22)
mns = replicate(1000, mean(rexp(40, 0.2)))
hist(mns, xlab = "Mean", main = "Histogram of Means of Simulated Data")
```



We can see that the sample means are centered around 5.

Sample Mean vs. Theoretical Mean

We now calculate the exact mean of our simulated data.

```
simMean <- mean(mns)
```

This results in a sample mean of **4.989191**. The expected, theoretical mean of the exponential distribution is 1/lambda. The lambda used in our simulation was 0.2, which results in an expected mean of **5**. Hence the mean of our simulated data is very close to the expected value.

Sample Variance vs. Theoretical Variance

Next, we calculate the variance of our simulated data.

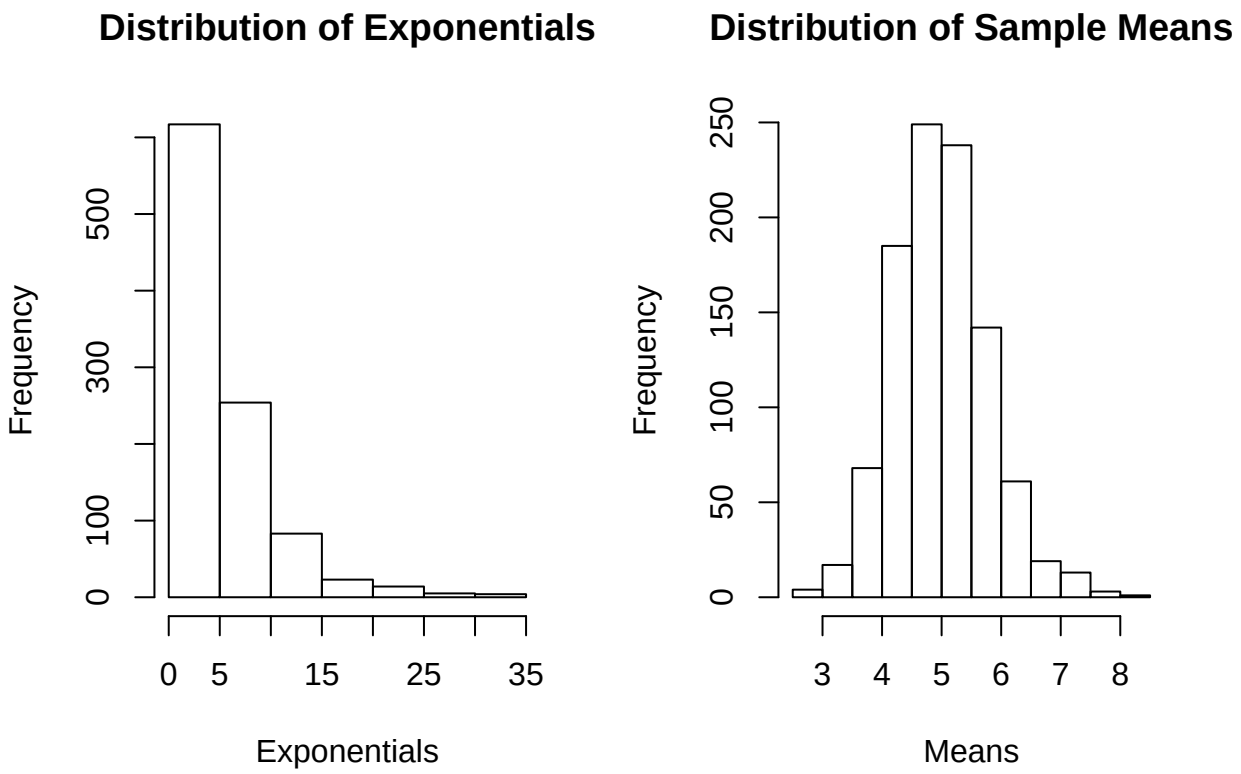
```
simVar <- var(mns)
```

This results in a sample variance of **0.6337023**. The expected, theoretical variance of the exponential distribution (taking into account the n = 40 draws from the population) is (1/n) * (1/lambda^2), which results in an expected variance of **0.625**. Once again, the variance of our simulated data is very close to the expected value.

Distribution

Finally, we want to see if the distribution of our sample means is approximately normal. We first show a side-by-side comparison of a plot of 1000 random exponentials (using our same lambda of 0.2) and the same histogram of our means of our simulations of 40 samples.

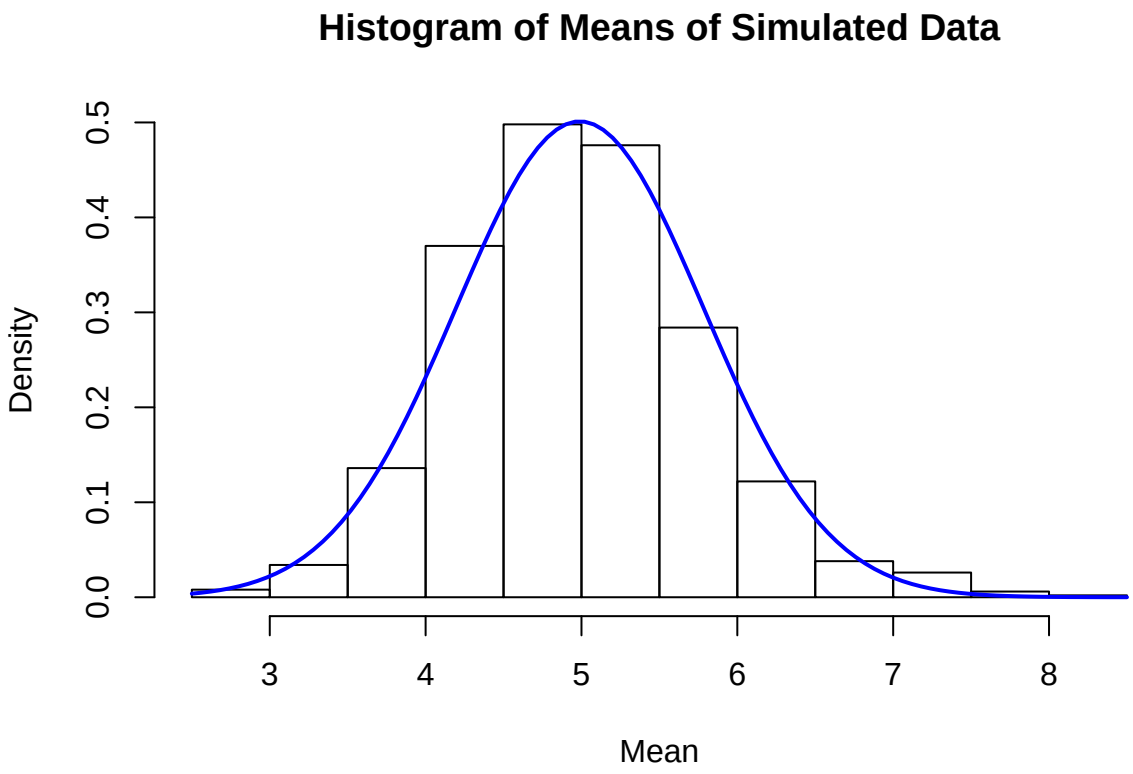
```
par(mfrow=c(1,2))
hist(rexp(1000, 0.2), xlab = "Exponentials", main = "Distribution of Exponentials")
hist(mns, xlab = "Means", main = "Distribution of Sample Means")
```



As expected, the first plot shows the frequency of values decreasing exponentially, with the majority of values near 5. Our second plot also shows the means centered around 5, however in this plot we see that our distribution of means takes on a Gaussian shape.

As an additional check, we plot the density of the simulated means in the figure below, and we have overlayed a normal curve.

```
hist(mns, xlab = "Mean", main = "Histogram of Means of Simulated Data", prob = TRUE)
curve(dnorm(x, mean = simMean, sd = sqrt(simVar)), col = "blue", add = TRUE, lwd = 2)
```



We can see that the distribution of our sample means is indeed approximately normal.

Summary

In summary, our analysis shows that the simulated data does behave as predicted by the CLT. Our mean and variance approach the theoretical values, and the distribution of our sample means is approximately normal.

Part 2

Overview

In Part 2 of this report, we will examine a fairly small dataset that shows tooth growth results from Vitamin C, using one of two delivery methods and given at three different doses. We will examine whether the data appears to be normally distributed, and if so, determine if the differences in resutls appear to be statistically significant.

Data Summary

We first load the ToothGrowth dataset and use the stat.desc() function to perform some initial analysis of the data.

```
library(datasets)
library(pastecs)
```

```
data("ToothGrowth")
head(ToothGrowth)
```

```
##    len supp dose
## 1  4.2   VC  0.5
## 2 11.5   VC  0.5
## 3  7.3   VC  0.5
## 4  5.8   VC  0.5
## 5  6.4   VC  0.5
## 6 10.0   VC  0.5
```

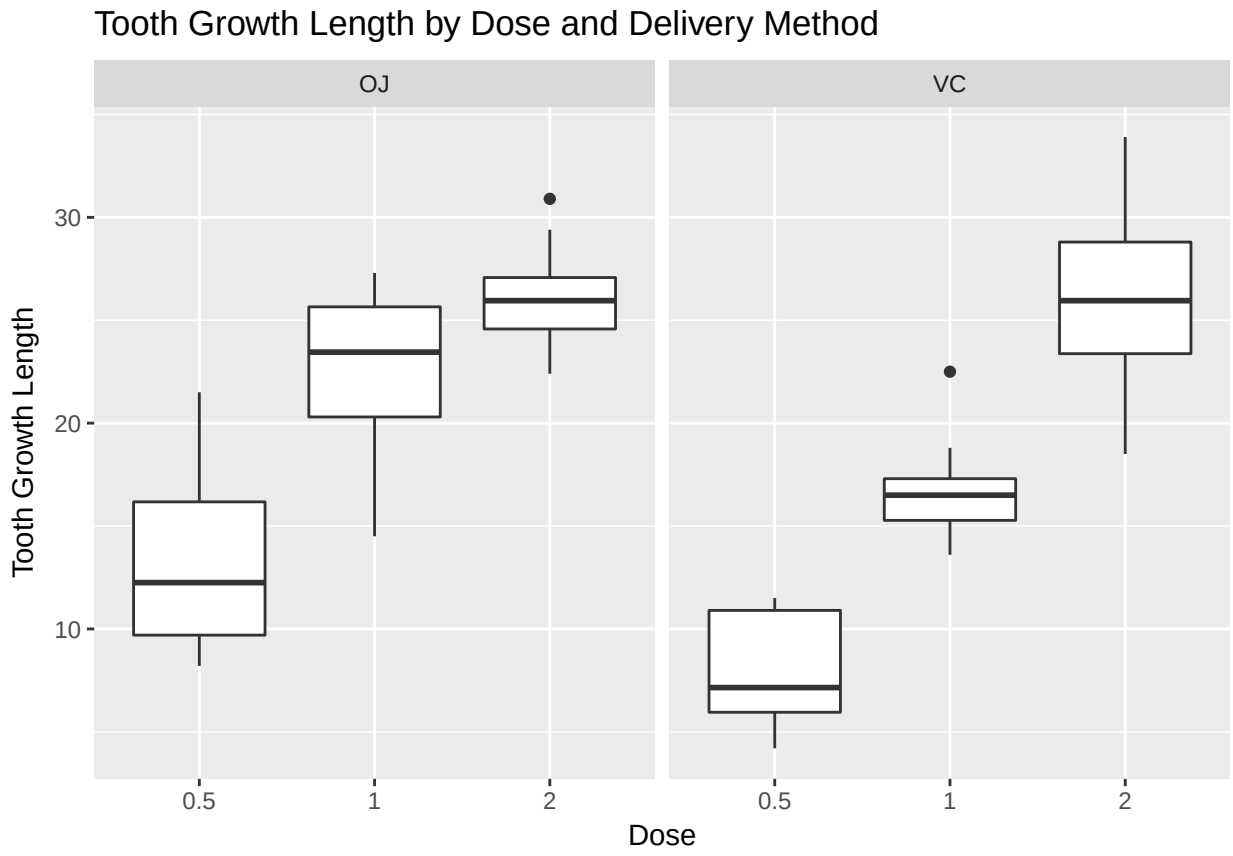
```
stat.desc(ToothGrowth, norm = TRUE)
```

```
##                      len supp          dose
## nbr.val        60.0000000   NA  6.000000e+01
## nbr.null        0.0000000   NA  0.000000e+00
## nbr.na          0.0000000   NA  0.000000e+00
## min             4.2000000   NA  5.000000e-01
## max            33.9000000   NA  2.000000e+00
## range          29.7000000   NA  1.500000e+00
## sum          1128.8000000   NA  7.000000e+01
## median         19.2500000   NA  1.000000e+00
## mean           18.8133333   NA  1.166667e+00
## SE.mean         0.9875223   NA  8.118705e-02
## CI.mean.0.95    1.9760276   NA  1.624549e-01
## var            58.5120226   NA  3.954802e-01
## std.dev         7.6493152   NA  6.288722e-01
## coef.var        0.4065901   NA  5.390333e-01
## skewness       -0.1425376   NA  3.722966e-01
## skew.2SE       -0.2308721   NA  6.030190e-01
## kurtosis       -1.0425144   NA -1.549583e+00
## kurt.2SE       -0.8566377   NA -1.273298e+00
## normtest.W      0.9674286   NA  7.649050e-01
## normtest.p      0.1091005   NA  1.990132e-08
```

Based on the Shapiro-Wilk normality test statistic p-value of 0.11 calculated for this dataset and an assumed significance level of 0.05, we will not reject the hypothesis that the data is normally distributed and can proceed with confidence estimates using a t-distribution.

We start by plotting the results by dose and delivery method. Dose values are 0.5, 1, and 2. Delivery methods are OJ and VC, which stand for Orange Juice and Ascorbic Acid, respectively.

```
library(ggplot2)
qplot(factor(dose), len, data = ToothGrowth, geom = "boxplot", facets = .~supp, xlab = "Dose", ylab = "
```



There doesn't appear to be significant differences by delivery method, but the dose does seem to affect the length of tooth growth. Let's further examine this using t-test confidence intervals.

Testing

We start by subsetting the data by dose so that we can then compare two at a time. We then compare the 0.5 and 1.0 doses. We look at the confidence interval, which will show with 95% confidence what the difference in the mean is between the two data sets.

```
dose05 <- subset(ToothGrowth, dose == 0.5)
dose1 <- subset(ToothGrowth, dose == 1.0)
dose2 <- subset(ToothGrowth, dose == 2.0)
t.test(dose05$len, dose1$len)$conf.int
```

```
## [1] -11.983781  -6.276219
## attr(,"conf.level")
## [1] 0.95
```

Since 0 is not in the interval, we can say with 95% confidence that the difference in the means between the 0.5 and 1.0 doses is significant (non-zero).

We continue the analysis by comparing the 1.0 and 2.0 dose data, as well as the 0.5 and 2.0 dose data (to cover all combinations, though logic tells us that there is a significant difference in the means between the 0.5 and 2.0 dose data.)

```r
t.test(dose1$len, dose2$len)$conf.int
```

```
## [1] -8.996481 -3.733519
## attr(,"conf.level")
## [1] 0.95
```

```r
t.test(dose05$len, dose2$len)$conf.int
```

```
## [1] -18.15617 -12.83383
## attr(,"conf.level")
## [1] 0.95
```

Once again, with both of these tests, 0 is not in the confidence interval, so the differences in the means between the doses appears to be significant (non-zero).