Duncan McKinnon

West

HW 4

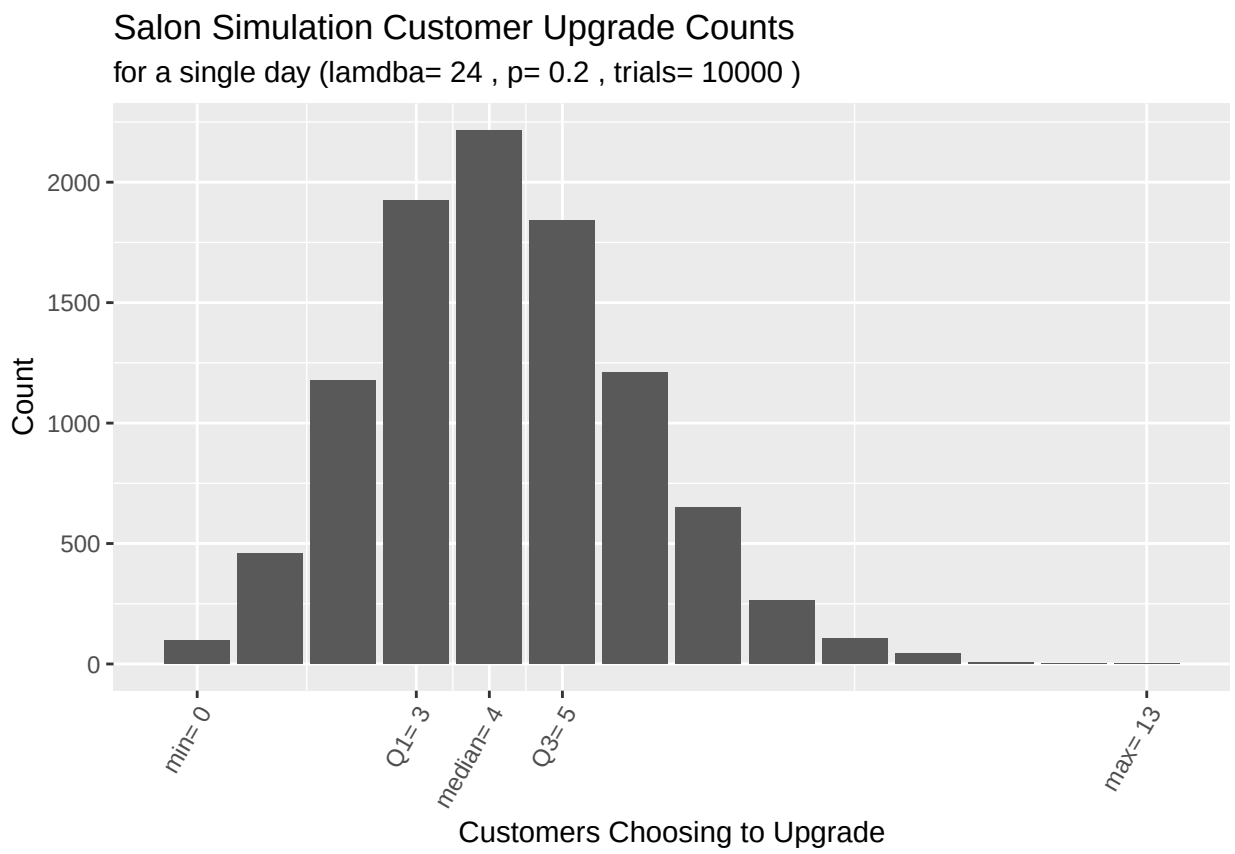
QA).

```
#load libraries
suppressPackageStartupMessages({
  library(tidyverse)
  library(mosaic)
  library(broom)
})
seed = 1947

# function to run simulation and sensitivity analysis
# with different #'s for lamdba, probability and iterations
simulate <- function(lambda, p, n, seed = 1947) {

  # set random sampling seed value
  set.seed(seed)

  # create simulation of number of customers choosing to upgrade each day for n days
  sim <- rbinom(n, rpois(1, lambda), p)

  # summarize distribution and save in table
  sum <- glance(summary(sim))

  # set label values for plot
  labels <- c(
    paste('min=', sum$minimum),
    paste('Q1=', sum$q1),
    paste('median=', sum$median),
    paste('Q3=', sum$q3),
    paste('max=', sum$maximum)
  )

  # create histogram plot of simulation results with summary labels
  ggp <- ggplot( as.data.frame(sim) ) +
    geom_histogram( aes(x = sim),
      stat = 'count',
      binwidth = 1) +
    labs( title = 'Salon Simulation Customer Upgrade Counts',
      subtitle = paste('for a single day (lamdba=', lambda, ', p=', p, ', trials=',n,')'),
      x = 'Customers Choosing to Upgrade',
      y = 'Count') +
    scale_x_continuous( breaks = unlist(sum[c('minimum', 'q1', 'median', 'q3', 'maximum')]),
      labels = labels) +
    theme( axis.text.x = element_text(angle = 60, hjust = 1) )
```

```
  # return list of simulation results
  return(list(simulation = sim, summary = sum, plot = ggp))
}
```

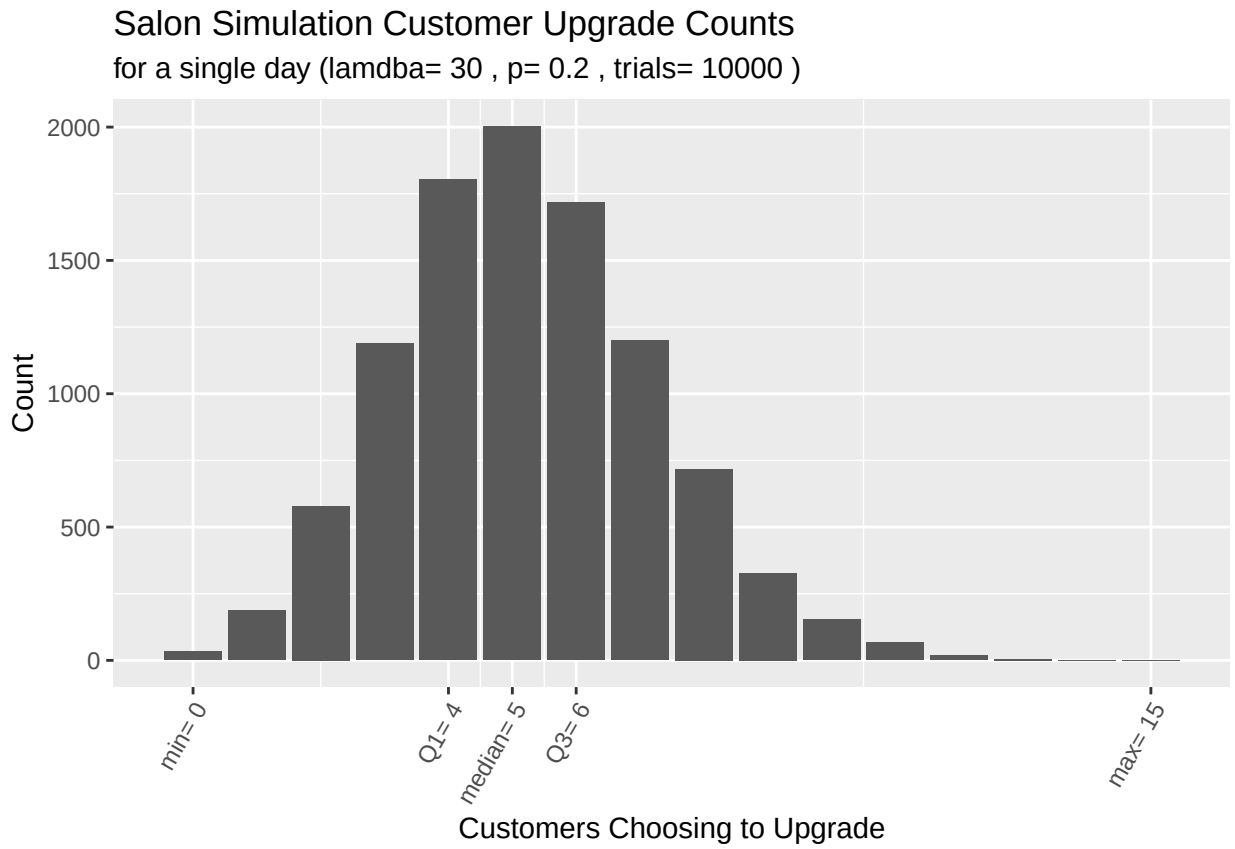
1).

```
# set parameters for simulating population
s1 <- simulate(24, 0.2, 10000)
s1$plot
```



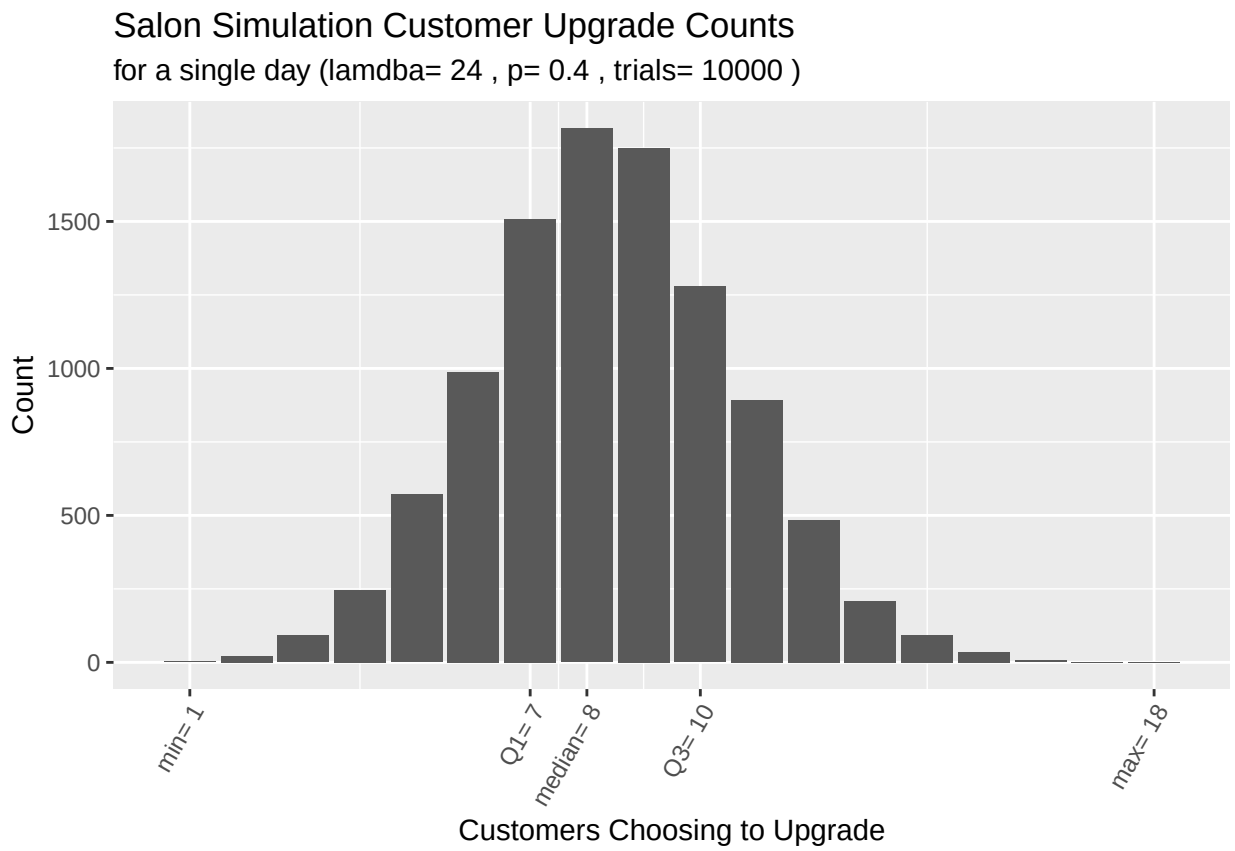
2).

```
# update parameters for sensitivity analysis
s2 <- simulate(30, 0.2, 10000)
s2$plot
```



3).

```
# update parameters for sensitivity analysis
s3 <- simulate(24, 0.4, 10000)
s3$plot
```



4).

```
# get the probability of having 120 customers in a day,
# given that the daily customer distribution follows a poisson distribution with mean 24
dpois(120, 24)
```

```
## [1] 2.381703e-44
```

5).

Although the poisson distribution has no upper limit and the salon's customer capacity does have an upper limit, the lambda value chosen for the distribution is low enough that the probability of it generating a number of customers that is above the daily capacity of the salon is much less than 1 in a billion. If anything, the owner should be concerned that extremely high customer counts are hardly ever present in this model, while in reality there is the distinct possibility of having many more customers than expected during a single day, (especially if a promotion is successful).

QB).

```
L <- function(x) { return(( min( x ) + max( x ) ) / 2 ) }
```

```
# find a gamma distribution with right skew, mean ~ 60 and P(q > 100) ~ 0.15
```

```r
# gamma distributions with shape / scale == 60 will have mean 60
shape <- c(1, 2, 3, 6, 10, 12, 15, 20, 30, 60, 120)
stats <- array(dim = c(1, length(shape)), dimnames = list('P(x >= 100)'))
colnames(stats) <- shape

# get the percentage of the distribution that is > 100 for each set of distribution parameters
for(i in 1 : length(shape)){
  n <- shape[i]
  stats[1, i] <- 1 - pgamma(100, n, n/60)
}

stats
```

```
##                      1         2        3          6        10         12
## P(x >= 100) 0.1888756 0.1545873 0.124652 0.06708596 0.0310044 0.02138682
##                     15          20           30           60           120
## P(x >= 100) 0.01240206 0.005115981 0.0009168289 6.341503e-06 4.005161e-10
```

```r
# find the value of shape that corresponds to a gamma distribution where
# rgamma(n, shape, shape/60) -> P( x > 100 ) ~ 0.15
shape_val <- shape[ abs(( stats * 100 ) - 15) < 1 ]
shape_val
```

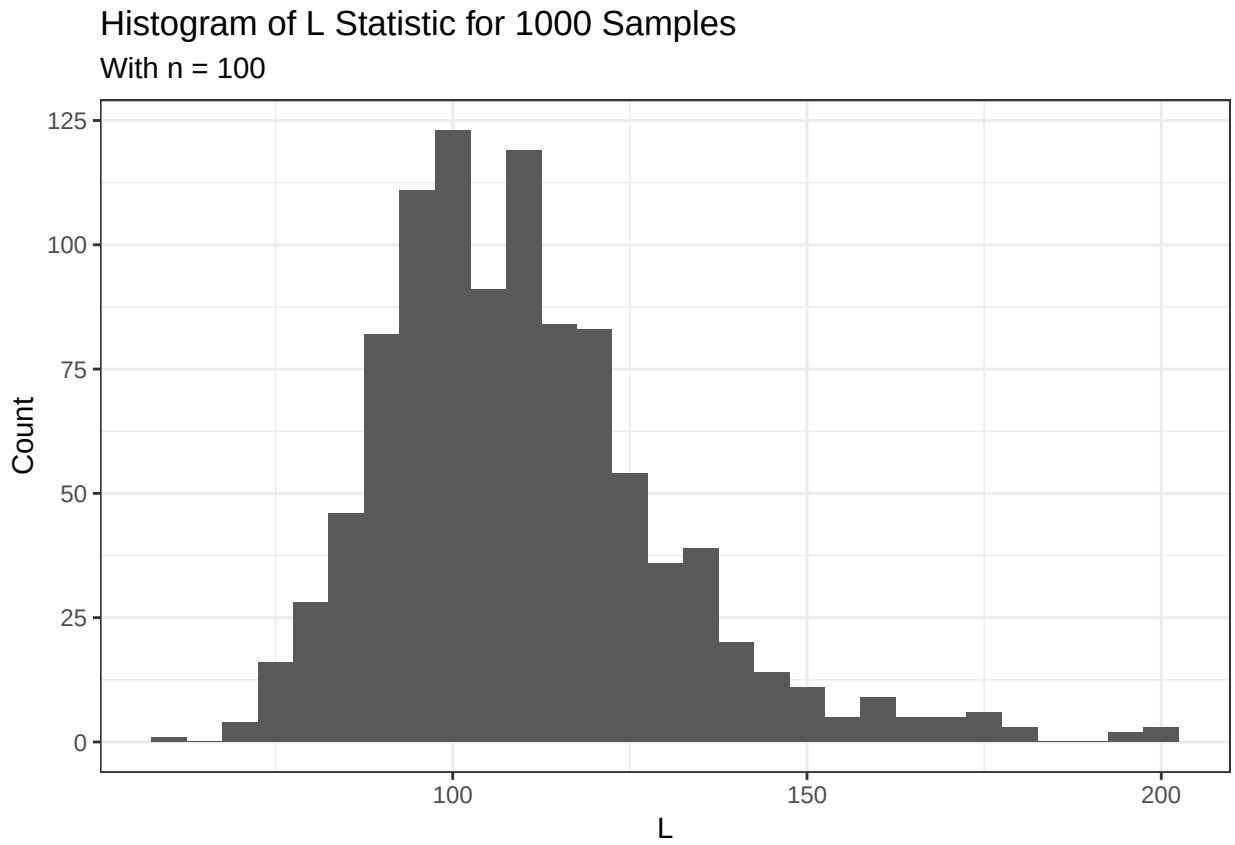
```
## [1] 2
```

1).

```r
# set random sampling seed value
set.seed(seed)

# create the simulated population with 100,000 values
sim <- as.data.frame ( rgamma(100000, shape_val, shape_val / 60) )
```

2).

```r
# set random sampling seed value
set.seed(seed)

# get 1000 samples of 100
samples <- replicate(1000, sim %>% sample_n(100))

# get the L statistic for each sample
L_stat <- map_dbl(samples, L)

# plot the histogram of L stats from each sample
ggplot() +
  geom_histogram(aes(x = L_stat), binwidth = 5) +
  labs( title = 'Histogram of L Statistic for 1000 Samples',
      subtitle = 'With n = 100',
      x = 'L',
      y = 'Count') +
  theme_bw()
```




```
# summarize the distribution of the L stat
favstats(L_stat)
```

```
##       min       Q1   median       Q3      max     mean       sd    n
##  61.01339 95.06631 107.4603 119.7404 198.7998 109.7661 20.26204 1000
##  missing
##        0
```

3).

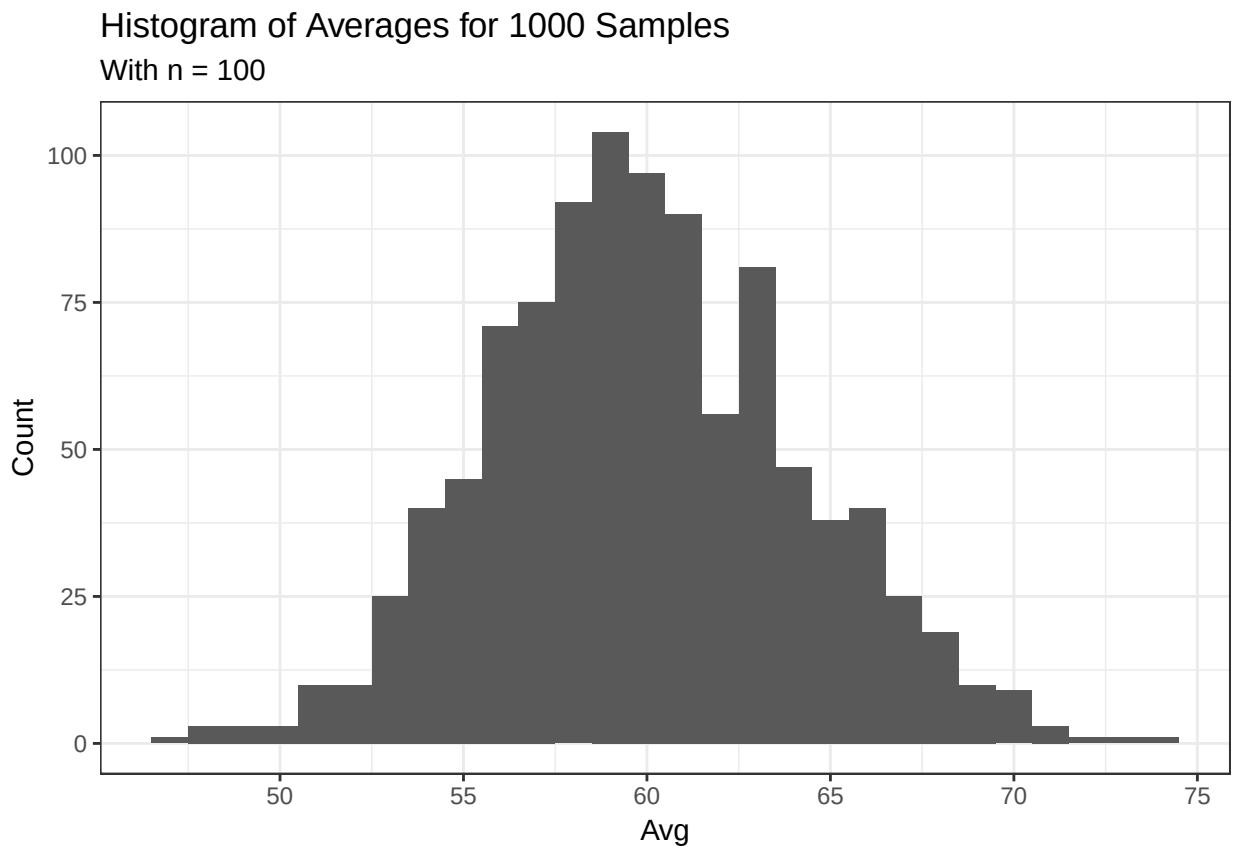
```
# calculate the standard error of the L stat
se <- sd( L_stat ) / sqrt(length( L_stat ))
se
```

```
## [1] 0.6407419
```

4).

```
# get the mean for each sample
mu_stat <- map_dbl(samples, mean)

# plot the histogram of means from each sample
ggplot() +
  geom_histogram(aes(x = mu_stat), binwidth = 1) +
  labs( title = 'Histogram of Averages for 1000 Samples',
```

```
      subtitle = 'With n = 100',
      x = 'Avg',
      y = 'Count') +
  theme_bw()
```



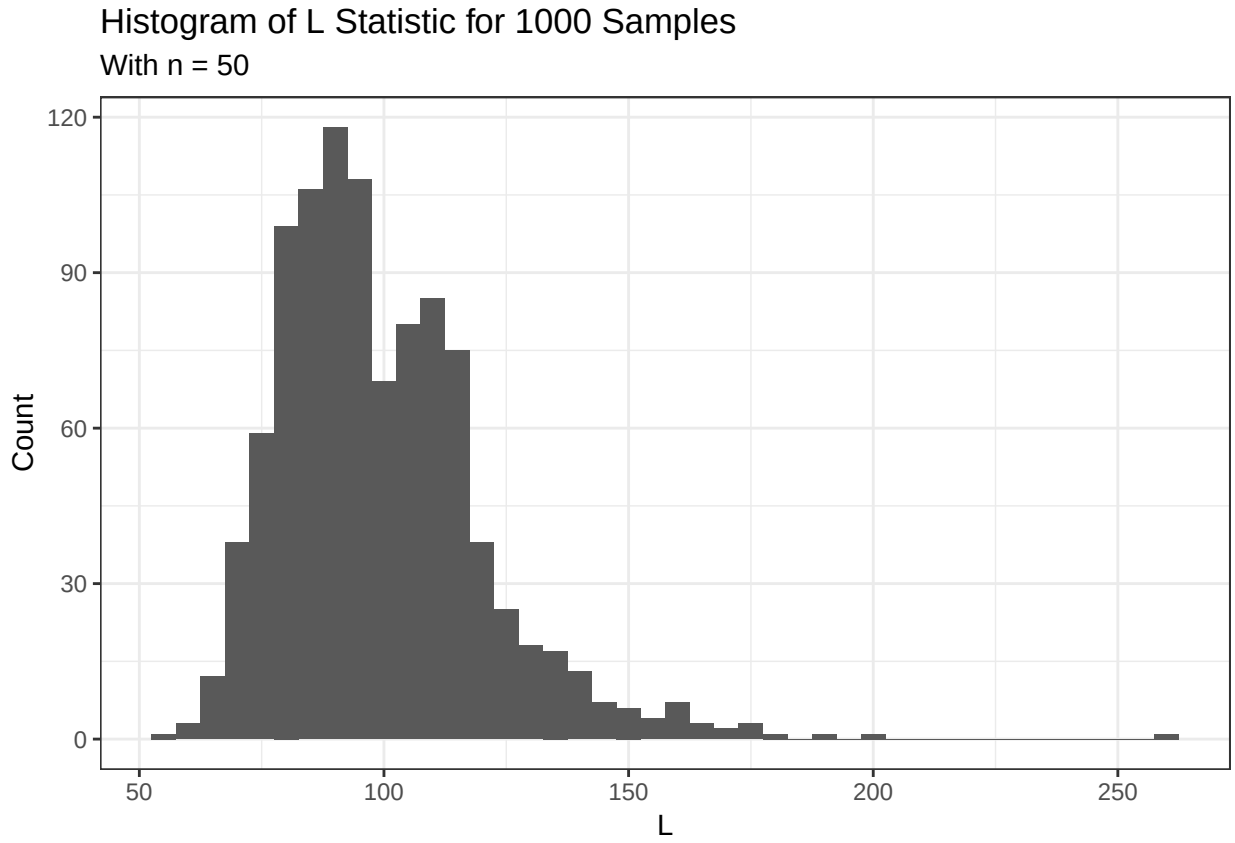

```
# summarize the distribution of the mean
favstats(mu_stat)
```

```
##       min       Q1  median       Q3      max     mean       sd    n
##  47.43793 56.96586 59.6567 62.75043 73.80533 59.92928 4.237507 1000
##  missing
##        0
```

5).

```
# set random sampling seed value
set.seed(seed)

# get 1000 samples of 50
samples50 <- replicate(1000, sim %>% sample_n(50))

# get the L statistic for each sample
L_stat50 <- map_dbl(samples50, L)

# plot the histogram of L stats from each sample
```

```
ggplot() +
  geom_histogram(aes(x = L_stat50), binwidth = 5) +
  labs( title = 'Histogram of L Statistic for 1000 Samples',
      subtitle = 'With n = 50',
      x = 'L',
      y = 'Count') +
  theme_bw()
```




```
# summarize the distribution of the L stat
favstats(L_stat50)
```

```
##       min       Q1   median       Q3      max     mean       sd    n
##  57.26805 84.41481 95.87627 110.9666 258.2429 99.19862 20.95059 1000
##  missing
##        0
```

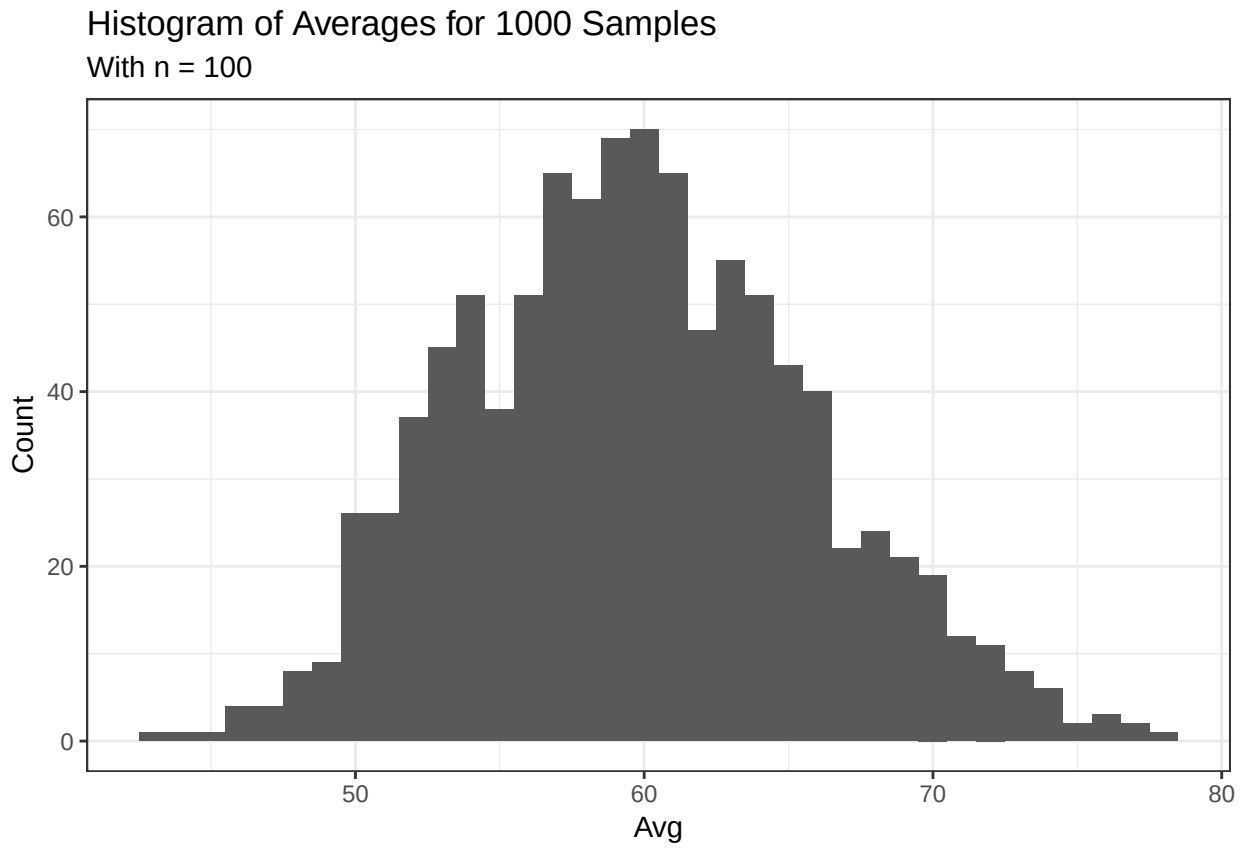
6).

```
# calculate the standard error of the L stat
se <- sd( L_stat50 ) / sqrt(length( L_stat50 ))
se
```

```
## [1] 0.6625158
```

7).

```
# get the mean for each sample
mu_stat50 <- map_dbl(samples50, mean)

# plot the histogram of means from each sample
ggplot() +
  geom_histogram(aes(x = mu_stat50), binwidth = 1) +
  labs( title = 'Histogram of Averages for 1000 Samples',
      subtitle = 'With n = 100',
      x = 'Avg',
      y = 'Count') +
  theme_bw()
```




```
# summarize the distribution of the mean
favstats(mu_stat50)
```

```
##       min       Q1   median       Q3      max    mean       sd    n
##  42.94775 55.46189 59.55411 63.83138 77.65045 59.7498 6.003243 1000
##  missing
##        0
```

8).

The L statistic provides a different perspective on the distribution of the data, but the information it captures is very limited. Using the maximum and minimum values in the statistic focuses on the values that correlate

the least with the mean of the distribution, so any correlation between the mean and the L stat can be attributed to random variation. The L statistic is also going to be right-skewed, while the distribution of means should be normal. Because there is a hard limit on the minimum value in the distribution (0), very large values of the maximum should occur with much greater frequency than equally small values of the minimum. Since the L statistic depends on the very highest and lowest values in any given sample, this skew effect should dominate the distribution.

The results of the simulation using the L statistic show this right skewed effect. In the charts of the distribution of the L statistic with sample n of 50 the distribution high is more than 3 times the mean, while the low is more than half the mean, meaning the distribution overall is extremely right skewed. This effect was slightly smaller with the larger sample size, but still very present. In both simulations, the mean of the distribution of the L statistic was close to 100, while the mean of the population overall was 60. Any relation between the L statistic mean and the sample mean is difficult to discern, especially given that the statistic have such different distributions.

In summary, the L statistic might be useful in understanding relations among the values with the greatest variance from the mean, but any information about the mean and distribution of the sample is lost in the random variability of maximum and minimum values.