Sampling With Markov Chain Monte Carlo

2025-10-03

1.

```r
set.seed(3701)
n <- 30
dens <- function(x,y) {
  return(dbinom(x,n, y))
}
simnum <- 10000

x <- vector()
current <- runif(1)
x[1] <- current

for(i in 1:simnum){
  new <- runif(1)
  likelihood_current <- dens(12, current)
  likelihood_new <- dens(12, new)

  if(runif(1) < likelihood_new/likelihood_current){
    current <- new
  }
  x[i] <- current
}
mean(x)
```

```
## [1] 0.4043903
```

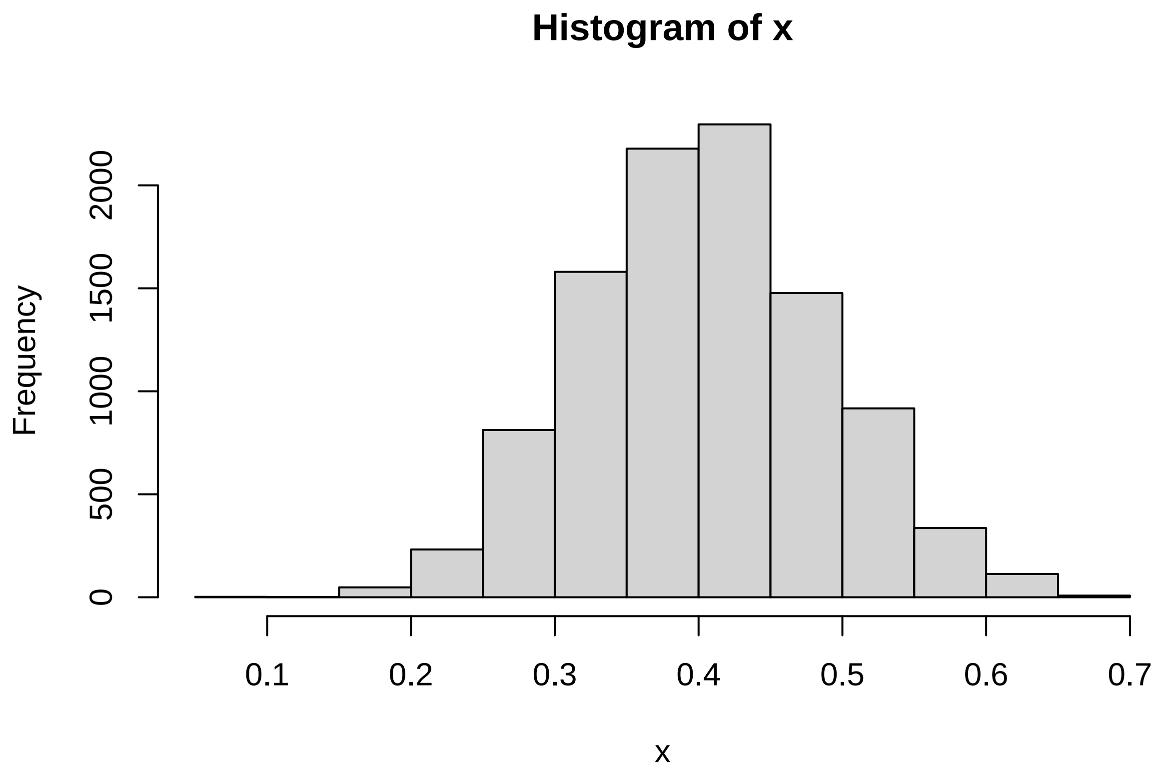
```r
median(x)
```

```
## [1] 0.4031942
```

```r
confint <- quantile(x, c(0.05, 0.95))
confint
```

```
##        5%       95%
## 0.2695944 0.5464456
```

```r
hist(x)
```



2.

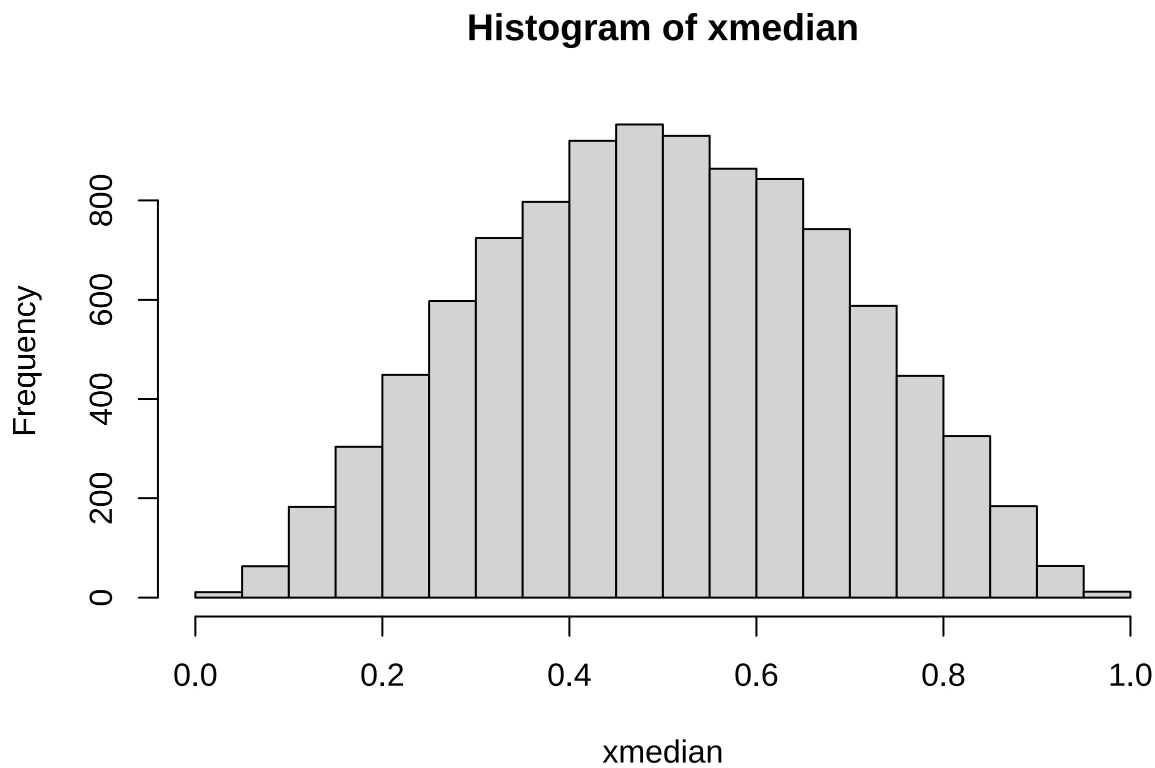
```
#n = 5 uniform
set.seed(1234)
simnum <- 10000
n <- 5
xmedian <- vector()
for(i in 1:simnum){
  x <- runif(n)
  xmedian[i] <- median(x)
}
mean(xmedian)
```

```
## [1] 0.5010251
```

```
var(xmedian)
```

```
## [1] 0.03539914
```

```
hist(xmedian)
```



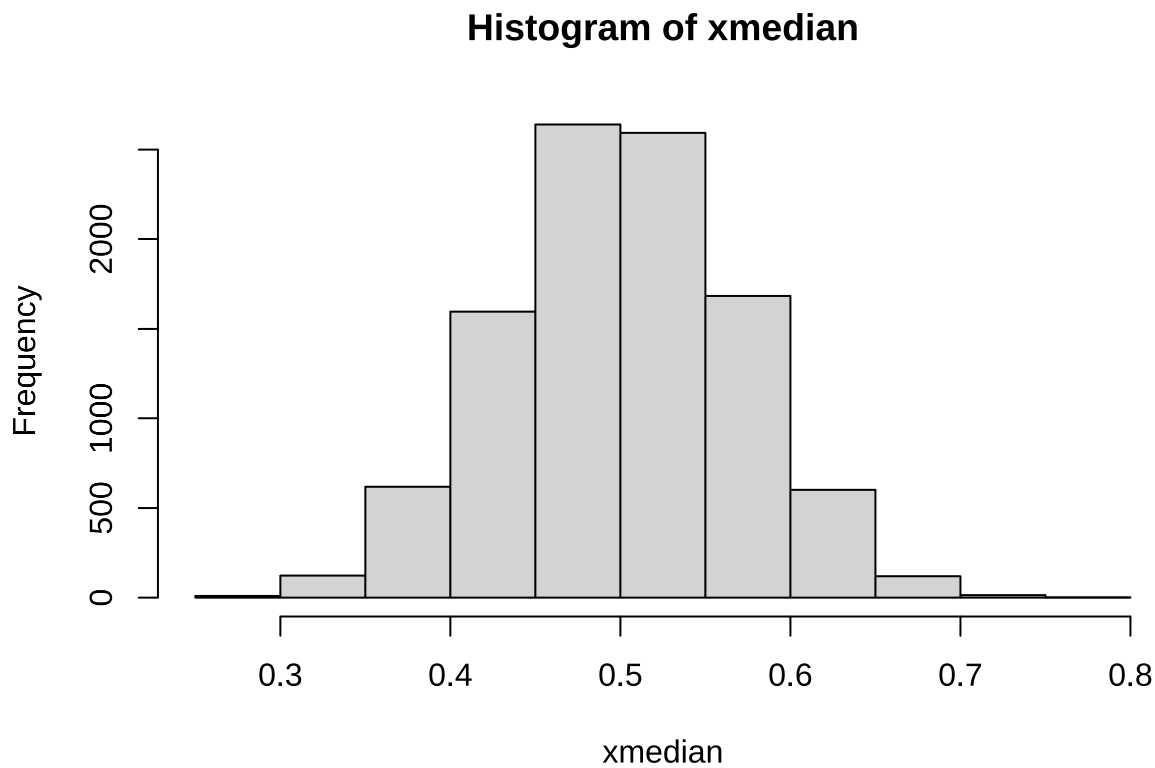

```r
#n = 50 uniform
set.seed(1234)
simnum <- 10000
n <- 50
xmedian <- vector()
for(i in 1:simnum){
  x <- runif(n)
  xmedian[i] <- median(x)
}
mean(xmedian)
```

```
## [1] 0.5003552
```

```r
var(xmedian)
```

```
## [1] 0.004757597
```

```r
hist(xmedian)
```



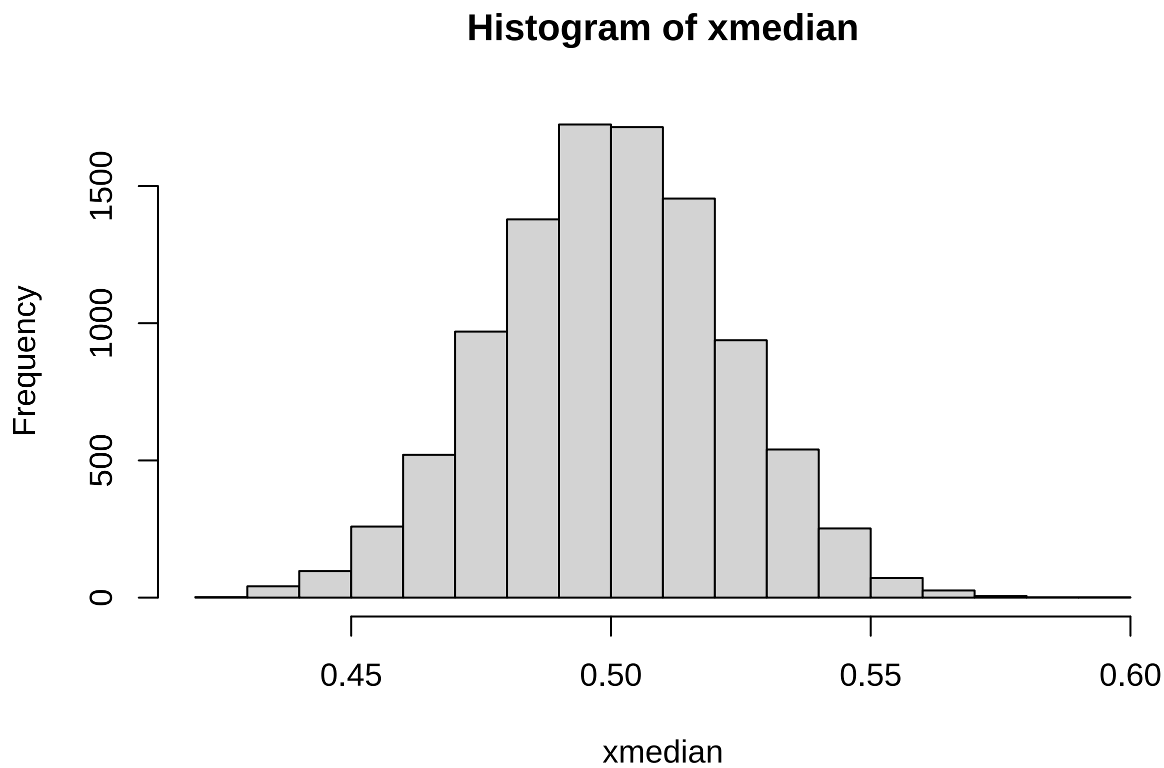

```r
#n = 500 uniform
set.seed(1234)
simnum <- 10000
n <- 500
xmedian <- vector()
for(i in 1:simnum){
  x <- runif(n)
  xmedian[i] <- median(x)
}
mean(xmedian)
```

```
## [1] 0.4999126
```

```r
var(xmedian)
```

```
## [1] 0.0005007589
```

```r
hist(xmedian)
```



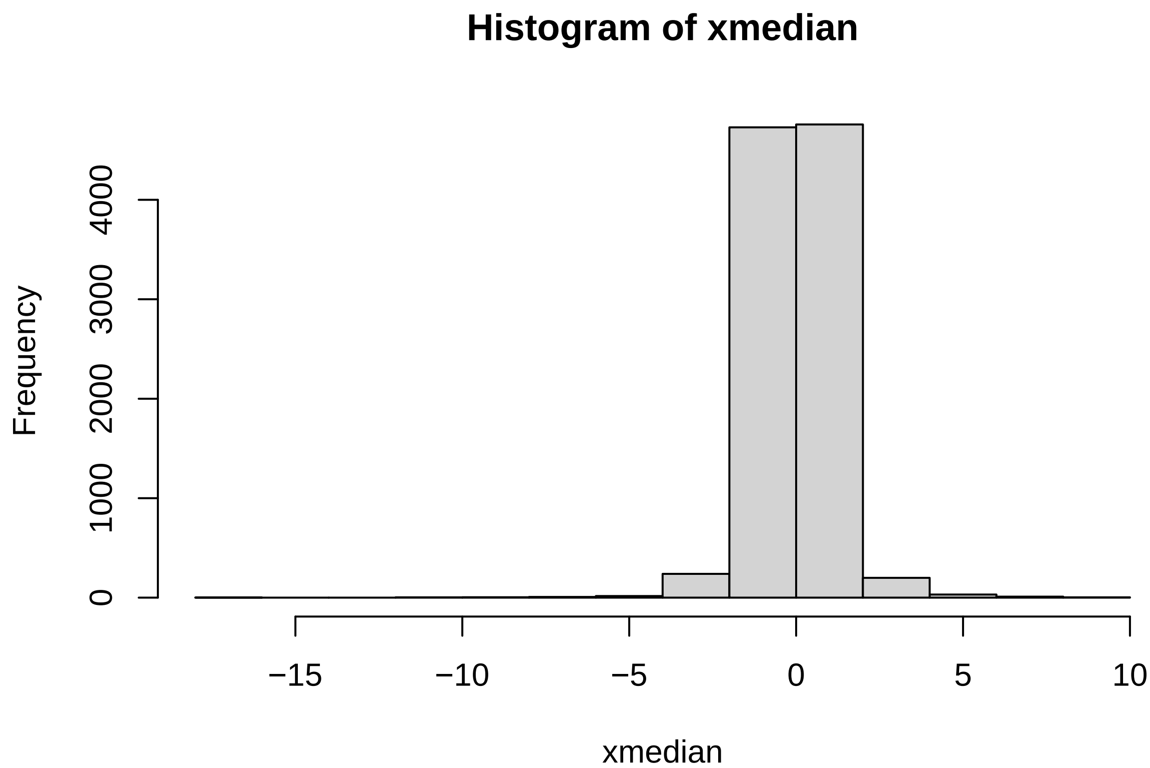

```
#n = 5 cauchy
set.seed(1234)
simnum <- 10000
n <- 5
xmedian <- vector()
for(i in 1:simnum){
  x <- rcauchy(n)
  xmedian[i] <- median(x)
}
mean(xmedian)
```

```
## [1] -0.0133133
```

```
var(xmedian)
```

```
## [1] 1.088838
```

```
hist(xmedian)
```



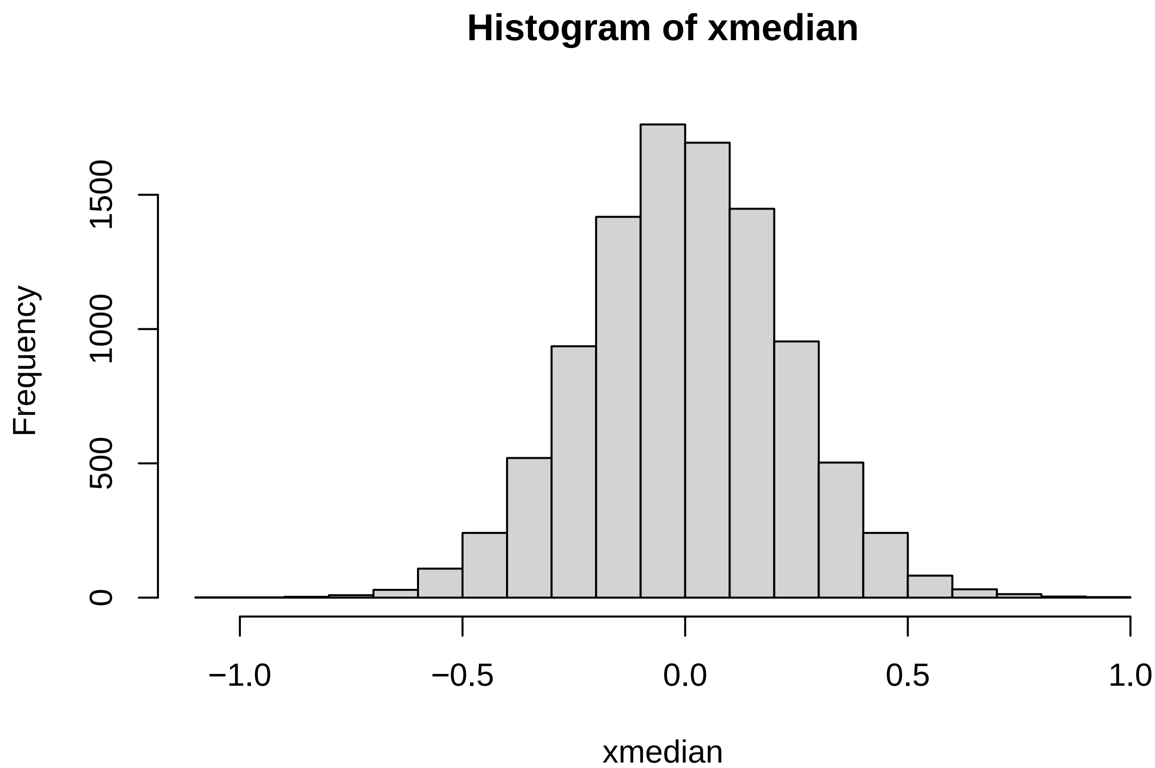

```r
#n = 50 cauchy
set.seed(1234)
simnum <- 10000
n <- 50
xmedian <- vector()
for(i in 1:simnum){
  x <- rcauchy(n)
  xmedian[i] <- median(x)
}
mean(xmedian)
```

```
## [1] -0.0007552526
```

```r
var(xmedian)
```

```
## [1] 0.05111425
```

```r
hist(xmedian)
```



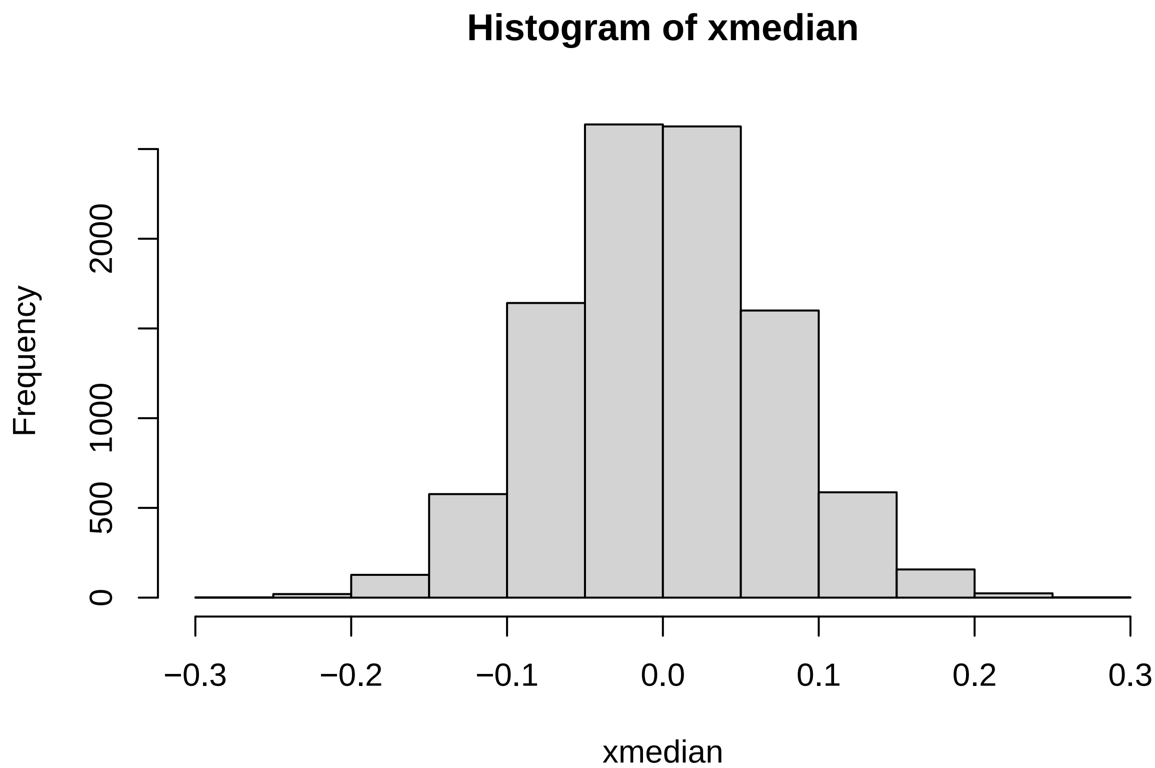

```r
#n = 500 cauchy
set.seed(1234)
simnum <- 10000
n <- 500
xmedian <- vector()
for(i in 1:simnum){
  x <- rcauchy(n)
  xmedian[i] <- median(x)
}
mean(xmedian)
```

```
## [1] 0.0002725443
```

```r
var(xmedian)
```

```
## [1] 0.004889038
```

```r
hist(xmedian)
```



The medians of the uniform distribution for all n values are all around 0.5, while the medians of the cauchy distribution for all n values are near zero. I believe this is because unlike the uniform distribution, the cauchy distribution is like a normal distribution, where mean and median tend to be the same value. The variance also shows that as our n value gets bigger, the variance decreases for both distributions.

3.

```
set.seed(3701)
#coffee shop t.test
coffee_shop <- read.csv("coffee_shop.csv")
data <- data.frame(coffee_shop)
t.test(data$Customers)
```

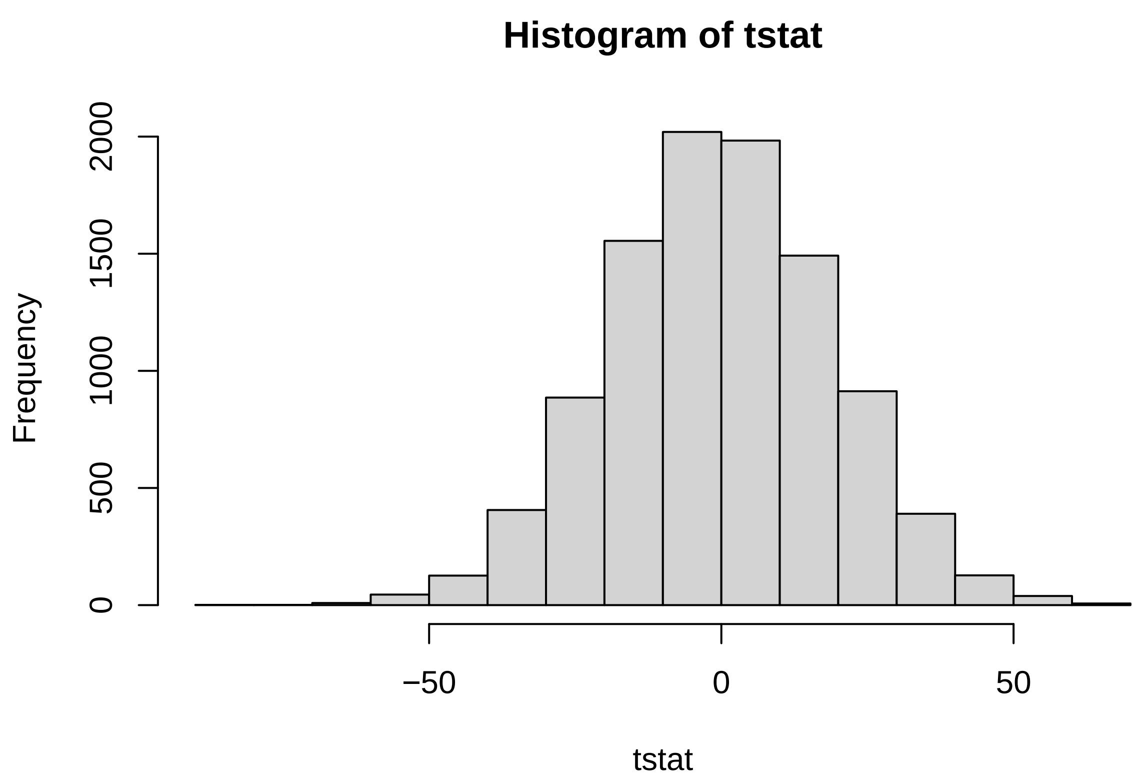
```
##
##  One Sample t-test
##
## data:  data$Customers
## t = 151.45, df = 364, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  60.09707 61.67827
## sample estimates:
## mean of x
##  60.88767
```

```r
#Monte Carlo t test
set.seed(3701)
simnum <- 10000
lambda <- 60
data <- coffee_shop$Customers
n <- length(data)
tstat <- vector()

for (i in 1:simnum){
  x <- rpois(n, lambda)
  sampmean <- mean(x)
  sampvar <- var(x)
  sampSD <- sqrt(sampvar/n)
  tstat[i]<- (sampmean-lambda)/(sampSD/sqrt(n))
}
pvalue <- mean(2*(1-pt(abs(tstat), df = n-1)))
pvalue
```

```
## [1] 0.03339738
```

```r
hist(tstat)
```



Since the p value is 0.033, and $0.033 < 0.05$, we reject the null which states that lambda is 0. This matches up with the previous t test, which also had a p value which was less than the alpha, thus rejecting the null hypothesis. I don't really think this has practical significance, since the amount of customers for a coffee shop would probably be over 0 or they would close.