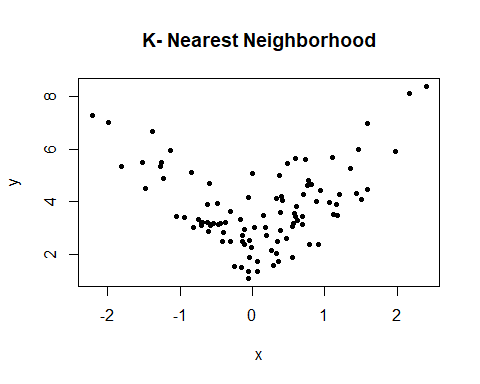
R-Plot

JeongHo LEE

## R-Plot

set.seed(1)  
x <- sort(rnorm(100))  
y<- 3+x^2 + rnorm(100)  
library(FNN)  
eval.n = 100  
eval.point = seq(-3,3, length= eval.n)  
plot(x, y, main = "K- Nearest Neighborhood", pch = 20)

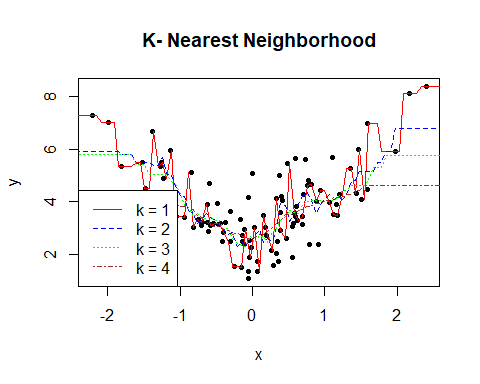


FNN라이브러리를 불러오고, eval.point를 plot에 찍어낸다.

idx.mat1 <- knnx.index(x, eval.point , k = 1)  
idx.mat2 <- knnx.index(x, eval.point , k = 5)  
idx.mat3 <- knnx.index(x, eval.point , k = 10)  
idx.mat4 <- knnx.index(x, eval.point , k = 30)  
yh1 = rep(0,eval.n)  
yh2 = rep(0, eval.n)  
yh3 = rep(0, eval.n)  
yh4 = rep(0, eval.n)  
for (i in 1:eval.n) yh1[i]<-mean(y[idx.mat1[i,]])  
for (i in 1:eval.n) yh2[i]<-mean(y[idx.mat2[i,]])  
for (i in 1:eval.n) yh3[i]<-mean(y[idx.mat3[i,]])  
for (i in 1:eval.n) yh4[i]<-mean(y[idx.mat4[i,]])

k = 1, k = 5, k = 10, k = 30 인 경우에 eval.point에 가장 가까운 점들을 찾아내 idx.mat에 할당하고, 그 평균들을 각각의 yhat에 저장한다.

plot(x, y, main = "K- Nearest Neighborhood", pch = 20)  
lines(eval.point , yh1, type= 'l', lty = 1, col = 'red')  
lines(eval.point , yh2, type= 'l', lty = 2, col = 'blue')  
lines(eval.point , yh3, type= 'l', lty = 3, col = 'green')  
lines(eval.point , yh4, type= 'l', lty = 4, col = 'brown')  
legend("bottomleft", c("k = 1", "k = 2", "k = 3", "k = 4"),   
 col = c("red", "blue", "green", "brown"), lty = c(1, 2, 3, 4))



k = 1 인 경우, 가장 가까운 점은 각 포인트 자기 자신이므로 모든 포인트를 지나는 빨간 그래프가 그려진다. k = 5, 10, 30인 경우는 가까운 값들의 평균들을 지나는 그래프가 각각 파란색, 초록색, 갈색으로 그려진다. 이에 대한 범례는 왼쪽 아래에 들어갔다.