Johns Hopkins Coursera: Statistical Inference Peer Reviewed Project

Part 1: Simulation Exercise

Overview

In this project, we want to use simulation to make a discovery about the exponential distribution. An exponential distribution of 40 observances was simulated 1000 times. The simulated distribution had a mean of 5 and a standard deviation of 5. We then compare the sample mean to the theoretical mean, the sample variance and standard deviation to the theoretical variance and standard deviation, and finally show that the distribution of the mean is approximately normal through visual aids.

Simulation, Means, and Variances

```
library(ggplot2)
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.1.1
```

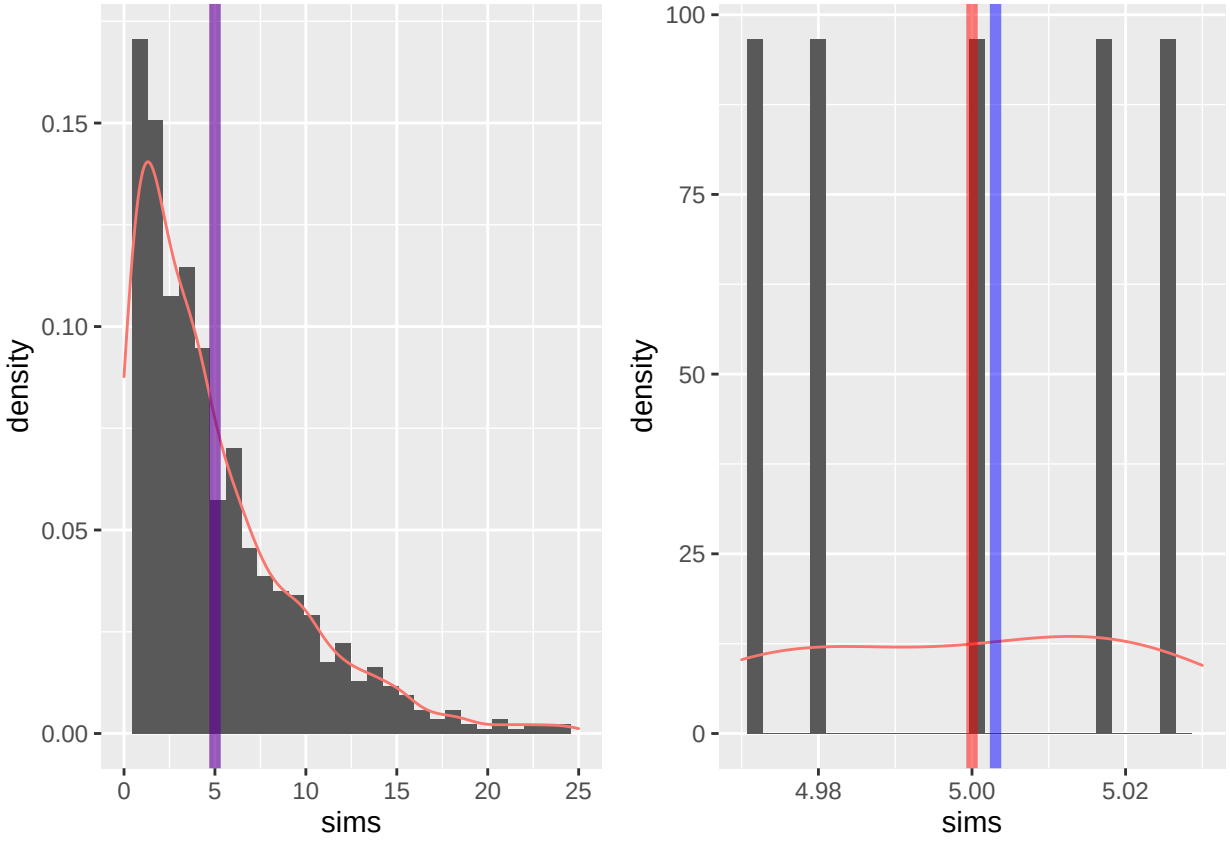
```
#Set seed for reproducibility.
set.seed(1234)

#Generate a dataframe of exponential values
sims <- data.frame("sims" = rexp(1000, rate = .2))

#The first plot containing the entire distribution
plt1 <- ggplot(data = sims, mapping = aes(x = sims, y = ..density..)) +
        geom_histogram() +
        geom_density(mapping = aes(colour = "red")) +
        geom_vline(xintercept = mean(sims$sims), color = "blue", alpha = .5, size = 2)+
        geom_vline(xintercept = 1/.2, color = "red", alpha = .25, size = 2) +
        xlim(0,25) +
        theme(legend.position = "none")

#The second plot, zooming into where the differences are.
plt2 <- ggplot(data = sims, mapping = aes(x = sims, y = ..density..)) +
        geom_histogram() +
        geom_density(mapping = aes(colour = "red")) +
        geom_vline(xintercept = mean(sims$sims), color = "blue", alpha = .5, size = 2)+
        geom_vline(xintercept = 1/.2, color = "red", alpha = .5, size = 2) +
        xlim(4.97,5.03) +
        theme(legend.position = "none")

#Plotting
plot_grid(plt1, plt2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 8 rows containing non-finite values (stat_density).
```

```
## Warning: Removed 2 rows containing missing values (geom_bar).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 995 rows containing non-finite values (stat_bin).
## Warning: Removed 995 rows containing non-finite values (stat_density).
## Warning: Removed 2 rows containing missing values (geom_bar).
```



On the left is the entire distribution, and on the right is the distribution zoomed in to see the lines diverge.

```
print(c("Means",1/.2, mean(sims$sims)))
```

```
## [1] "Means"           "5"               "5.0030668729407"
```

```
print(c("Standard Deviations",1/.2, sd(sims$sims)))
```

```
## [1] "Standard Deviations" "5"                   "5.05671826956297"
```

$\mu - \bar{x} = 0.003067$, % difference from $\mu = 0.0006134$ %

$\sigma - s = .05672$, % difference from $\sigma = 0.011344$ %

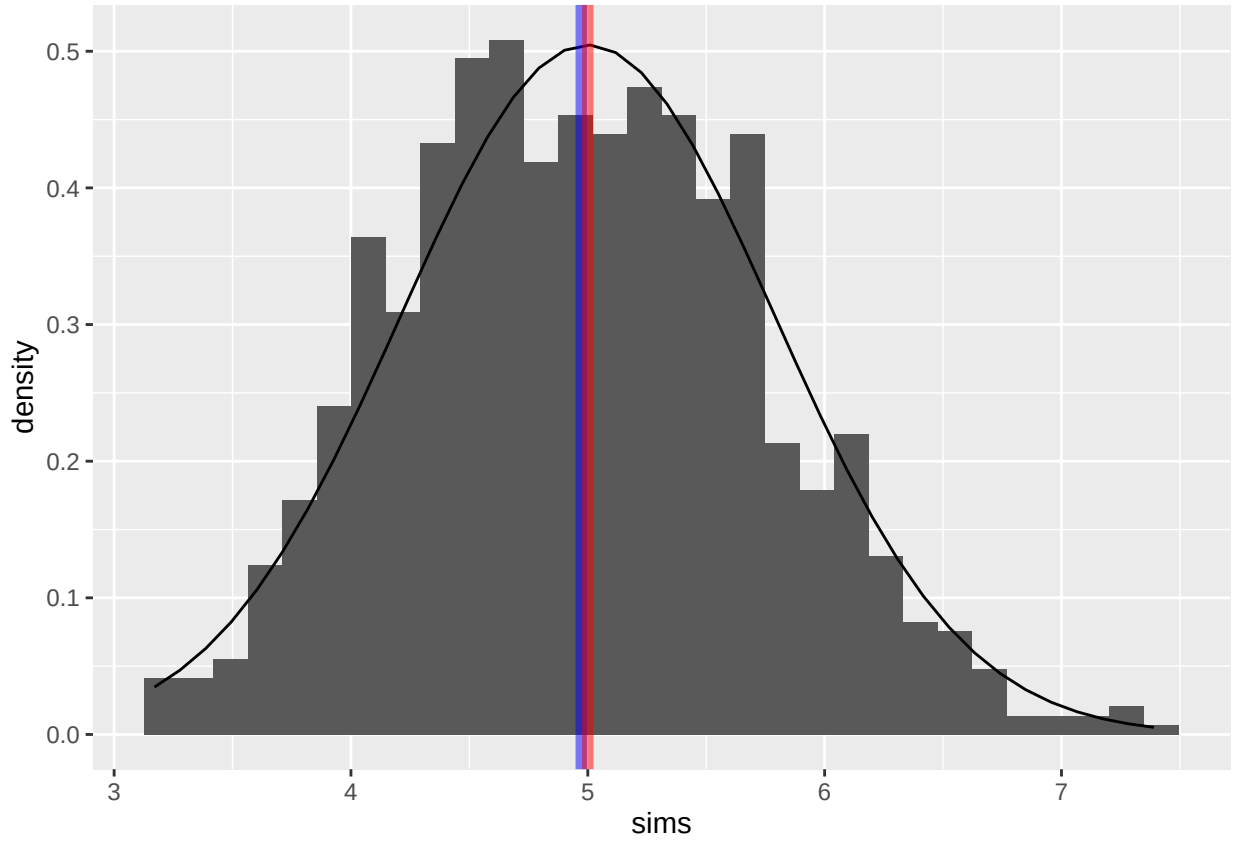
At n = 1000, we are over 98% accurate when measuring mean and variance. This is due to the Central Limit Theorem, which states that as n increases, our accuracy in approximating a distribution also increases.

Distribution

Now, let's generate 1000 sample distributions, each with 40 observations, and take the mean of each distribution, and plot them.

```r
#Initializing the list
means = NULL

#Declaring size externally for cleaner code
size = 40

#Generating the distributions 1000 times
for (i in 1:1000){
        means <- c(means, mean(rexp(size, rate = .2)))
}

#Writing the list to a dataframe
sims <- data.frame("sims" = means)

#Plotting the distribution of the means
ggplot(data = sims) +
        geom_histogram(mapping = aes(x = sims, y = ..density..)) +
        stat_function(fun = dnorm, n = 40, args = list(mean = 1/.2, sd = 1/.2/sqrt(40))) +
        geom_vline(xintercept = mean(sims$sims), color = "blue", alpha = .5, size = 2)+
        geom_vline(xintercept = 1/.2, color = "red", alpha = .5, size = 2) +
        theme(legend.position = "none")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
print(c("Mean", mean(sims$sims)))
```

```
## [1] "Mean"             "4.97231912836865"
```

```
print(c("Standard Deviation", sd(sims$sims)))
```

```
## [1] "Standard Deviation" "0.76022251264432"
```

The means appear to follow a normal distribution, but how can we check?

We know a few things:

$\mu = 1/\lambda$

$\sigma = \sqrt{(1/\lambda^2)}$

Given: $\lambda = .2$

We can check against what we already know about normal distributions

1. $\mu = 1/\lambda = 5$, $\bar{X} = 4.972$
2. $\sigma = .79057$, $s = .7602$

Discussion

- Means: Our simulated mean was almost perfectly in line with the population mean, due to the central limit theorem
- Variance: As n increases, the variance in our sample mean distribution will decrease.

Part 2: Analysis of the Tooth Growth Dataset

The requirements of this section are: * Provide a basic summary of the tooth growth dataset * Compare tooth growth by supp and dose(using only techniques we learned during the course) * State conclusions and assumptions

The ToothGrowth Dataset

The ToothGrowth dataset was published in *The Statistics of Bioassay* by C.I. Bliss in 1952.

From the description in the datasets library:

> The response is the length of odontoblasts (cells responsible for tooth growth) in 60 guinea pigs. Each animal received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods, orange juice or ascorbic acid (a form of vitamin C and coded as VC).

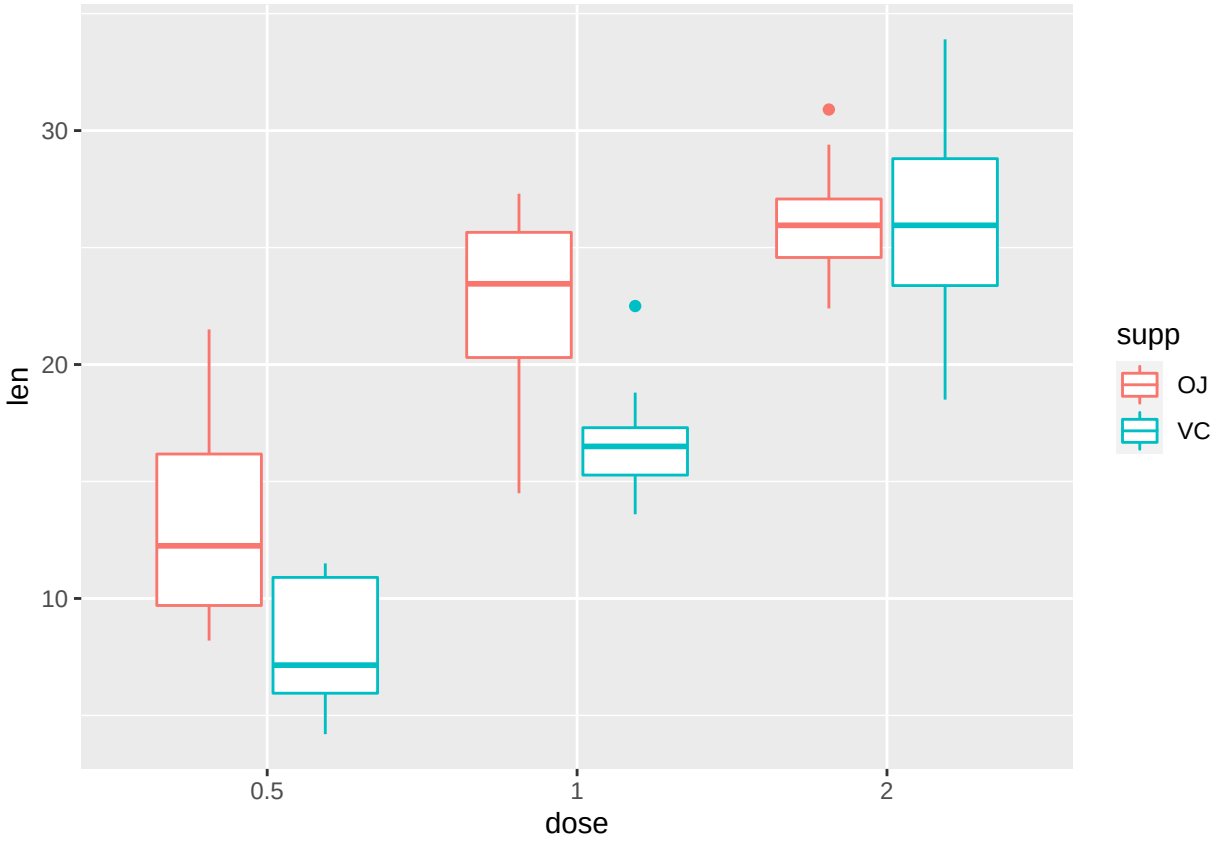
```
#Read in data
library(datasets)
teeth <- ToothGrowth
#Converting the dose to a factor variable
teeth$dose <- as.factor(teeth$dose)
str(teeth)
```

```
## 'data.frame':    60 obs. of  3 variables:
##  $ len : num  4.2 11.5 7.3 5.8 6.4 10 11.2 11.2 5.2 7 ...
##  $ supp: Factor w/ 2 levels "OJ","VC": 2 2 2 2 2 2 2 2 2 2 ...
##  $ dose: Factor w/ 3 levels "0.5","1","2": 1 1 1 1 1 1 1 1 1 1 ...
```

Exploratory Analysis

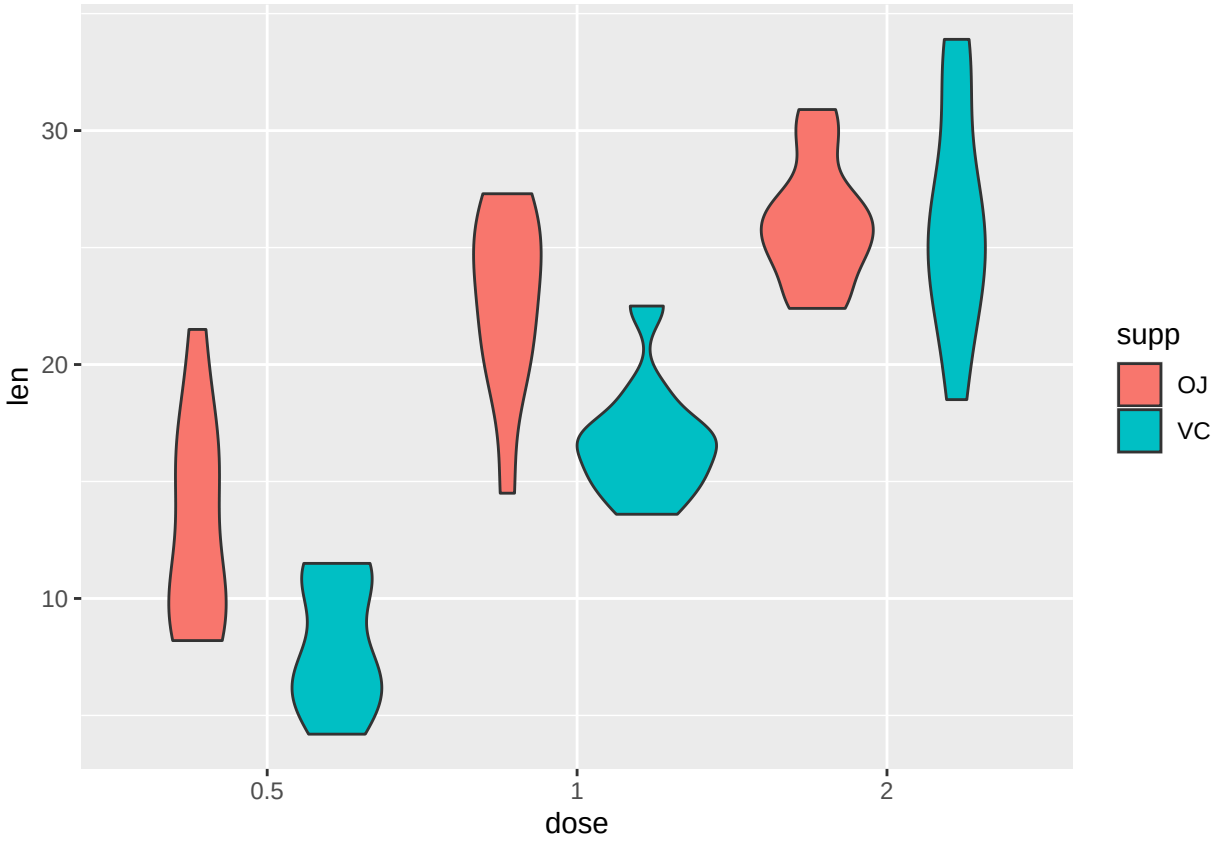
Let's start with a boxplot of the len by dose and supp type.

```
library(ggplot2)

ggplot(teeth, mapping = aes(x = dose, y = len, colour = supp)) + geom_boxplot()
```



It generally looks like the cells performed better on Orange juice until the dose became 2mg. It also generally appears higher doses are correlated to higher response lengths of the cells.

```
ggplot(teeth, mapping = aes(x = dose, y = len, fill = supp)) + geom_violin()
```



This is just another way to visualize the boxplot. I think it gives it a bit more gravitas as to the scale of the means, and just how spread out the boxs are at certain points.

Hypothesis Testing

Power First, let's calculate our power, assuming we want 90% power, at a 1 delta, at our current standard deviation.

```
std_dev = sd(teeth$len)
power.t.test(n = 30, delta = 1, sd = std_dev, sig.level = .95, type = "two.sample")
```

```
##
##      Two-sample t test power calculation
##
##               n = 30
##           delta = 1
##              sd = 7.649315
##       sig.level = 0.95
##           power = 0.6713349
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

.67 isn't great, but we have two different dials we can turn, delta and significance level. I would like to keep the .95 significance, so let's turn up the delta and see what we can land at.

```
power.t.test(n = 30, delta = 2, sd = std_dev, sig.level = .95, type = "two.sample")
```

```
##
##      Two-sample t test power calculation
##
##               n = 30
##           delta = 2
##              sd = 7.649315
##       sig.level = 0.95
##           power = 0.8289213
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

At a delta of 2, which is a little more than a quarter of a standard deviation, we have a power of .82, which is acceptable.

T-tests Two t-test will be deployed to help us understand our data. The first will to test the null hypothesis that a change in dosage does not change cell response. For this, we will use the dosage at .5 and 2

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
less_vitamins <- teeth %>%
        subset(dose == .5)

more_vitamins <- teeth %>%
        subset(dose == 2)

t.test(more_vitamins$len, less_vitamins$len, paired = FALSE, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  more_vitamins$len and less_vitamins$len
## t = 11.799, df = 38, p-value = 2.838e-14
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  12.83648 18.15352
## sample estimates:
## mean of x mean of y
##    26.100    10.605
```

From the output, we can see that the p value is incredibly small (and therefore significant). H_a was that the difference in means was not equal, however it might be more apt to test if a higher dose yields higher cell response.

```
t.test(more_vitamins$len, less_vitamins$len, paired = FALSE, var.equal = TRUE, alternative = "greater")
```

```
##
```

```
##  Two Sample t-test
##
## data:  more_vitamins$len and less_vitamins$len
## t = 11.799, df = 38, p-value = 1.419e-14
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  13.28093      Inf
## sample estimates:
## mean of x mean of y
##    26.100    10.605
```

It can be concluded that a higher dosage yields a higher mean.

Now, let's test if there exists a difference in cell response between orange Juice and ascorbic acid as the delivery method.

```
library(tidyselect)

oj <- teeth %>%
        filter(supp == "OJ")

vc <- teeth %>%
        filter(supp == "VC")

t.test(oj$len, vc$len, paired = FALSE, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  oj$len and vc$len
## t = 1.9153, df = 58, p-value = 0.06039
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1670064  7.5670064
## sample estimates:
## mean of x mean of y
##  20.66333  16.96333
```

The p-value is greater than .05, and the 95% confidence interval contains 0. Now we need to bring back power into the discussion. We have enough power to NOT be able to reject the null hypothesis that the means of the two groups are not different.

Other analysis

It would be reasonable to use an ANOVA and a linear model to attempt to predict if the delivery method or dosage had an effect, and be able to measure that effect. Unfortunantly, that is outside the project scope.