P_M6_1

October 31, 2021

1 Module 6 Peer Review Assignment

2 Problem 1

Suppose X and Y are independent normal random variables with the same mean μ and the same variance σ^2. Do the random variables $W = X + Y$ and $U = 2X$ have the same distribution? Explain.

No they do not have the same distribution. First, $W \sim N(2\mu, 2\sigma^2)$ since we're adding X and Y with the same distribution. However, $U \sim (2\mu, 4\sigma^2)$ We know the variance of U is $4\sigma^2$ because $Var(aU) = a^2 Var(U)$. Thus, $Var(2X) \Rightarrow 4Var(X)$.

3 Problem 2: Central Limit Theorem and Simulation

a) For this problem, we will be sampling from the Uniform distribution with bounds [0, 100]. Before we simulate anything, let's make sure we understand what values to expect. If $X \sim U(0, 100)$, what is $E[X]$ and $Var(X)$?

$$E[X] = \frac{100 + 0}{2} = 50$$
$$Var[X] = \frac{(100 + 0)^2}{2} = 5000$$

b) In real life, if we want to estimate the mean of a population, we have to draw a sample from that population and compute the sample mean. The important questions we have to ask are things like:

- Is the sample mean a good approximation of the population mean?
- How large does my sample need to be in order for the sample mean to well-approximate the population mean?

Complete the following function to sample n rows from the $U(0, 100)$ distribution and return the sample mean. Start with a sample size of 10 and draw a sample mean from your function. Is the estimated mean a good approximation for the population mean we computed above? What if you increase the sample size?

[116]:
```
uniform.sample.mean = function(n){

    # Your Code Here
    sample.mean = mean(runif(n, min=0, max=100))
    return(sample.mean)
}
set.seed(12093)
uniform.sample.mean(10)
```

43.2234968408011

[117]:
```
uniform.sample.mean(100)
```

48.1945376440417

Using a sample size of 10, the mean is not a good representation of the population mean. Once we increase the sample size from 10 to 100, the sample mean improves, but still isn't close to 50.

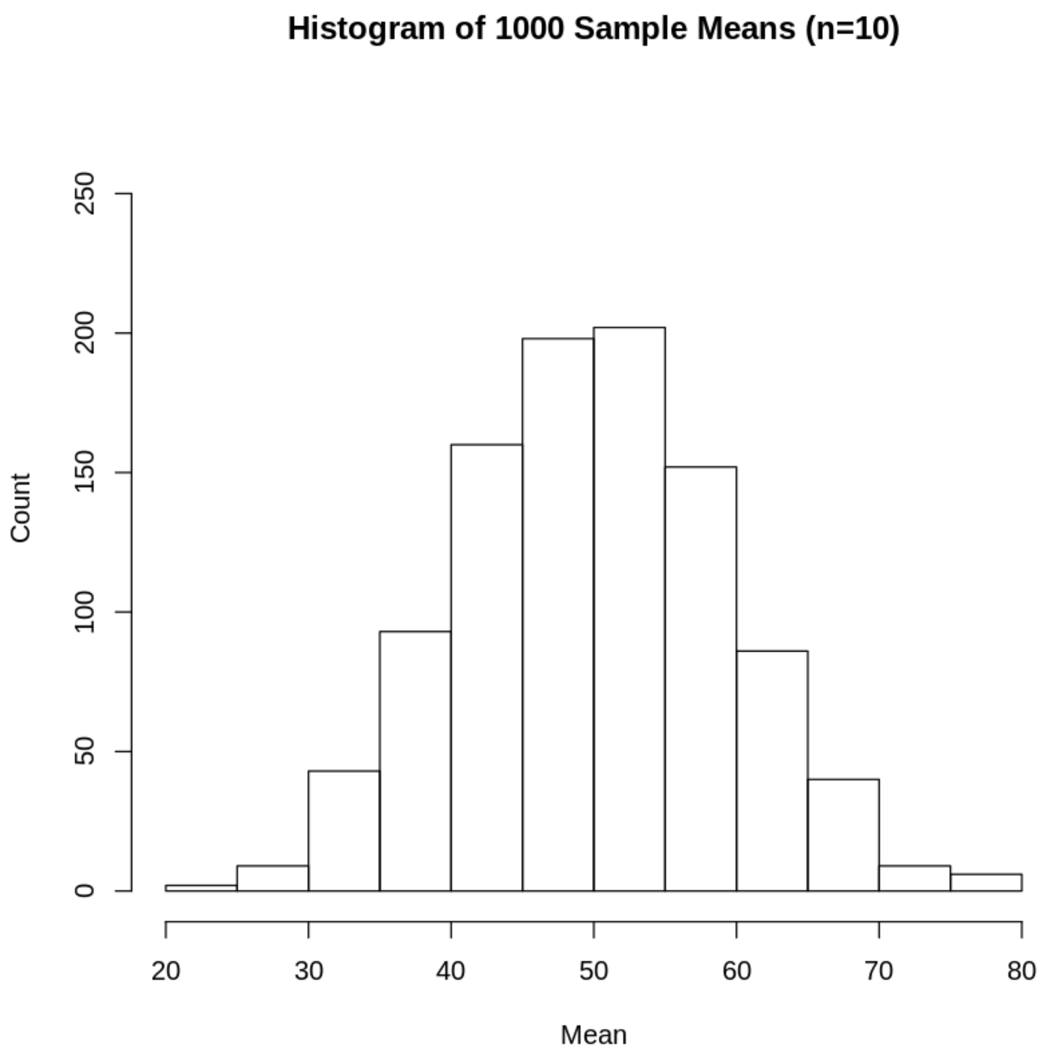
c) Notice, for a sample size of n, our function is returning an estimator of the form

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

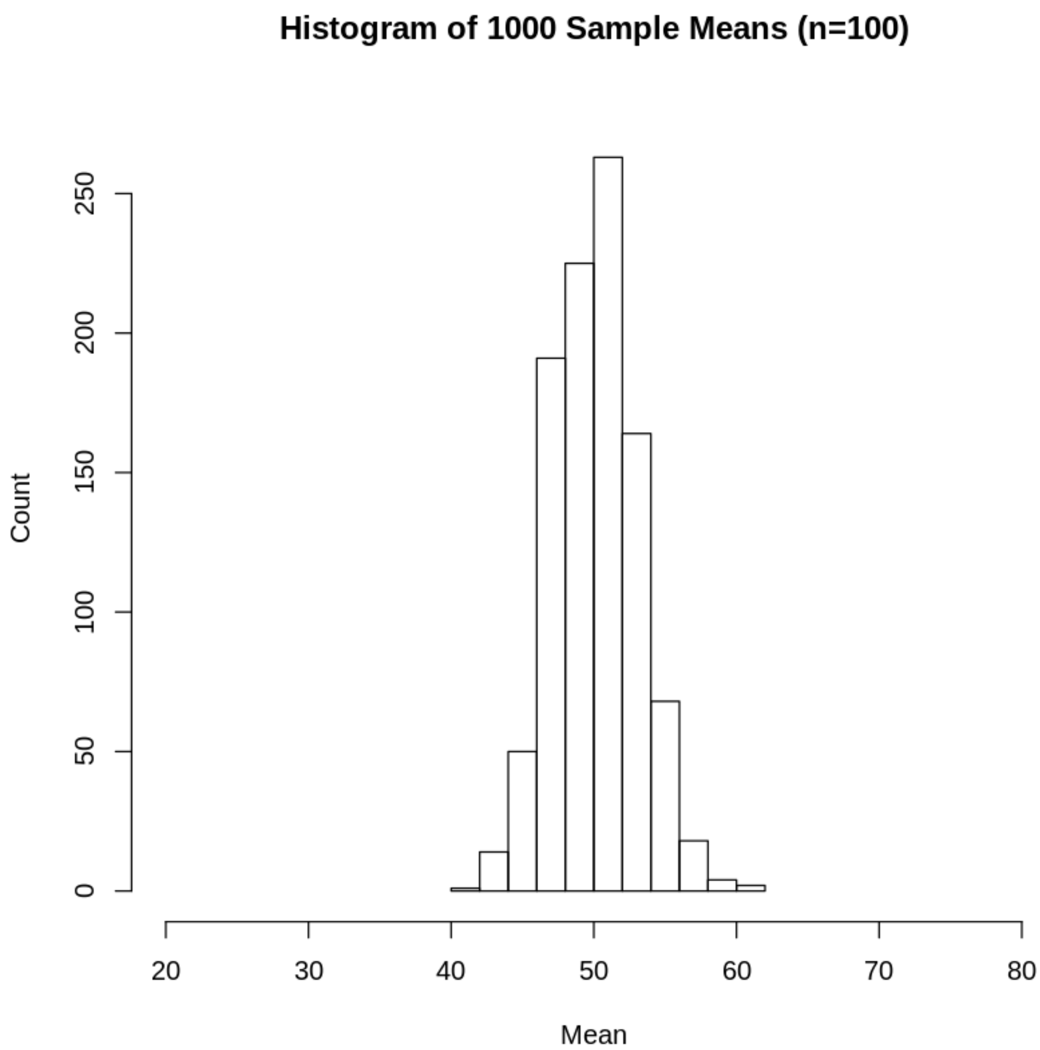
That means, if each X_i is a random variable, then our sample mean is also a random variable with its own distribution. We call this disribution the sample distribution. Let's take a look at what this distribution looks like.

Using the `uniform.sample.mean` function, simulate $m = 1000$ sample means, each from a sample of size $n = 10$. Create a histogram of these sample means. Then increase the value of n and plot the histogram of those sample means. What do you notice about the distribution of $\bar{X}$? What is the mean μ and variance σ^2 of the sample distribution?

[118]:
```
# Your Code Here
set.seed(12094)
sample_count =1000
mean_list = vector('list', length = sample_count)
for(i in 1:sample_count){
    mean_list[i] = uniform.sample.mean(10)
    mean_list = unlist(mean_list, use.names = FALSE)
}
hist(mean_list, main='Histogram of 1000 Sample Means (n=10)', xlim=c(20,80),␣
 ↪ylim=c(0,275), xlab = 'Mean', ylab='Count')
```



[119]:
```
set.seed(12095)
sample_count =1000
mean_list = vector('list', length = sample_count)
for(i in 1:sample_count){
    mean_list[i] = uniform.sample.mean(100)
    mean_list = unlist(mean_list, use.names = FALSE)
}
hist(mean_list, main='Histogram of 1000 Sample Means (n=100)', xlim=c(20,80),␣
↪ylim=c(0,275), xlab = 'Mean', ylab='Count')
```



[120]:
```
print(paste0('The mean of the n=100 distribution is: ', mean(mean_list)))
print(paste0('The variance of the n=100 distribution is: ', var(mean_list)))
```

```
[1] "The mean of the n=100 distribution is: 50.0906563463761"
[1] "The variance of the n=100 distribution is: 8.44914632701546"
```

Once we increased n to 100, the variance of the sample distribution decreased and more values were centralized around a mean of 100. In other words, the spread of the graph decreased and more values of the list are centralized around the expected value, 50.

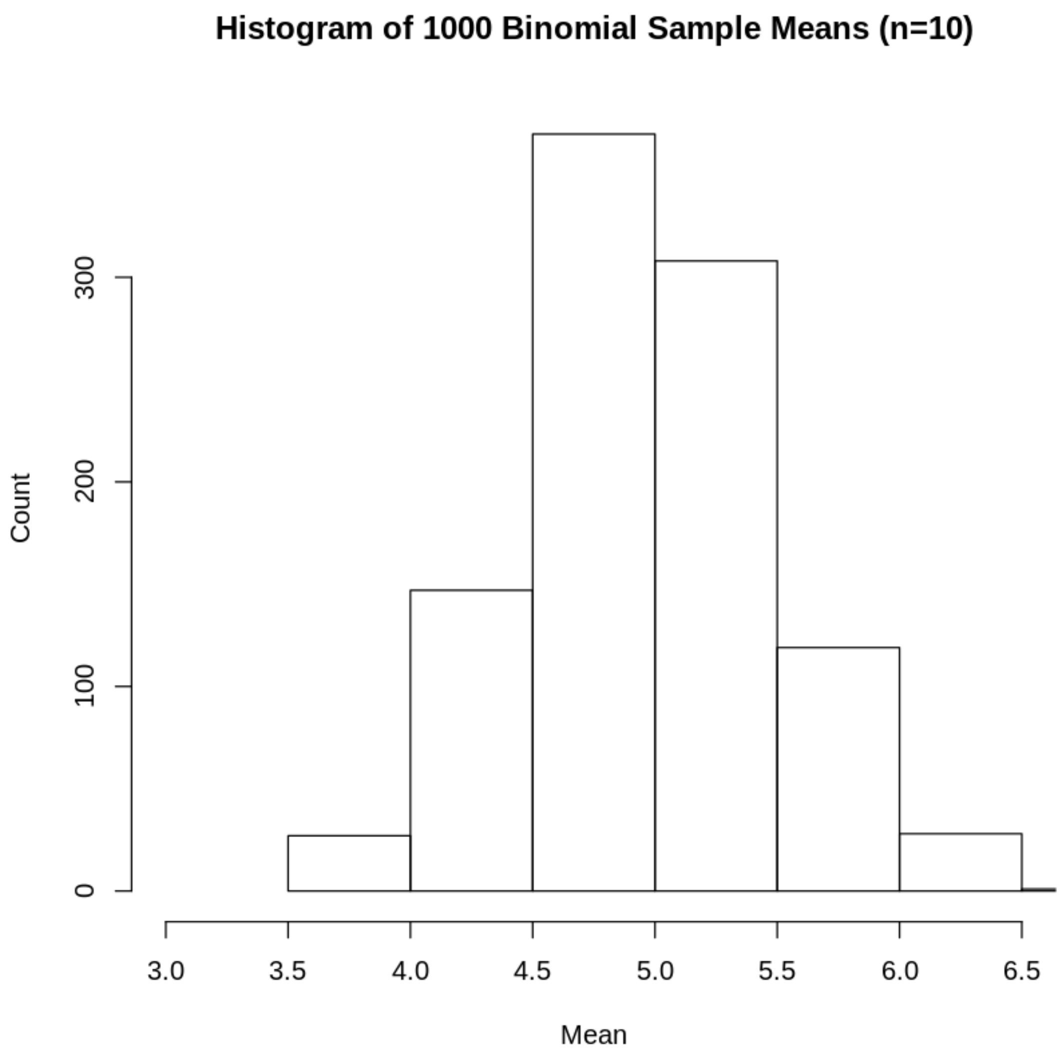
d) Recall that our underlying population distribution is $U(0, 100)$. Try changing the underlying distribution (For example a binomial(10, 0.5)) and check the sample distribution. Be sure to explain what you notice.

[121]:
```
binomial.sample.mean = function(n){

    # Your Code Here
    sample.bin.mean = mean(rbinom(n=10, size=10, prob=0.5))
    return(sample.bin.mean)
}
```

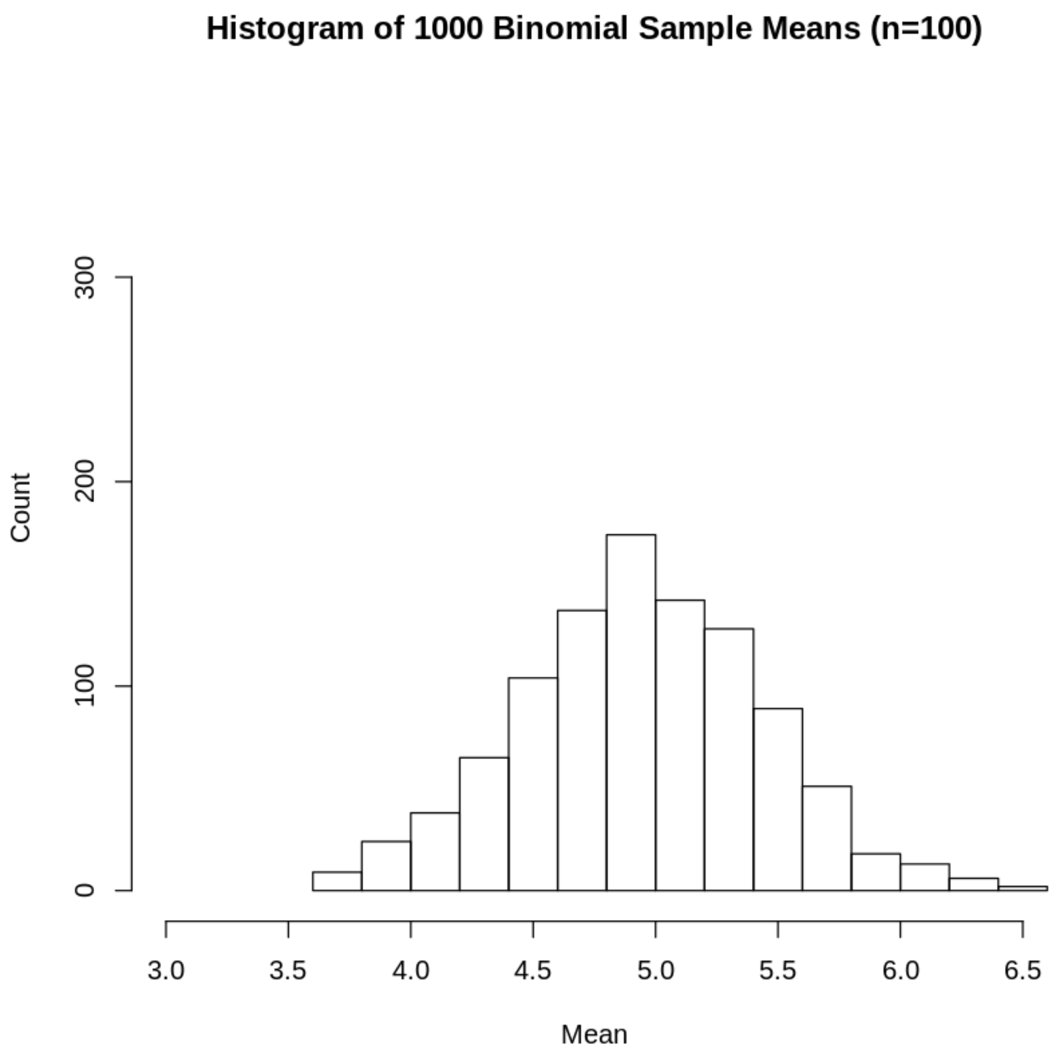
[122]:
```
set.seed(1111)

sample_count = 1000
mean_list2a = vector('list', length = sample_count)
for(i in 1:sample_count){
    mean_list2a[i] = binomial.sample.mean(10)
    mean_list2a = unlist(mean_list2a, use.names = FALSE)
}
hist(mean_list2a, main='Histogram of 1000 Binomial Sample Means (n=10)',␣
↪xlim=c(3, 6.5), ylim=c(0, 375), xlab = 'Mean', ylab='Count')
```



[123]:
```
set.seed(1112)

sample_count = 1000
mean_list2b = vector('list', length = sample_count)
for(i in 1:sample_count){
    mean_list2b[i] = binomial.sample.mean(100)
    mean_list2b = unlist(mean_list2b, use.names = FALSE)
}
hist(mean_list2b, main='Histogram of 1000 Binomial Sample Means (n=100)',␣
 ↪xlim=c(3, 6.5), ylim=c(0, 375), xlab = 'Mean', ylab='Count')
```



Both graphs seem to represent a normal distribution. However, increasing the trials in each sample doesn't seem to decrease the variance.

4 Problem 3

Let X be a random variable for the face value of a fair d-sided die after a single roll. X follows a discrete uniform distribution of the form unif$\{1, d\}$. Below is the mean and variance of unif$\{1, d\}$.

$$E[X] = \frac{1+d}{2} \qquad Var(X) = \frac{(d-1+1)^2 - 1}{12}$$

a) Let $\bar{X}_n$ be the random variable for the mean of n die rolls. Based on the Central Limit Theorem, what distribution does $\bar{X}_n$ follow when $d = 6$.

Based on the Central Limit Theorem, when n is small, $\bar{X}_n$ represents a discrete uniform distribution $\rightarrow \bar{X}_n \sim Unif(\frac{1+d}{2}, \frac{(d-1+1)^2-1}{12})$. However, when n is large, typically when n is 30 or larger, $\bar{X}_n$ follows a Normal distribution where $\bar{X}_n \sim N(\frac{1+d}{2}, \frac{(d-1+1)^2-1}{12})$.

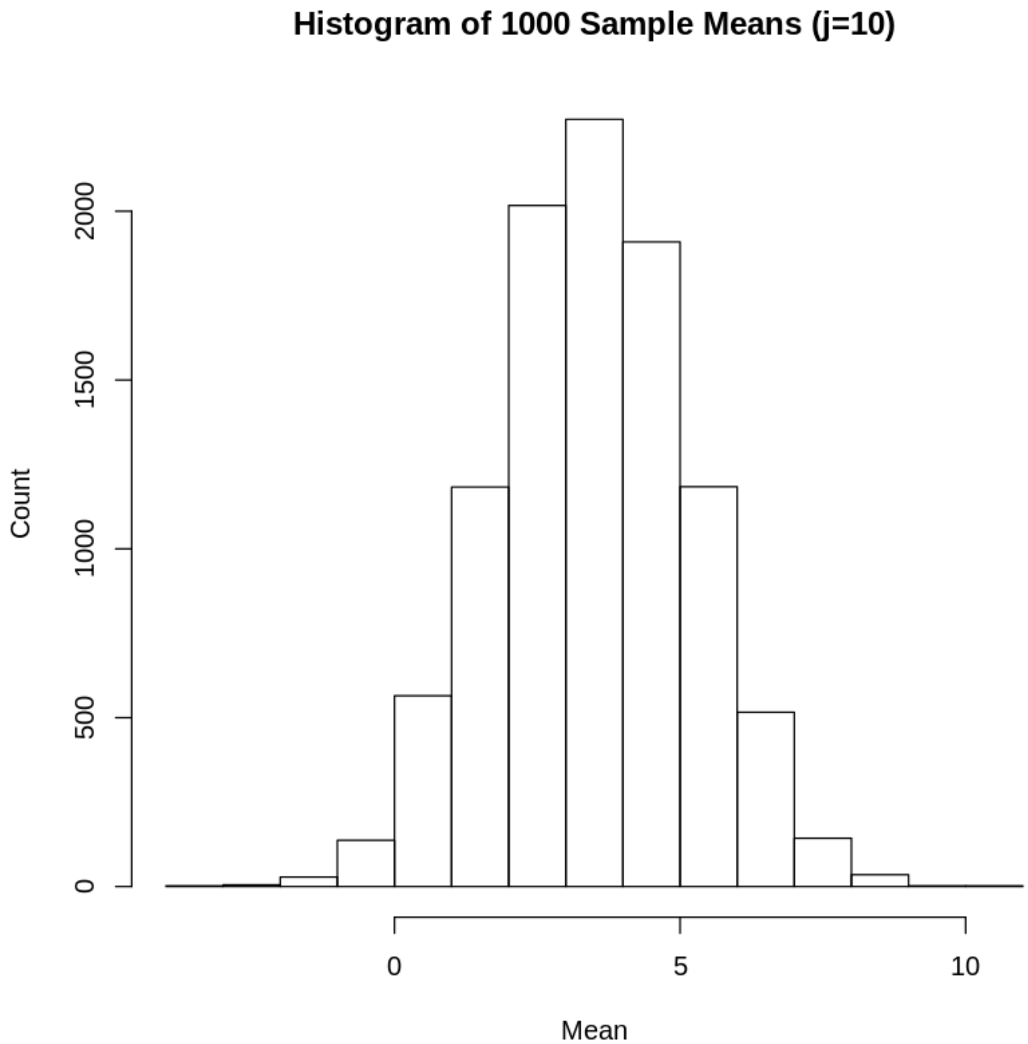
b) Generate $n = 1000$ die values, with $d = 6$. Calculate the running average of your die rolls. In other words, create an array r such that:

$$r[j] = \sum_{i=1}^{j} \frac{X_i}{j}$$

Finally, plot your running average per the number of iterations. What do you notice?

[124]:
```
# Your Code Here
set.seed(1113)
d = 6
n = 1000 # number of samples
j = 10 # trials per sample
e.x = (1 + 6)/2
mean3_list = c()
var.x = ((d-1+1)^2 - 1)/12
for(i in 1:n){
    mean3_list = append(mean3_list, rnorm(j, mean=e.x, sd=sqrt(var.x)))
}
hist(mean3_list, main='Histogram of 1000 Sample Means (j=10)', xlab = 'Mean',␣
 ↪ylab='Count')
print(paste0('The mean of this new distribution is: ', round(mean(mean3_list),␣
 ↪2)))
print(paste0('The variance of this new distribution is: ',␣
 ↪round(var(mean3_list), 2)))
```

```
[1] "The mean of this new distribution is: 3.48"
[1] "The variance of this new distribution is: 2.89"
```



Overall, instead of a disrete distribution, I notice the distribution represents the following normal distribution: $\bar{X}_n \sim N(3.48, 2.89)$.