

R입문__기말고사__201922050

한진석

12/21/2021

#1. 회귀분석

#1-1.

```
a= c(rnorm(1000,0,1))
head(a)
```

```
## [1]  0.4721210 -0.4745230 -0.2339239  1.7665515  1.1098864 -0.9689571
```

#1-2.

```
t=c()
for(i in 1:1000){
  t[i] = 2*i*pi/1000
}
head(t)
```

```
## [1] 0.006283185 0.012566371 0.018849556 0.025132741 0.031415927 0.037699112
```

```
x1 = c()
x2 = c()
for(i in 1:1000){
  x1[i] = sin(t[i])
  x2[i] = cos(4*t[i])
}
head(x1)
```

```
## [1] 0.006283144 0.012566040 0.018848440 0.025130095 0.031410759 0.037690183
```

```
head(x2)
```

```
## [1] 0.9996842 0.9987370 0.9971589 0.9949510 0.9921147 0.9886517
```

#1-3.

```
y = c()
for(i in 1:1000){
  y[i] = 1.5+5*x1[i] + 3*x2[i] + a[i]
}
```

```

head(y)

## [1] 5.002589 4.084518 4.351795 6.377055 5.743284 3.685449

plot(t,y, col='gray60')

#1-4.
X= cbind(1,x1,x2)
head(X)

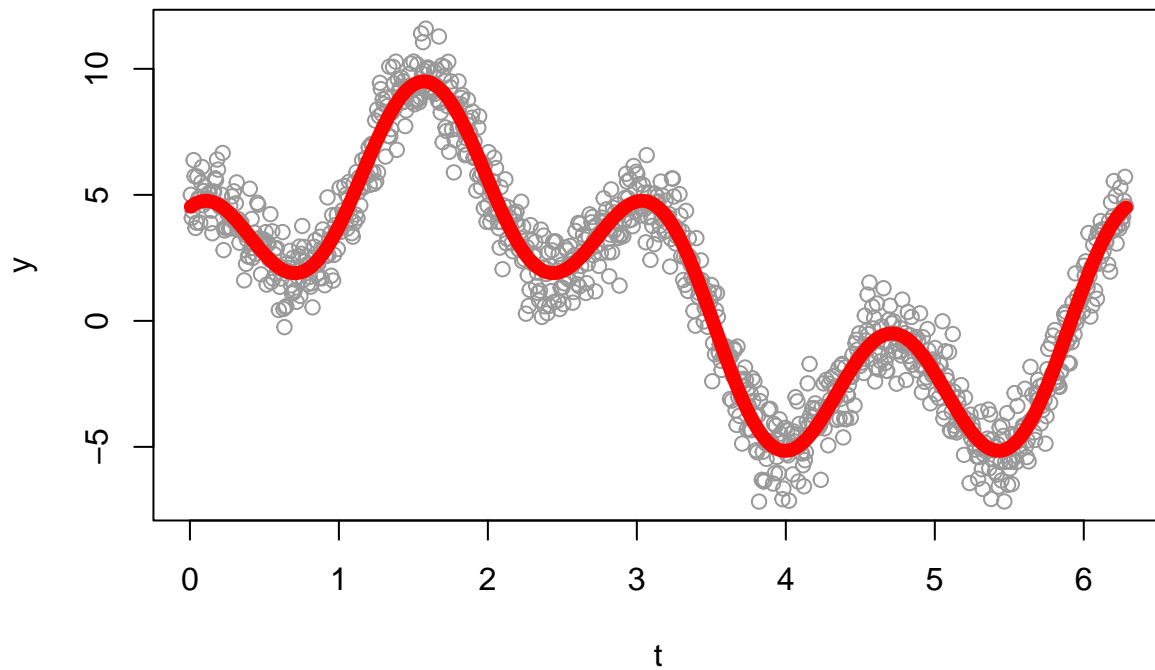
##           x1      x2
## [1,]  1 0.006283144 0.9996842
## [2,]  1 0.012566040 0.9987370
## [3,]  1 0.018848440 0.9971589
## [4,]  1 0.025130095 0.9949510
## [5,]  1 0.031410759 0.9921147
## [6,]  1 0.037690183 0.9886517

#1-5.
Beta=c(1.5,5,3)
dim(Beta)=c(3,1)
head(Beta)

##      [,1]
## [1,]  1.5
## [2,]  5.0
## [3,]  3.0

XB = X %*% Beta
plot(t,y, col='gray60')
lines(t,XB, col='red', lwd=7)

```



#1-6.

```
B_hat = solve(t(X)%*%X) %*% t(X) %*% y
head(B_hat)
```

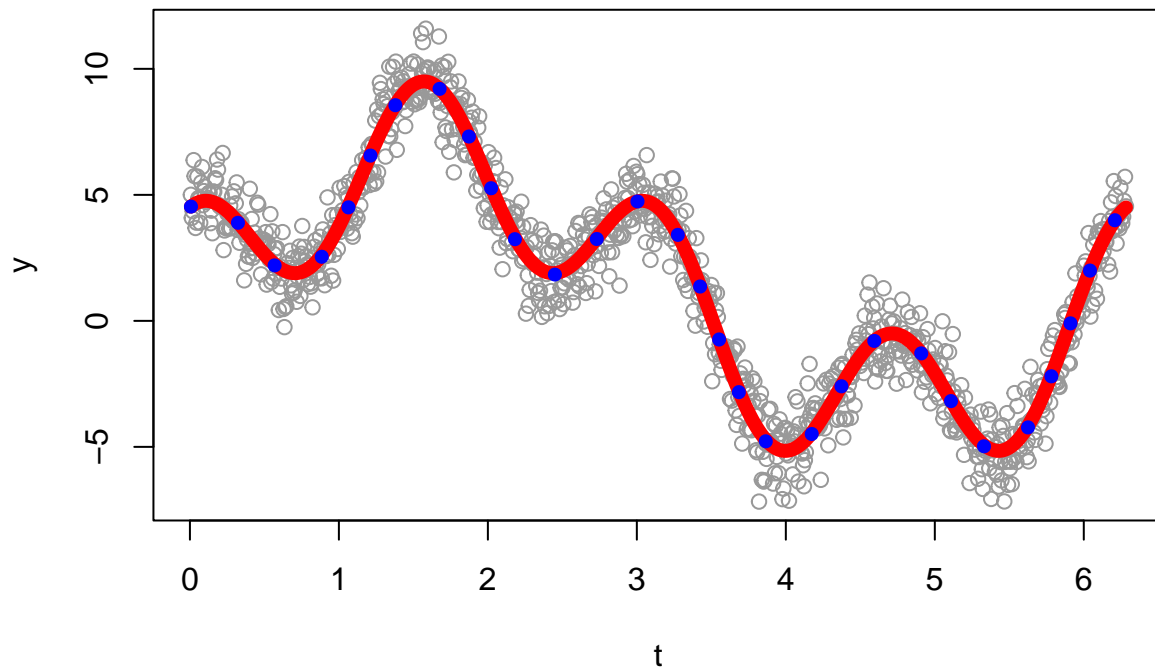
```
##          [,1]
## 1.469494
## x1 4.996549
## x2 3.030334
```

#1-7.

```
XB_hat = X %*% B_hat
head(XB_hat)
```

```
##          [,1]
## [1,] 4.530264
## [2,] 4.558786
## [3,] 4.585395
## [4,] 4.610091
## [5,] 4.632877
## [6,] 4.653759
```

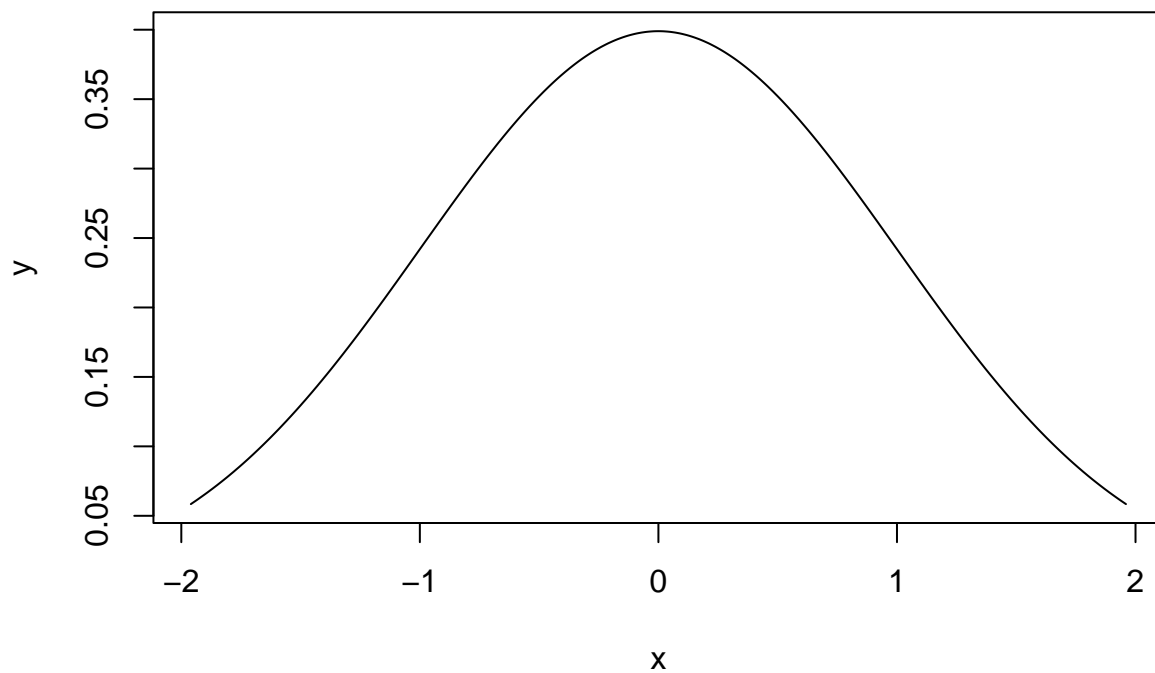
```
plot(t,y, col='gray60')
lines(t,XB, col='red', lwd=7)
lines(t,XB_hat, col='blue', lty=3, lwd=7)
```



#2. 몬테카를로 적분

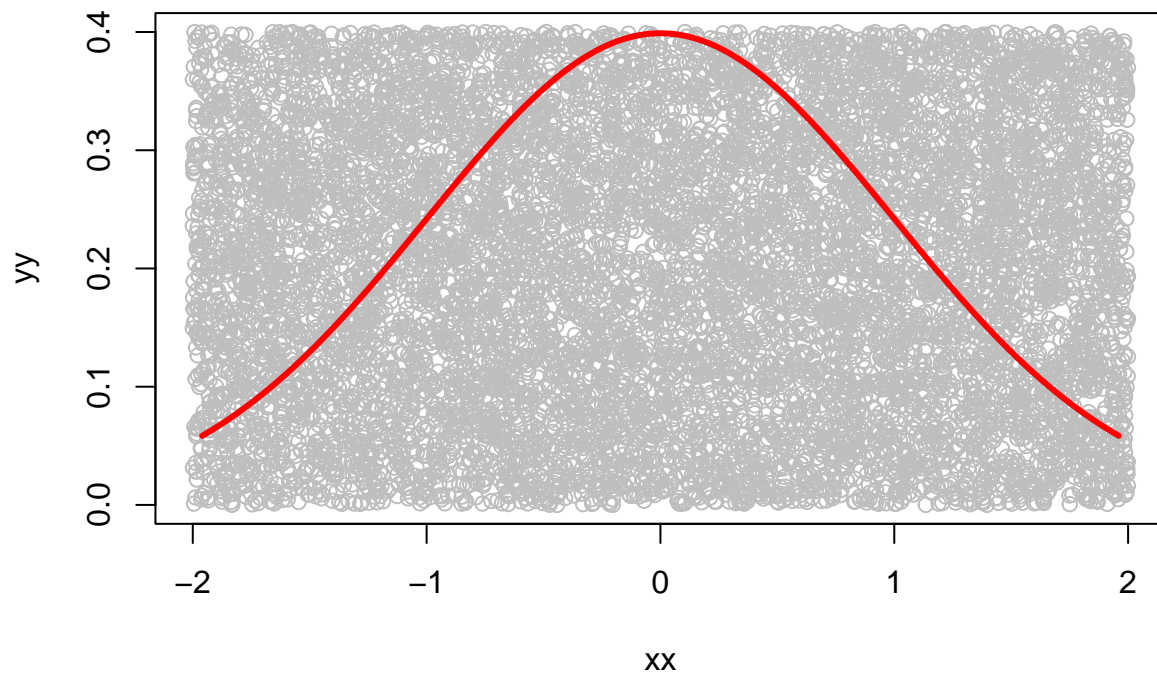
#2-1.

```
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(10000, min=-2, max =2)
yy=runif(10000)*0.4
```

```
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])
}
plot(xx,yy,col='gray')
lines(x,y,col='red',lwd=3)
points(xx[tst],yy[tst],col='red')
sum(tst)
```

```
## [1] 5906
```

```
(sum(tst)/10000) * 4*0.4
```

```
## [1] 0.94496
```

#2-2.

```
x = rnorm(1000,0,1)
a=ifelse(x<=1.96&x>=-1.96,1,0)
head(a)
```

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

```
sum(a)
```

```
## [1] 955
```

```
#3. 징검다리
```

```
library(tidyverse)
```

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

```
## v ggplot2 3.3.5      v purrr  0.3.4
```

```
## v tibble  3.1.6      v dplyr  1.0.7
```

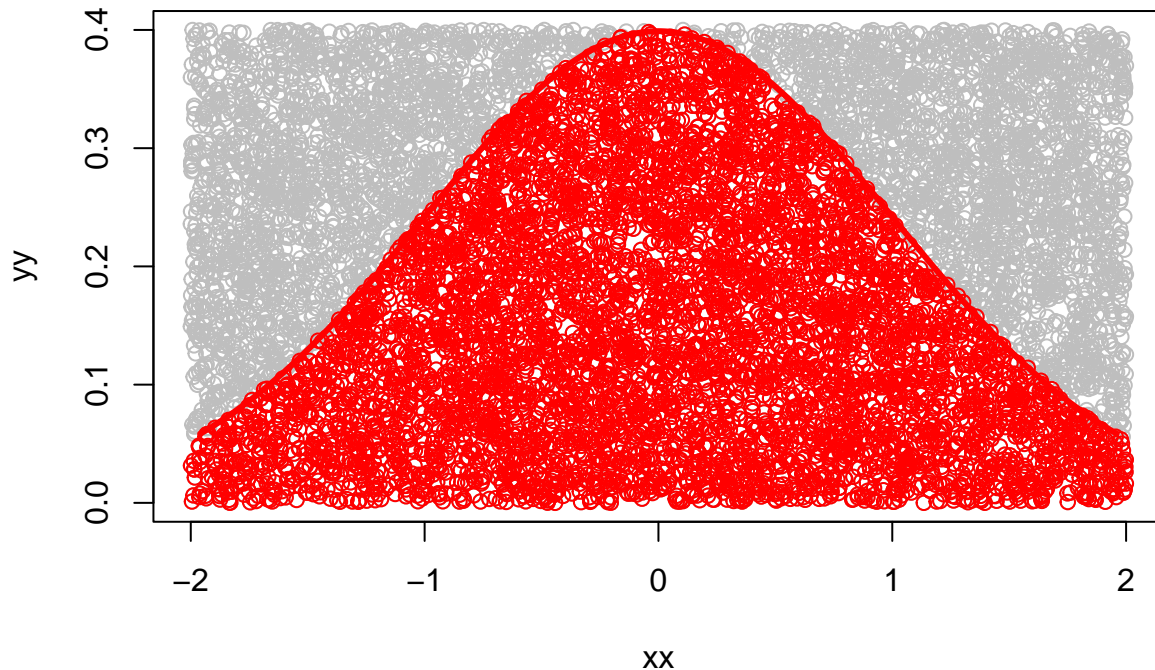
```
## v tidyr   1.1.4      v stringr 1.4.0
```

```
## v readr   2.1.1      v forcats 0.5.1
```

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

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

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



```
#3. 징검다리
```

```
library(tidyverse)
```

```
#type A
```

```
m=c()
```

```
for(i in 1:10000){
```

```
  chance=2
```

```
  success=0
```

```
  while(TRUE){
```

```
    if(chance!=1 && chance!=0){
```

```
      if(runif(1)<=0.5){
```

```

        succeess=succeess+1
    }
    else{
        chance=chance-1
        succeess=succeess+1
    }
}
else if(chance ==1){
    if(runif(1)<=0.95){
        succeess=succeess+1
    }
    else{
        chance=chance-1
        succeess=succeess+1
    }
}
else{
    if(runif(1)<=0.5){
        succeess=succeess+1
    }
    else {
        chance=chance-1
    }
}
if(chance== -1){
    m[i]=0
    break
}
if(succeess==20){
    m[i]=1
    break
}
}
}
perc_saveA = sum(m)/10000
perc_saveA

```

```
## [1] 0.4412
```

```

#Tyoe B
t=c()
k=c()
for(s in 1:10000){
    a = 20
    d=c()

```

```

for(i in 1:8){
  b = rbinom(a,size=1,0.5)
  d[i] = sum(cumprod(b))
  a = a-d[i]-1
  if(a<=0) break
}
if(a<=0){
  g=1} else{g=0
}
k[s]=(g)
}
perc_saveB = sum(k)/10000
perc_saveB

```

```
## [1] 0.18
```

```
library(tidyverse)
```

```
#4. COVID19
```

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

```
## Rows: 12294 Columns: 5
```

```
## -- Column specification -----
```

```
## Delimiter: ","
```

```
## chr (1): prov
```

```
## dbl (4): year, month, day, cases
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
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
```

```
#4-1.
```

```
df2020 = df[df$year == 2020,]
```

```
df2020
```

```
## # A tibble: 6,246 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
##  7  2020     1    20 울산     0
##  8  2020     1    20 세종     0
##  9  2020     1    20 경기     0
## 10  2020     1    20 강원     0
## # ... with 6,236 more rows
```

```
sum(df2020$cases)
```

```
## [1] 60726
```

```
df2021 = df[df$year == 2021,]
df2021
```

```
## # A tibble: 6,048 x 5
##      year month   day prov  cases
##      <dbl> <dbl> <dbl> <chr> <dbl>
##  1  2021     1     1 서울    357
##  2  2021     1     1 부산     57
##  3  2021     1     1 대구     43
##  4  2021     1     1 인천     65
##  5  2021     1     1 광주     17
##  6  2021     1     1 대전     11
##  7  2021     1     1 울산     43
##  8  2021     1     1 세종      1
##  9  2021     1     1 경기    284
## 10  2021     1     1 강원     30
## # ... with 6,038 more rows
```

```
sum(df2021$cases)
```

```
## [1] 396886
```

#4-2.

```
df2020_2 = df2020[df2020$month==2,]
df2.1_2.15=df2020_2[df2020_2$day<=15,]
```

```
df2.1_2.15 %>%
  group_by(prov) %>%
  summarise(count = n(),
            case = sum(cases))
```

```
## # A tibble: 18 x 3
##   prov   count  case
##   <chr> <int> <dbl>
## 1 강원     15     0
## 2 검역     15     0
## 3 경기     15     9
## 4 경남     15     0
## 5 경북     15     0
## 6 광주     15     2
## 7 대구     15     0
## 8 대전     15     0
## 9 부산     15     0
## 10 서울     15     5
## 11 세종     15     0
## 12 울산     15     0
## 13 인천     15     0
## 14 전남     15     1
## 15 전북     15     0
## 16 제주     15     0
## 17 충남     15     0
## 18 충북     15     0
```

```
#4-3.
df2.15_2.29 = df2020_2[df2020_2$day>15,]
df2.15_2.29 %>%
  group_by(prov) %>%
  summarise(count=n(),
            case = sum(cases))
```

```
## # A tibble: 18 x 3
##   prov   count  case
##   <chr> <int> <dbl>
## 1 강원     14     7
## 2 검역     14     0
## 3 경기     14    65
## 4 경남     14    59
## 5 경북     14   472
## 6 광주     14     7
## 7 대구     14  2055
## 8 대전     14    13
## 9 부산     14    75
## 10 서울     14    62
## 11 세종     14     1
## 12 울산     14    17
## 13 인천     14     5
```

```
## 14 전남      14      1
## 15 전북      14      4
## 16 제주      14      2
## 17 충남      14     48
## 18 충북      14     10
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

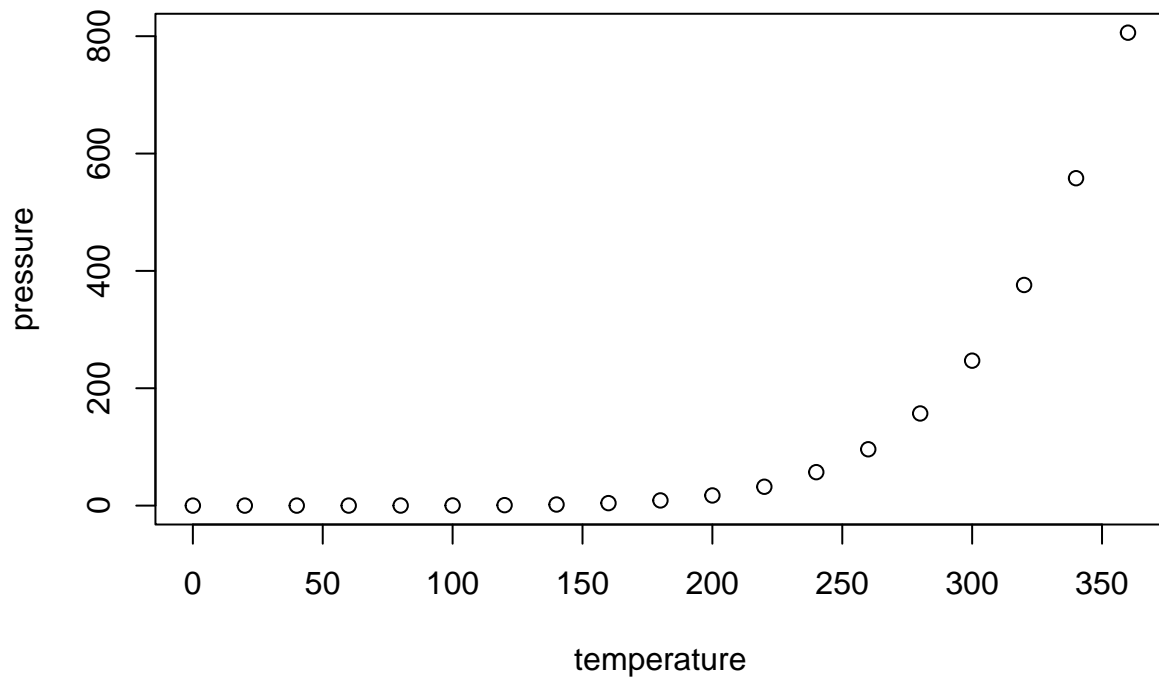
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0    Min.   :  2.00
##  1st Qu.:12.0    1st Qu.: 26.00
##  Median :15.0    Median : 36.00
##  Mean   :15.4    Mean    : 42.98
##  3rd Qu.:19.0    3rd Qu.: 56.00
##  Max.   :25.0    Max.    :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.