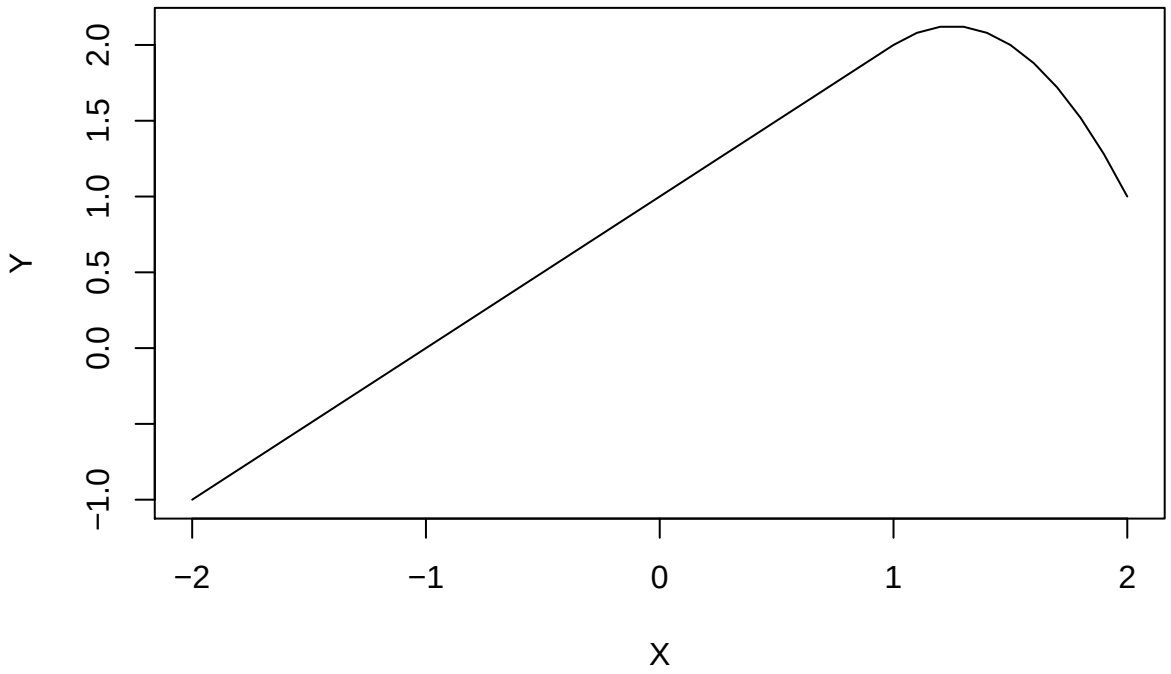
MA679-GAM hw

Ziyi Bai

2/26/2021

7.3

```
X <- seq(-2,2,0.1)
Y = 1+1*X-2*((X-1)^2*I(X>=1))
plot(X,Y,type = "l")
```



7.9

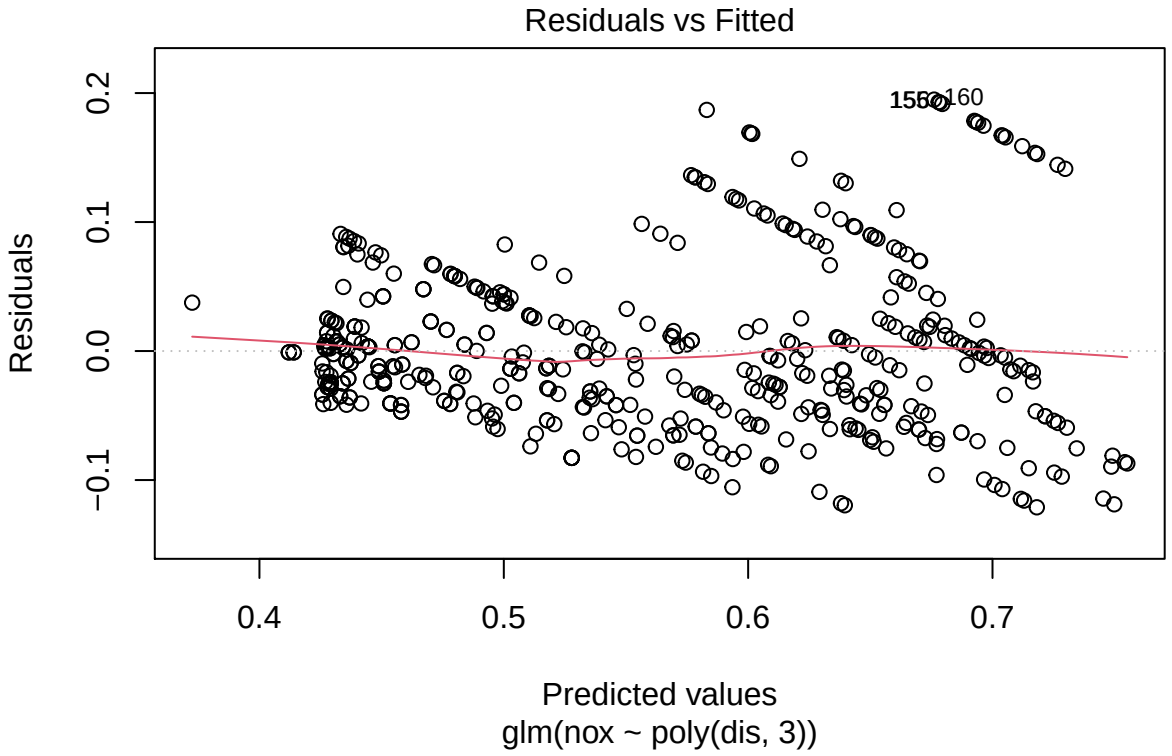
(a)

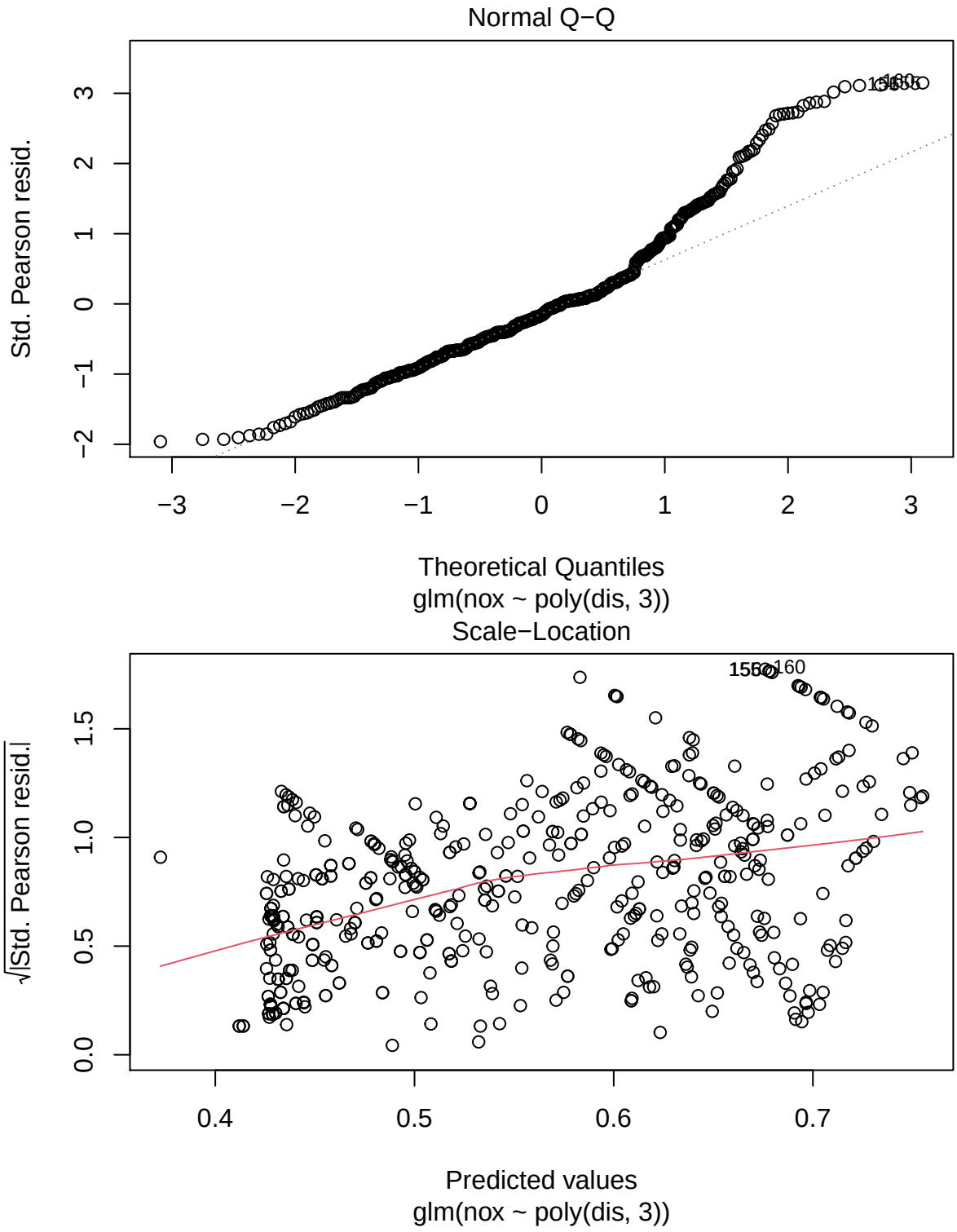
```
data("Boston")
fit_7.9a <- glm(nox~poly(dis,3),data=Boston)
summary(fit_7.9a)
```

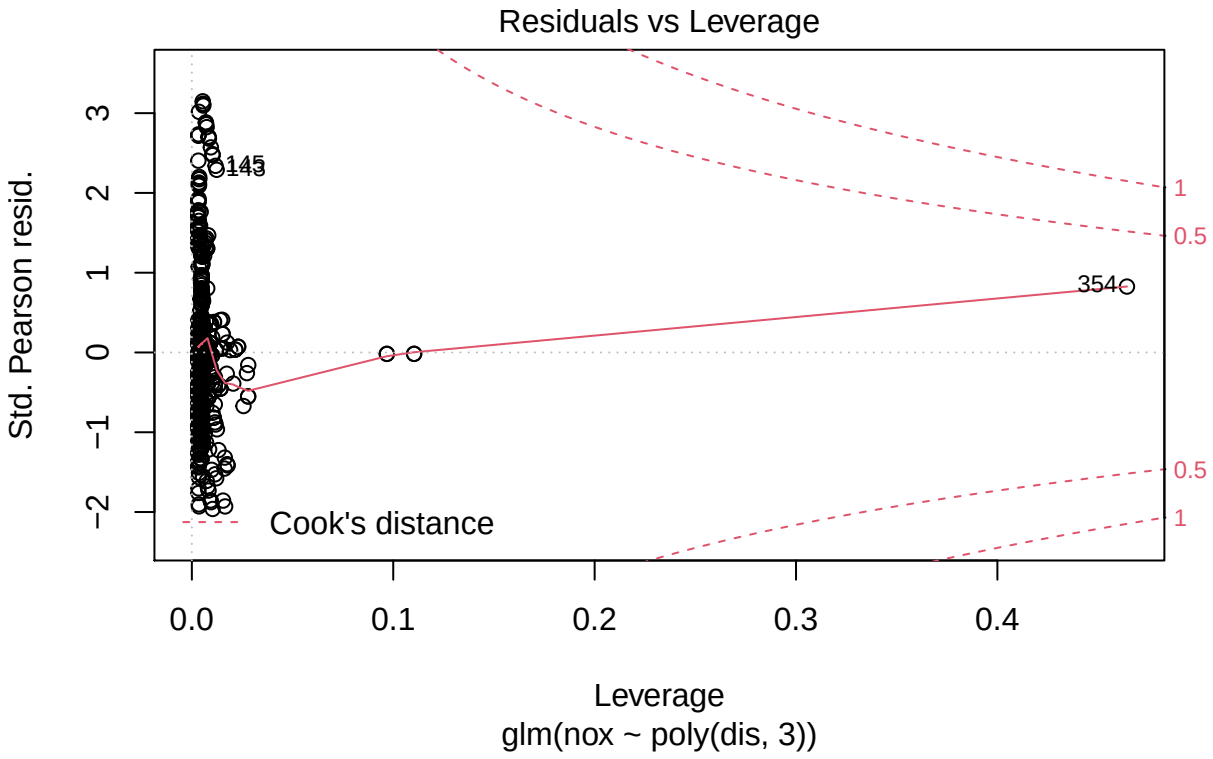
```
##
## Call:
## glm(formula = nox ~ poly(dis, 3), data = Boston)
##
## Deviance Residuals:
##       Min         1Q     Median         3Q        Max
## -0.121130  -0.040619  -0.009738   0.023385   0.194904
```

```
##
## 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
##
## (Dispersion parameter for gaussian family taken to be 0.003852802)
##
##     Null deviance: 6.7810  on 505  degrees of freedom
## Residual deviance: 1.9341  on 502  degrees of freedom
## AIC: -1370.9
##
## Number of Fisher Scoring iterations: 2
```

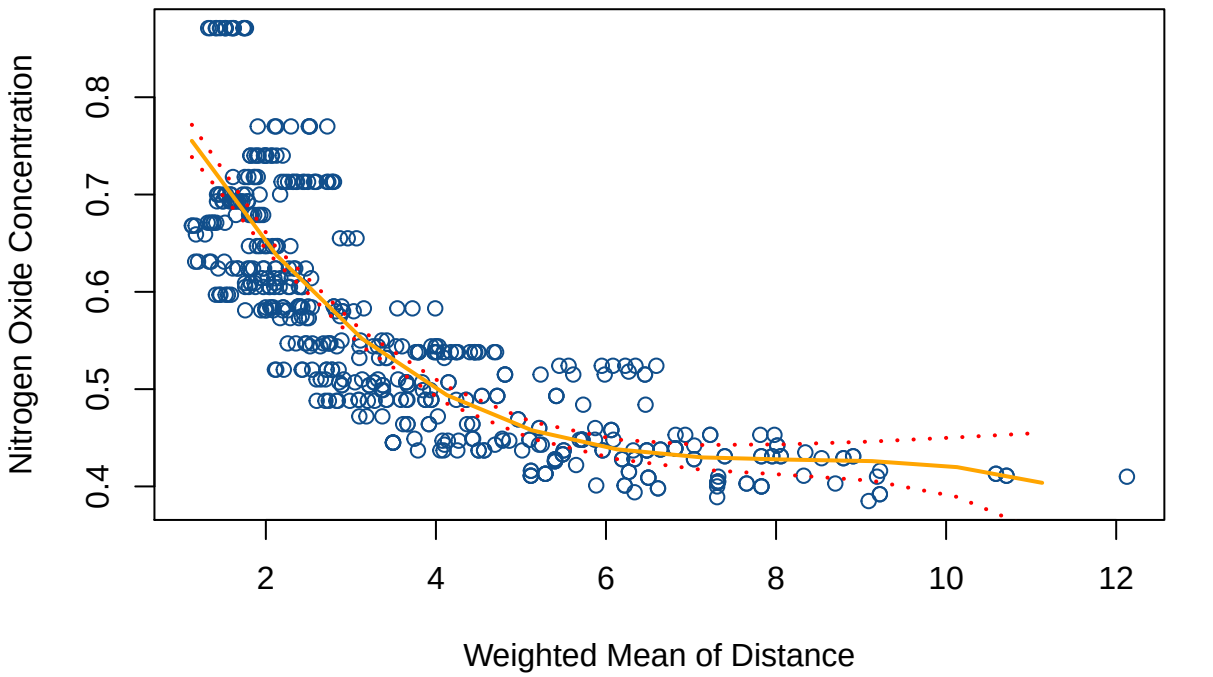
```
plot(fit_7.9a)
```




Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(nox ~ poly(dis, 3))
Scale−Location
√|Std. Pearson resid.|
Predicted values
glm(nox ~ poly(dis, 3))

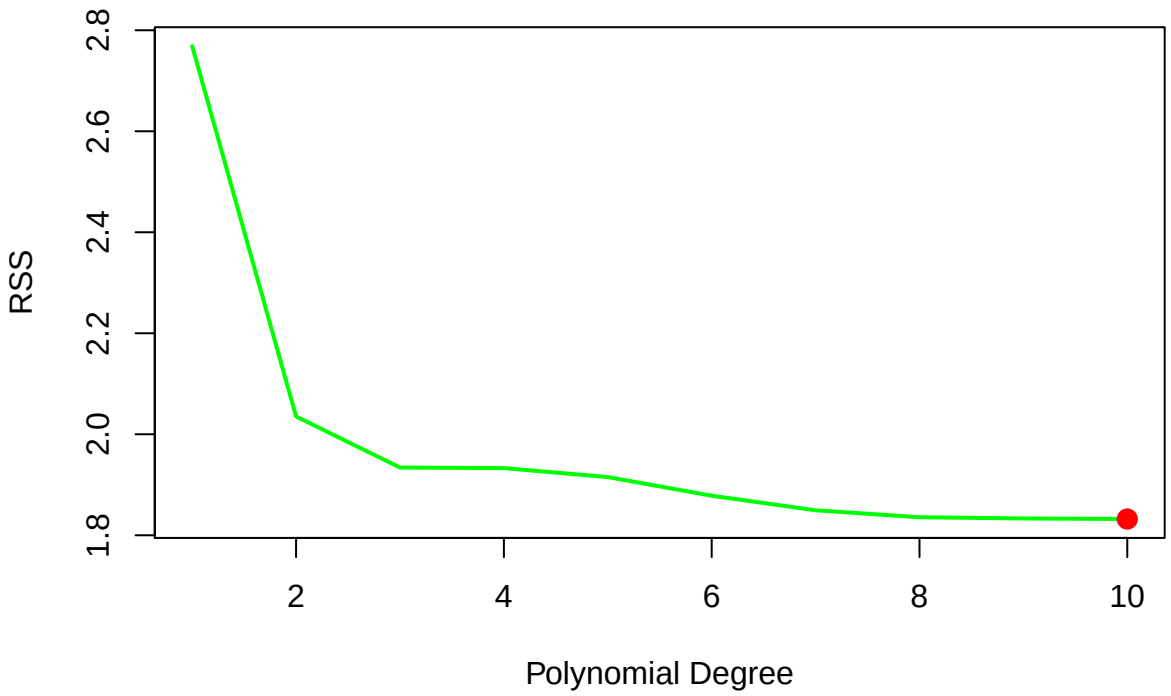



```
lims_dis <- range(Boston$dis)
grid_dis <- seq(lims_dis[1],lims_dis[2])
pred1 <- predict(fit_7.9a,list(dis=grid_dis), se=T)
se_lines <- cbind(pred1$fit+2*pred1$se.fit,pred1$fit-2*pred1$se.fit)
plot(Boston$dis,Boston$nox,xlab="Weighted Mean of Distance",ylab = "Nitrogen Oxide Concentration", col=
lines(grid_dis,pred1$fit,col="orange",lwd=2)
matlines(grid_dis,se_lines,lwd=2,col="red",lty=3)
```



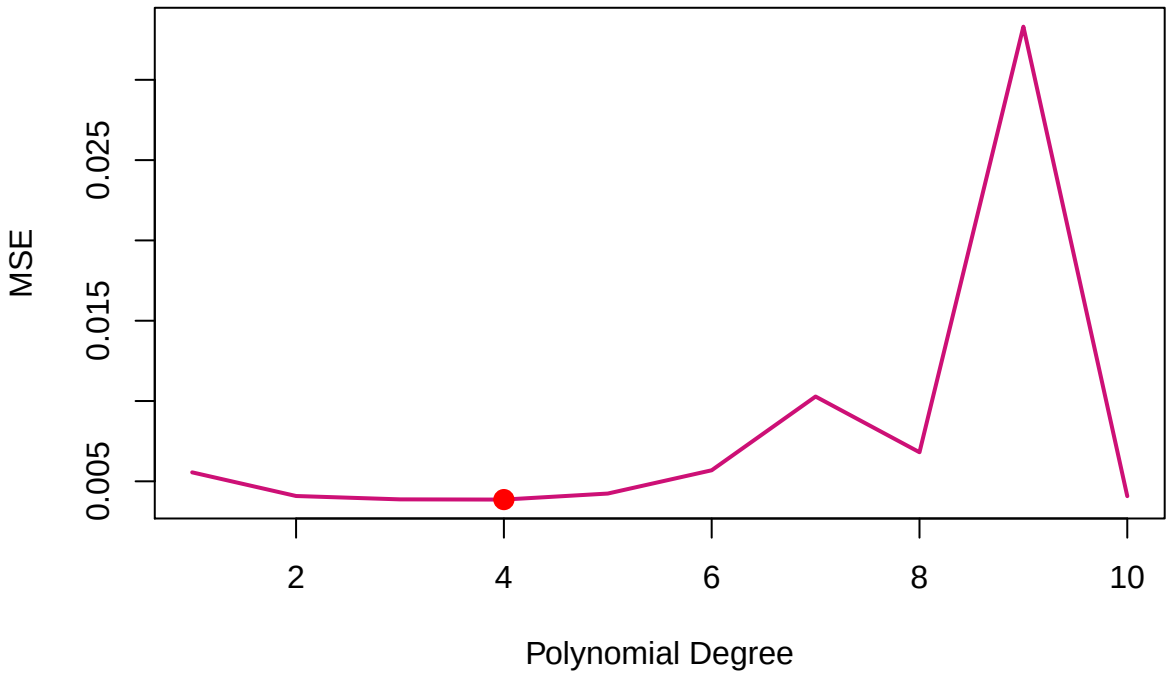
##(b)

```
set.seed(1)
rss <- rep(NA,10)
for (i in 1:10){
  fit_7.9b <- glm(nox~poly(dis,i),data=Boston)
  rss[i] <- sum(fit_7.9b$residuals^2)
}

plot(1:10,rss,xlab = "Polynomial Degree", ylab = "RSS", type="l",col="green",lwd=2)
points(which.min(rss),rss[which.min(rss)],col='red',pch=20,cex=2)
```



##(c)

```
err <- rep(NA,10)
for (i in 1:10) {
  fit_7.9c <- glm(nox~poly(dis,i),data=Boston)
  err[i] <- cv.glm(Boston,fit_7.9c,K=10)$delta[1]
}

plot(1:10,err,xlab = "Polynomial Degree",ylab = "MSE",type="l",col="deeppink3",lwd=2)
points(which.min(err),err[which.min(err)],col='red',pch=20,cex=2)
```



##(d)

```
range(Boston$dis)
```

```
## [1]  1.1296 12.1265
```

```
summary(Boston$dis)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.130   2.100   3.207   3.795   5.188  12.127
```

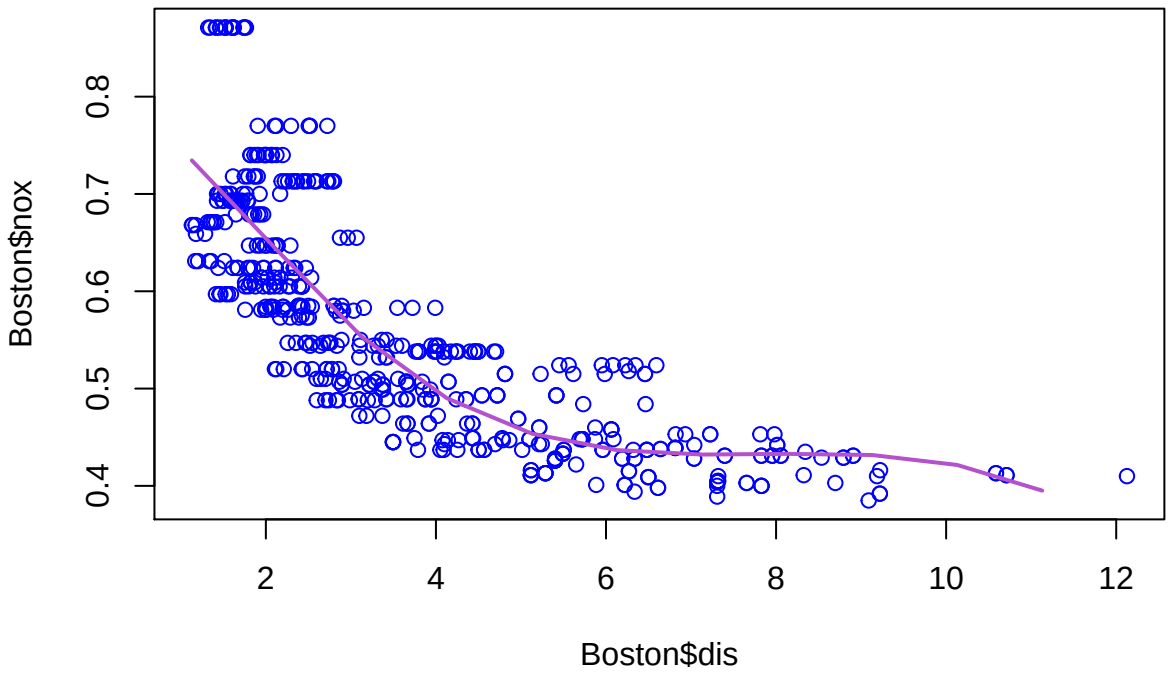
```
fit_7.9d <- lm(nox~bs(dis, df=4),data=Boston)
summary(fit_7.9d)
```

```
##
## Call:
## lm(formula = nox ~ bs(dis, df = 4), data = Boston)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.124622 -0.039259 -0.008514  0.020850  0.193891
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.73447    0.01460  50.306  < 2e-16 ***
## bs(dis, df = 4)1 -0.05810    0.02186  -2.658  0.00812 **
## bs(dis, df = 4)2 -0.46356    0.02366 -19.596  < 2e-16 ***
## bs(dis, df = 4)3 -0.19979    0.04311  -4.634 4.58e-06 ***
## bs(dis, df = 4)4 -0.38881    0.04551  -8.544  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06195 on 501 degrees of freedom
## Multiple R-squared:  0.7164, Adjusted R-squared:  0.7142
## F-statistic: 316.5 on 4 and 501 DF,  p-value: < 2.2e-16
```

```r
attr(bs(Boston$dis,df=4),"knots")
```

```
##     50%
## 3.20745
```

```r
x<- seq(min(Boston$dis),max(Boston$dis))
y <- predict(fit_7.9d,data.frame(dis=x))
plot(Boston$dis,Boston$nox,col="blue")
lines(x,y,col="mediumorchid3",lwd=2)
```

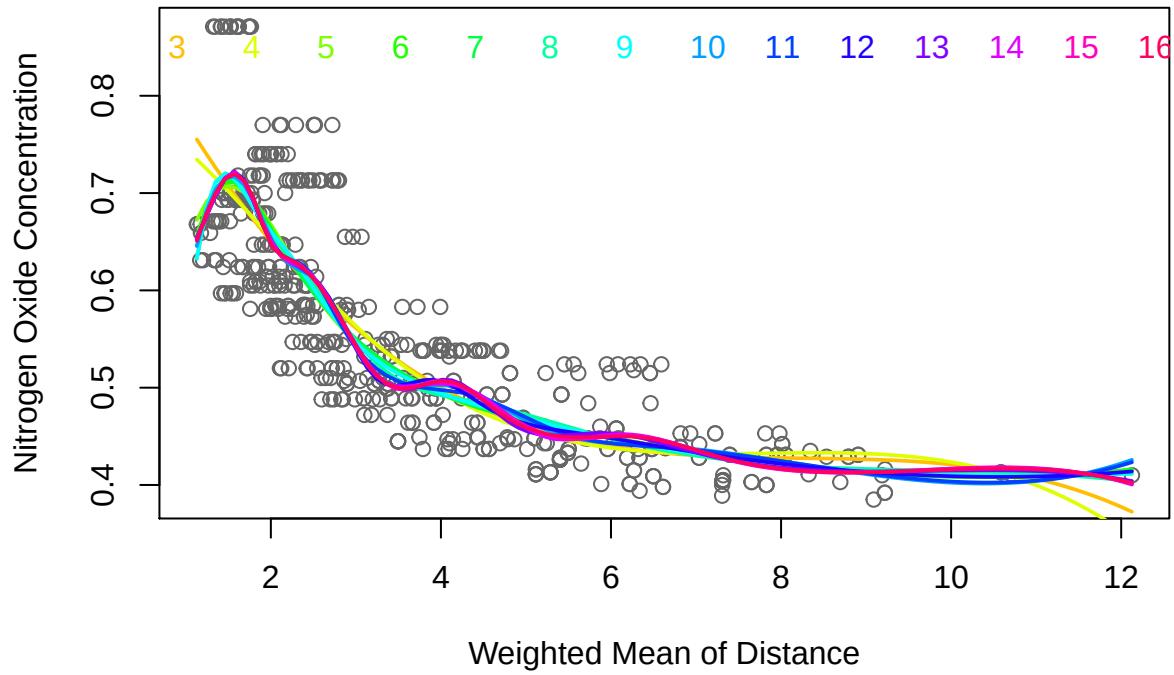


R

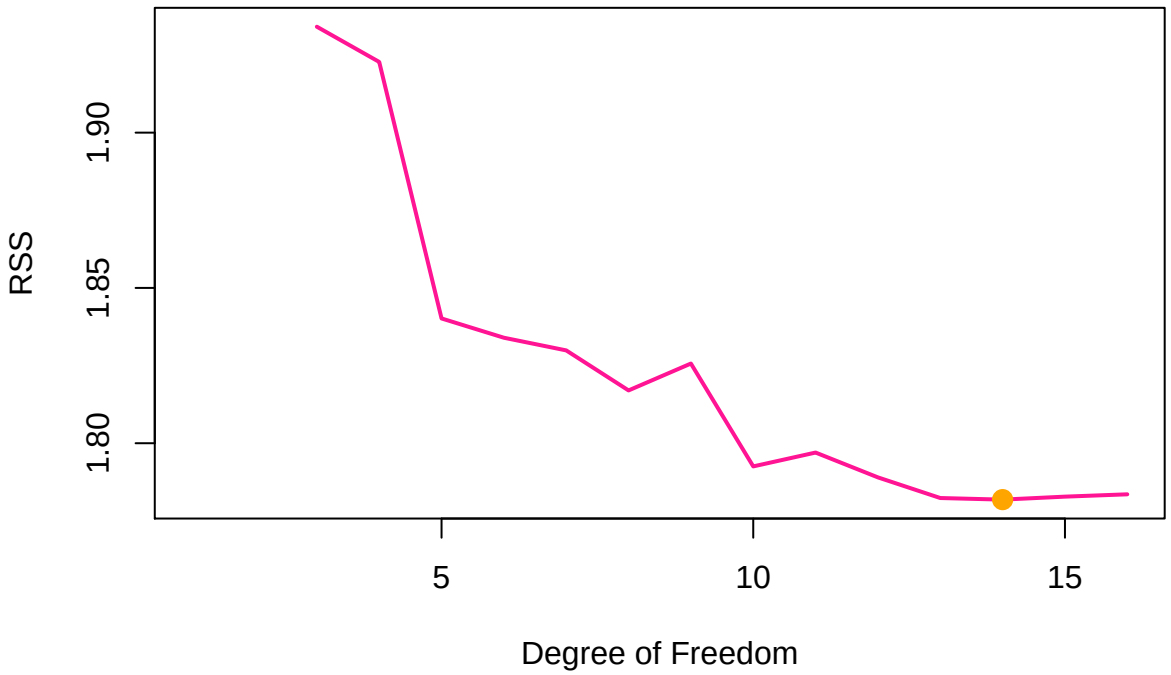
chosses the knot of 3.207

##(e)

```r
plot(Boston$dis,Boston$nox,xlab = "Weighted Mean of Distance", ylab="Nitrogen Oxide Concentration",col=
clrs <- rainbow(16)
legend(x="topright",legend = 3:16,text.col = clrs[3:16],text.width = 0.2,bty="n",horiz=T)
x <- seq(min(Boston$dis),max(Boston$dis),length.out=100)
rss_df <- c()
for (i in 3:16){
  fit_7.9e <- lm(nox~bs(dis,df=i),data=Boston)
  pred <- predict(fit_7.9e,data.frame(dis=x))
  lines(x,pred,col=clrs[i],lwd=1.85)
  rss_df[i] <- sum(fit_7.9e$residuals^2)
}
```




```
plot(1:16,rss_df,xlab = "Degree of Freedom", ylab="RSS",type="l",col="deeppink1",lwd=2)
points(which.min(rss_df),rss_df[which.min(rss_df)],col='orange',pch=20,cex=2)
```



##(f)

```
set.seed(9)
cv <- rep(NA,16)
for (i in 3:16) {
  fit_7.9f <- glm(nox~bs(dis,df=i),data = Boston)
  cv[i] <- cv.glm(Boston,fit_7.9f,K=10)$delta[1]
}
```

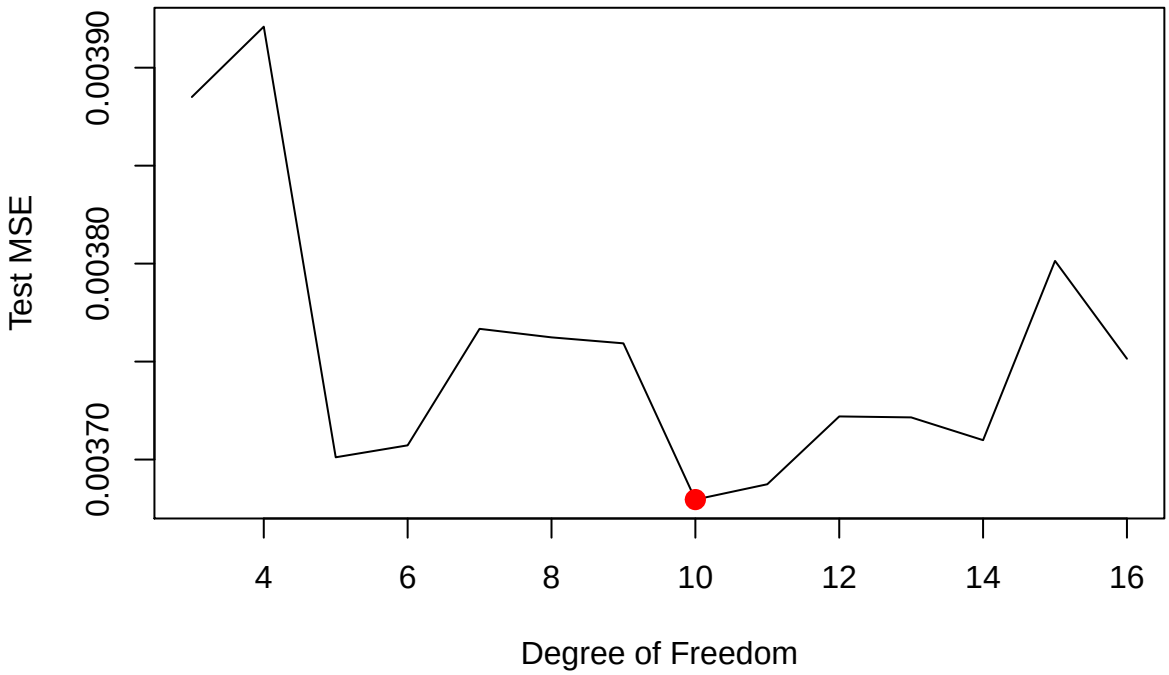
```
## Warning in bs(dis, degree = 3L, knots = numeric(0), Boundary.knots = c(1.1691, :
```

```
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = numeric(0), Boundary.knots = c(1.1691, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = numeric(0), Boundary.knots = c(1.1296, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = numeric(0), Boundary.knots = c(1.1296, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`50%` = 3.1992), Boundary.knots =
## c(1.1296, : some 'x' values beyond boundary knots may cause ill-conditioned
## bases

## Warning in bs(dis, degree = 3L, knots = c(`50%` = 3.1992), Boundary.knots =
## c(1.1296, : some 'x' values beyond boundary knots may cause ill-conditioned
## bases

## Warning in bs(dis, degree = 3L, knots = c(`50%` = 3.1323), Boundary.knots =
## c(1.137, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`50%` = 3.1323), Boundary.knots =
## c(1.137, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`33.33333%` = 2.3817, `66.66667%` =
## 4.418: some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`33.33333%` = 2.3817, `66.66667%` =
## 4.418: some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`33.33333%` = 2.38876666666667, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`33.33333%` = 2.38876666666667, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`25%` = 2.1084, `50%` = 3.2721, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`25%` = 2.1084, `50%` = 3.2721, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`25%` = 2.08585, `50%` = 3.1057, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`25%` = 2.08585, `50%` = 3.1057, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`20%` = 1.96376, `40%` = 2.6439, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`20%` = 1.96376, `40%` = 2.6439, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`20%` = 1.9265, `40%` = 2.6403, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases
```

```
## Warning in bs(dis, degree = 3L, knots = c(`20%` = 1.9265, `40%` = 2.6403, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`16.66667%` = 1.86223333333333, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`16.66667%` = 1.86223333333333, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`16.66667%` = 1.82203333333333, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`16.66667%` = 1.82203333333333, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`14.28571%` = 1.7936, `28.57143%` =
## 2.2044, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`14.28571%` = 1.7936, `28.57143%` =
## 2.2044, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`14.28571%` = 1.7936, `28.57143%`
## = 2.16972857142857, : some 'x' values beyond boundary knots may cause ill-
## conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`14.28571%` = 1.7936, `28.57143%`
## = 2.16972857142857, : some 'x' values beyond boundary knots may cause ill-
## conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`12.5%` = 1.7519375, `25%` =
## 2.087875, : some 'x' values beyond boundary knots may cause ill-conditioned
## bases

## Warning in bs(dis, degree = 3L, knots = c(`12.5%` = 1.7519375, `25%` =
## 2.087875, : some 'x' values beyond boundary knots may cause ill-conditioned
## bases

## Warning in bs(dis, degree = 3L, knots = c(`12.5%` = 1.751575, `25%` = 2.10035, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`12.5%` = 1.751575, `25%` = 2.10035, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`11.11111%` = 1.66397777777778, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`11.11111%` = 1.66397777777778, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`11.11111%` = 1.72676666666667, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`11.11111%` = 1.72676666666667, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`10%` = 1.63564, `20%` = 1.92404, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases
```

```
## Warning in bs(dis, degree = 3L, knots = c(`10%` = 1.63564, `20%` = 1.92404, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`10%` = 1.63564, `20%` = 1.9648, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`10%` = 1.63564, `20%` = 1.9648, :
## some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`9.090909%` = 1.59590909090909, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`9.090909%` = 1.59590909090909, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`9.090909%` = 1.64131818181818, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`9.090909%` = 1.64131818181818, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`8.333333%` = 1.58949166666667, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`8.333333%` = 1.58949166666667, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.692308%` = 1.5539, `15.38462%` =
## 1.8195, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.692308%` = 1.5539, `15.38462%` =
## 1.8195, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.692308%` = 1.57991538461538, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.692308%` = 1.57991538461538, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.142857%` = 1.54201428571429, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.142857%` = 1.54201428571429, : some
## 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.142857%` = 1.52245, `14.28571%` =
## 1.7573, : some 'x' values beyond boundary knots may cause ill-conditioned bases

## Warning in bs(dis, degree = 3L, knots = c(`7.142857%` = 1.52245, `14.28571%` =
## 1.7573, : some 'x' values beyond boundary knots may cause ill-conditioned bases
```

```r
plot(3:16, cv[3:16],xlab = "Degree of Freedom", ylab="Test MSE",type="l")
points(which.min(cv),cv[which.min(cv)],col="red",pch=20,cex=2)
```



7.10

```
data("College")
set.seed(1)
train_id <- sample(1:nrow(College),500)
train <- College[train_id,]
test <- College[-train_id]

fit_fwd <- regsubsets(Outstate~., train, nvmax = ncol(College)-1,method="forward")
fwd_summary <- summary(fit_fwd)

test_mat <- model.matrix(Outstate~., test, nvmax=ncol(College)-1)
```

```
err_fwd <- rep(NA,ncol(College)-1)
for (i in 1:(ncol(College)-1)) {
  coeff <- coef(fit_fwd,id=i)
  pred_fwd <- test_mat[,names(coeff)] %*% coeff
  err_fwd[i] <- mean((test$Outstate-pred_fwd)^2)
}
par(mfrow = c(2,2))
plot(err_fwd, type = "b", main = "Test MSE", xlab = "Number of Predictors")
min_mse <- which.min(err_fwd)
points(min_mse, err_fwd[min_mse], col = "red", pch = 4, lwd = 5)
plot(fwd_summary$adjr2, type = "b", main = "Adjusted R^2", xlab = "Number of Predictors")
max_adjr <- which.max(fwd_summary$adjr2)
points(max_adjr, fwd_summary$adjr2[max_adjr], col = "red", pch = 4, lwd = 5)
plot(fwd_summary$cp, type = "b", main = "Cp", xlab = "Number of Predictors")
min_cp <- which.min(fwd_summary$cp)
points(min_cp, fwd_summary$cp[min_cp], col = "red", pch = 4, lwd = 5)
plot(fwd_summary$bic, type = "b", main = "BIC", xlab = "Number of Predictors")
min_bic <- which.min(fwd_summary$bic)
points(min_bic, fwd_summary$bic[min_bic], col = "red", pch = 4, lwd = 5)
```

##(b)

```
#install.packages("gam")
library(gam)
```

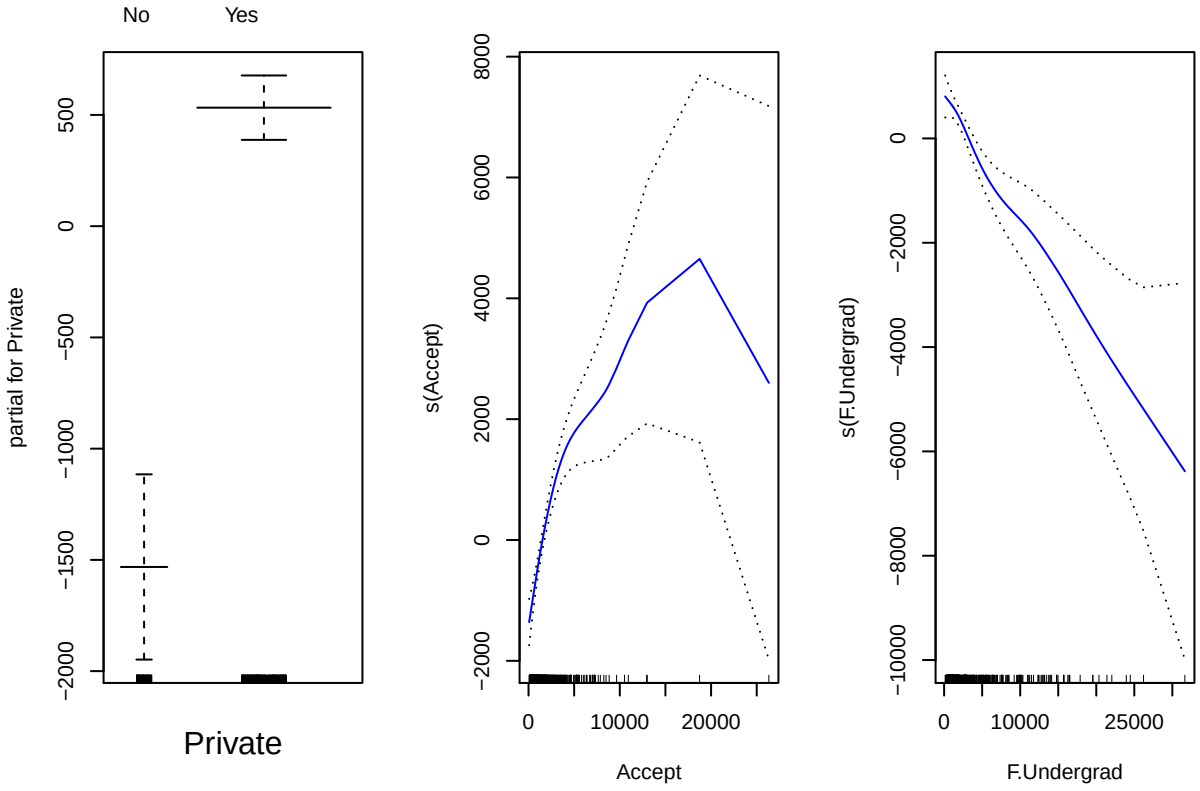
```
## Loading required package: foreach
```

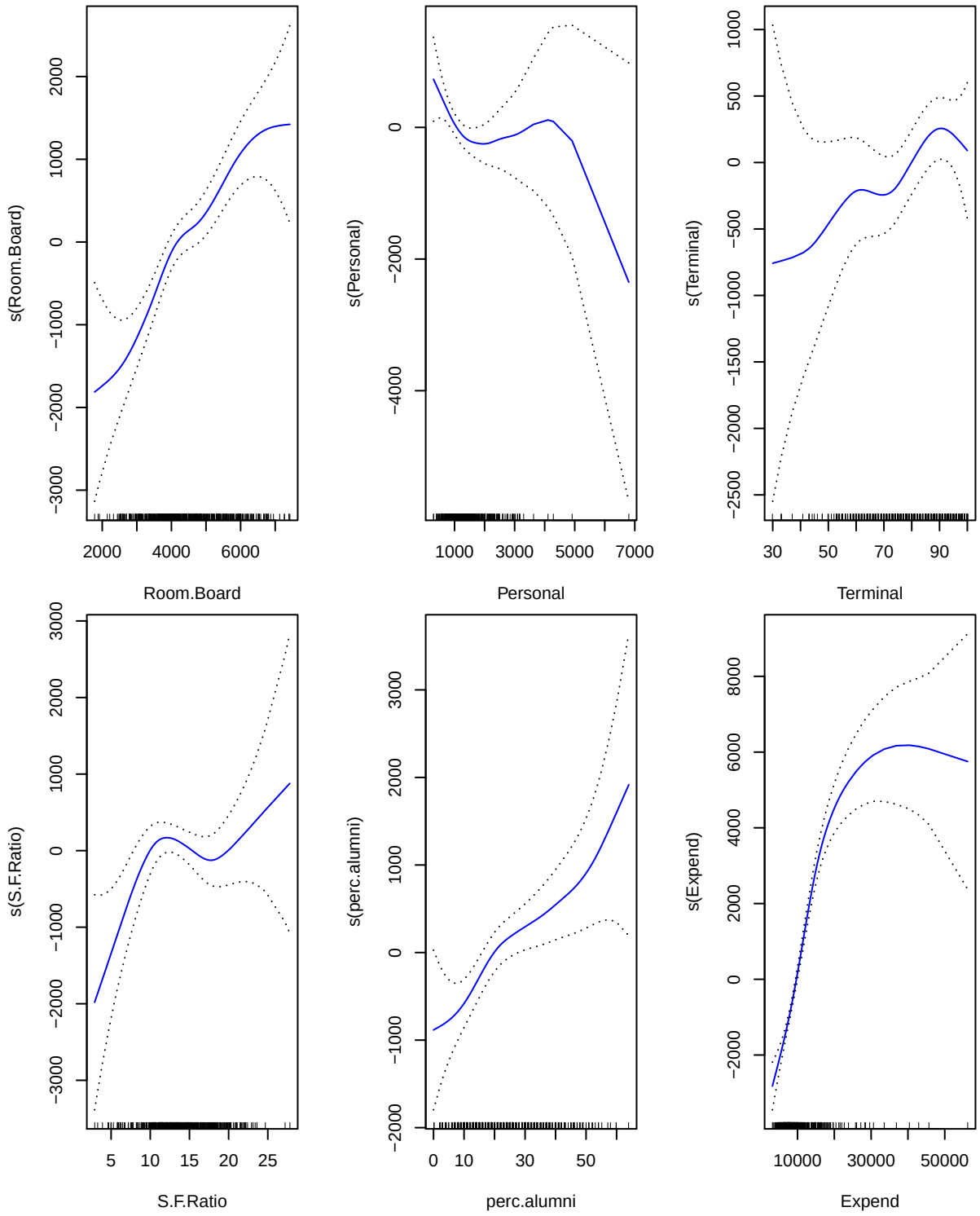
```
## Loaded gam 1.20
```

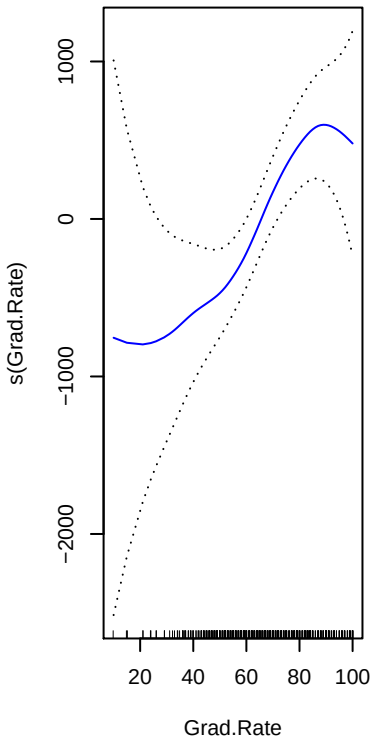
```
##
## Attaching package: 'gam'
```

```
## The following objects are masked from 'package:mgcv':
##
##     gam, gam.control, gam.fit, s
```

```
gam1 <- gam(Outstate ~ Private + s(Accept) + s(F.Undergrad) + s(Room.Board) + s(Personal) + s(Terminal)
par(mfrow = c(1,3))
plot(gam1, se = TRUE, col = "blue")
```




s(Room.Board)
Room.Board
s(Personal)
Personal
s(Terminal)
Terminal
s(S.F.Ratio)
S.F.Ratio
s(perc.alumni)
perc.alumni
s(Expend)
Expend



##(c)

pred_gam <- predict(gam1,test) err_gam <- mean((test$Outstate-pred_gam)^2) err_gam

tss_gam <- mean(((test$Outstate) − mean(test$Outstate))^2) rss_gam <- 1-err_gam/tss_gam rss_gam

##(d)

```
summary(gam1)
```

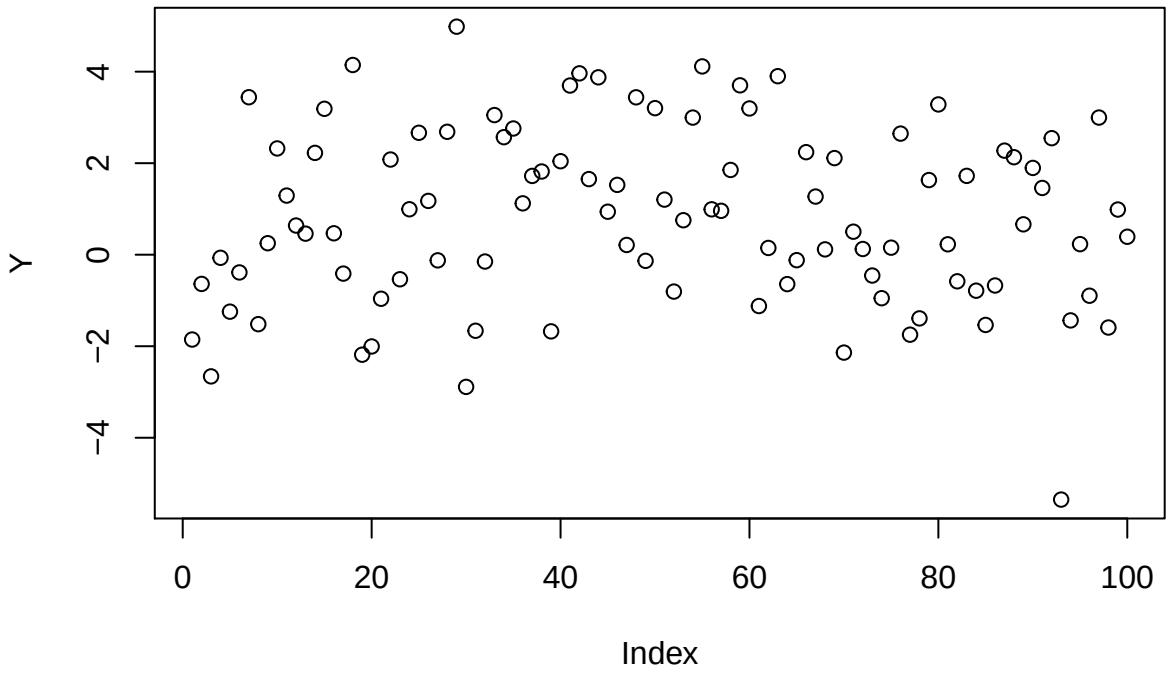
```
##
## Call: gam(formula = Outstate ~ Private + s(Accept) + s(F.Undergrad) +
##     s(Room.Board) + s(Personal) + s(Terminal) + s(S.F.Ratio) +
##     s(perc.alumni) + s(Expend) + s(Grad.Rate), data = train)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -6354.35 -1155.70    80.98  1166.22  7146.19
##
## (Dispersion Parameter for gaussian family taken to be 3239316)
##
##     Null Deviance: 8686699532 on 499 degrees of freedom
## Residual Deviance: 1496562646 on 461.9996 degrees of freedom
## AIC: 8952.854
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##                  Df     Sum Sq    Mean Sq F value    Pr(>F)
## Private           1 2298947726 2298947726 709.702 < 2.2e-16 ***
## s(Accept)         1  568225500  568225500 175.415 < 2.2e-16 ***
## s(F.Undergrad)    1  161250662  161250662  49.779 6.329e-12 ***
## s(Room.Board)     1 1256674967 1256674967 387.945 < 2.2e-16 ***
## s(Personal)       1   69570045   69570045  21.477 4.666e-06 ***
## s(Terminal)       1  374083666  374083666 115.482 < 2.2e-16 ***
```

```
## s(S.F.Ratio)     1  268164925  268164925  82.784 < 2.2e-16 ***
## s(perc.alumni)   1  246623307  246623307  76.134 < 2.2e-16 ***
## s(Expend)        1  725498255  725498255 223.966 < 2.2e-16 ***
## s(Grad.Rate)     1   49858577   49858577  15.392 0.0001006 ***
## Residuals      462 1496562646    3239316
## ---
## 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(Accept)            3 12.0063 1.394e-07 ***
## s(F.Undergrad)       3  1.5835  0.192553
## s(Room.Board)        3  2.5238  0.057089 .
## s(Personal)          3  2.4493  0.062979 .
## s(Terminal)          3  1.3173  0.268075
## s(S.F.Ratio)         3  4.2559  0.005567 **
## s(perc.alumni)       3  0.9564  0.413178
## s(Expend)            3 27.6450 2.220e-16 ***
## s(Grad.Rate)         3  1.2384  0.295210
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

7.11

##(a)

```
set.seed(99)
X1 <- rnorm(100)
X2 <- rnorm(100)
eps <- rnorm(1:100,sd=1)

b_0=0.8
b_1=-1.5
b_2=1
Y=b_0+b_1*X1+b_2*X2+eps
plot(Y)
```



##(b)

```
b_h1 <-3
```

##(c)

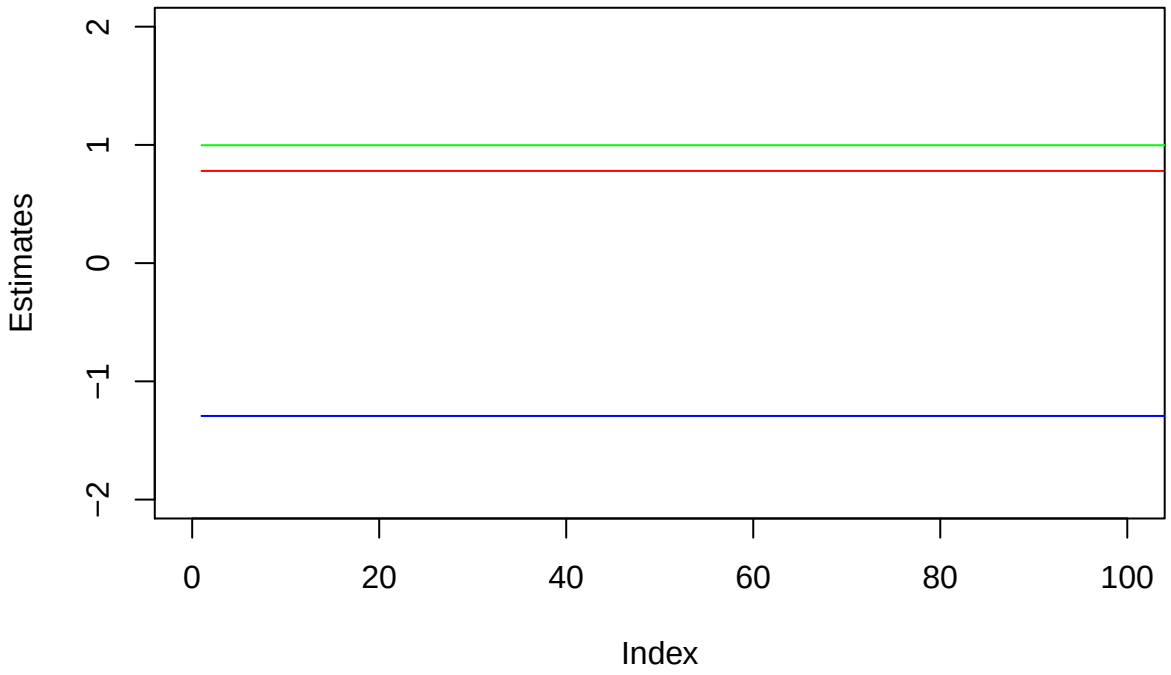
```
a=Y-b_h1*X1
b_h2=lm(a~X2)$coef[2]
```

##(d)

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

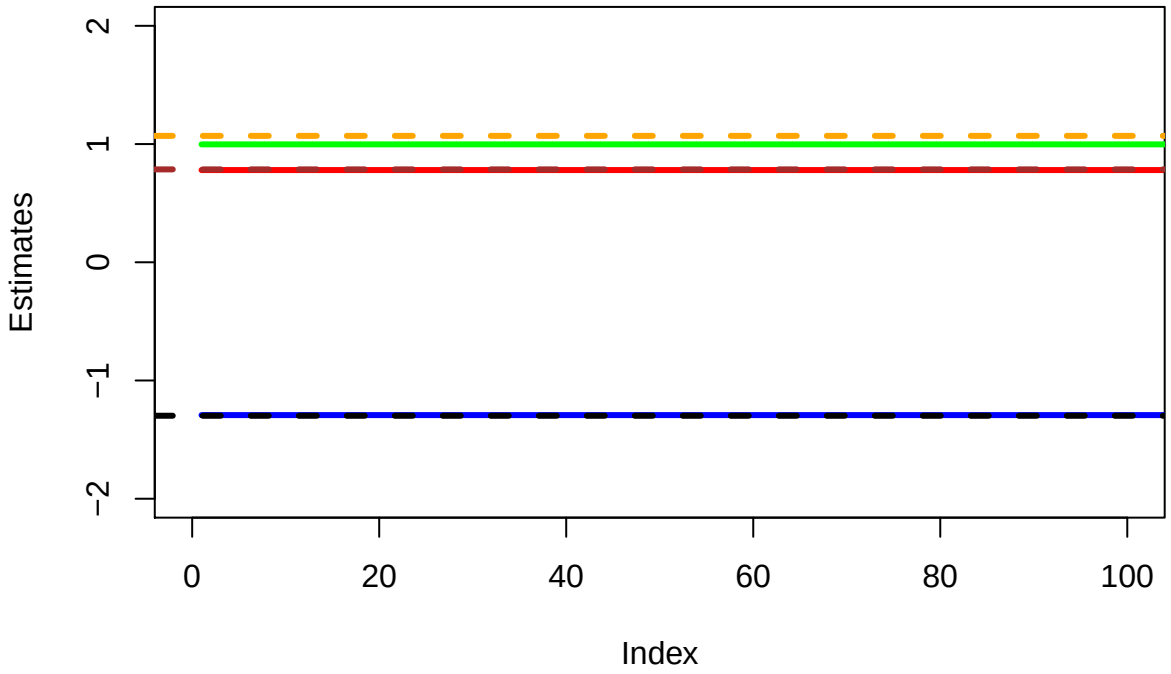
##(e)

```
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")
```



##(f)

```
fit_7.11f <- 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(fit_7.11f)[1], lty = "dashed", col = "brown", lwd = 3)
abline(h = coef(fit_7.11f)[2], lty = "dashed", col = "black", lwd = 3)
abline(h = coef(fit_7.11f)[3], lty = "dashed", col = "orange", lwd = 3)
```



##(g)

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

```
##      b_hat0    b_hat1    b_hat2
## 1 0.7799804 -1.292655 0.9972832
## 2 0.7799804 -1.292655 0.9972832
## 3 0.7799804 -1.292655 0.9972832
## 4 0.7799804 -1.292655 0.9972832
## 5 0.7799804 -1.292655 0.9972832
## 6 0.7799804 -1.292655 0.9972832
```

One is enough.