BEE552 Biometry Week 3

Maria Feiler

2/9/2022

My Learning Journey

Over the last week, I participated in Biometry in the following ways:

- I asked / answered **4** questions posed in class.
- I asked **4** questions in Slack.
- I answered **0** questions posed by other students on Slack.
- I came to Heather's office hours: **No**
- I came to Jose's office hours: **Yes**
- I met with Heather or Jose separately from office hours: **No**

Anything not falling into one of the above categories?

> **No**

On a scale of 1 (no knowledge) to 10 (complete expert), how would I rate my comfort with R programming after this week?

> **5**

Any topics from last week that you are still confused about?

> **Doing fine for now**

Problem Set

Question 1

Let X be a discrete random variable whose pdf is described in the table given here:

x	f(x)
-1	1/8
0	6/8
1	1/8

Find the following:

a. $E[X] = x_1 f(x_1) + x_2 f(x_2) + x_3 f(x_3) = -1(1/8) + 0(6/8) + 1(1/8) = 0$

b. $P(X = 0) = 6/8$

c. $P(X \leq 1) = P(X = -1) + P(X = 0) + P(X = 1) = 1$

d. $F(1) = P(X \leq 1) = 1$

e. $F^{-1}(7/8) = 0$

Question 2

X and Y are independent random variables. Write the following expressions in terms of E[X] and E[Y]

a.

$$\begin{aligned} E\left[\frac{X+35}{10}\right] &= E\left[\frac{X}{10} + \frac{35}{10}\right] \\ &= E\left[\frac{X}{10}\right] + \frac{35}{10} \\ &= \frac{1}{10}E[X] + \frac{35}{10} \end{aligned}$$

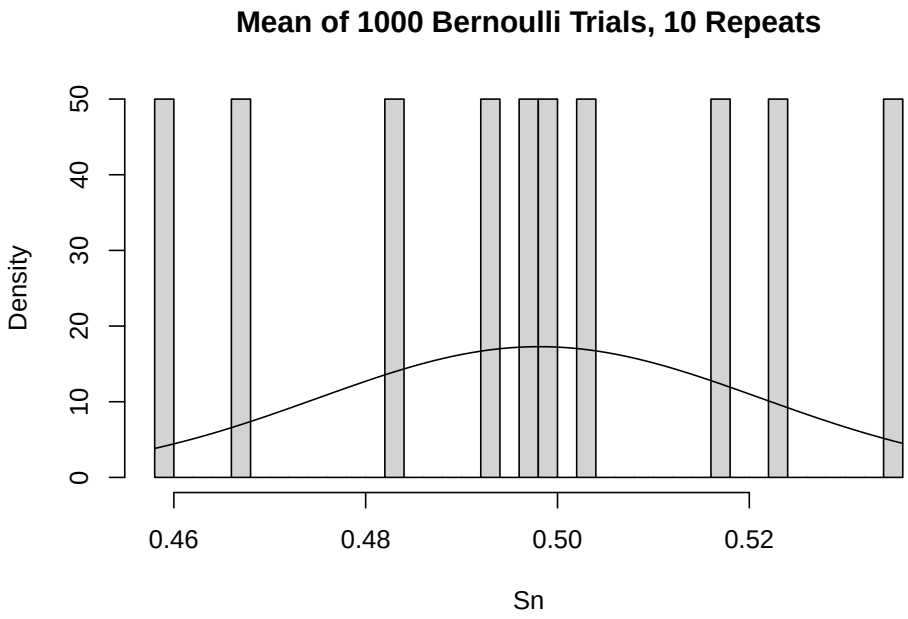
b.

$$\begin{aligned} E[X - 14Y + E[Y] + 7] - E[X + Y + 5] &= E[X] - E[14Y] + E[Y] - E[X] - E[Y] - 12 \\ &= 2 - 14E[Y] \end{aligned}$$

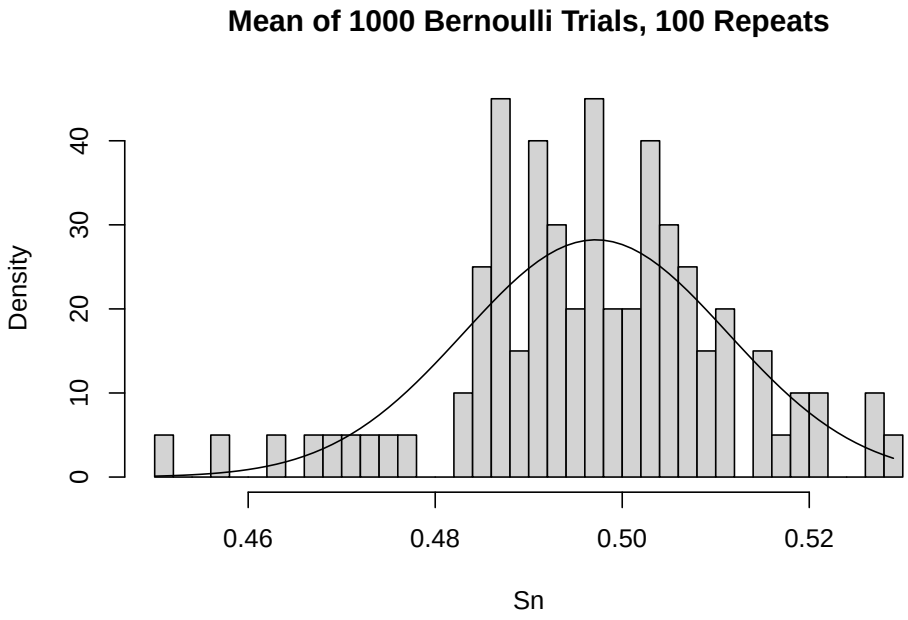
Question 3

Use R to convince yourself of the Central Limit Theorem using draws from any distribution that is not the Normal distribution. Briefly (2-3 sentences) explain your process and provide your code as well.

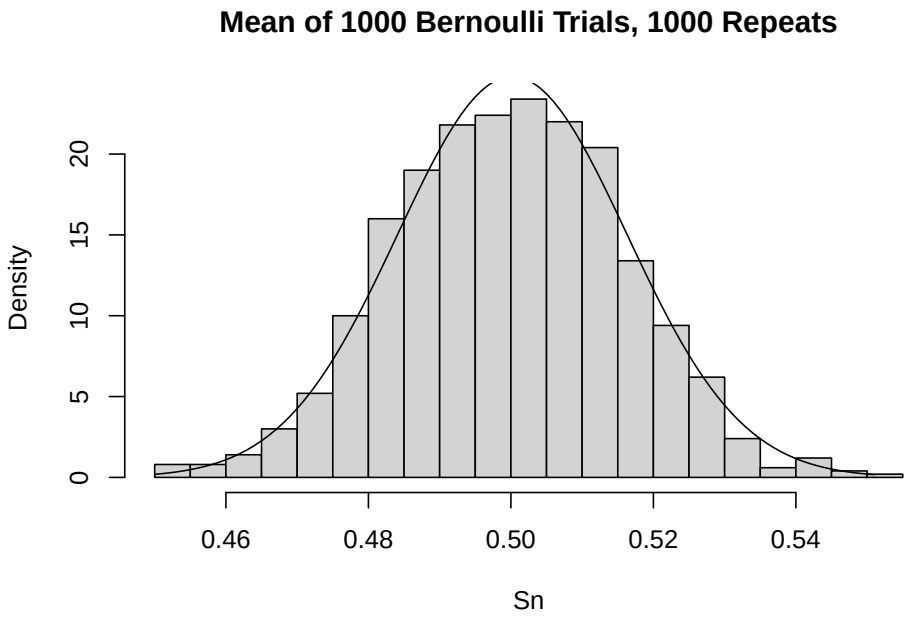
```
# Define the test iterations of 1000 Bernoulli trials to be run
# Code only shows the first test, but replace n[#] with the desired value in
# nit to produce the second, third, and fourth histograms.
nit <- c(10, 100, 1000, 10000)

# Matrix to catch n[#] iterations of 1000 coin flips
bern <- matrix(0,
               nrow = nit[1],
               ncol = 1000)

# Run the iterations
for (i in 1:nit[1]){
        bern[i,] <- rbinom(n = 1000,
                           size = 1,
                           prob = 0.5)
}

# Get the mean of each iteration
Sn <- rowMeans(bern)

# Fit the means to a normal distribution
fit <- fitdistr(Sn, "normal")

# Define x values to plot the pdf over
xVals <-  seq(from = min(Sn),
               to = max(Sn),
               by = 0.001
               )

# Make histogram
hist(x = Sn,
     # To plot density so the pdf is scaled properly
     freq = FALSE,
     breaks = 30,
     main = "Mean of 1000 Bernoulli Trials, 10 Repeats"
     )

# Plot pdf
lines(xVals, dnorm(xVals,
                   mean = fit$estimate[1],
                   sd = fit$estimate[2]
                   )
      )
```



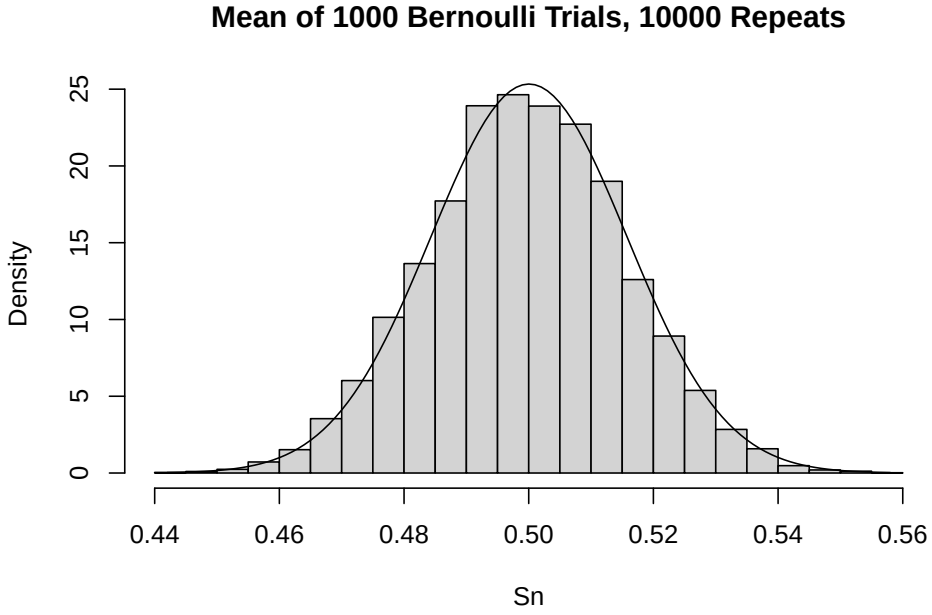

```
The log likelihood of the normal distribution N(0.498, 0.023) is 23.
```




```
The log likelihood of the normal distribution N(0.497, 0.014) is 284.
```




```
The log likelihood of the normal distribution N(0.5, 0.016) is 2711.
```




```
The log likelihood of the normal distribution N(0.5, 0.016) is 27322.
```

After running these Bernoulli tests with more and more iterations, the log likelihood values show that the normal distribution fits better the more iterations used. Using repeated Bernoulli trials, which about are as far from Normal as you can get, to produce a normal distribution of means convinces me of the Central Limit Theorem.

Question 4

Assume that leaf biomass (in grams) from the plant Salix arctica can be described as having the following probability density function:

$$f(x) = \frac{2}{(x+1)^3} for x > 0$$

a. Another way of saying that this pdf is restricted to $x > 0$ is to say that there is no support for $x \leq 0$. Why would this be the case here?

If you are measuring a mass, you cannot have a "negative" mass. Mass is entirely additive and nonnegative in the same way that height is. Therefore, there is no support for any PDF at $x < 0$ because it is impossible to have a probability for negative mass, something that cannot exist.

b. Prove (using the two requirements for a valid pdf) that this is a valid pdf.

Requirement 1: Everywhere non-negative. Given the condition $x > 0$, there is no instance where this pdf will be negative.

Requirement 2: Integrates to 1.

$$\begin{aligned}
P(0 < x < 3) &= \int_0^{\infty} \left(\frac{2}{(x+1)^3}\right)\, dx \\
&= 2\int_0^{\infty} \left(\frac{1}{u^3}\right)\, du \quad where\ u = x+1 \\
&= 2\int_0^{\infty} (u^{-3})\, du \\
&= 2\left(\frac{u^{-3+1}}{-2+1}\right) \\
&= 2\left(\frac{u^{-2}}{-2}\right) \\
&= -u^{-2} \\
&= \frac{-1}{(x+1)^2} \quad solved\ over\ 0\ to\ \infty \\
&= \frac{-1}{(\infty+1)^2} - \left(-\frac{1}{(0+1)^2}\right) \quad \frac{-1}{(\infty+1)^2}\ approaches\ 0 \\
&= 1
\end{aligned}$$

c. Manually (i.e., using calculus) calculate the probability that a leaf has biomass between 0 g and 3 g. In other words, find $P(0 < X < 3)$.

$$
\begin{aligned}
P(0 < x < 3) &= \int_0^3 \left(\frac{2}{(x+1)^3} \right) \, dx \\
&= 2 \int_0^3 \left(\frac{1}{u^3} \right) \, du \quad \textit{where } u = x + 1 \\
&= 2 \int_0^3 (u^{-3}) \, du \\
&= 2 \left(\frac{u^{-3+1}}{-2+1} \right) \\
&= 2 \left(\frac{u^{-2}}{-2} \right) \\
&= -u^{-2} \\
&= \frac{-1}{(x+1)^2} \quad \textit{solved over } 0 \textit{ to } 3 \\
&= \frac{-1}{(3+1)^2} - \left(-\frac{1}{(0+1)^2} \right) \\
&= -1/16 + 1 \\
&= 15/16 \\
&= 0.9375
\end{aligned}
$$

d. Use the integrate function in R to confirm this result numerically.

```
integrate(f = function(x){2/(x+1)^3},
          lower = 0,
          upper = 3
          )
```

```
0.9375 with absolute error < 5.3e-08
```

Question 5

Bliss and R.A. Fisher (1953) examined female European red mite counts (Panonychus ulmi) on McIntosh apple trees [Malus domestica [McIntosh]). Counts of the mites on 150 leaves are shown here:

Mites per Leaf	0	1	2	3	4	5	6	7	8
Leaves Observed	70	38	17	10	9	3	2	1	0

a. What is the expected value E[X] of mites per leaf in this sample (show your work, either an R script or a calculation)?

```
# Define vectors of the mites per leaf and number of leaves observed
mites <- c(0, 1, 2, 3, 4, 5, 6, 7, 8)
leaves <- c(70, 38, 17, 10, 9, 3, 2, 1, 0)

# Make vector of the mites observed on each leaf
obs <- c()

for (i in 1:length(mites)) {
        temp <- rep(mites[i], leaves[i])
        obs <- c(obs, temp)
}
```

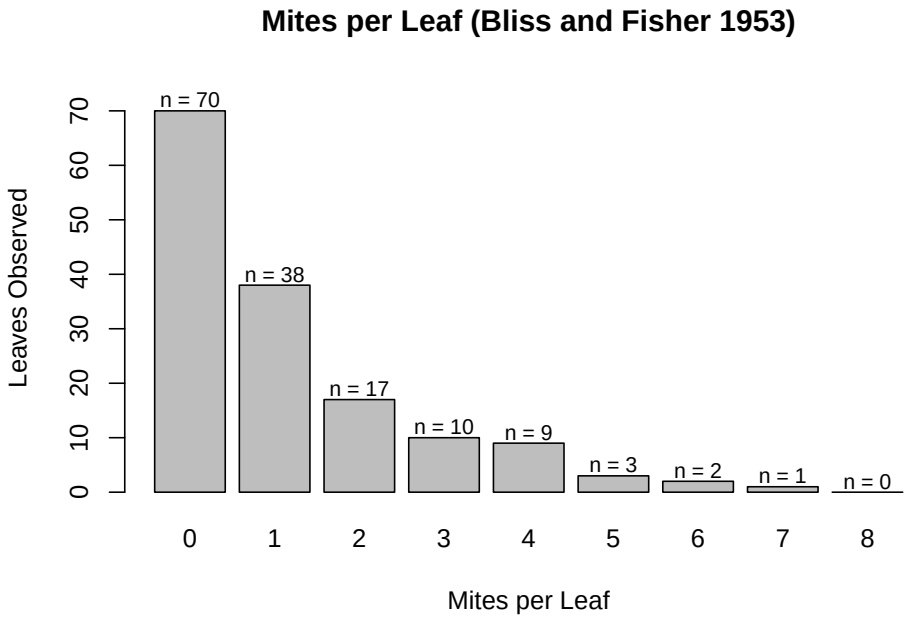
```
# Expected value (E[X]) is the mean of all the observations
mean(obs)
```

```
lambda = 1.15
```

b. Assume for the moment that Bliss and Fisher used a Poisson distribution to describe the number of mites per leaf X ($X \sim Pois(\lambda)$). What would be the most reasonable value for the Poisson parameter λ and why?

> Lambda (λ) for a Poisson distribution is defined as the mean of the observations. As the sample size increases, λ will converge on a normal distribution, or $\lim_{x \to \infty} (X \sim Pois(\lambda)) \to N(\lambda, \lambda)$, and as a result λ will converge on $E[X]$. Therefore, $\lambda = 1.15$.

c. Make a barplot (the R function histogram should work, but as these are discrete variables, we would normally refer to this as a bar plot) of Bliss and Fisher's data. Is the expected value also the most common value? Why or why not?



The expected value is not the most common value because the expected value is not an integer. Since this distribution is defined by discrete observations, the expected value 1.15 will not be represented by the data at all. If we were to round to the nearest integer (1), it still does not represent the most common value (0). This is because it is impossible to get a mean of 0 if there are any observations that are greater than zero.

d. What is the standard deviation of the number of mites per leaf? What is the standard error of the mean (SEM) number of mites per leaf? In one sentence, describe the interpretation of the standard error of the mean?

```
# To calculate the standard deviation
sd(obs)

# To calculate the standard error of the mean
sd(obs)/sqrt(length(obs))
```

```
The standard deviation of the mites per leaf is 1.507862.

The standard error of the mean mites per leaf is 0.1231164.
```

The standard error is a measure of how accurate of a representation the mean of a sample from a given population is of the true population mean; the larger the standard error, the less likely your sample is representative of the true population.

e. To gain better intuition for the meaning of the standard error of the mean, create 1000 new datasets by sampling with replacement from the original dataset. Use those 1000 simulated datasests to calculate the standard error of the mean. Because of sampling error, this will not be exactly the same as what you calculated in part d, but it should be close.

```
# Define number of iterations
nit = 1000

# Create matrix to catch the 1000 trials
bootObs <- matrix(data = NA,
                  nrow = nit,
                  ncol = length(obs),
                  dimnames = list(c(1:nit),
                                  c(1:length(obs))
                                  )
                )

# Create 1000 more datasets sampling with replacement
for (i in 1:nit) {
        bootObs[i,] <- sample(x = obs,
                              size = length(obs),
                              replace = TRUE
                              )
}
```

```
# Calculate the standard error of the mean of the bootstrap datasets
sd(rowMeans(bootObs))
```

```
The standard error of the mean mites per leaf is 0.1206135.
```

Question 6

We are going to get some practice using these statistical distributions in a biological context, and in the process gain some extra practice writing R code. Read Viswanathan et al. (2008) for the biological context of this problem, but don't worry too much about the mathematical details (you can skim over Section 4).

This question is designed to get you writing some more sophisticated R scripts, with more emphasis on loops and logic, but there are some statistical and biological goals as well. Before getting to the actual question, I want to emphasize the following "take-home" messages:

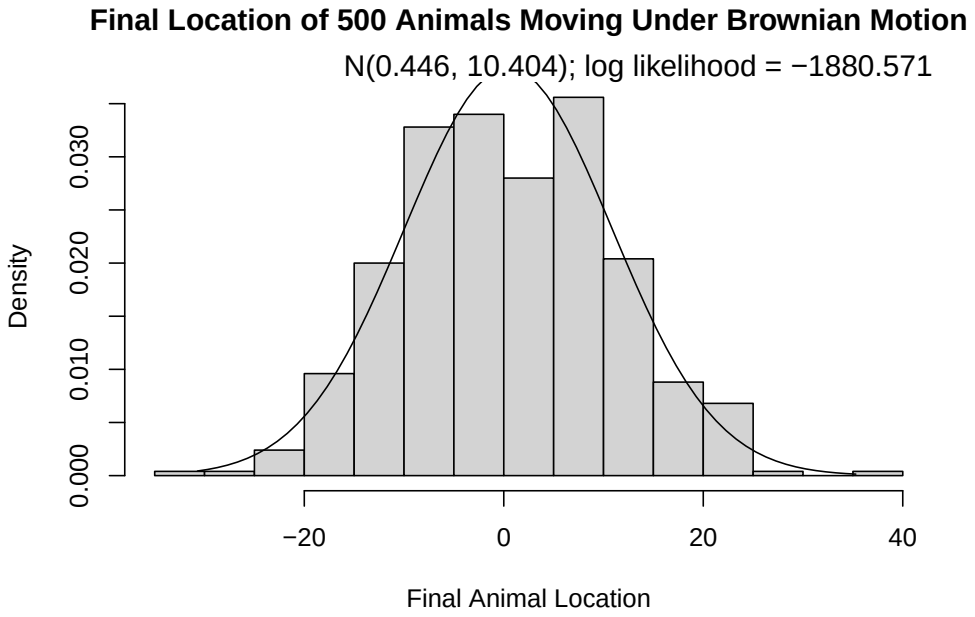
1. *There are many, many statistical distributions. We have learned only a small subset of all the statistical distributions. Long-tailed distributions (such as the L'evy, Cauchy, and Pareto) have many uses in ecology and evolution.*
2. *Simulations provide a straightforward way of studying complex stochastic phenomena.*
3. *Understanding 'pattern' goes a long way to understanding the underlying 'process'. Statistical distributions provide a way to quantify the pattern of your data. In many cases, only certain biological or ecological mechanisms can generate those patterns.*

Write a short script to simulate the movement of animals operating according to the movement rules described below. For simplicity, we will assume only one-dimensional motion along a line. Animals can move forward, or backwards, along a line. For each animal, simulate 500 individuals each moving for 100 time steps each.

Animal 1 *This animal moves by "Brownian" motion. In each time step, the individual chooses a random direction (50% probability of moving forward or backwards [this is a Binomial process!]) and moves a distance given by (the absolute value of)* $N(\mu = 0, \sigma^2 = 1)$. *Cut and paste your script and a histogram of the final location for each of the 500 individuals. Fit a normal distribution to that distribution - what is* $\hat{\mu}$ *and* $\hat{\sigma^2}$*?*

```
# Define function that describes the movement of this animal over 100 steps
# Steps default value = 100 as defined by question, but other values can be
# substituted as needed
animal1 <- function(steps = 100){
        # Vector of random distances, mean and sd as defined by N(mu = 0, sigma = 1)
        distance <- abs(rnorm(steps,
                              mean = 0,
                              sd = 1)
                        )

        # Vector to catch the true distances traveled (accounting for length and
        # direction, where moving right is a positive movement and moving left
        # is a negative movement)
        result <- c()

        # Loop to determine distance and direction of each step
        for (i in 1:steps){
                # Conducting a Bernoulli trial to determine right (success)
                # or left (failure), then appending the distance traveled based
                # on the ith distance in the previously defined vector of
                # normally distributed random distances
                if(rbinom(n = 1, size = 1, prob = 0.5) == 1) {
                        result[i] <- distance[i]
                }
                else{
                        result[i] <- -distance[i]
                }
        }

        # Vector with distances the animal traveled
        return(result)
}

# Run the animal1() function for 500 animals
simAnimal1 <- matrix(data = 0,
                     nrow = 500,                    # Number of animals
                     ncol = 100,                    # Number of steps
                     dimnames = list(seq(1, 500),
                                     seq(1, 100)
                                     )
                     )

for (i in 1:nrow(simAnimal1)){
        simAnimal1[i,] <- animal1()
}

# Get the final location of each animal
finalAnimal1 <- rowSums(simAnimal1)
```

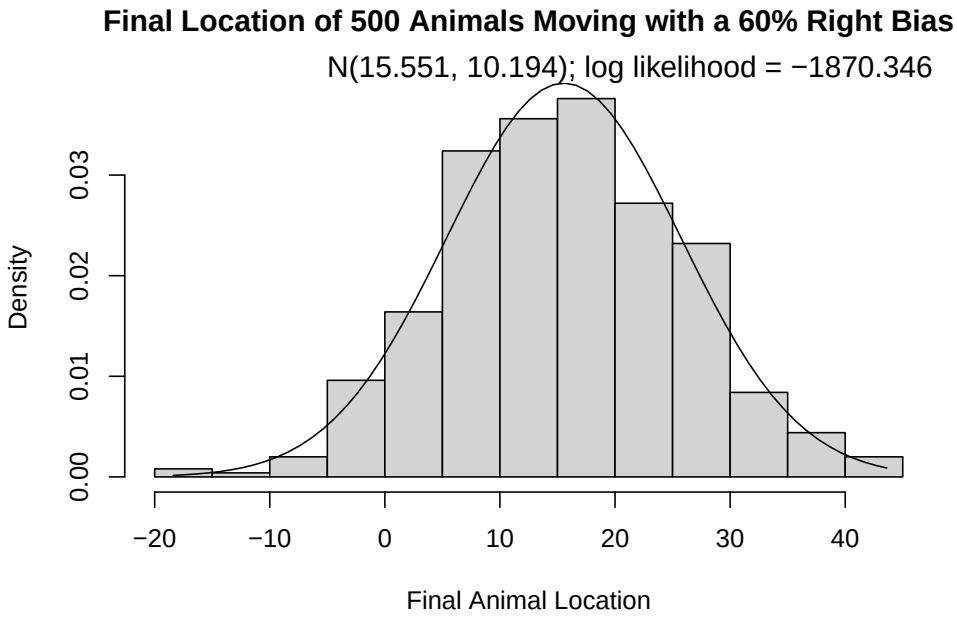
```r
# Fit normal distribution to the animal final locations
fit <- fitdistr(finalAnimal1, "normal")
```



Animal 2 *This animal is similar to Animal #1 but it preferentially moves to the right. In each time step, there is a 60% chance the animal will move to the right, the distance moved is still given by* $N(\mu = 0, \sigma^2 = 1)$. *Cut and paste your script and a histogram of the final location for each of the 500 individuals. Fit a normal distribution to that distribution - what is* $\hat{\mu}$ *and* $\hat{\sigma^2}$ *now? Note that this is still a random walk, albeit one with drift.*

```
# Define function that describes the movement of this animal over 100 steps
# Steps default value = 100 as defined by question, but other values can be
# substituted as needed
animal2 <- function(steps = 100){
        # Vector of random distances, mean and sd as defined by N(mu = 0, sigma = 1)
        distance <- abs(rnorm(steps,
                              mean = 0,
                              sd = 1)
                        )

        # Vector to catch the true distances traveled (accounting for length and
        # direction, where moving right is a positive movement and moving left
        # is a negative movement)
        result <- c()

        # Loop to determine distance and direction of each step
        for (i in 1:steps){
                # Conducting a Bernoulli trial to determine right (success)
                # or left (failure), assuming a 60% chance that the animal will
                # choose right instead of left, then appending the distance
                # traveled based on the ith distance in the previously defined
                # vector of normally distributed random distances
                if(rbinom(n = 1, size = 1, prob = 0.6) == 1) {
                        result[i] <- distance[i]
                }
                else{
                        result[i] <- -distance[i]
                }
        }

        # Vector with distances the animal traveled
        return(result)
}

# Run the animal1() function for 500 animals
simAnimal2 <- matrix(data = 0,
                     nrow = 500,                    # Number of animals
                     ncol = 100,                    # Number of steps
                     dimnames = list(seq(1, 500),
                                     seq(1, 100)
                                     )
                     )

for (i in 1:nrow(simAnimal2)){
        simAnimal2[i,] <- animal2()
}

# Get the final location of each animal
```

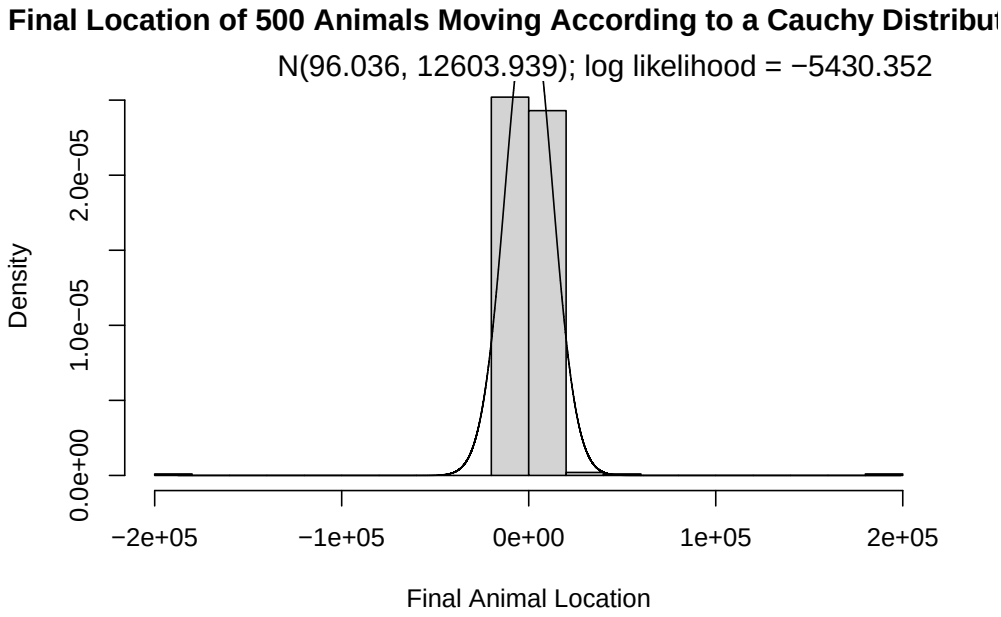
```r
finalAnimal2 <- rowSums(simAnimal2)

# Fit normal distribution to the animal final locations
fit <- fitdistr(finalAnimal2, "normal")
```



Animal 4 *This animal has no preference for direction, so has a 50/50 chance of moving left or right at each time step. However, distances travelled are now NOT normally distributed, but are governed by the Cauchy distribution with location=0 and scale=2 [Hint: Use 'rcauchy']. The Cauchy distribution is 'pathological' in that is has no mean and no variance; in fact, none of the moments of the Cauchy are defined. The Cauchy is one of the so-called 'heavy-tailed' distributions, in that the probability of very large values declines more slowly than the exponential distribution. (Translation: Very large numbers are more likely with heavy-tailed distributions.) In this context, the Cauchy distribution will yield a lot of small moves and some very large moves. Cut and paste your script and a histogram of the final location for each of the 500 individuals. Is the resulting distribution still normally distributed? If not, how does the final distribution of individuals differ from a normal? Why?*

```
# Define function that describes the movement of this animal over 100 steps
# Steps default value = 100 as defined by question, but other values can be
# substituted as needed
animal3 <- function(steps = 100){
        # Vector of random distances, mean and sd as defined by N(mu = 0, sigma = 1)
        distance <- abs(rcauchy(steps,
                                location = 0,
                                scale = 2)
                        )

        # Vector to catch the true distances traveled (accounting for length and
        # direction, where moving right is a positive movement and moving left
        # is a negative movement)
        result <- c()

        # Loop to determine distance and direction of each step
        for (i in 1:steps){
                # Conducting a Bernoulli trial to determine right (success)
                # or left (failure), then appending the distance traveled based
                # on the ith distance in the previously defined vector of
                # normally distributed random distances
                if(rbinom(n = 1, size = 1, prob = 0.5) == 1) {
                        result[i] <- distance[i]
                }
                else{
                        result[i] <- -distance[i]
                }
        }

        # Vector with distances the animal traveled
        return(result)
}

# Run the animal1() function for 500 animals
simAnimal3 <- matrix(data = 0,
                     nrow = 500,                    # Number of animals
                     ncol = 100,                    # Number of steps
                     dimnames = list(seq(1, 500),
                                     seq(1, 100)
                                     )
                     )

for (i in 1:nrow(simAnimal3)){
```

```
        simAnimal3[i,] <- animal3()
}

# Get the final location of each animal
finalAnimal3 <- rowSums(simAnimal3)

# Fit normal distribution to the animal final locations
fit <- fitdistr(finalAnimal3, "normal")
```



Since the log likelihood of a normal distribution is -5430.352, this animal does not exhibit as normally distributed movements as Animal 1 and Animal 2. This is probably due to the nature of the Cauchy distribution that, wherein there are higher likelihoods of very large movements. It differs from normal in that the spread of the data is skewed (very long negative tail, no positive tail at all) and very wide. It's almost a "negative" logarithmic distributio in shape.