PS8

Prateek Kumar

3 November 2018

# Problem 1

training <- rep(c(1,1,1,0,0,0,-1,-1,-1),each=3)  
feedback <- rep(c(1,0,-1,1,0,-1,1,0,-1),each=3)  
  
group <- rep(c(1,-1,0,-1,0,1,0,1,-1),each=3)  
  
training %\*% group

## [,1]  
## [1,] 0

cor(training, group)

## [1] 0

feedback %\*% group

## [,1]  
## [1,] 0

cor(feedback, group)

## [1] 0

set.seed(100)  
group1 <- 1:27  
group2 <- rep(-1:1, 9)  
group3 <- 1:27-mean(1:27)  
group4 <- rep(1:3, 9)  
group5 <- rep(-1:1,each=9)  
group6 <- runif(27)-.5  
group7 <- rep(1:9,each=3)  
  
training %\*% group2

## [,1]  
## [1,] 0

cor(training, group2)

## [1] 0

training %\*% group4

## [,1]  
## [1,] 0

cor(training, group4)

## [1] 0

feedback %\*% group2

## [,1]  
## [1,] 0

cor(feedback, group2)

## [1] 0

feedback %\*% group4

## [,1]  
## [1,] 0

cor(feedback, group4)

## [1] 0

feedback %\*% group5

## [,1]  
## [1,] 0

cor(feedback, group5)

## [1] 0

# Problem2

data <- data.frame(buyerid = group7,  
 timeframe=group4)  
  
set.seed(103)  
  
buyerbase <- runif(9)  
timebase <- c(1,2,2.5)  
agebase <- 20+runif(9)\* 40  
  
data$age <- round(agebase[data$buyerid])  
data$purchase <- round( 25 + buyerbase[data$buyerid] \* 50 +  
 data$age \* .5+   
 timebase[data$timeframe] \* 7 +  
 rnorm(27)\*4,2)  
  
  
lm(purchase ~ timeframe+age, data = data)$coef

## (Intercept) timeframe age   
## 46.3220369 5.3900000 0.3200048

lm(purchase ~ timeframe, data = data)$coef

## (Intercept) timeframe   
## 58.55333 5.39000

lm(purchase ~ age, data = data)$coef

## (Intercept) age   
## 57.1020369 0.3200048

age1 <- data$age-mean(data$age)  
tf1 <- data$timeframe-mean(data$timeframe)  
  
lm(purchase ~ tf1+age1, data = data)$coef

## (Intercept) tf1 age1   
## 69.3333333 5.3900000 0.3200048

lm(purchase ~ tf1, data = data)$coef

## (Intercept) tf1   
## 69.33333 5.39000

lm(purchase ~ age1, data = data)$coef

## (Intercept) age1   
## 69.3333333 0.3200048

data2 <- data[-1,]  
lm(purchase ~ timeframe+age, data = data2)$coef

## (Intercept) timeframe age   
## 47.3474132 5.1137764 0.3124496

lm(purchase ~ timeframe, data = data2)$coef

## (Intercept) timeframe   
## 59.556327 5.013878

lm(purchase ~ age, data = data2)$coef

## (Intercept) age   
## 58.0842835 0.3043129

age2 <- data2$age-mean(data2$age)  
tf2 <- data2$timeframe-mean(data2$timeframe)  
  
lm(purchase ~ tf2+age2, data = data2)$coef

## (Intercept) tf2 age2   
## 69.7769231 5.1137764 0.3124496

lm(purchase ~ tf2, data = data2)$coef

## (Intercept) tf2   
## 69.776923 5.013878

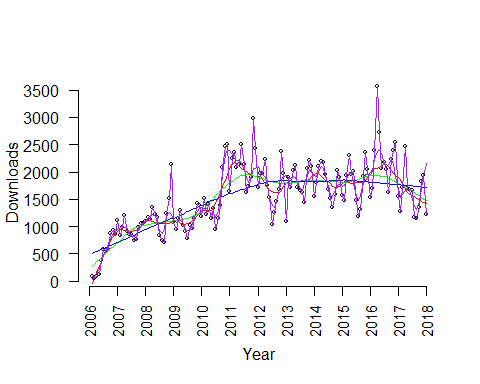
lm(purchase ~ age2, data = data2)$coef

## (Intercept) age2   
## 69.7769231 0.3043129

# Problem 3

# Q1

down <- read.csv("DownloadData.csv")  
down$Month <- as.factor(substr(down$Date,6,8))  
down$MonthNumber <- 1:nrow(down)  
  
plot(as.numeric(down$Date),down$Downloads,xaxt="n",bty="n",pch=21,cex=.5,type="p",las=1,  
 ylab="Downloads",xlab="Year")  
axis(1,0:12\*12,2006:2018,las=3,cex.axis=.95)  
  
lines(lowess(down$Downloads~down$MonthNumber, f=0.10), col="red")  
lines(lowess(down$Downloads~down$MonthNumber, f=0.25), col="green")  
lines(lowess(down$Downloads~down$MonthNumber, f=0.50), col="blue")  
lines(lowess(down$Downloads~down$MonthNumber, f=0.01), col="purple")  
lines(lowess(down$Downloads~down$MonthNumber, f=0.05), col="purple")



plot(as.numeric(down$Date),down$Downloads,xaxt="n",bty="n",pch=21,cex=.5,type="p",las=1,  
 ylab="Downloads",xlab="Year")  
axis(1,0:12\*12,2006:2018,las=3,cex.axis=.95)  
  
lmodel <- loess(down$Downloads~down$MonthNumber)  
xs <- 0:50000/100  
points(xs,predict(lmodel,xs),type="l",col="red",lwd=2)  
summary(lmodel)

## Call:  
## loess(formula = down$Downloads ~ down$MonthNumber)  
##   
## Number of Observations: 144   
## Equivalent Number of Parameters: 4.34   
## Residual Standard Error: 405.6   
## Trace of smoother matrix: 4.73 (exact)  
##   
## Control settings:  
## span : 0.75   
## degree : 2   
## family : gaussian  
## surface : interpolate cell = 0.2  
## normalize: TRUE  
## parametric: FALSE  
## drop.square: FALSE

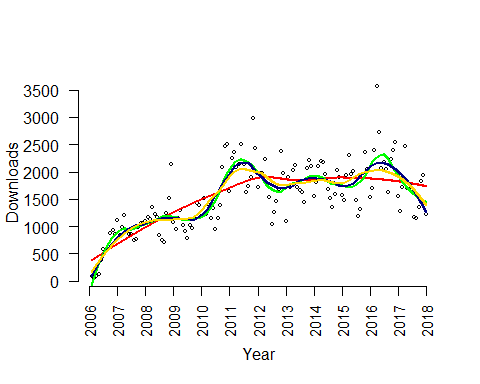
lmodel2 <- loess(down$Downloads~down$MonthNumber,span=.2)  
points(xs,predict(lmodel2,xs),type="l",col="green",lwd=2)  
summary(lmodel2)

## Call:  
## loess(formula = down$Downloads ~ down$MonthNumber, span = 0.2)  
##   
## Number of Observations: 144   
## Equivalent Number of Parameters: 14.98   
## Residual Standard Error: 351.1   
## Trace of smoother matrix: 16.57 (exact)  
##   
## Control settings:  
## span : 0.2   
## degree : 2   
## family : gaussian  
## surface : interpolate cell = 0.2  
## normalize: TRUE  
## parametric: FALSE  
## drop.square: FALSE

lmodel3 <- loess(down$Downloads~down$MonthNumber,span=.3)  
points(xs,predict(lmodel3,xs),type="l",col="navy",lwd=2)  
summary(lmodel3)

## Call:  
## loess(formula = down$Downloads ~ down$MonthNumber, span = 0.3)  
##   
## Number of Observations: 144   
## Equivalent Number of Parameters: 10.19   
## Residual Standard Error: 358.9   
## Trace of smoother matrix: 11.26 (exact)  
##   
## Control settings:  
## span : 0.3   
## degree : 2   
## family : gaussian  
## surface : interpolate cell = 0.2  
## normalize: TRUE  
## parametric: FALSE  
## drop.square: FALSE

lmodel4 <- loess(down$Downloads~down$MonthNumber,span=.4)  
points(xs,predict(lmodel4,xs),type="l",col="gold",lwd=2)

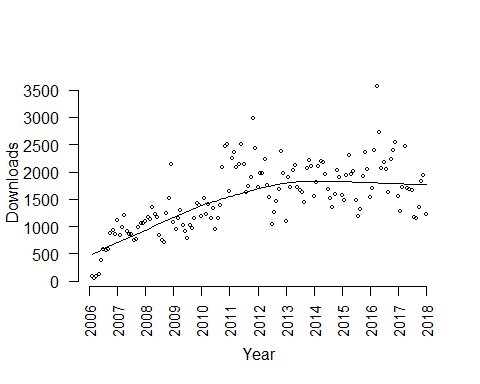


summary(lmodel4)

## Call:  
## loess(formula = down$Downloads ~ down$MonthNumber, span = 0.4)  
##   
## Number of Observations: 144   
## Equivalent Number of Parameters: 7.71   
## Residual Standard Error: 365.4   
## Trace of smoother matrix: 8.5 (exact)  
##   
## Control settings:  
## span : 0.4   
## degree : 2   
## family : gaussian  
## surface : interpolate cell = 0.2  
## normalize: TRUE  
## parametric: FALSE  
## drop.square: FALSE

# Q2

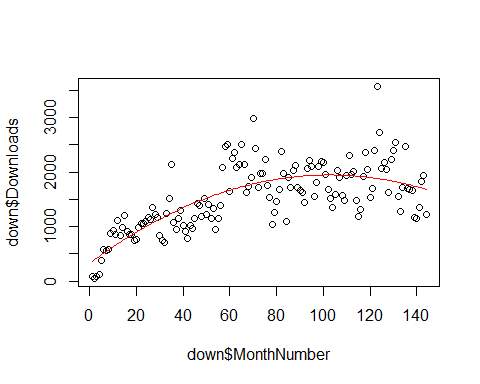
plot(down$MonthNumber,down$Downloads,xaxt="n",bty="n",pch=21,cex=.5,type="p",las=1,  
 ylab="Downloads",xlab="Year")  
axis(1,0:12\*12,2006:2018,las=3,cex.axis=.95)  
  
  
lines(lowess(down$Downloads~down$MonthNumber))



poly2 <- lm(Downloads ~ poly(MonthNumber, 2), data = down)   
summary(poly2)

##   
## Call:  
## lm(formula = Downloads ~ poly(MonthNumber, 2), data = down)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -807.14 -299.35 -2.19 202.95 1685.16   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1541.67 33.95 45.406 < 2e-16 \*\*\*  
## poly(MonthNumber, 2)1 4616.69 407.43 11.331 < 2e-16 \*\*\*  
## poly(MonthNumber, 2)2 -2846.36 407.43 -6.986 1.03e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 407.4 on 141 degrees of freedom  
## Multiple R-squared: 0.5569, Adjusted R-squared: 0.5506   
## F-statistic: 88.6 on 2 and 141 DF, p-value: < 2.2e-16

pred<-predict(poly2)  
  
  
plot(down$MonthNumber,down$Downloads)  
lines(down$MonthNumber,y=pred,col="red")



poly1 <- lm(Downloads ~ poly(MonthNumber, 1), data = down)  
poly3 <- lm(Downloads ~ poly(MonthNumber, 3), data = down)  
poly4 <- lm(Downloads ~ poly(MonthNumber, 4), data = down)  
poly5 <- lm(Downloads ~ poly(MonthNumber, 5), data = down)  
  
  
data.frame(model = paste ("lm" ,1:5 , sep =""),  
 rbind ( extractAIC ( poly1 ),  
 extractAIC ( poly2 ),  
 extractAIC ( poly3 ),  
 extractAIC ( poly4 ),  
 extractAIC ( poly5 )))

## model X1 X2  
## 1 lm1 2 1774.614  
## 2 lm2 3 1733.811  
## 3 lm3 4 1735.810  
## 4 lm4 5 1737.027  
## 5 lm5 6 1738.086

extractBIC <- function (model)  
{  
 extractAIC (model ,k= log ( length ( model $ residuals )))  
}  
  
data.frame( model = paste ("lm" ,1:5 , sep =""),  
 rbind ( extractBIC ( poly1 ),  
 extractBIC ( poly2 ),  
 extractBIC ( poly3 ),  
 extractBIC ( poly4 ),  
 extractBIC ( poly5 )))

## model X1 X2  
## 1 lm1 2 1780.554  
## 2 lm2 3 1742.721  
## 3 lm3 4 1747.690  
## 4 lm4 5 1751.876  
## 5 lm5 6 1755.905

# Q3

fit3 <- lm(Downloads ~ poly(MonthNumber,2)+as.factor(Month)+0, data = down)   
pred<-predict(fit3)  
plot(down$MonthNumber,down$Downloads)  
lines(down$MonthNumber,y=pred,col="red")

