Tutorial 4 Solutions

STAT 3013/8027

---

1. Rice Chapter 6: **Question 9**. **See the handwritten solutions.**

2. Rice Chapter 8: **Question 5 (a, b, c)** also find the expectation and variance of the two estimators; **Question 52 (a, b)**. **See the handwritten solutions.**

3. Write down an algorithm (pseudo-code) and implement that code in R to generate 1,000 independent samples from the following distributions:

   - $Y \sim$ binomial$(8, 2/3)$. Do not use the `rbinom()` function. You may use `qbinom()`.
   - $Y \sim$ negative binomial$(5, 1/3)$. Do not use the `rnbinom()` function. You may use `qnbinom()`.

- **Solutions:** (a)
    - Set $S = 10,000$.
    - Create a storage vector called *out* with length S.
    - Generate S random uniform values u.
    - Based on the possible values of $Y : \{y_1 < y_2 < \cdots\}$ for each u if $F_y(y_i) < u \leq F_y(y_{i+1})$ populate *out*[s] with y_{i+1}. Here

$$F_Y(y) = P(Y \leq y) = \sum_{i=0}^{y} \binom{8}{i} (2/3)^i (1 - 2/3)^{8-i}$$

Define $y_0 = -\infty$ and $F_Y(y_0) = 0$. From here we can determine Y.

$$Y = \begin{cases} 0 & \text{if } 0 < U \leq 2 \times 10^{-4} \\ 1 & \text{if } 2 \times 10^{-4} < U \leq 0.0026 \\ 2 & \text{if } 0.0026 < U \leq 0.0197 \\ 3 & \text{if } 0.0197 < U \leq 0.0879 \\ 4 & \text{if } 0.0879 < U \leq 0.2586 \\ 5 & \text{if } 0.2586 < U \leq 0.5318 \\ 6 & \text{if } 0.5318 < U \leq 0.8049 \\ 7 & \text{if } 0.8049 < U \leq 0.961 \\ 8 & \text{if } 0.961 < U \leq 1 \end{cases}$$

```r
set.seed(1001)
n <- 8
p <- 2/3
S <- 1000
u <- runif(S)
y <- qbinom(u, size=n, prob=p)

mean(y)
```

```
## [1] 5.298
```

```
    true.mean <- n*p
    true.mean
```

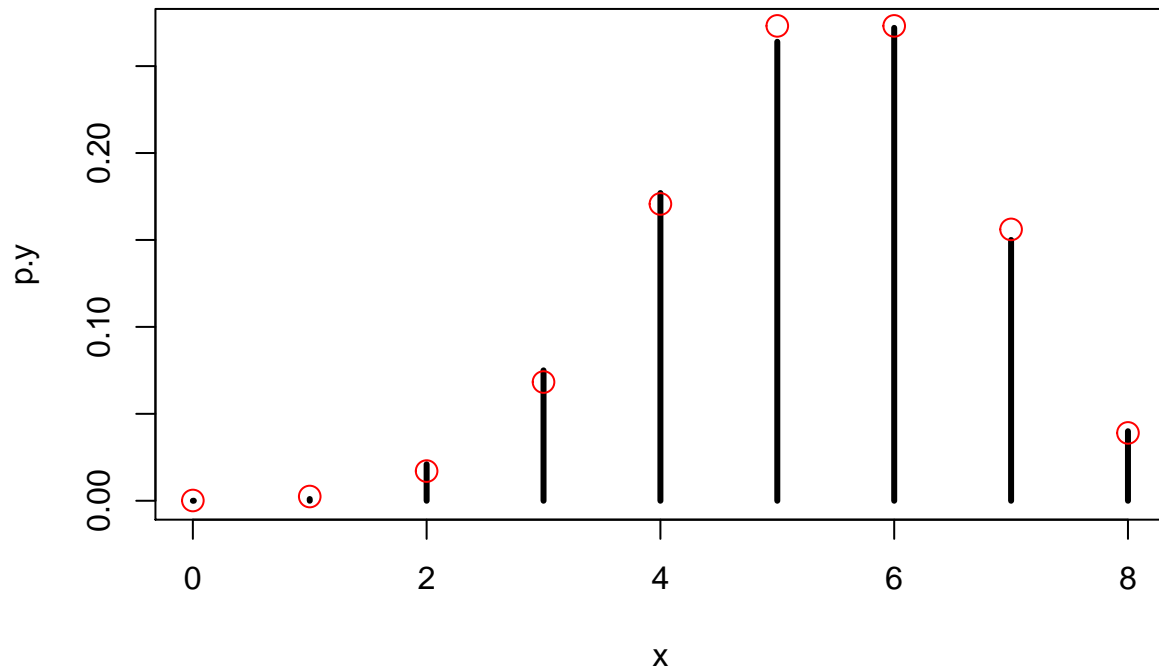## [1] 5.333333

```
    var(y)
```

[1] 1.827023

```
    true.var <- n*p*(1-p)
    true.var
```

[1] 1.777778

```
p.y <- table(y)/S
p.y <- c(0, p.y)
x <- 0:n
plot(x, p.y, type="h", lwd=3)
points(x, dbinom(x, n, p), col="red", cex=1.5)
```


Black bars are from the random sample. Red circles are from the distribution. They match quite closely.

- (b)
  - Set $S = 10,000$.
  - Create a storage vector called *out* with length S.
  - Generate S random uniform values u.
  - Based on the possible values of $Y : \{y_1 < y_2 < \cdots\}$ for each u if $F_y(y_i) < u \leq F_y(y_{i+1})$ populate *out*[s] with y_{i+1}. Here

$$F_Y(y) = P(Y \le y) = \sum_{i=0}^{y} \binom{r+i-1}{i} (2/3)^r (1 - 2/3)^i$$

Define $y_0 = -\infty$ and $F_Y(y_0) = 0$. From here we can determine Y.

```r
set.seed(1001)
n <- 5
p <- 1/3

S <- 1000
u <- runif(S)
y <- qnbinom(u, n, p)

mean(y)
```

```
## [1] 9.847
```

```r
true.mean <- (n*(1-p))/p
true.mean
```

```
## [1] 10
```

```r
var(y)
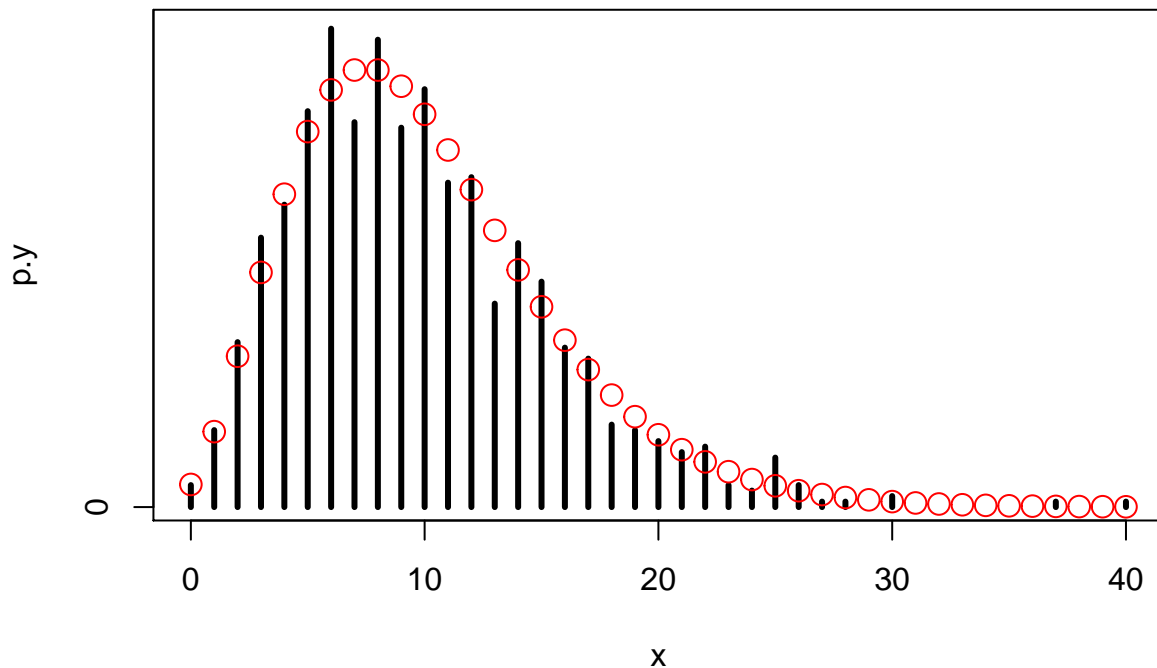```

```
## [1] 30.3079
```

```r
true.var <- (n*(1-p))/p^2
true.var
```

```
## [1] 30
```

```r
p.y <- table(y)/S
x <- as.numeric(names(table(y)))
plot(x, p.y, type="h", lwd=3)
x <- 0:max(as.numeric(names(table(y))))
points(x, dnbinom(x, n, p), col="red", cex=1.5)
```

Black bars are from the random sample. Red circles are from the distribution. They match fairly closely. Due to the large number of bins we ought to increase the sample size (S).

4. Let's consider drawing 10,000 random samples from a t-distribution with degrees of freedom $\nu = 10$ through a few different methods.

- Use a Metropolis algorithm with a symetric distirbution that is a standard normal.
- Use an accept/reject algorithm based on being able to draw samples from Cauchy random variable with location $= 0$ and scale $= 1$.
- Show how to use a tansformation method to directly generate random samples from the t-distribution.
- Which of the three methods do you prefer? Can you rank them?

- **Solutions:** (a): Here I will use the Metropolis algorithm to dengerate samples from a student's t distribution with $\nu = 10$. I will use a symmetric proposal distirbution based on a $n(0,1)$ random variable. So if Y is the current value then a proposal will be $Y^* = Y + Z$, where $Z \sim n(0,1)$.

```r
set.seed(1001)
S <- 10000
out <- rep(0, S)
acc <- 0

## starting value
y <- 5
out[1] <- y

## tuning parameter
sigma <- 1
```
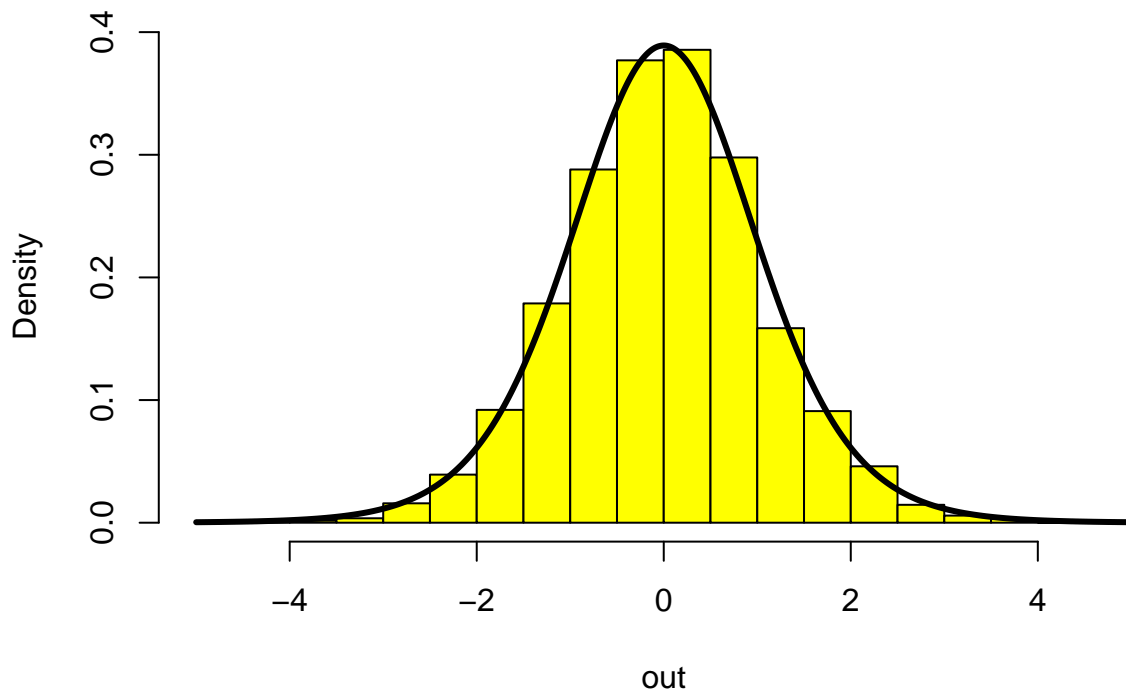
```
## MCMC
for(i in 2:S){

  y.star <- y + rnorm(1, 0, sigma)
  r <-  dt(y.star, 10)/dt(y, 10)
  rho <- min(r,1)

  if(runif(1) <= rho){
    y <- y.star
    acc <- acc + 1
    }

  out[i] <- y
  }

hist(out, col="yellow", xlim=c(-5,5), prob=TRUE,
     main="Student's t distribution with dof 10 - M-H Alg.")
y <- seq(-5, 5, by=0.01)
lines(y, dt(y,10), type="l", lwd=3)
```
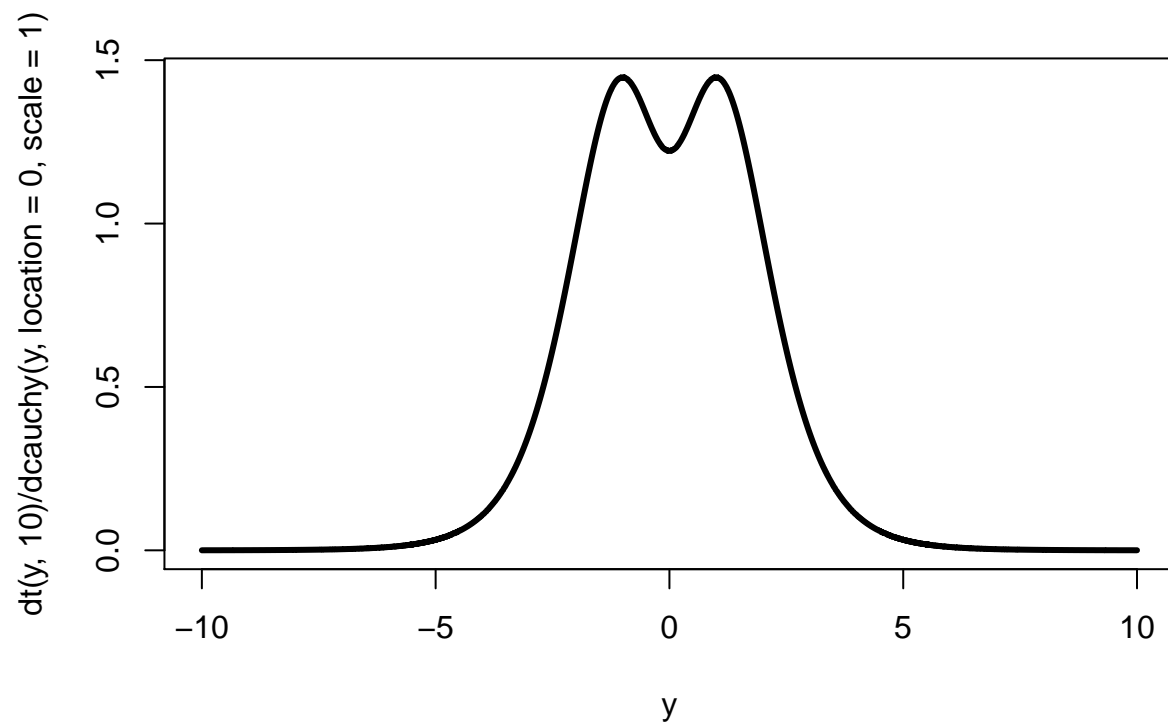
Student's t distribution with dof 10 – M–H Alg.

- (b): Let's do the same thing but use the Accept/Reject algorithm. We will propose values based on a Cauchy random variable with location $= 0$ and scale $= 1$. For the algorithm let's first determine M (in an exam you may need to do this analytically):

```
y <- seq(-10, 10, by=0.001)
plot(y, dt(y, 10)/dcauchy(y, location = 0, scale = 1), lwd=3, type="l")
```
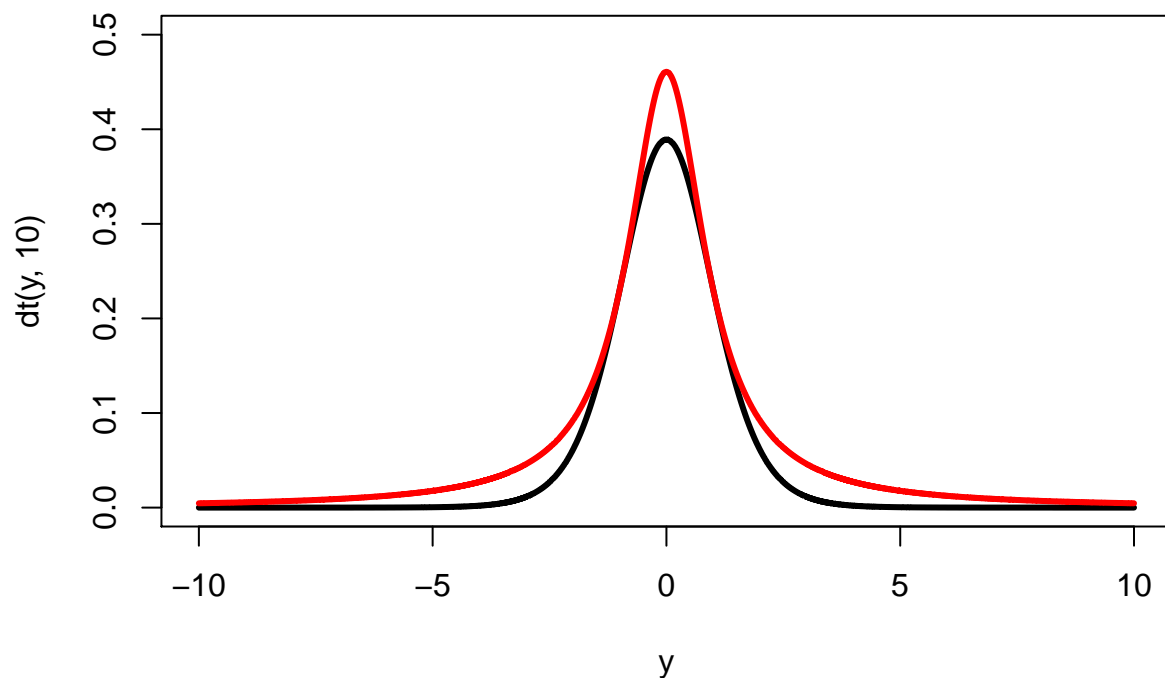
```r
M <- max(dt(y, 10)/dcauchy(y, location = 0, scale = 1))
M
```

```
## [1] 1.447407
```

Now let's check to see that we have a proper envelope. This looks good.

```r
y <- seq(-10, 10, by=0.001)
plot(y,  dt(y, 10), type="l", lwd=3, ylim=c(0,0.5))
lines(y, M*dcauchy(y, location = 0, scale = 1), lwd=3, col="red")
```

```
mean(M*dcauchy(y, location = 0, scale = 1) >= dt(y, 10))
```
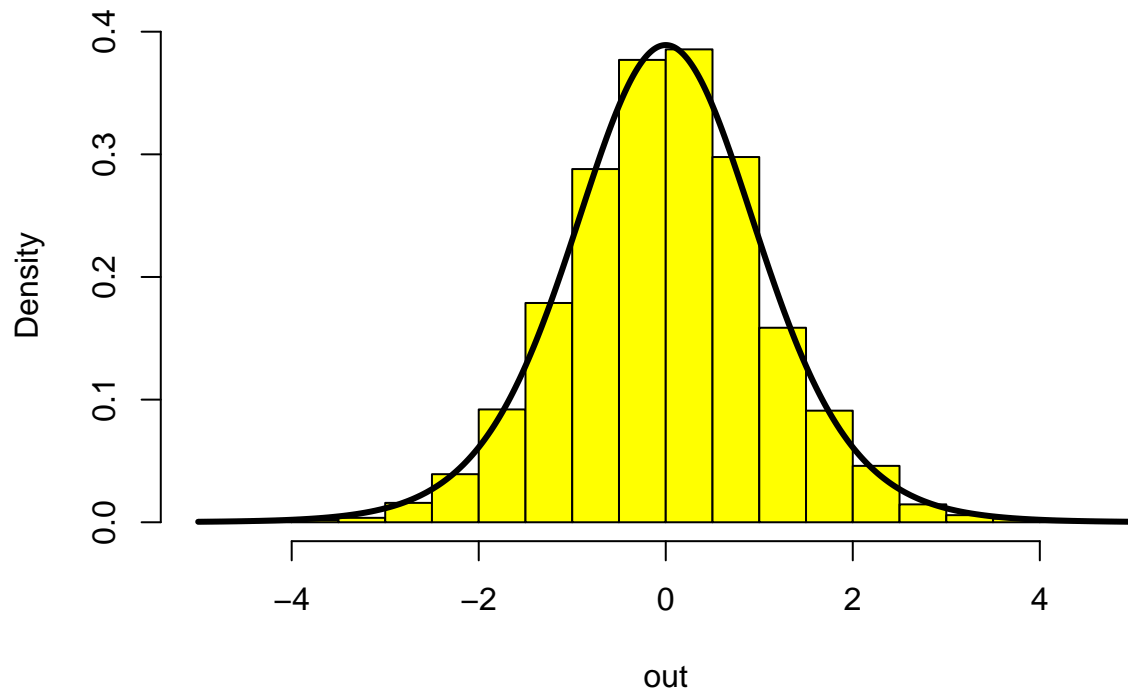
```
## [1] 1
```

Now let's run the algorithm.

```
n <- 10000
y <- NULL

for(i in 1:n){
u <- runif(1, 0, 1)
v <- rcauchy(1, 0, 1)
if(u < (1/M)*(dt(v, 10)/dcauchy(v, 0, 1))){
 y.i <- v
 y <- c(y, y.i)
  }}

hist(out, col="yellow", xlim=c(-5,5), prob=TRUE,
     main="Student's t distribution with dof 10 - A/R Alg.")
y <- seq(-5, 5, by=0.01)
lines(y, dt(y,10), type="l", lwd=3)
```

Student's t distribution with dof 10 – A/R Alg.

- (c): Now let's sample directly. Based on last weeks tutorial we know we can generate a student's t distribution with $\nu = 10$ ($Y \sim t_{\nu=10}$) through the following standard normal distributions ($Z_1, \ldots, Z_{11}$):

$$Y = Z_1 / \sqrt{\sum_{i=2}^{11} Z_i^2 / 10}$$

We could do generate the Zs through the Box-Muller transform, but I will cheat and use the standard **rnorm** function in R. Now let's generate from the t distribution.

```r
set.seed(1001)
S <- 10000
out <- rep(0, S)

for(s in 1:S){
  z <- rnorm(11)
  out[s] <- z[1]/sqrt(sum(z[2:11]^2)/10)
  }

hist(out, col="yellow", xlim=c(-5,5), prob=TRUE,
    main="Student's t distribution with dof 10 - Direct Alg.")
y <- seq(-5, 5, by=0.01)
lines(y, dt(y,10), type="l", lwd=3)
```

Student's t distribution with dof 10 – Direct Alg.
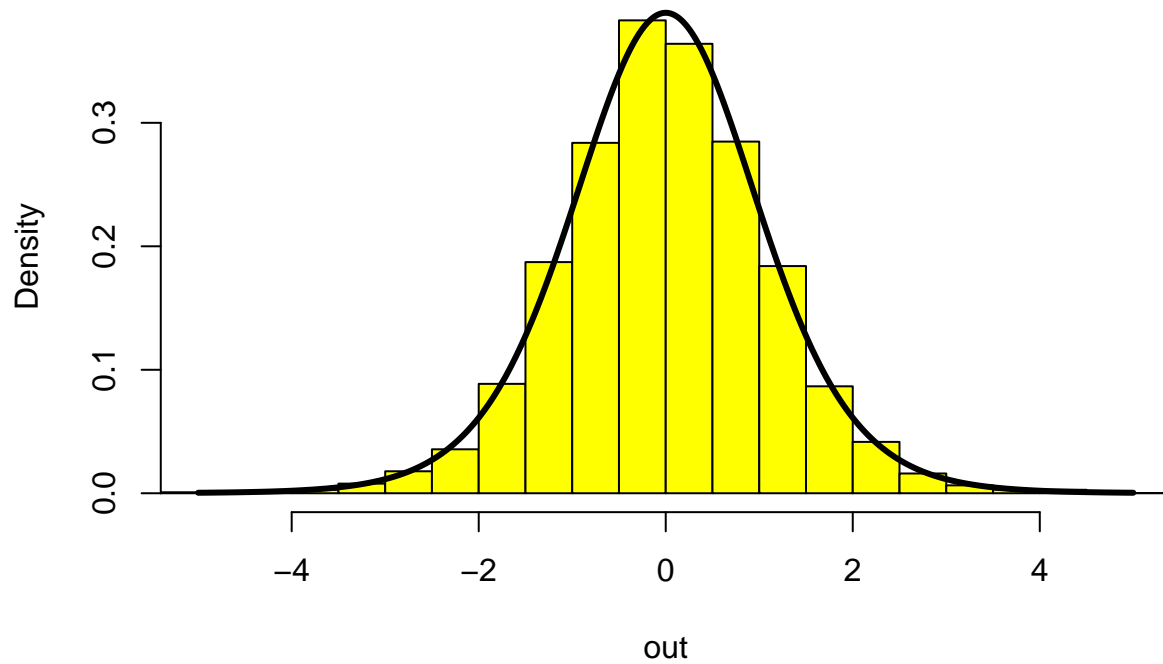
- (d): I prefer to use the direct method as we achieve independent samples and there is no rejection process. For this specific problem, I would next prefer the accept/reject ajgorithm as the samples are independent. Finally, the MCMC approach, which has dependent samples and does reject proposals.