

# R Final

박세은(202015635)

12/21/2021

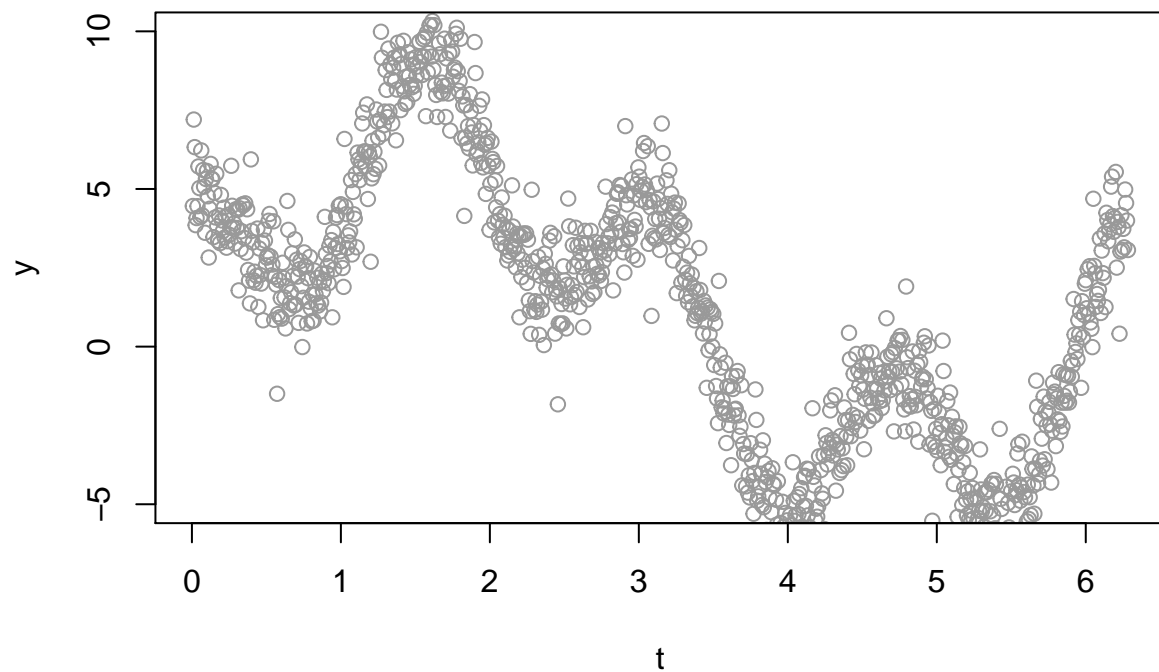
## 1번 문제풀이

```
#(1)
epsilon=rnorm(100)
head(epsilon)
```

```
## [1] -0.07166462  2.64302042  1.74174067 -0.75750580 -0.55606773 -0.20056171
```

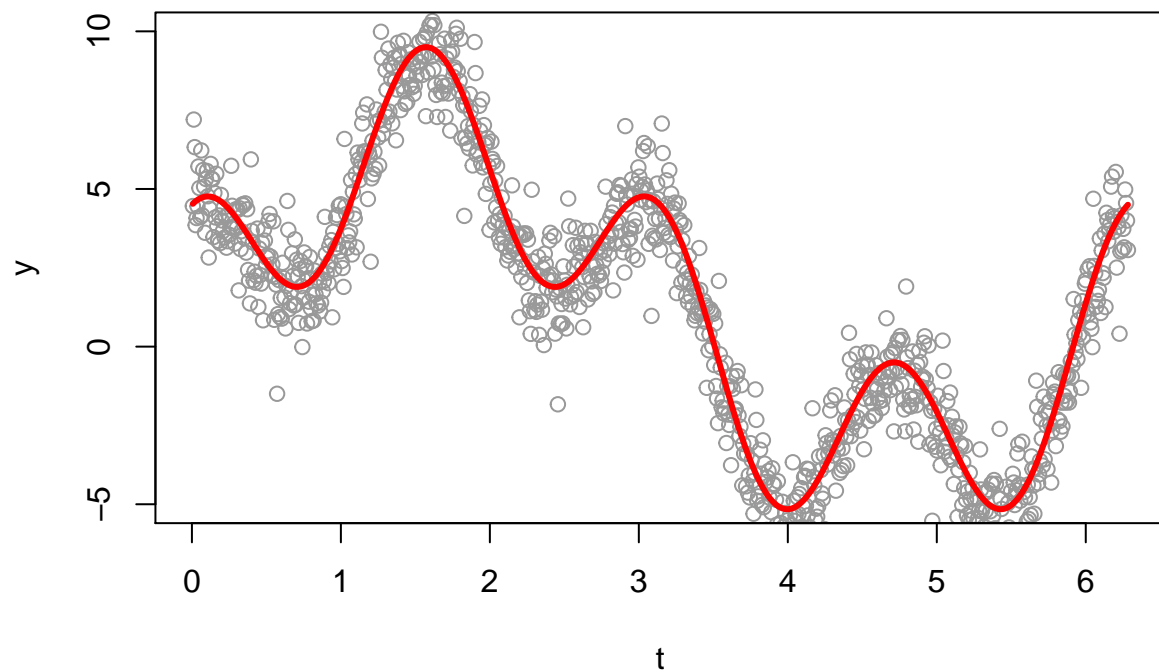
```
#(2)
i=1:1000
t=c()
t[i]=(2*pi*i)/1000
x_1=c()
x_1=sin(t[i])
x_2=c()
x_2[i]=cos(4*t[i])
```

```
#(3)
i=1:1000
t=c()
t[i]=(2*pi*i)/1000
y=c()
y[i]=1.5+5*x_1+3*x_2+epsilon
plot(t,y,col='gray60',ylim=c(-5,10))
```



```
#(4)
X=cbind(1,x_1,x_2)
dim(X)=c(1000,3)
```

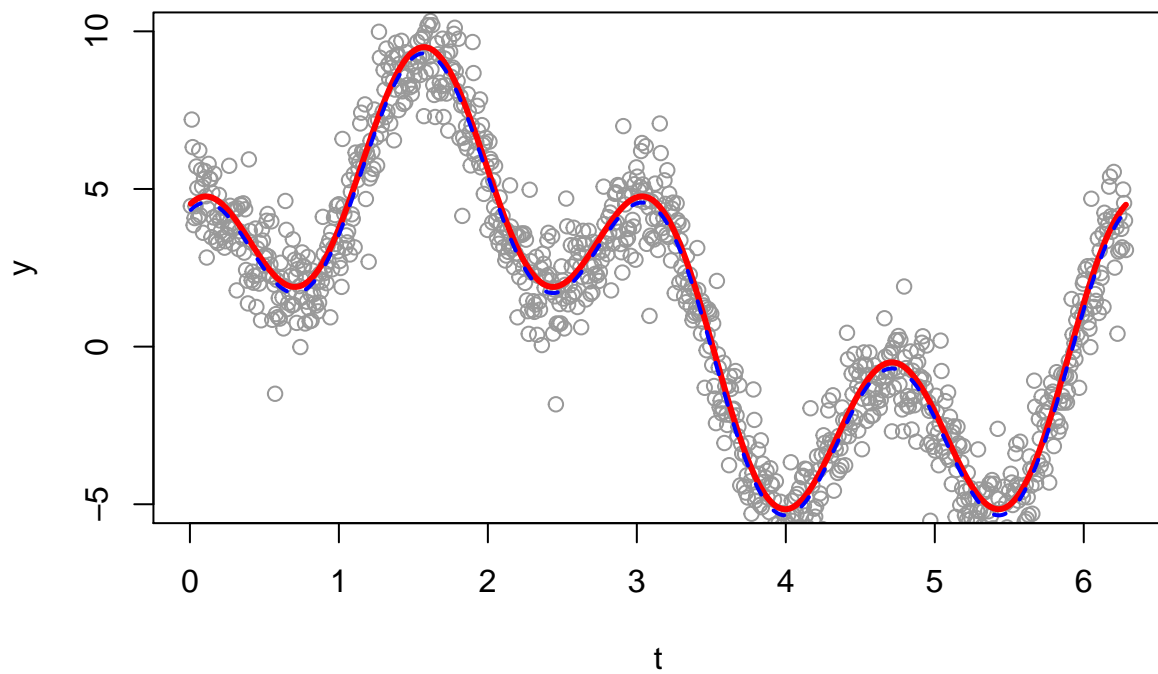
```
#(5)
beta = rbind(1.5,5,3)
XB <- X %*% beta
plot(t,y,col='gray60',ylim=c(-5,10))
lines(t,XB,col='red',lwd=3)
```



```
#(6)
y<-matrix(y)
B_hat = solve(t(X) %*% X)%*% t(X) %*% y
B_hat
```

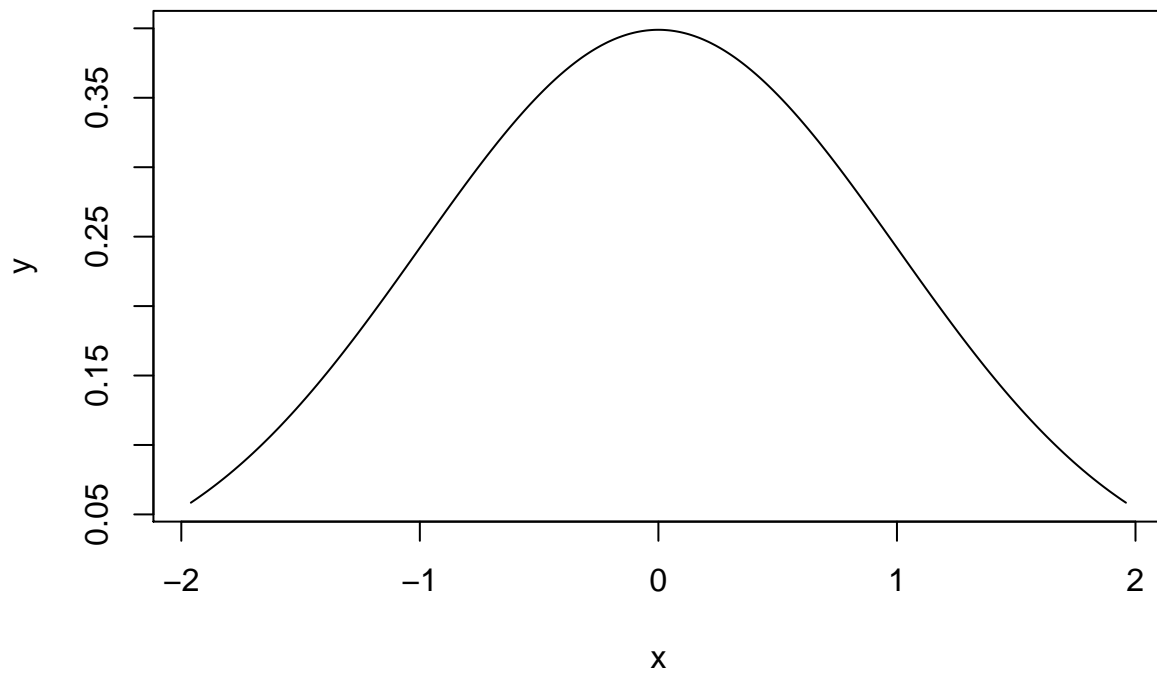
```
##           [,1]
## [1,]  1.304838
## [2,]  5.000000
## [3,]  3.000000
```

```
#(7)
XB_hat <- X %*% B_hat
plot(t,y,col='gray60',ylim=c(-5,10))
lines(t,XB,col='red',lwd=3)
lines(t,XB_hat,col='blue',lty=2,lwd=2)
```

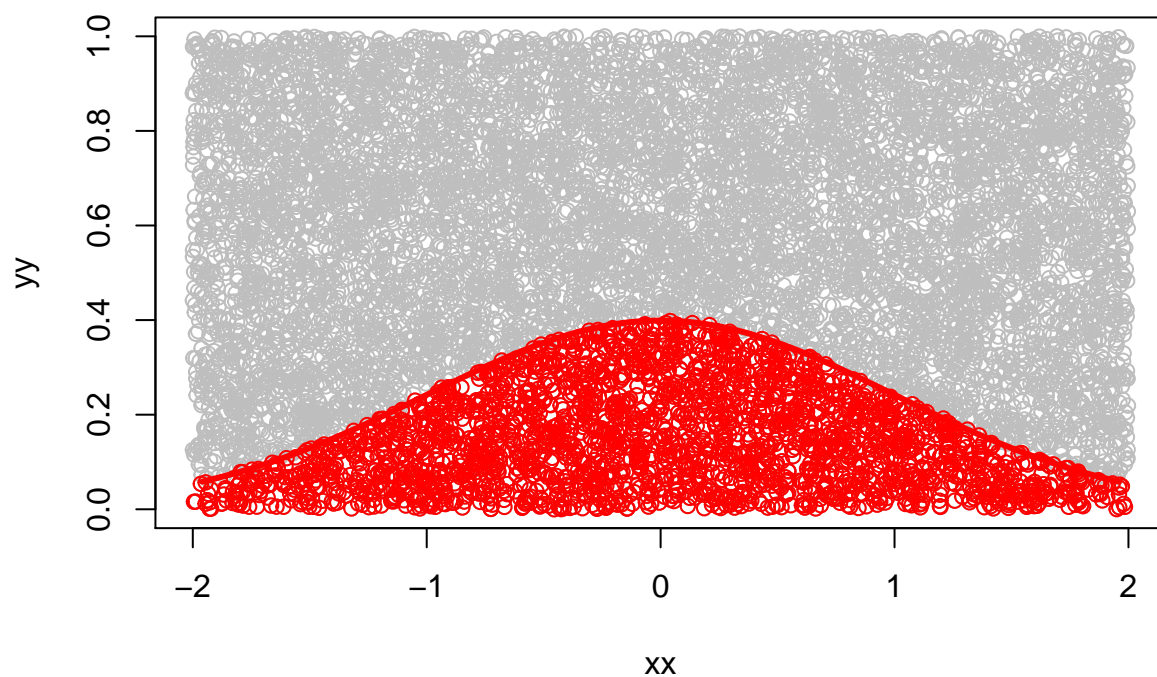


## 2번 문제풀이

```
#(1)
x=seq(from=-1.96,to=1.96,by=0.01)
y= 1/sqrt(2*pi)*exp(-1/2 * x^2)
plot(x,y,type='l')
```



```
xx=runif(n=10000,min=-2,max=2)
yy=runif(n=10000)
plot(xx,yy,col='gray')
lines(x,y,col='red',lwd=3)
test = function(xx,yy) yy < 1/sqrt(2*pi) * exp(-1/2 * xx^2)
tst=c()
for(i in 1:10000) tst[i]=test(xx[i],yy[i])
points(xx[tst],yy[tst],col='red')
```



```
sum(tst)/10000*4
```

```
## [1] 0.9608
```

```
##(2)
```

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.3      v purrr  0.3.4
## v tibble  3.1.1      v dplyr  1.0.6
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   1.4.0      v forcats 0.5.1
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
var=rnorm(1000)
test_2 = function(x){
  x>=-1.96 & x<=1.96
}
test_2(var) %>% sum
```

```
## [1] 945
```

### 3번 문제풀이

```
# Type A
rslt <- c()
for(i in 1:10000){
  x = sum(cumprod(rbinom(20,size=1,prob=0.5)))
  x_ = 20-x-1
  if(x_>0) y = sum(cumprod(rbinom(x_,size=1,prob=0.95)))
  y_ = x_-y-1
  if(y_>0) z = sum(cumprod(rbinom(y_,size=1,prob=0.5)))
  rslt[i] <- z >= y_
}
sum(rslt)/10000
```

```
## [1] 0.4347
```

```
# Type B
rslt_2 <- c()
for(i in 1:10000){
  a = sum(cumprod(rbinom(20,size=1,prob=0.5)))
  a_ = 20-a-1
  if(a_>0) b = sum(cumprod(rbinom(a_,size=1,prob=0.5)))
  b_ = a_-b-1
  if(b_>0) c = sum(cumprod(rbinom(b_,size=1,prob=0.5)))
  c_ = b_-c-1
  if(c_>0) d = sum(cumprod(rbinom(c_,size=1,prob=0.5)))
  d_ = c_-d-1
  if(d_>0) e = sum(cumprod(rbinom(d_,size=1,prob=0.5)))
  e_ = d_-e-1
  if(e_>0) f = sum(cumprod(rbinom(e_,size=1,prob=0.5)))
  f_ = e_-f-1
  if(f_>0) g = sum(cumprod(rbinom(f_,size=1,prob=0.5)))
  g_ = f_-g-1
  if(g_>0) h = sum(cumprod(rbinom(g_,size=1,prob=0.5)))
  rslt_2[i] <- h >= g_
}
sum(rslt_2)/10000
```

```
## [1] 0.1311
```

# 따라서 Type A에서 8번 참가자가 살아남을 확률이 높다.



## 4번 문제풀이

```
df=read_csv('https://raw.githubusercontent.com/guebin/2021IR/master/_notebooks/covid19.c
```

```
##
## -- Column specification -----
## cols(
##   year = col_double(),
##   month = col_double(),
##   day = col_double(),
##   prov = col_character(),
##   cases = col_double()
## )
```

```
head(df)
```

```
## # A tibble: 6 x 5
##   year month   day prov  cases
##   <dbl> <dbl> <dbl> <chr> <dbl>
## 1  2020     1    20 서울     0
## 2  2020     1    20 부산     0
## 3  2020     1    20 대구     0
## 4  2020     1    20 인천     1
## 5  2020     1    20 광주     0
## 6  2020     1    20 대전     0
```

```
##(1)
df %>% filter(year == 2020) %>% summarise(sum_2020=sum(cases))
```

```
## # A tibble: 1 x 1
##   sum_2020
##   <dbl>
## 1    60726
```

```
df %>% filter(year == 2021) %>% summarise(sum_2021=sum(cases))
```

```
## # A tibble: 1 x 1
##   sum_2021
##   <dbl>
## 1   396886
```

```
#(2)
df_fh <- df %>%
  group_by(prov) %>%
  filter(month==2 & day<16) %>%
  summarise(sum_cases=sum(cases))
df_fh$prov[which.max(df_fh$sum_cases)]
```

```
## [1] "서울"
```

```
#(3)
df_sh <- df %>%
  group_by(prov) %>%
  filter(month==2 & day>=16) %>% summarise(sum_cases=sum(cases))
df_sh$prov[which.max(df_sh$sum_cases)]
```

```
## [1] "대구"
```