data\_analysis\_2.R

khagen

Sat Feb 22 11:05:56 2020

# $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  
# Data Analysis with R  
# Khagendra Adhikari  
# 2016  
#  
# $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  
  
# Use packages   
library(DAAG)

## Loading required package: lattice

library(lattice)  
library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:DAAG':  
##   
## hills

library(leaps)  
  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# Q.1  
# The following code is designed to explore effects that can result from the omission  
# of explanatory varaibles:  
# > x1<-runif(10)  
# > x2<-rbinom(10,1,1-x1)  
# > y<-5\*x1+x2+rnorm(10,sd=0.1) # simulated model, coef of x2 is positive  
# > y.lm<-lm(y~factor(x2)) # model fitting to observed data without x1  
# > coef(y.lm)  
# (Intercept) factor(x2)1  
# 3.2429990 -0.3789021  
# > y.lm2<-lm(y~x1+factor(x2)) # correct model  
# > coef(y.lm2)  
# (Intercept) x1 factor(x2)1  
# 0.034054df.b 5.0017938 1.0232510  
#  
# What happens if x2 is generated according to  
# x2 <- rbinom(10,1,x1)?  
# x2 <- rbinom(10,1, .5)?  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
set.seed(100)  
x1.0 <- runif(10)  
x2.0 <- rbinom(10, 1, 1-x1.0)  
  
## This function comapreModels compare the coefficent of omitted  
## model with the full model.  
  
compareModels <- function(x2, x1 = x1.0)  
{  
 y <- 5\*x1 + x2 + rnorm(10, sd=0.1) # simulated model, coef of x2 is positive  
 y.lm <- lm(y ~ factor(x2)) # model fitting to observed data without x1  
 print("Omited model")  
 print(coef(y.lm))  
 y.lm2 <- lm(y ~ x1 + factor(x2)) # correct model  
 print("correct model")  
 print(coef(y.lm2))  
}  
  
compareModels(x2.0) # given in question

## [1] "Omited model"  
## (Intercept) factor(x2)1   
## 2.5791641 -0.3569772   
## [1] "correct model"  
## (Intercept) x1 factor(x2)1   
## 0.06075099 4.84112189 1.06645652

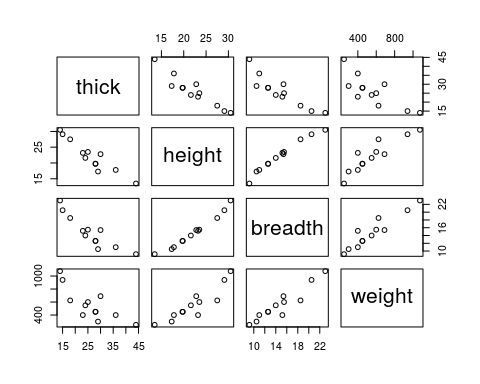
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## The coefficient of x2.0 has opposite sign.  
##   
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
x2.1 <- rbinom(10, 1, x1.0) # First model  
compareModels(x2.1)

## [1] "Omited model"  
## (Intercept) factor(x2)1   
## 1.527577 1.948628   
## [1] "correct model"  
## (Intercept) x1 factor(x2)1   
## -0.1490426 5.3378385 1.0038321

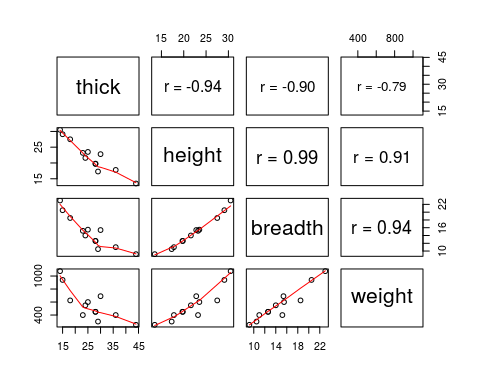
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## Here, x2.1 depends on x1.0. So, the cofficient of x2.1 has correct sign  
## but not the value.The omitted variable bias.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
x2.2 <- rbinom(10, 1, 0.5) # Second model  
compareModels(x2.2)

## [1] "Omited model"  
## (Intercept) factor(x2)1   
## 1.206932 2.152508   
## [1] "correct model"  
## (Intercept) x1 factor(x2)1   
## 0.017888 4.848618 1.062489

# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## Here, x2.2 does not depent on x1.0. So, the coeficient of estimation  
## of x2.2 is reasonable in comparision to correct(full) model.   
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# Q.2  
# For the data frame oddbooks (DAAG),  
# (a) Use the function pairs( ) to display the scatterplot matrix. Which paris of   
# variables show evidence of a strong relationships?  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
pairs(oddbooks)



# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## Height and breadth shows the evidence of strong relationship. And other variables  
## have also significant correlation.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
## Full model  
oddbooks.all.lm <- lm(weight ~ thick + height + breadth, data = oddbooks)  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (b) In each panel of the scatterplot matrix, record the correlation for the panel and  
# add the smooth fitting curve in the panel. Hint: look for functions in help page  
# of pairs().  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
## This function panel.cor is imported from lecture code multilr.R.   
panel.cor <- function(x, y, digits = 2, prefix = "r = ", cex.cor, ...)  
{  
 usr <- par("usr"); on.exit(par(usr))  
 par(usr = c(0, 1, 0, 1))  
 r <- cor(x,y)  
 ar <- abs(r)  
 txt <- format(c(r, 0.123456789), digits = digits)[1]  
 txt <- paste0(prefix, txt)  
 if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)  
 text(0.5, 0.5, txt, cex = cex.cor \* ar)  
}  
  
pairs(oddbooks,lower.panel = panel.smooth, upper.panel = panel.cor)



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (c) Use the backward selection method to select the best linear regression model  
# based on AIC.  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
## stepAIC {MASS}: Performs stepwise model selection by AIC.  
## smallar AIC value gives better model.  
  
stepAIC(oddbooks.all.lm, direction = "backward") # backward selection

## Start: AIC=110.44  
## weight ~ thick + height + breadth  
##   
## Df Sum of Sq RSS AIC  
## - height 1 1749 62954 108.78  
## - thick 1 1772 62977 108.79  
## <none> 61205 110.44  
## - breadth 1 33919 95124 113.74  
##   
## Step: AIC=108.78  
## weight ~ thick + breadth  
##   
## Df Sum of Sq RSS AIC  
## - thick 1 11348 74302 108.77  
## <none> 62954 108.78  
## - breadth 1 189724 252678 123.46  
##   
## Step: AIC=108.77  
## weight ~ breadth  
##   
## Df Sum of Sq RSS AIC  
## <none> 74302 108.77  
## - breadth 1 597239 671542 133.19

##   
## Call:  
## lm(formula = weight ~ breadth, data = oddbooks)  
##   
## Coefficients:  
## (Intercept) breadth   
## -272.95 56.21

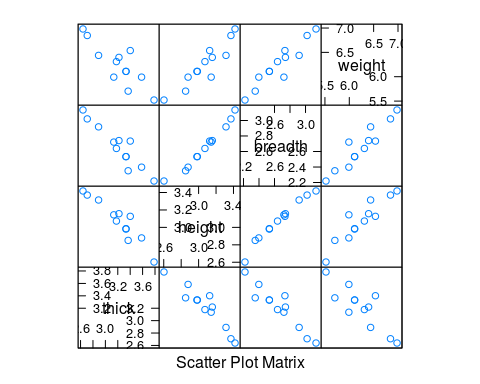
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## Based on AIC, weight ~ breadth is the best linear regression model.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (d) Search for the best model on all possible models based on the criterion adjust R^2.  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
## The error of regression goes down as adjusted R^ goes up.  
## uses regression subset selection packages leaps based on adjusted R^2.  
  
precditors.matrix <- data.matrix(oddbooks)[, 1:3]  
leaps(precditors.matrix, oddbooks$weight, method = "adjr2")

## $which  
## 1 2 3  
## 1 FALSE FALSE TRUE  
## 1 FALSE TRUE FALSE  
## 1 TRUE FALSE FALSE  
## 2 TRUE FALSE TRUE  
## 2 FALSE TRUE TRUE  
## 2 TRUE TRUE FALSE  
## 3 TRUE TRUE TRUE  
##   
## $label  
## [1] "(Intercept)" "1" "2" "3"   
##   
## $size  
## [1] 2 2 2 3 3 3 4  
##   
## $adjr2  
## [1] 0.8782912 0.8070387 0.5861073 0.8854222 0.8853805 0.8268728 0.8746817

# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## The observation shows that the that includes thick (column 1) and  
## breadth (columns 3) of precditors matrix has larger(close to 1).   
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
## compare adj.r.sruqred with the summary of that model.  
oddbooks.tb.lm <- lm(weight ~ thick + breadth, data = oddbooks)  
summary(oddbooks.tb.lm)$adj.r.squared

## [1] 0.8854222

# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## Based on adjusted R square, the model weight ~ thick + breadth is the best.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (e) Fit the following regression relationships with log transformations. To do such  
# transformation is due to that we might expect weight proportional to volume(tbh).  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
## scatterplot matrix in log scale  
splom(~log(oddbooks))



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# e.i. log(weight) on log(thick), log(height) and log(breadth).  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
oddbooks.log.lm <- lm(log(weight) ~ log(thick) + log(height) + log(breadth), data = oddbooks)  
summary(oddbooks.log.lm)

##   
## Call:  
## lm(formula = log(weight) ~ log(thick) + log(height) + log(breadth),   
## data = oddbooks)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.33818 -0.02858 0.06164 0.07445 0.12585   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -0.7191 3.2162 -0.224 0.829  
## log(thick) 0.4648 0.4344 1.070 0.316  
## log(height) 0.1537 1.2734 0.121 0.907  
## log(breadth) 1.8772 1.0696 1.755 0.117  
##   
## Residual standard error: 0.1611 on 8 degrees of freedom  
## Multiple R-squared: 0.8978, Adjusted R-squared: 0.8595   
## F-statistic: 23.43 on 3 and 8 DF, p-value: 0.000257

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# e.ii. log(weight) on log(thick) and 0.5\*(log(height)+log(breadth)).  
# Hint: the model formal is log(weight) ∼ log(thick) +I(0.5\*(log(height)+log(breadth))).  
# The reason for the use of the wrapper function I( ) is to prevent the parser  
# from giving \* the special meaning that it would otherwise have in a model formula.  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
oddbooks.log.lm1 <- lm(log(weight) ~ log(thick)   
 + I(0.5\*(log(height) + log(breadth))), data = oddbooks)  
summary(oddbooks.log.lm1)

##   
## Call:  
## lm(formula = log(weight) ~ log(thick) + I(0.5 \* (log(height) +   
## log(breadth))), data = oddbooks)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.35537 -0.02640 0.04914 0.08207 0.12986   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -1.5930 2.9316 -0.543 0.60007  
## log(thick) 0.4834 0.4234 1.142 0.28309  
## I(0.5 \* (log(height) + log(breadth))) 2.1905 0.5622 3.897 0.00364  
##   
## (Intercept)   
## log(thick)   
## I(0.5 \* (log(height) + log(breadth))) \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1573 on 9 degrees of freedom  
## Multiple R-squared: 0.8905, Adjusted R-squared: 0.8661   
## F-statistic: 36.58 on 2 and 9 DF, p-value: 4.764e-05

# What feature of the scatterplot matrix suggests that this might make sense to use  
# this form of equation?  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## The height and breath are strongly correlated. From scatter plot, we can see that  
## weight has similar corelation with both height and breadth, so we don't need to  
## include both as different predictor variables. It seems to consider the geometric   
## mean of height and breadth as a single variable.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
# Removing the intercept term gives better fittings? Explain why?  
oddbooks.log.lm1.no.intercept <- lm(log(weight)~ -1 + log(thick)   
 + I(0.5\*(log(height) + log(breadth))), data = oddbooks)  
summary(oddbooks.log.lm1.no.intercept)

##   
## Call:  
## lm(formula = log(weight) ~ -1 + log(thick) + I(0.5 \* (log(height) +   
## log(breadth))), data = oddbooks)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.35421 -0.02292 0.04219 0.10197 0.12676   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## log(thick) 0.25720 0.07497 3.431 0.00643  
## I(0.5 \* (log(height) + log(breadth))) 1.88876 0.08412 22.453 6.91e-10  
##   
## log(thick) \*\*   
## I(0.5 \* (log(height) + log(breadth))) \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1516 on 10 degrees of freedom  
## Multiple R-squared: 0.9995, Adjusted R-squared: 0.9994   
## F-statistic: 1.021e+04 on 2 and 10 DF, p-value: < 2.2e-16

# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## Yes, removing the intercept term gives better fitting. The p small value of the model   
## without intercept inticates that the estimated coefficeitns are signigicant. And, this  
## model has also smallar Std. Error compare to the previous model.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (f) Check multicolinearity with variance inflation factor. If the multicolinearity problem  
# exists, remove some variable to deal with it.  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
vif(oddbooks.all.lm)

## thick height breadth   
## 11.106 76.842 46.827

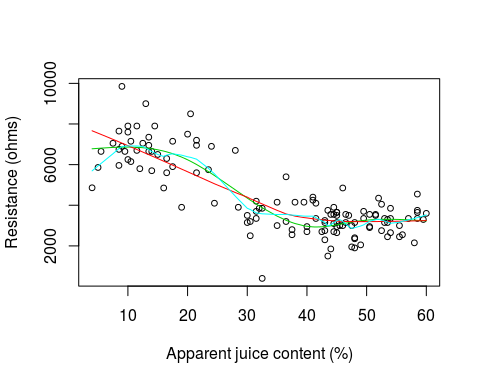
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## The scatter plot shows that height and breadth are highly correlated.   
## So, include one of these. Here, remove height variable with big vif.  
## i.e Weight ~ thick + breadth  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
vif(oddbooks.tb.lm)

## thick breadth   
## 5.1694 5.1694

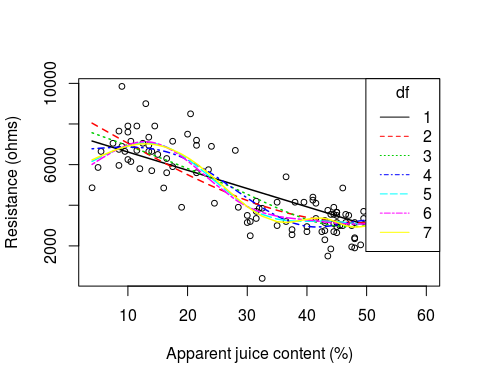
# here vif are ok.  
  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# Q.3.  
# The following compares regression splines with locally weighted regression (lowess), for  
# scatterplot smoothing, using the fruitohms data frame in DAAG.  
library(splines)  
plot(ohms ~ juice, cex = 0.8, xlab = "Apparent juice content (%)",  
 ylab = "Resistance (ohms)", data = fruitohms)  
hat <- with(fruitohms, fitted(lm(ohms ~ ns(juice, 4))))  
with(fruitohms, lines(juice, hat, col=3))  
  
attributes(ns(fruitohms$juice, 4))$knots

## 25% 50% 75%   
## 18.625 41.000 48.000

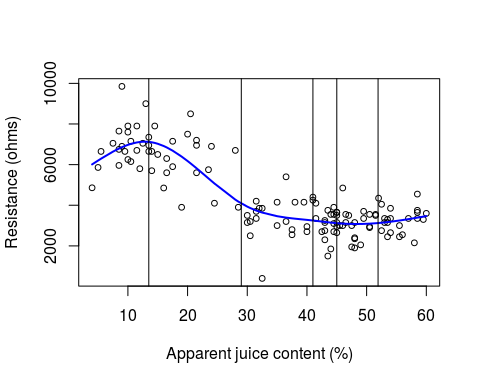
with(fruitohms, lines(lowess(juice, ohms), col = "red"))  
with(fruitohms, lines(lowess(juice, ohms, f=0.2), col = "cyan"))



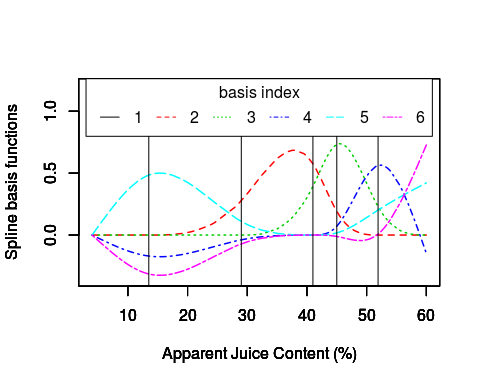
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (a) Experiment with different choices for number of degrees of freedom.  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
plot(ohms ~ juice, cex=0.8, xlab="Apparent juice content (%)",  
 ylab="Resistance (ohms)", data=fruitohms)  
dF <- c(1:7) # list of degree of freedom  
for (dOf in dF) {  
 hat <- with(fruitohms, fitted(lm(ohms ~ ns(juice, df = dOf))))  
 with(fruitohms, lines(juice, hat, col=dOf, lty=dOf, lwd = 1.5))  
}  
legend("topright", legend = dF, col = dF, lty = dF, title = "df" )



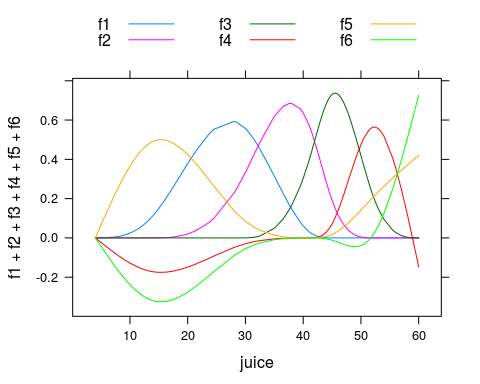
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (b) How many degrees of freedom seem needed to adequately capture the pattern of change?  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
## "  
## 6 degrees of freedom seem needed to adequately capture the pattern of the change.  
## "  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
df.b <- 6 # best degree of freedom  
plot(ohms ~ juice, cex = 0.8, xlab = "Apparent juice content (%)",  
 ylab = "Resistance (ohms)", data = fruitohms)  
hat.df.b <- with(fruitohms, fitted(lm(ohms ~ ns(juice, df=df.b))))  
with(fruitohms, lines(juice, hat.df.b, col="blue", lty=1, lwd = 2))  
knots <- as.data.frame(attributes(ns(fruitohms$juice,df=df.b))$knots)  
for (knot in knots) {abline(v=knot)}



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# (c) Plot the spline basic functions. Add vertical lines to the plot that show the knot  
# locations.  
# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
## natural splines basis curve.  
  
fruit.best.df.lm <- lm(ohms ~ ns(juice, df.b), data = fruitohms)  
model.m <- model.matrix(fruit.best.df.lm)  
for(index in 2 : df.b + 1){  
 plot(model.m[, index] ~ juice, xlab = "Apparent Juice Content (%)",  
 ylab="Spline basis functions", type="l", lty= index-1, lwd=1.5,   
 col = index-1, data=fruitohms, ylim=c(-0.35,1.2))  
 par(new=T)  
}  
for (knot in knots) {abline(v=knot)} # verticle lines at knot locations.  
legend("top",legend=1:df.b, col=1:df.b, lty = 1:df.b, seg.len = 1.3,   
 horiz = TRUE, title = "basis index" )



## Alternative method gives the same plot.  
## basis plot using lattice.  
nsdf.b <- as.data.frame(with(fruitohms, ns(juice, df.b))[, 1:df.b])  
names(nsdf.b) <- c("f1", "f2", "f3","f4", "f5", "f6")  
nsdf.b$juice <- fruitohms$juice  
xyplot(f1 + f2 + f3 + f4 + f5 + f6 ~ juice, type="l", data= nsdf.b,  
 auto.key = list(columns=3, points=FALSE, lines=TRUE))



# $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$