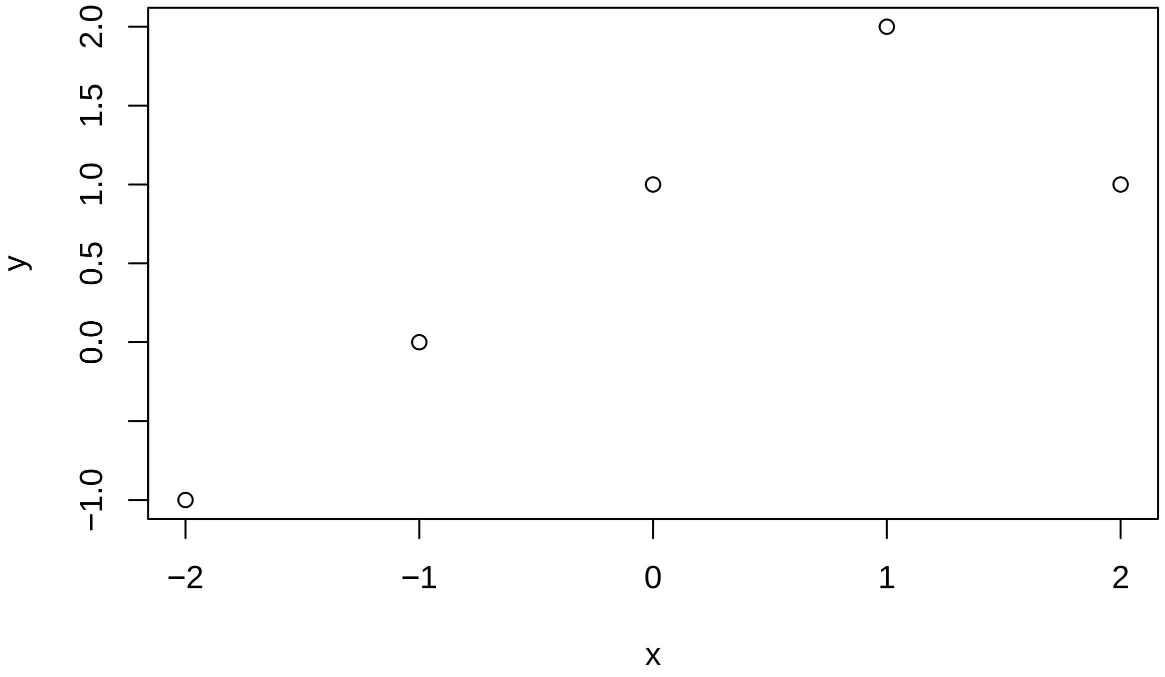
chp7 gam

Wendy Liang

2/28/2021

7.3

```
x = -2:2
y = 1 + x + -2 * (x-1)^2 * I(x>1)
plot(x, y)
```



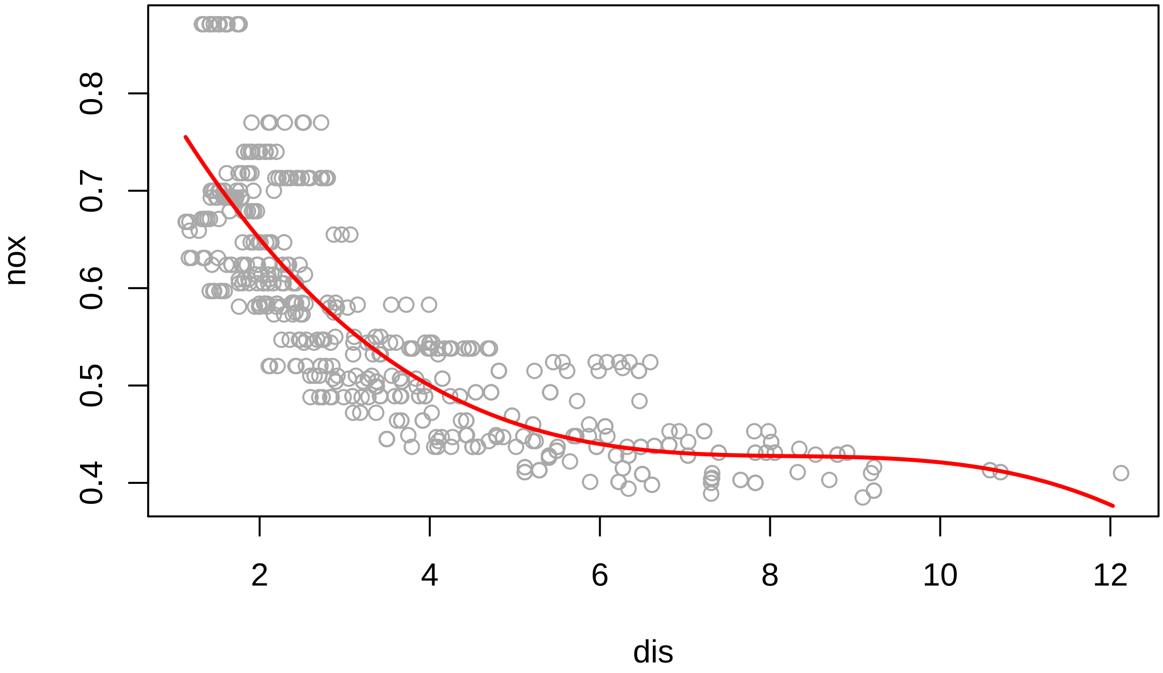
7.9

```
library(MASS)
set.seed(1)
fit <- lm(nox ~ poly(dis, 3), data = Boston)
summary(fit)
```

a

```
##
## Call:
## lm(formula = nox ~ poly(dis, 3), data = Boston)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.121130 -0.040619 -0.009738  0.023385  0.194904
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.554695   0.002759 201.021  < 2e-16 ***
## poly(dis, 3)1 -2.003096   0.062071 -32.271  < 2e-16 ***
## poly(dis, 3)2  0.856330   0.062071  13.796  < 2e-16 ***
## poly(dis, 3)3 -0.318049   0.062071  -5.124 4.27e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06207 on 502 degrees of freedom
## Multiple R-squared:  0.7148, Adjusted R-squared:  0.7131
## F-statistic: 419.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

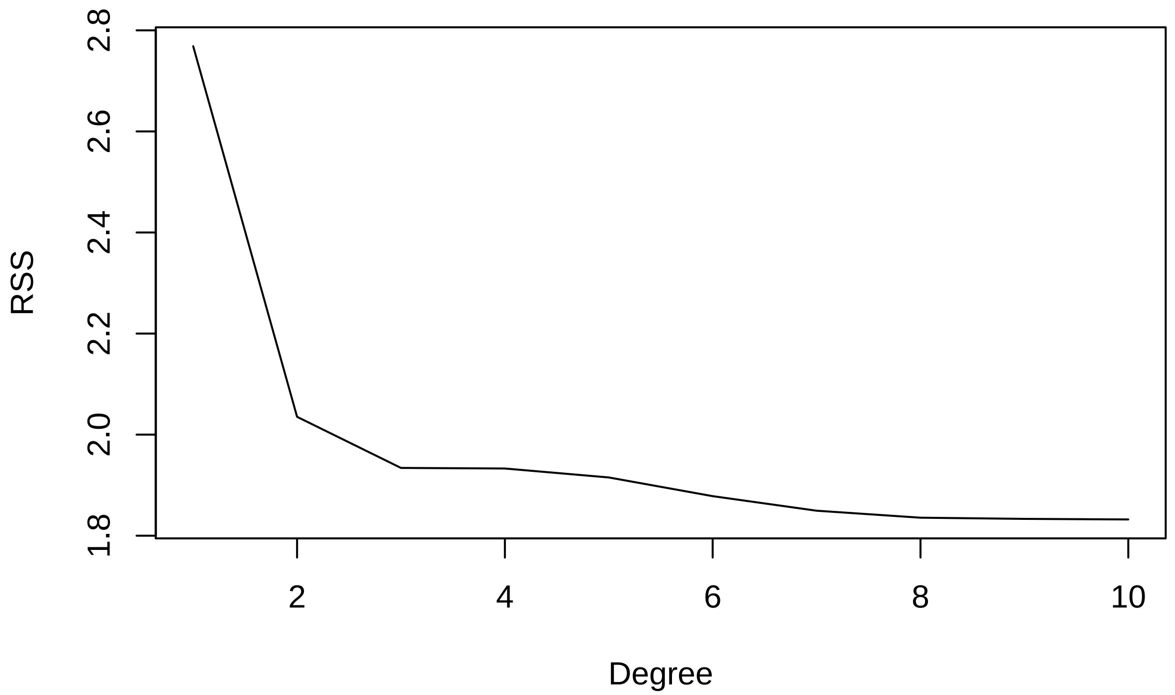
```
dislims <- range(Boston$dis)
dis.grid <- seq(from = dislims[1], to = dislims[2], by = 0.1)
preds <- predict(fit, list(dis = dis.grid))
plot(nox ~ dis, data = Boston, col = "darkgrey")
lines(dis.grid, preds, col = "red", lwd = 2)
```



The polynomial terms are significant.

```
rss <- rep(NA, 10)
for (i in 1:10) {
    fit <- lm(nox ~ poly(dis, i), data = Boston)
    rss[i] <- sum(fit$residuals^2)
}
plot(1:10, rss, xlab = "Degree", ylab = "RSS", type = "l")
```

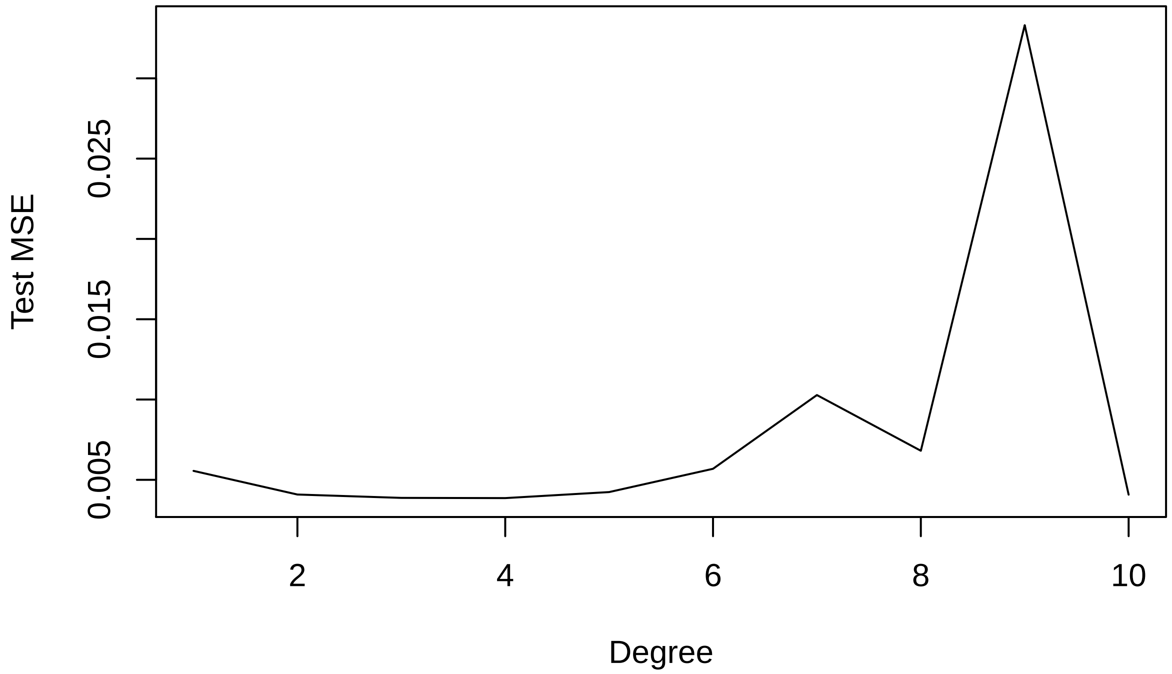
b



It seems that the RSS decreases with the degree of the polynomial, and so is minimum for a polynomial of degree 10.

```
library(boot)
deltas <- rep(NA, 10)
for (i in 1:10) {
    fit <- glm(nox ~ poly(dis, i), data = Boston)
    deltas[i] <- cv.glm(Boston, fit, K = 10)$delta[1]
}
plot(1:10, deltas, xlab = "Degree", ylab = "Test MSE", type = "l")
```

c



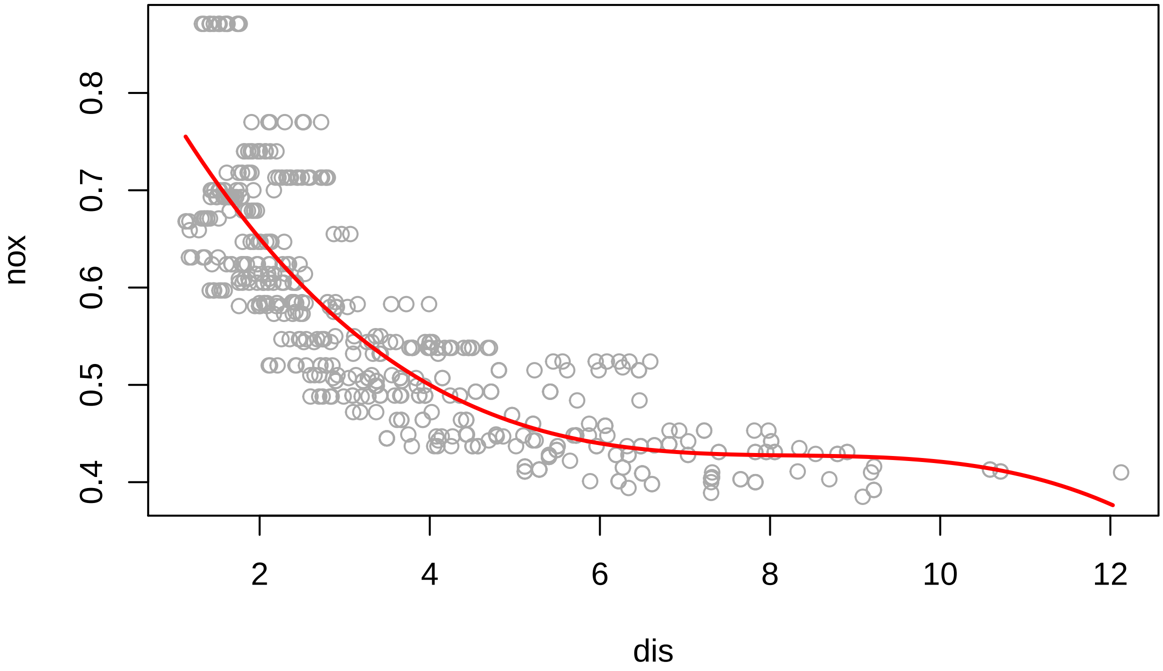
```r
library(splines)
fit <- lm(nox ~ bs(dis, knots = c(4, 7, 11)), data = Boston)
summary(fit)
```

d

```
##
## Call:
## lm(formula = nox ~ bs(dis, knots = c(4, 7, 11)), data = Boston)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.124567 -0.040355 -0.008702  0.024740  0.192920
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.73926    0.01331  55.537  < 2e-16 ***
## bs(dis, knots = c(4, 7, 11))1 -0.08861    0.02504  -3.539  0.00044 ***
## bs(dis, knots = c(4, 7, 11))2 -0.31341    0.01680 -18.658  < 2e-16 ***
## bs(dis, knots = c(4, 7, 11))3 -0.26618    0.03147  -8.459 3.00e-16 ***
## bs(dis, knots = c(4, 7, 11))4 -0.39802    0.04647  -8.565  < 2e-16 ***
## bs(dis, knots = c(4, 7, 11))5 -0.25681    0.09001  -2.853  0.00451 **
## bs(dis, knots = c(4, 7, 11))6 -0.32926    0.06327  -5.204 2.85e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06185 on 499 degrees of freedom
## Multiple R-squared:  0.7185, Adjusted R-squared:  0.7151
## F-statistic: 212.3 on 6 and 499 DF,  p-value: < 2.2e-16
```

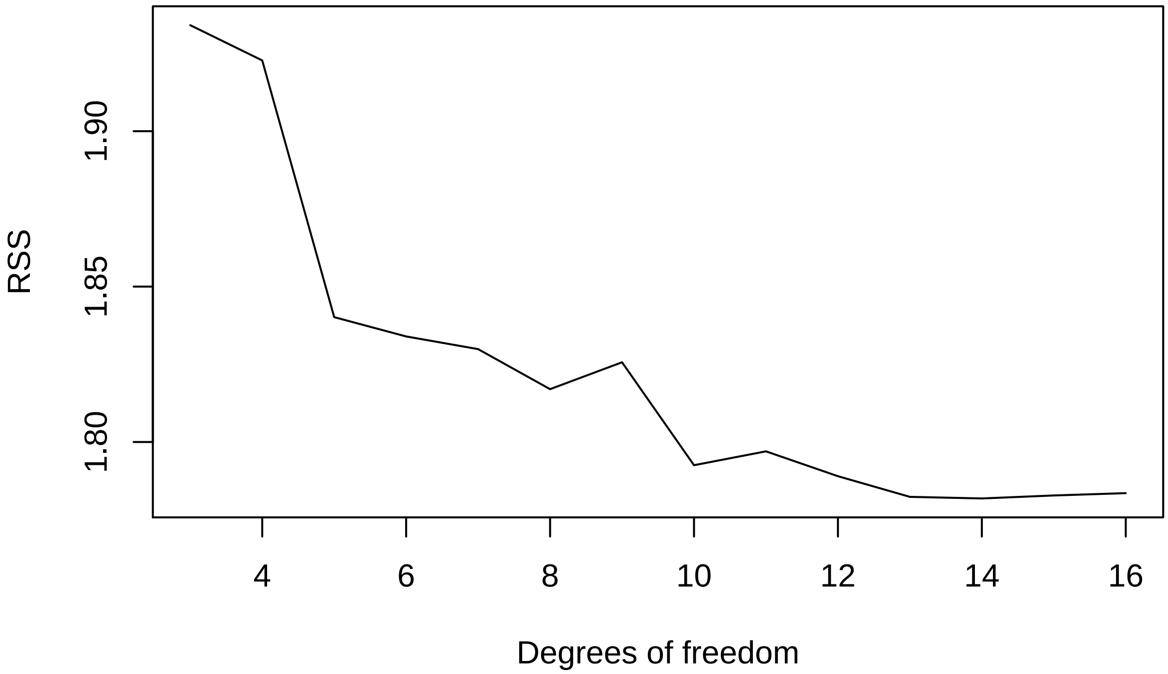
```r
pred <- predict(fit, list(dis = dis.grid))
plot(nox ~ dis, data = Boston, col = "darkgrey")
```

```
lines(dis.grid, preds, col = "red", lwd = 2)
```



```
rss <- rep(NA, 16)
for (i in 3:16) {
    fit <- lm(nox ~ bs(dis, df = i), data = Boston)
    rss[i] <- sum(fit$residuals^2)
}
plot(3:16, rss[-c(1, 2)], xlab = "Degrees of freedom", ylab = "RSS", type = "l")
```

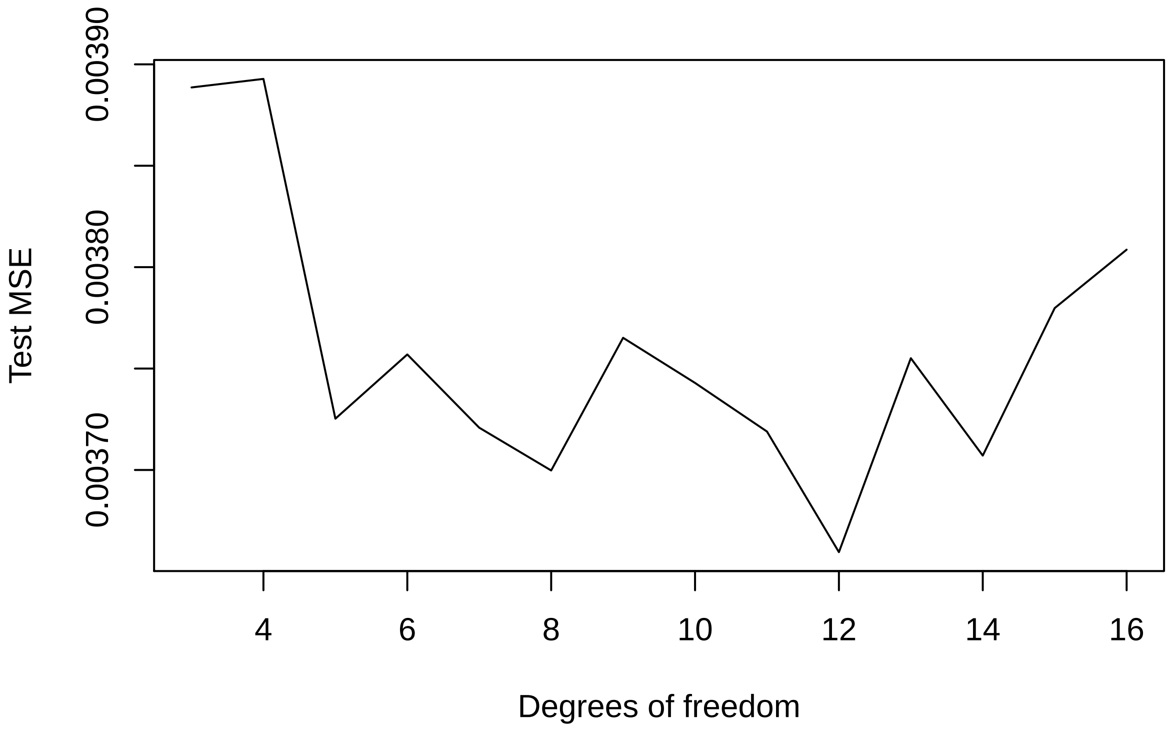
e



```
cv <- rep(NA, 16)
for (i in 3:16) {
    fit <- glm(nox ~ bs(dis, df = i), data = Boston)
    cv[i] <- cv.glm(Boston, fit, K = 10)$delta[1]
}
```

```
plot(3:16, cv[-c(1, 2)], xlab = "Degrees of freedom", ylab = "Test MSE", type = "l")
```

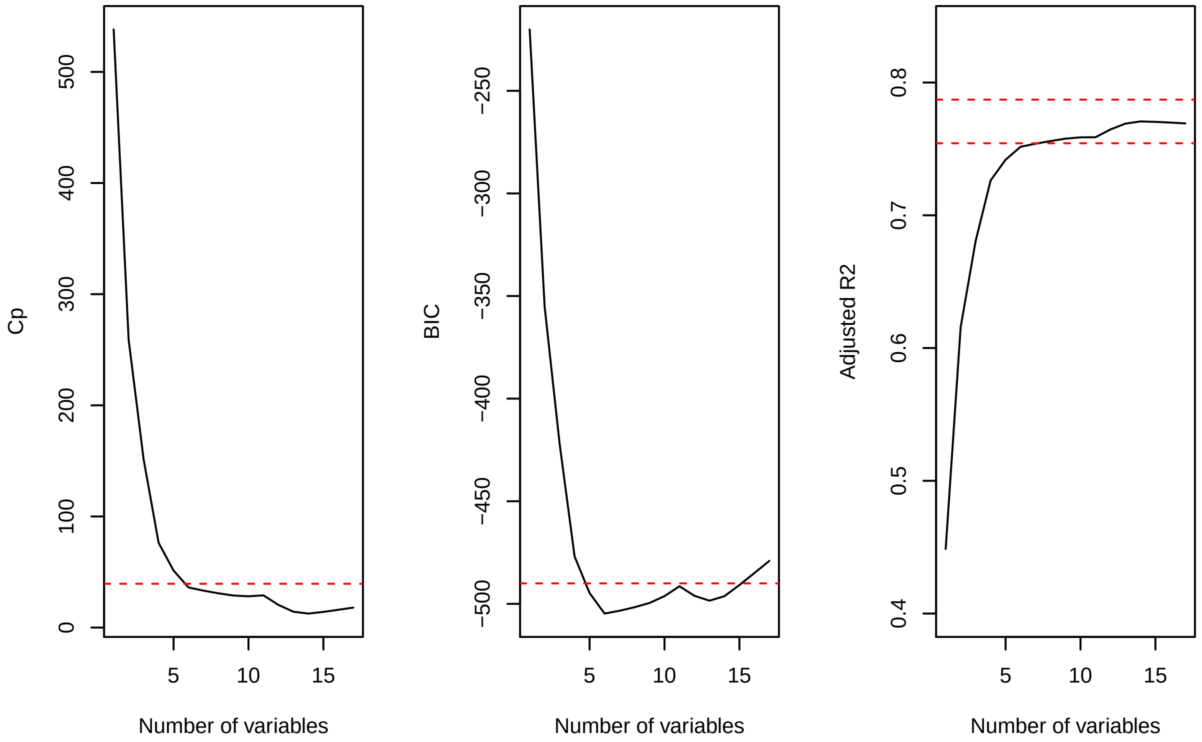
f



10

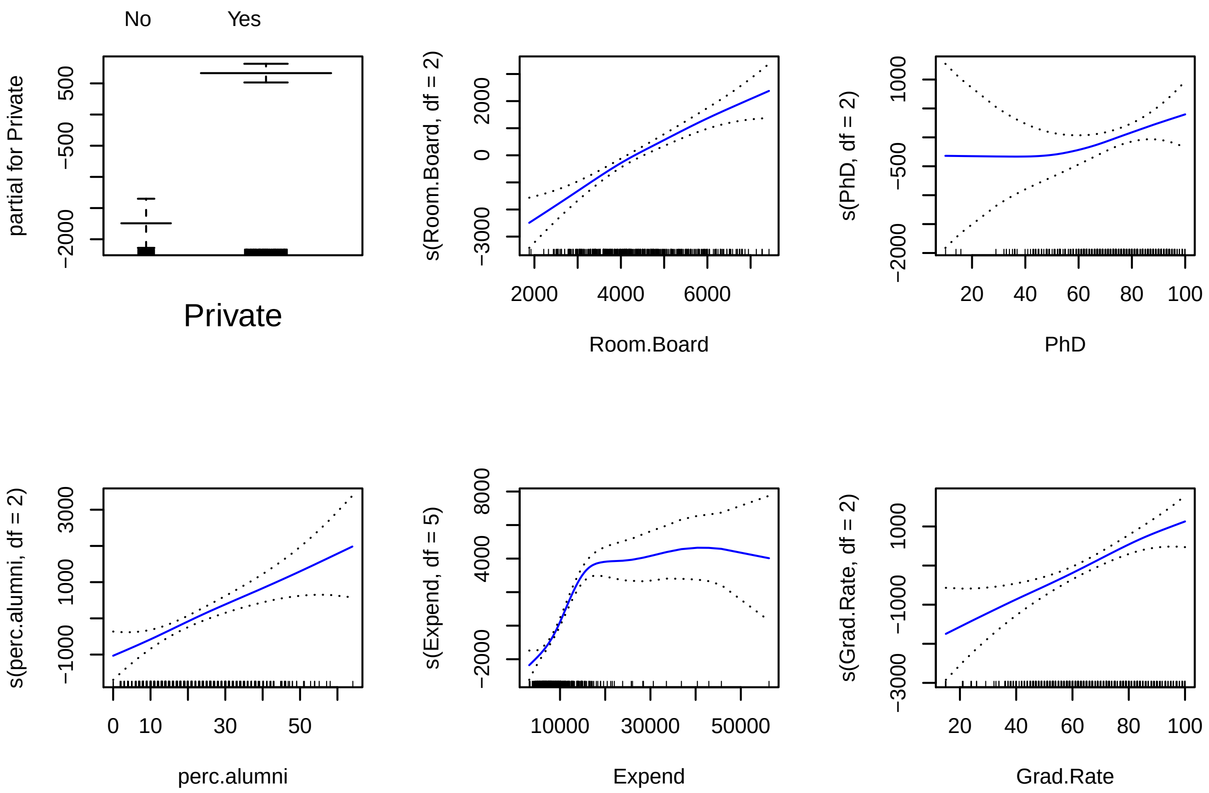
a

```
library(leaps)
set.seed(1)
attach(College)
train <- sample(length(Outstate), length(Outstate) / 2)
test <- -train
College.train <- College[train, ]
College.test <- College[test, ]
fit <- regsubsets(Outstate ~ ., data = College.train, nvmax = 17, method = "forward")
fit.summary <- summary(fit)
par(mfrow = c(1, 3))
plot(fit.summary$cp, xlab = "Number of variables", ylab = "Cp", type = "l")
min.cp <- min(fit.summary$cp)
std.cp <- sd(fit.summary$cp)
abline(h = min.cp + 0.2 * std.cp, col = "red", lty = 2)
abline(h = min.cp - 0.2 * std.cp, col = "red", lty = 2)
plot(fit.summary$bic, xlab = "Number of variables", ylab = "BIC", type='l')
min.bic <- min(fit.summary$bic)
std.bic <- sd(fit.summary$bic)
abline(h = min.bic + 0.2 * std.bic, col = "red", lty = 2)
abline(h = min.bic - 0.2 * std.bic, col = "red", lty = 2)
plot(fit.summary$adjr2, xlab = "Number of variables", ylab = "Adjusted R2", type = "l", ylim = c(0.4, 0
max.adjr2 <- max(fit.summary$adjr2)
std.adjr2 <- sd(fit.summary$adjr2)
abline(h = max.adjr2 + 0.2 * std.adjr2, col = "red", lty = 2)
abline(h = max.adjr2 - 0.2 * std.adjr2, col = "red", lty = 2)
```



```r
library(gam)
fit <- gam(Outstate ~ Private + s(Room.Board, df = 2) + s(PhD, df = 2) + s(perc.alumni, df = 2) + s(Expe
par(mfrow = c(2, 3))
plot(fit, se = T, col = "blue")
```

b



```r
preds <- predict(fit, College.test)
err <- mean((College.test$Outstate - preds)^2)
err
```

c

```
## [1] 3349290
```

```r
tss <- mean((College.test$Outstate - mean(College.test$Outstate))^2)
rss <- 1 - err / tss
rss
```

```
## [1] 0.7660016
```

```r
summary(fit)
```

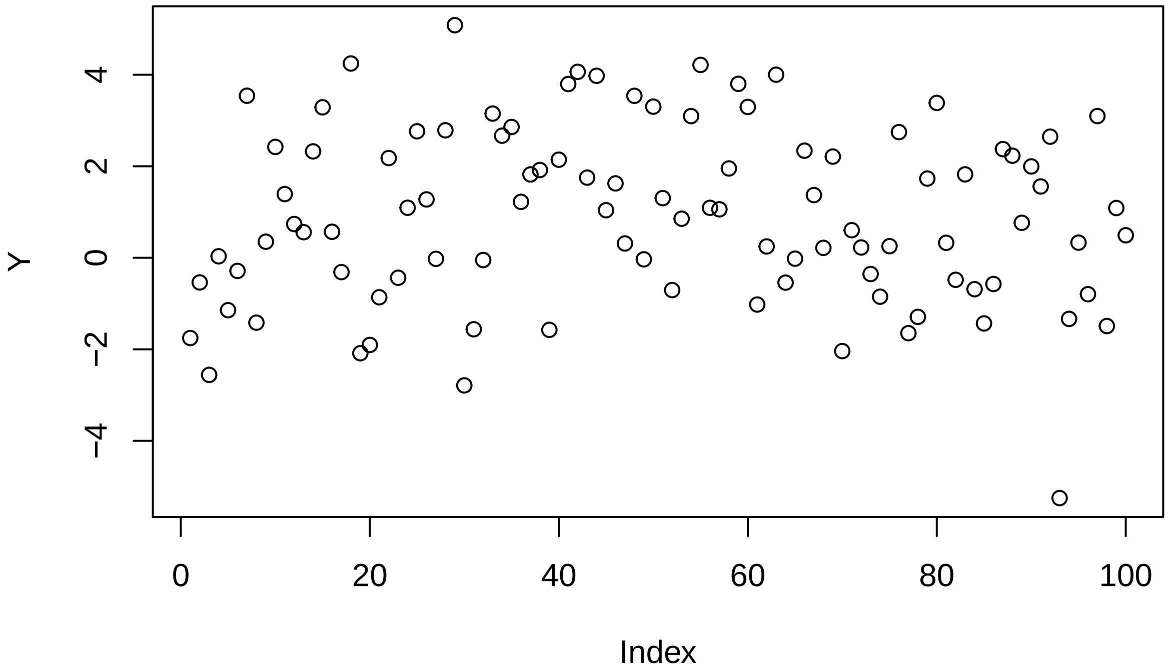
d

```
##
## Call: gam(formula = Outstate ~ Private + s(Room.Board, df = 2) + s(PhD,
##     df = 2) + s(perc.alumni, df = 2) + s(Expend, df = 5) + s(Grad.Rate,
##     df = 2), data = College.train)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -7402.89 -1114.45   -12.67  1282.69  7470.60
##
## (Dispersion Parameter for gaussian family taken to be 3711182)
##
##     Null Deviance: 6989966760 on 387 degrees of freedom
## Residual Deviance: 1384271126 on 373 degrees of freedom
## AIC: 6987.021
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##                          Df     Sum Sq    Mean Sq F value    Pr(>F)
## Private                   1 1778718277 1778718277 479.286 < 2.2e-16 ***
## s(Room.Board, df = 2)     1 1577115244 1577115244 424.963 < 2.2e-16 ***
## s(PhD, df = 2)            1  322431195  322431195  86.881 < 2.2e-16 ***
## s(perc.alumni, df = 2)    1  336869281  336869281  90.771 < 2.2e-16 ***
## s(Expend, df = 5)         1  530538753  530538753 142.957 < 2.2e-16 ***
## s(Grad.Rate, df = 2)      1   86504998   86504998  23.309 2.016e-06 ***
## Residuals               373 1384271126    3711182
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                        Npar Df  Npar F     Pr(F)
## (Intercept)
## Private
## s(Room.Board, df = 2)        1  1.9157    0.1672
## s(PhD, df = 2)               1  0.9699    0.3253
## s(perc.alumni, df = 2)       1  0.1859    0.6666
## s(Expend, df = 5)            4 20.5075 2.665e-15 ***
## s(Grad.Rate, df = 2)         1  0.5702    0.4506
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

11

```r
set.seed(99)
n <- 100
X1 <- rnorm(100)
X2 <- rnorm(100)
eps <- rnorm(1:100, sd = 1)
b_0 <- 0.9
b_1 <- -1.5
b_2 <- 1
Y = b_0 + b_1*X1 + b_2*X2 +eps
plot(Y)
```

a



```
b_h1 <- 1
```

b

```
a=Y-b_h1 *X1
b_h2=lm(a~X2)$coef [2]
```

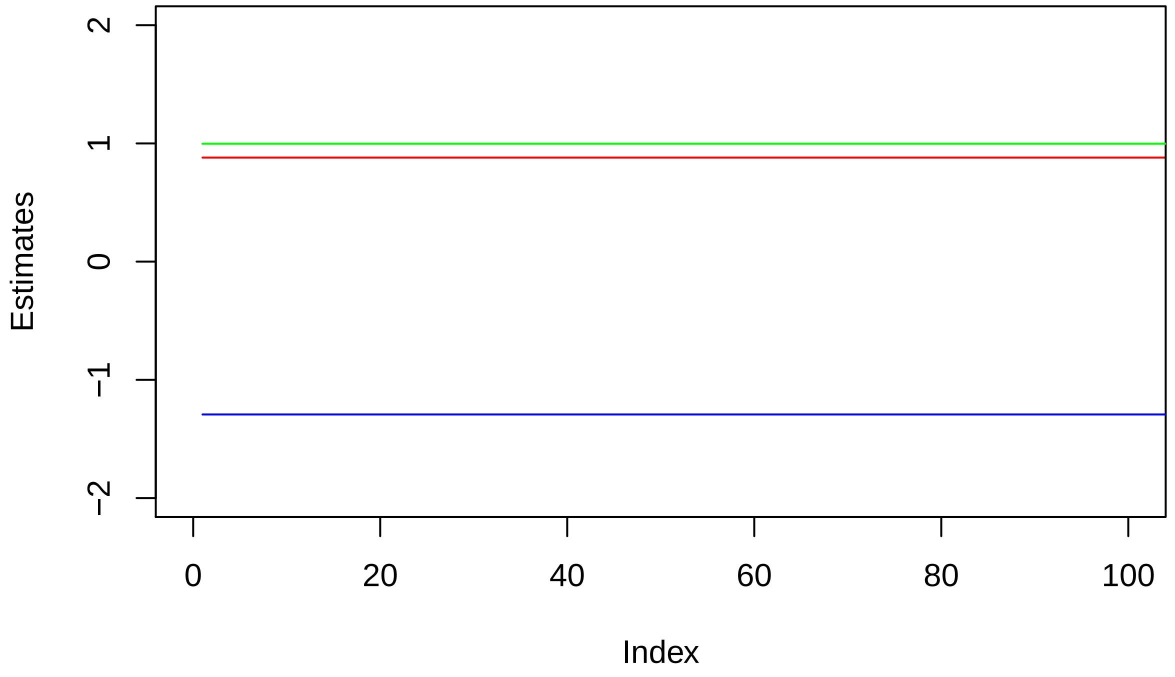
c

```
a=Y-b_h2 *X2
b_h1=lm(a~X1)$coef [2]
```

d

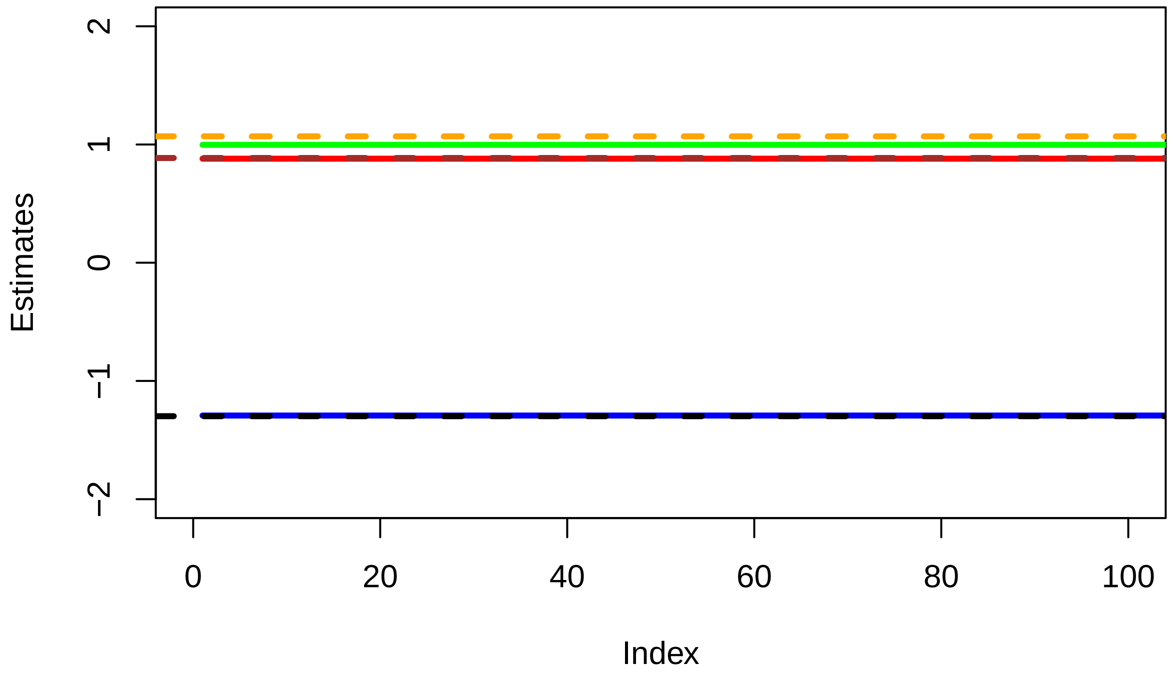
```
b_hat0 <- rep(0,1000)
b_hat1 <- rep(0,1000)
b_hat2 <- rep(0,1000)
for (i in 1:1000) {
  a <- Y - b_hat1[i]*X1
  b_hat2[i] <- lm(a ~ X2)$coef[2]
  a <- Y - b_hat2[i]*X2
  b_hat1[i] <- lm(a ~ X1)$coef[2]
  b_hat0[i] <- lm(a ~ X1)$coef[1]
}
plot(b_hat0, ylab = "Estimates", type = "l", col = "red", ylim = c(-2,2), xlim = c(0,100))
lines(b_hat1, col = "blue")
lines(b_hat2, col = "green")
```

e



```
fit3 <- lm(Y ~ X1 + X2)
plot(b_hat0, ylab = "Estimates", type = "l", col = "red", ylim = c(-2,2), xlim = c(0,100), lwd = 3)
lines(b_hat1, col = "blue", lwd = 3)
lines(b_hat2, col = "green", lwd = 3)
abline(h = coef(fit3)[1], lty = "dashed", col = "brown", lwd = 3)
abline(h = coef(fit3)[2], lty = "dashed", col = "black", lwd = 3)
abline(h = coef(fit3)[3], lty = "dashed", col = "orange", lwd = 3)
```

f



```
b <- data.frame(b_hat0, b_hat1, b_hat2)
head(b)
```

g

```
##      b_hat0    b_hat1    b_hat2
## 1 0.8799804 -1.292655 0.9972832
## 2 0.8799804 -1.292655 0.9972832
## 3 0.8799804 -1.292655 0.9972832
## 4 0.8799804 -1.292655 0.9972832
## 5 0.8799804 -1.292655 0.9972832
## 6 0.8799804 -1.292655 0.9972832
```