STAT 403 Spring 2018
HW05

Nan Tang
1662478

May 8, 2019

Q1

Q1-a

```
bern_p <- function(x) {
  return(exp(1 + 2 * x) / (1 + exp(1 + 2 * x)))
}
n <- 500
x_value <- rnorm(500)
y_value <- rbinom(n, size=1, p=bern_p(x_value))
xy_logic = glm(y_value~x_value, family = "binomial")

beta0 <- summary(xy_logic)$coefficient[1,1]
beta1 <- summary(xy_logic)$coefficient[2,1]

> beta0
[1] 0.9670936
> beta1
[1] 1.799514
```

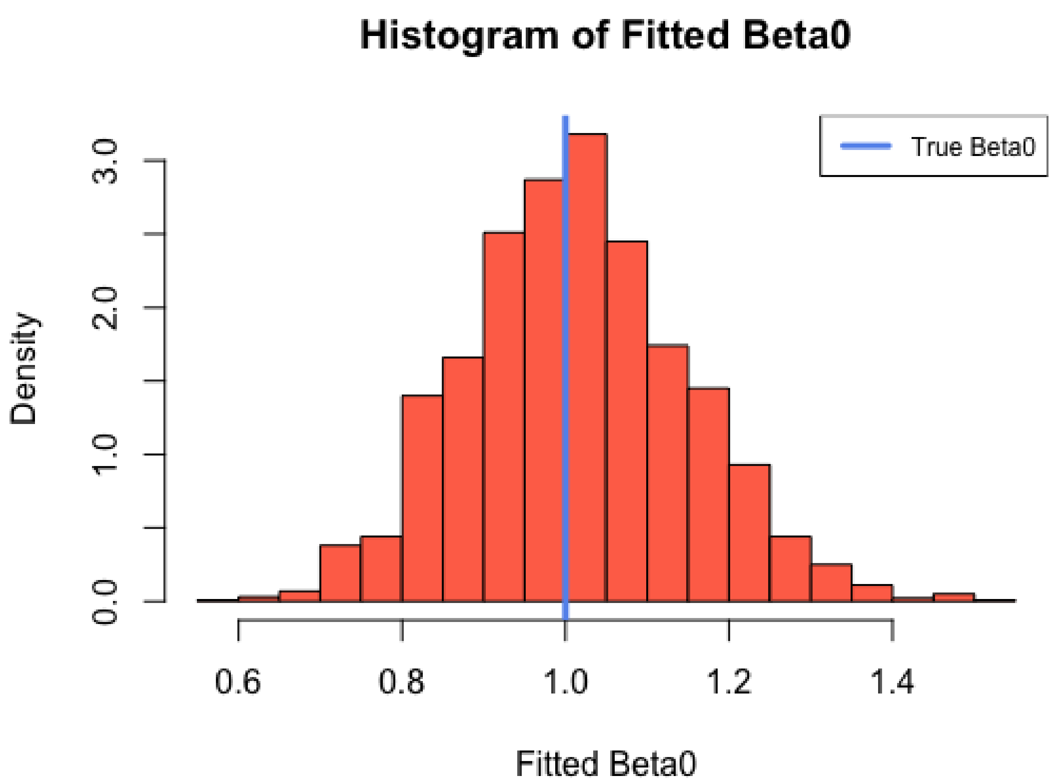
In one time simulation under sample size 500, $\widehat{\beta}_1$ and $\widehat{\beta}_0$ are respectively 1.799514 and 0.9670936.

Q1-b

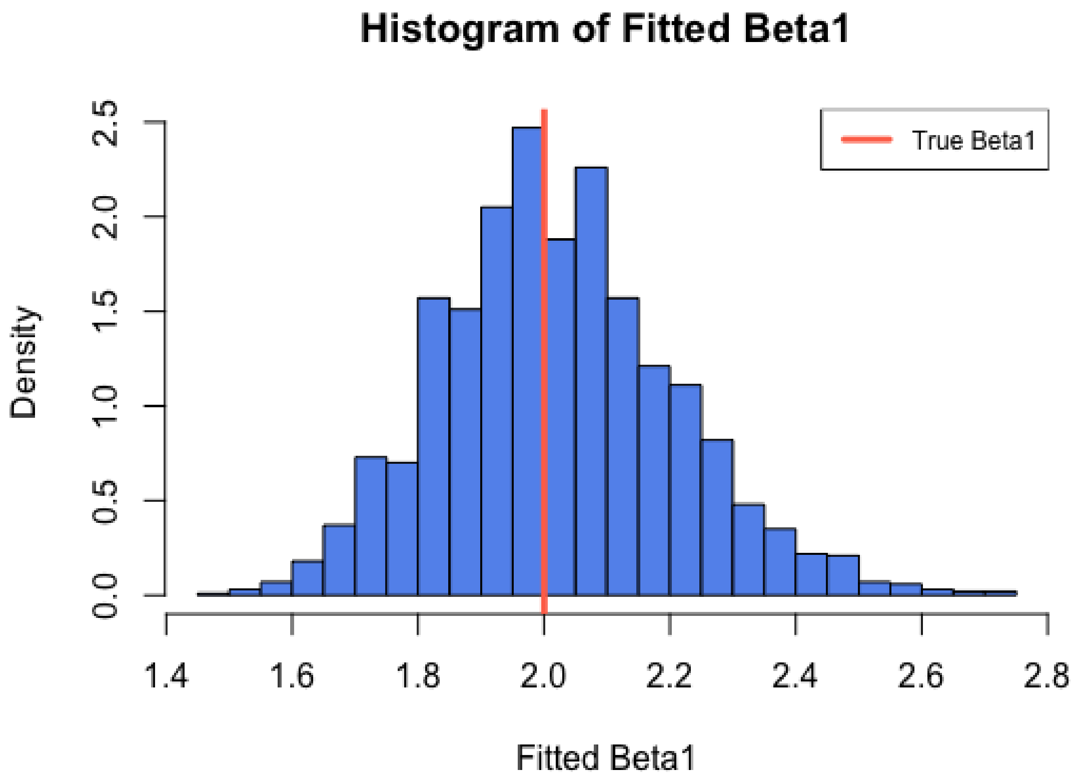
```
N = 2000
beta0_sim <- rep(NA, N)
beta1_sim <- rep(NA, N)

for (ii in 1:N) {
  x_value <- rnorm(n)
  y_value <- rbinom(n, size=1, p=bern_p(x_value))
  xy_logic = glm(y_value~x_value, family = "binomial")
  beta0_sim[ii] <- summary(xy_logic)$coefficient[1,1]
  beta1_sim[ii] <- summary(xy_logic)$coefficient[2,1]
}

hist(beta0_sim, probability=T, main='Histogram of Fitted Beta0', breaks=20,
     xlab='Fitted Beta0', col='coral1')
abline(v=1, lwd=3, col='cornflowerblue' )
legend('topright', 'True Beta0', col='cornflowerblue', lwd=3, cex=0.75)
```



```
hist(beta1_sim, probability=T, main='Histogram of Fitted Beta1', breaks=20,
     xlab='Fitted Beta1', col='cornflowerblue')
abline(v=2, lwd=3, col='coral1' )
legend('topright', 'True Beta1', col='coral1', lwd=3, cex=0.75)
```



Q1-c

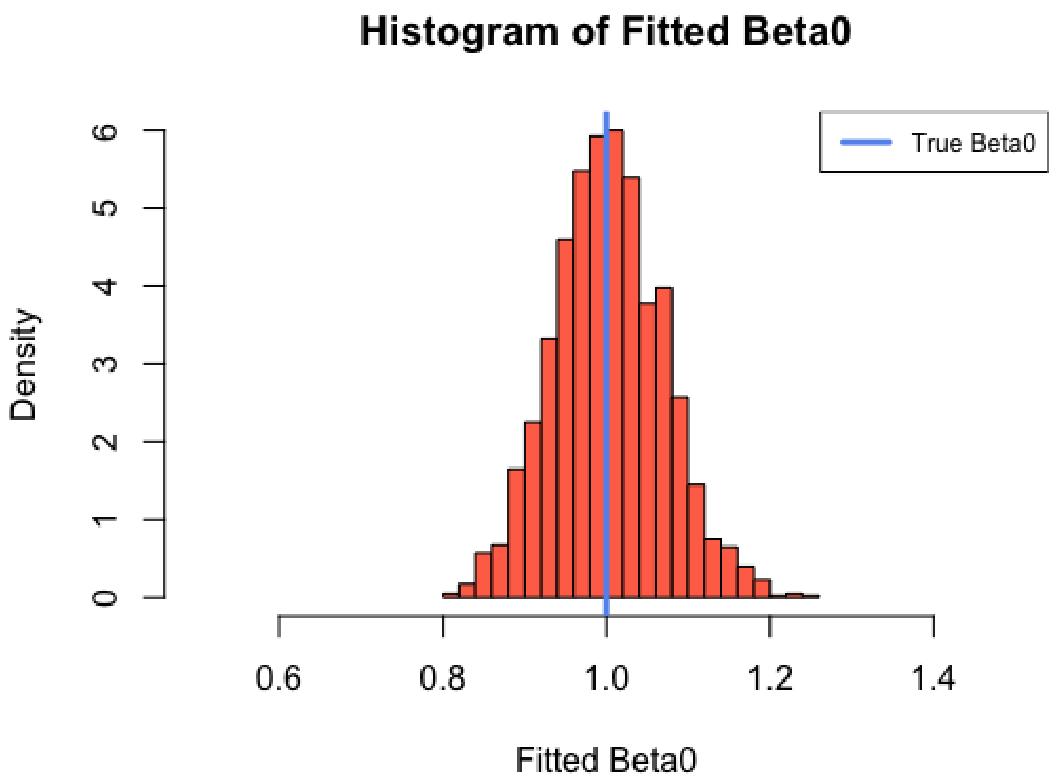
From previous histograms describing the distributions, we can perceive that $\widehat{\beta}_0$ and $\widehat{\beta}_1$ both follow normal distribution.

Because estimators $\bar{\beta}_1$ and $\bar{\beta}_0$ are derived from sample data which are generated based on true parameter value β_1 and β_0, by central limit theorem, values of the estimator should be normally distributed around true parameter.

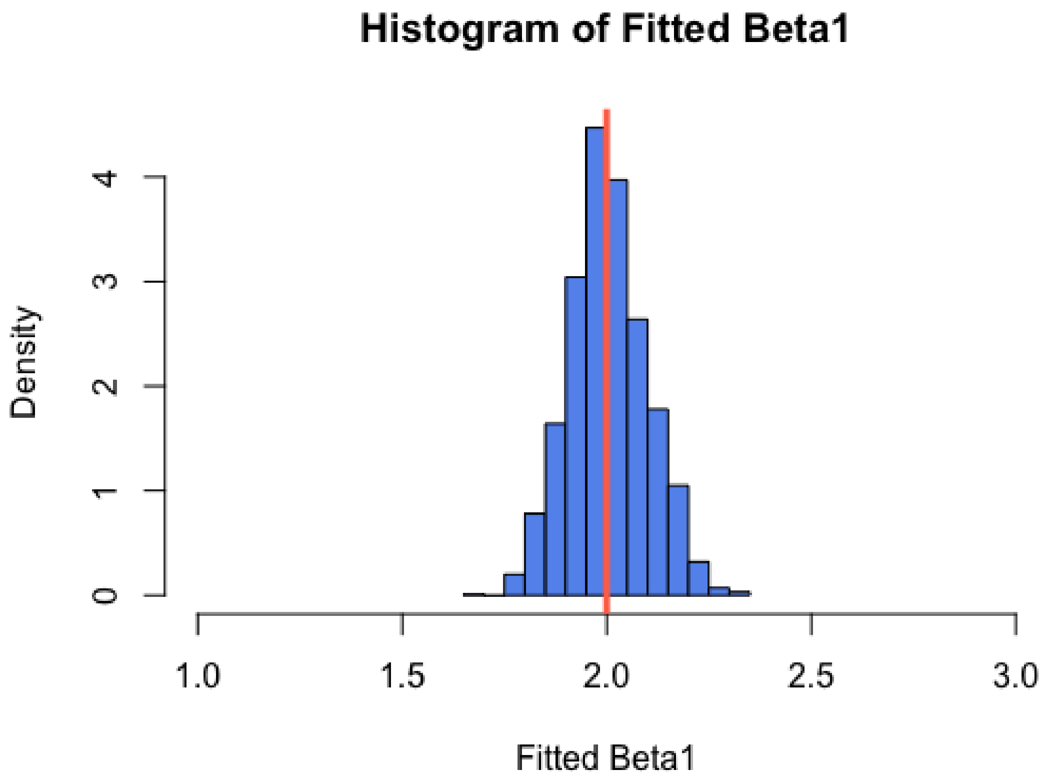
Q1-d

```
for (ii in 1:N) {
  x_value <- rnorm(n)
  y_value <- rbinom(n, size=1, p=bern_p(x_value))
  xy_logic = glm(y_value~x_value, family = "binomial")
  beta0_sim[ii] <- summary(xy_logic)$coefficient[1,1]
  beta1_sim[ii] <- summary(xy_logic)$coefficient[2,1]
}

hist(beta0_sim, probability=T, main='Histogram of Fitted Beta0', breaks=20,
     xlab='Fitted Beta0', col='coral1', xlim=c(0.5, 1.5))
abline(v=1, lwd=3, col='cornflowerblue' )
legend('topright', 'True Beta0', col='cornflowerblue', lwd=3, cex=0.75)
```




```
hist(beta1_sim, probability=T, main='Histogram of Fitted Beta1', breaks=20,
     xlab='Fitted Beta1', col='cornflowerblue', xlim=c(1, 3))
abline(v=2, lwd=3, col='coral1' )
legend('topright', 'True Beta1', col='coral1', lwd=3, cex=0.75)
```



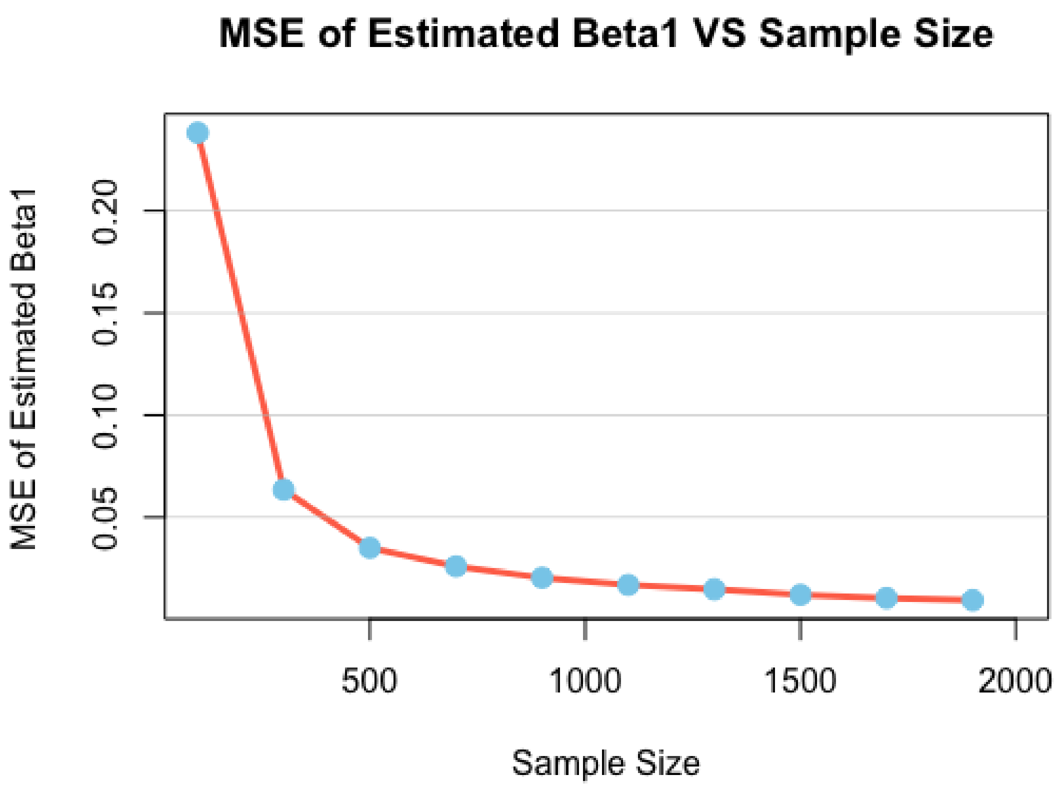
Both distributions concentrate more around true parameters, compare to sample size 500.

Q1-e

```
library(Metrics)
sample_sizes <- seq(from=100, to=2000, by=200)
beta1_mse <- rep(NA, length(sample_sizes))
N = 2000

for (ii in 1:length(sample_sizes)) {
  n <- sample_sizes[ii]
  beta1 <- rep(NA, N)
  for (jj in 1:N) {
    x_value <- rnorm(n)
    y_value <- rbinom(n, size=1, p=bern_p(x_value))
    xy_logic = glm(y_value~x_value, family = "binomial")
    beta1[jj] <- summary(xy_logic)$coefficient[2,1]
  }
  beta1_mse[ii] <- mse(beta1, 2)
}
```

```
plot(x=sample_sizes, y=beta1_mse, type='ln', lwd=3, col='coral1', xlim=c(100, 2000),
     xlab='Sample Size', ylab='MSE of Estimated Beta1',
     main='MSE of Estimated Beta1 VS Sample Size')
points(x=sample_sizes, y=beta1_mse, pch=20, cex=2, col="skyblue")
grid(nx=NA,ny=NULL,lty=1,lwd=0.5,col="gray")
```



This graph shows the MSE of $\widehat{\beta}_1$ converges to 0 when sample size increases.

Q2

Q2-a

$$
\begin{aligned}
\text{the cdf of X is } F_X(x) &= \frac{e^x}{1+e^x} \\
\text{the pdf of X } f_X(x) &= \frac{dF_X}{dx} \\
&= \frac{e^x}{(1+e^x)^2}
\end{aligned}
$$

Mean of random variable is $\mathbb{E}(X)$

$$
\begin{aligned}
\mathbb{E}(X) &= \int_{-\infty}^{\infty} f_X(x) \cdot x \, dx \\
&= -\frac{x}{1+e^x} + x - \log(e^x+1) \,|_{-\infty}^{\infty} \\
&= 0 - 0 = 0
\end{aligned}
$$

Median of random variable is the value of x where $F_X(x) = 0.5$

$$
\begin{aligned}
\frac{e^x}{1+e^x} &= 0.5 \\
e^x &= 1 \\
x &= \log(1) = 0
\end{aligned}
$$

Now we get the pdf of random variable X is $\frac{e^x}{(1+e^x)^2}$, and both mean and median equal to 0.

Q2-b

A rejection sampling method can be applied here. Choose Cauchy(0, 1) as proposal density p. M will be ratio of pdf of X to proposal density function.

$$
\begin{aligned}
M &\geq \sup \frac{f(x)}{p(x)} \\
&= \sup \frac{\frac{e^x}{(1+e^x)^2}}{\frac{1}{\pi}\frac{1}{1+x^2}} \\
&= \sup \frac{\pi e^x (1+x^2)}{(1+e^x)^2}
\end{aligned}
$$

The supremum of this ration is approximate 1.65, so choose $M = 1.7$.

Then generate random number Y from Cauchy(0, 1) and random number U from Uniform(0, 1)

If $U < \frac{f(Y)}{M \cdot p()Y}$, we may accept such Y value as a random variable X, otherwise, Y will not be accepted. We draw Y repeatedly from proposal density until we get enough random number X.

The probability of acceptance is $\frac{1}{M} \approx 0.6$.

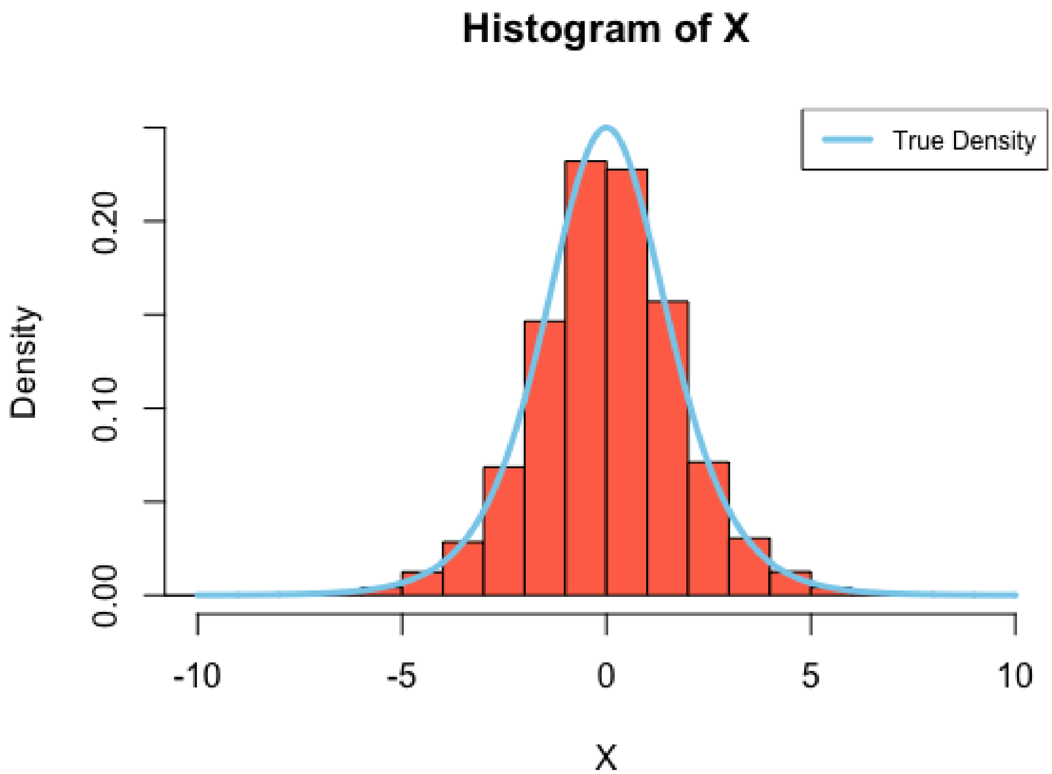
Q2-c

```
density_func <- function(x) {
  return(exp(x)/(1 + exp(x))^2)
}

M = 1.7
sim_size <- 20000
sim_U <- runif(sim_size)
sim_Y <- rcauchy(sim_size, 0, 1)
sim_X <- sim_Y[which(sim_U < density_func(sim_Y) / (M * dcauchy(sim_Y)))]

x_base <- seq(-10, 10, 0.01)
density_value <- density_func(x_base)

hist(sim_X, breaks=30, probability=T, xlim=c(-10,10), ylim=c(0, 0.25),
     xlab='X', main='Histogram of X', col='coral1')
lines(x_base, density_value, lwd=3, col='skyblue')
legend('topright', col='skyblue', legend='True Density', lwd=3, cex=0.75)
```



Since the acceptance rate is 0.6, 20000 simulated Y is enough to generate 10000 random number X. The histogram of X fits the density function.