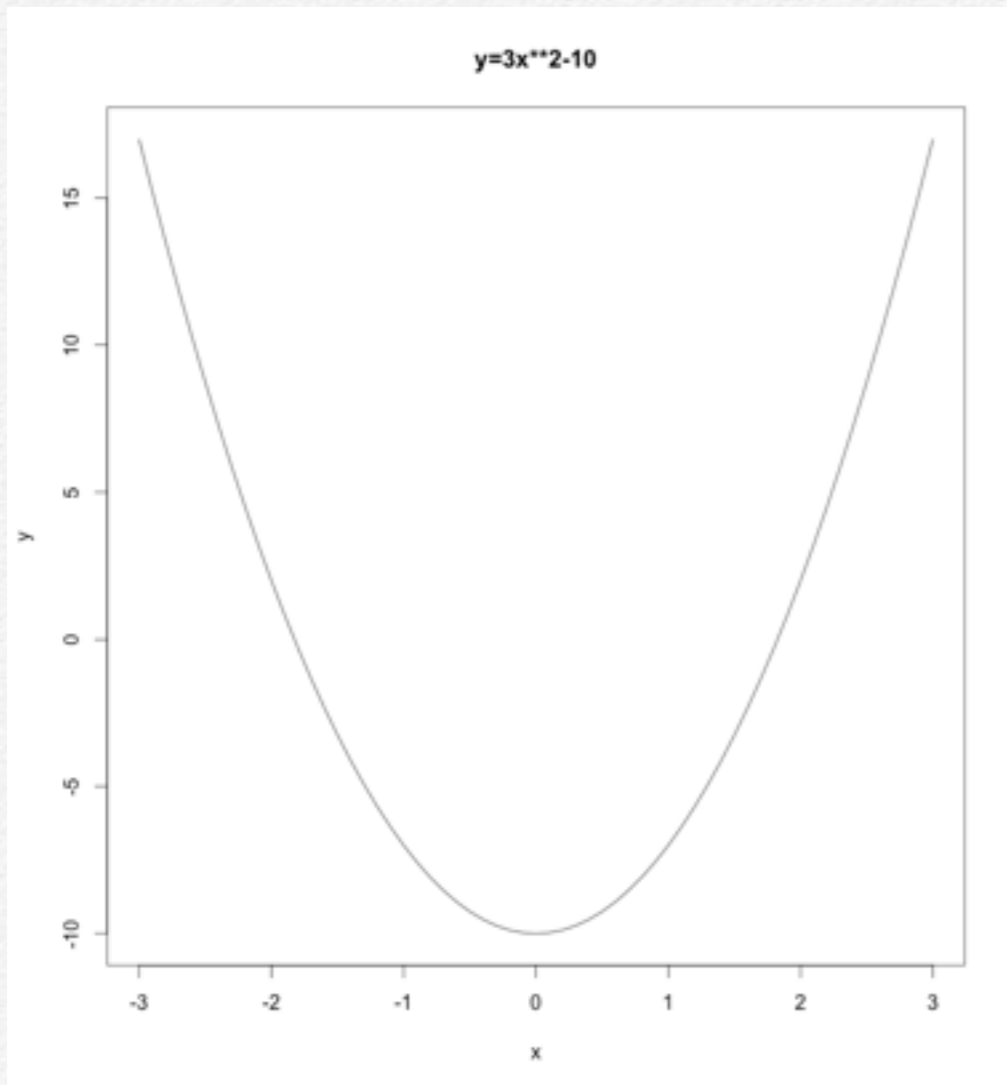


# R 활용 \_ 그래프

---

## 함수 그리기 - 함수 plot()

```
#y=3x^2-10 그래프 그리기
x=seq(-3,3,0.1) #x 범위 지정
y=3*x^2-10 #함수값
plot(x,y,main="y=3x**2-10",type="l")
```



## 스크립트 설명

- 함수 `seq(s,e,by)` - 시작 값  $s$ , 증가 값  $by$ 로, 종료 값  $e$ 까지 증가  
-3, -2.9, -2.8 ..., 2.9, 3 이 값을 오브젝트  $x$ 에 저장 -  $by$ 사용하지 않으면 1씩 증가,  $x$ -축의 범위지정
- $y=3*x^2+7$   
+ 오브젝트  $x$ 의 값을 입력하여 계산된 값을 오브젝트  $y$ 에 저장
- 함수 `plot(x, y, 옵션)` - 2차원 그래프 그리기

## 가능한 옵션

`main=""` - 그래프 제목

`ylab=""` - y-축 제목 붙이기      `xlab=""`

`ylim=c(s,e)` - y-축 시작 값  $s$ , 종료 값  $e$  지정      `ylim=c( , )`

`type=""`

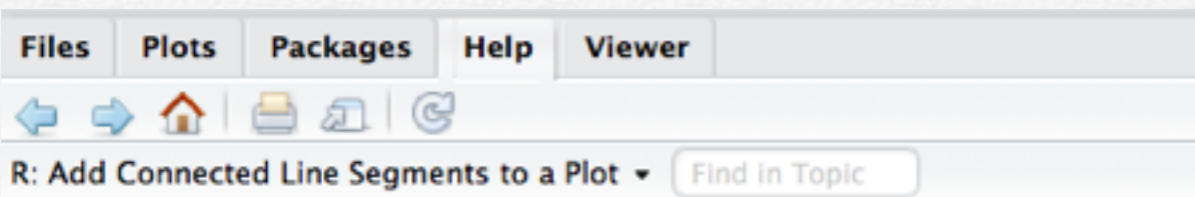
- "p" for points,
- "l" for lines,
- "b" for both,
- "c" for the lines part alone of "b",
- "o" for both 'overplotted',
- "h" for 'histogram' like (or 'high-density') vertical lines,
- "s" for stair steps,
- "S" for other steps, see 'Details' below,
- "n" for no plotting.

## 그래프 겹쳐 그리기 - lines() 함수 이용

```
x=seq(-3,3,0.1) #x 범위 지정
y=3*x^2-10 #함수값
plot(x,y,main="y=3x**2-10",type="l")
y2=x; lines(x,y2)
```

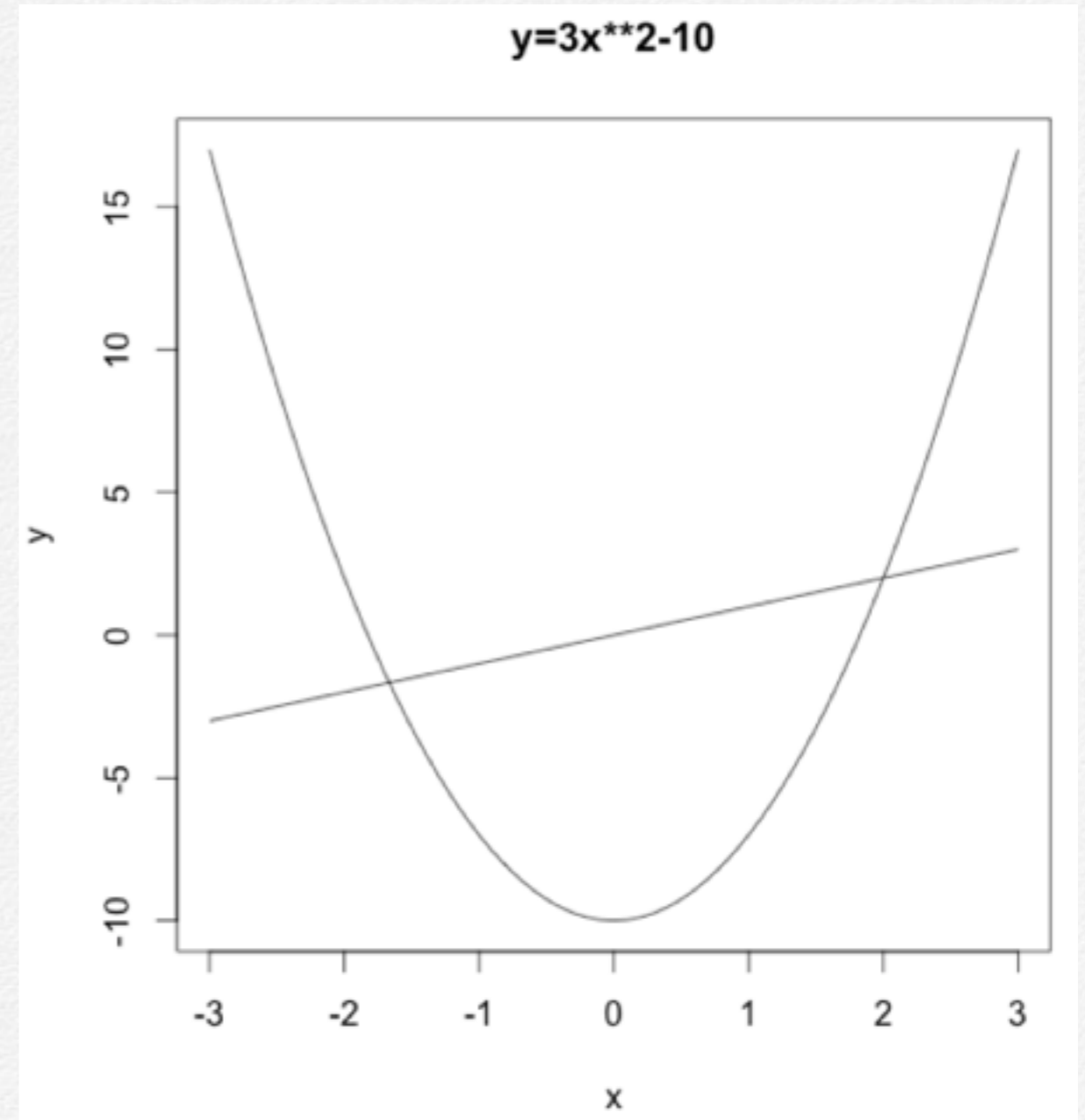
### 스크립트 설명

- # - 설명을 위하여 사용되는 것으로 이후 내용은 스크립트가 아닌 설명 문장으로 인식하여 실행하지 않는다
- ; - (세미콜론) R 스크립트는 라인(행) 단위로 실행됨, 짧은 라인의 경우 연속하여 쓰려면 라인 구별 “;”을 사용한다.
- 함수 lines()은 직선을 그리는 것으로 옵션은 plot() 함수 그대로 사용할 수 있다.
- 함수 help() - 도움말 기능, help(lines)을 실행하면 하단에 도움말 보임



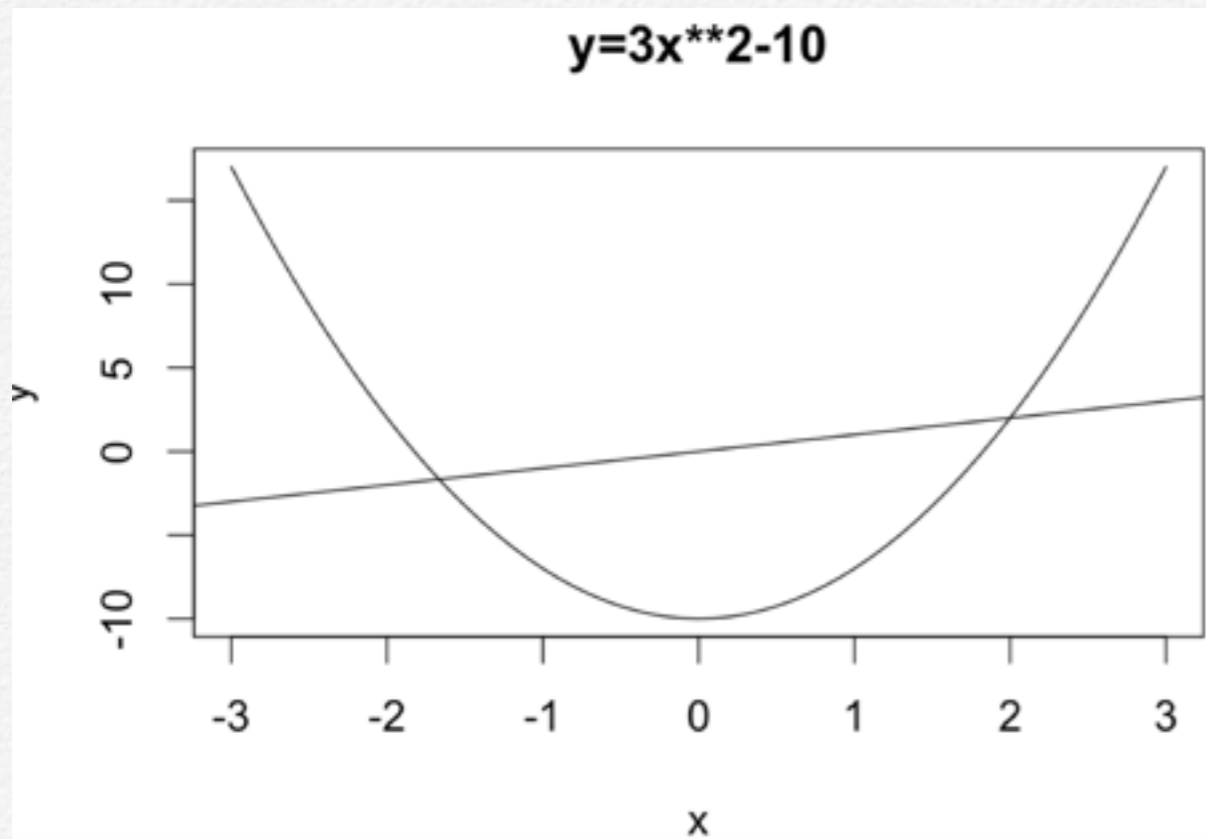
lines {graphics}

## Add Connected Line Segments to a Plot



## 그래프 겹쳐 그리기 - abline() 함수 이용

```
x=seq(-3,3,0.1) #x 범위 지정  
y=3*x^2-10 #함수값  
plot(x,y,main="y=3x**2-10",type="l")  
abline(a=0,b=1)
```

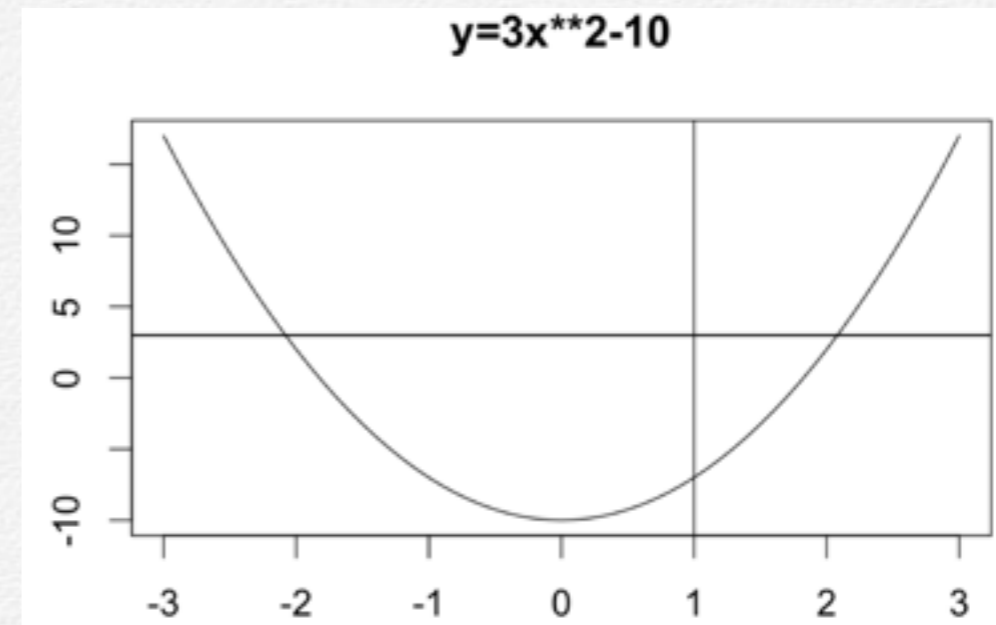


## 스크립트 설명

- 함수 abline() - 직선 그래프 겹쳐 그리기

### 옵션 설명

- a - 절편 값 설정
- b - 기울기 설정
- h - horizontal - 수평선 설정
- v - vertical - 수직선 설정 (예) `abline(h=3,v=1)`

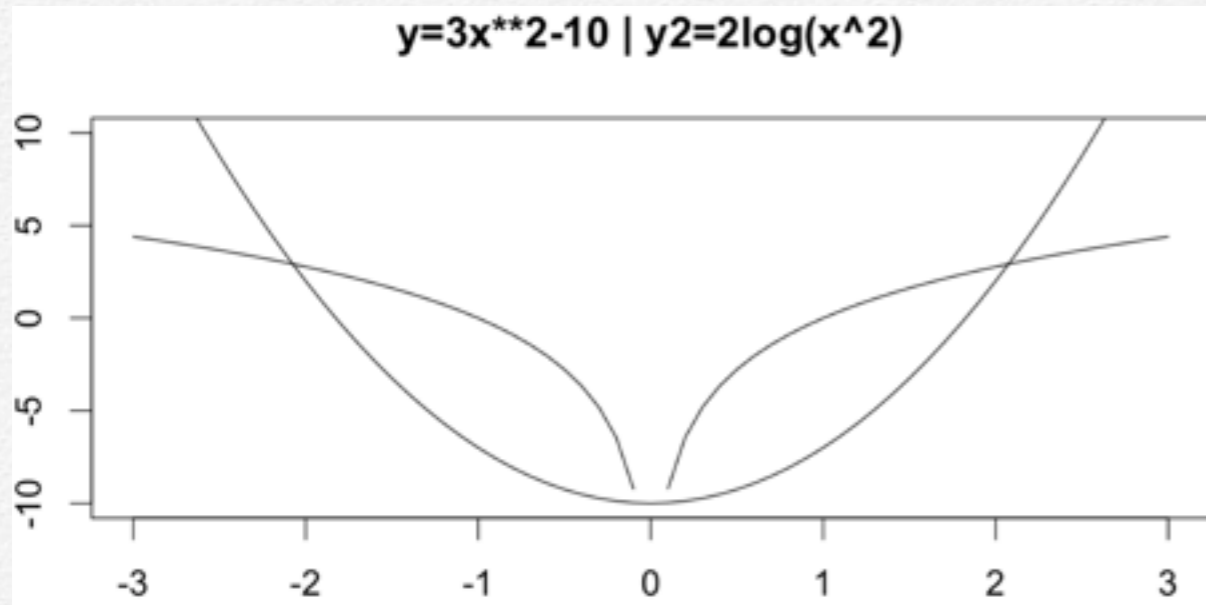


## 직선 아닌 두 그래프 동시에 그리기

```
x=seq(-3,3,0.1) #x 범위 지정
y=3*x^2-10 #함수값
y2=2*log(x^2) #함수값
plot(x,y,main="y=3x**2-10/y2=2log(x^)",type="l",ylim=c(-10,10))
lines(x,y2,type="l",ylim=c(-10,10))
```

### 설명

- 두 그래프의 y-축 값이 다르므로 축 값을 지정하는 ylim=c(s, e) 함수를 사용한다. (s=최저 값, e=최대값) - xlim=c(s, e) 사용가능
- main= 옵션은 첫 그래프에만 적어주면 된다.

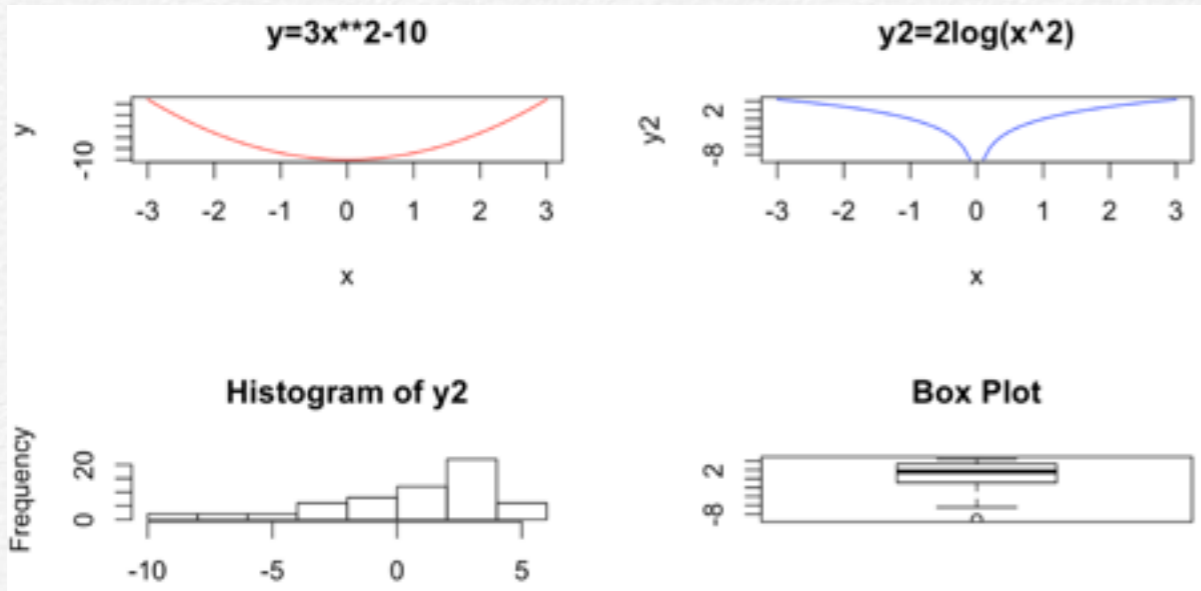


## 화면분할 그래프 그리기

```
x=seq(-3,3,0.1) #x 범위 지정
y=3*x^2-10 #함수값
y2=2*log(x^2) #함수값
par(mfrow=c(2,2))
plot(x,y,main="y=3x**2-10",type="l",col="red")
plot(x,y2,main="y2=2log(x^2)",type="l",col="blue")
hist(y2)
boxplot(y2, main="Box Plot of Y2")
dev.off()
```

### 함수 설명

- 함수 par() - 그래프 화면을 분할 하는 함수 => mfrow=c(2,2)는 화면을 2X2 4개 스크린으로 분할된다.
- 함수 dev.off() - 화면 분할 reset, 하나의 화면에 한 그래프로 전환
- 함수 hist() - 데이터(Y2 의 값)의 히스토그램을 그린다
- 함수 boxplot() - 데이터(Y2 의 값)의 나무상자를 그린다



### 함수 Split.plot() 사용

```
split.screen(c(2,2))
screen(1);plot(x,y,main="y=3x**2-10",type="l",col="red")
screen(2);plot(x,y2,main="y2=2log(x^2)",type="l",col="blue")
screen(3);hist(y2)
screen(4);boxplot(y2, main="Box Plot of Y2")
dev.off()
```

- 동일한 결과 - c(2,2) 2행 2열 화면 분할 옵션

### 함수 Layout() 사용

```
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
boxplot(y2, main="Box Plot of Y2")
plot(x,y,main="y=3x**2-10",type="l",col="red")
plot(x,y2,main="y2=2log(x^2)",type="l",col="blue")
```

- matrix() - 스크린 분할 개수 설정
- 2, 2 - 행 2개, 열 2개 화면 분할
- byrow=T - 행부터 먼저 그래프 출력, (반대) FALSE
- c(1, 1, 2, 3) - 1행1열 그래프1, 1행2열 그래프1, 2행1열 그래프2, 2행2열 그래프3

