L12Ex_Hardwood_Conc_Rick_Davila

Rick Davila

5/23/2020

Perform data housekeeping - upload, name columns, display to make sure it reads properly, etc.

```
knitr::opts_chunk$set(echo = TRUE)

library(e1071)
library("xlsx")
```

```
## Warning: package 'xlsx' was built under R version 4.0.3
```

```
library(xtable)
library(MASS) # Needed for ginv() function

rm(list = ls())

# load data
Ex71 <- read.xlsx("data-ex-7-1.xlsx",
  sheetIndex = 1, sheetName=NULL, rowIndex=NULL,
  startRow=NULL, endRow=NULL, colIndex= c(1,2),
  as.data.frame=TRUE, header=TRUE, colClasses=NA,
  keepFormulas=FALSE, encoding="unknown")

# Give labels to data columns
names(Ex71) <- c("Concentration",
                 "Strength")
attach(Ex71)

# Output data to make sure it reads properly
out <- as.data.frame(c(Ex71))
colnames(out) <- c("Concentration",
                   "Strength")
tab <- (xtable(out, digits=c(0,1,1)))
print(tab, type="html")
```

	Concentration	Strength
1	1.0	6.3
2	1.5	11.1
3	2.0	20.0
4	3.0	24.0
5	4.0	26.1
6	4.5	30.0
7	5.0	33.8
8	5.5	34.0

9	6.0	38.1
10	6.5	39.9
11	7.0	42.0
12	8.0	46.1
13	9.0	53.1
14	10.0	52.0
15	11.0	52.5
16	12.0	48.0
17	13.0	42.8
18	14.0	27.8
19	15.0	21.9

```
# Output data structure and dimensions
str(Ex71)
```

'data.frame': 19 obs. of 2 variables: $ Concentration: num 1 1.5 2 3 4 4.5 5 5.5 6 6.5 … $ Strength : num 6.3 11.1 20 24 26.1 30 33.8 34 38.1 39.9 …

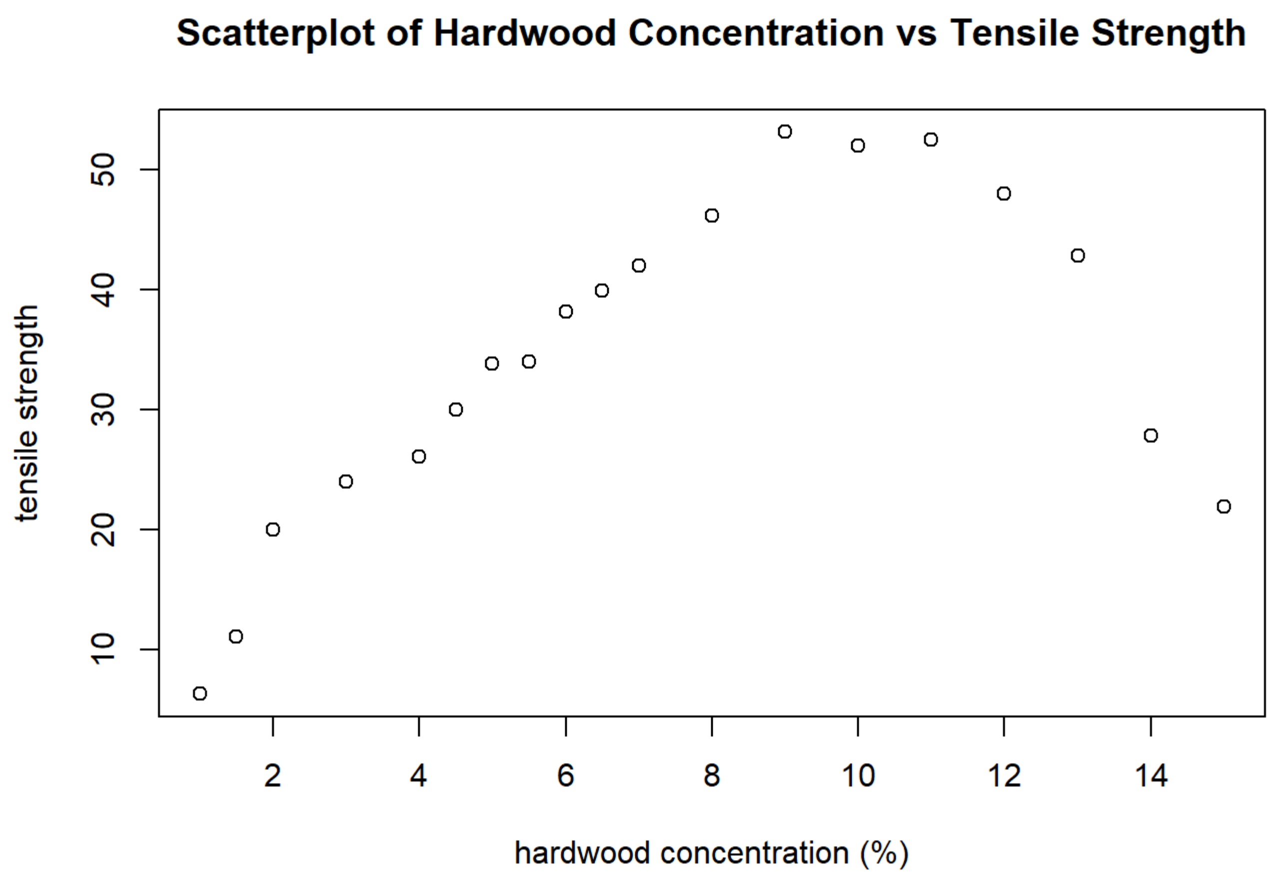
```
dim(Ex71)
```

[1] 19 2

Example 7.1 (p. 227-229)

Create scatterplot

```
plot(Concentration, Strength, main = "Scatterplot of Hardwood Concentration vs Tensile Strength"
,
     xlab = "hardwood concentration (%)",
     ylab = "tensile strength")
```



Fit a quadratic model on centered data; compare to p. 227

```
n <- length(Strength)
x_mean <- sum(Concentration)/n
x_1 <- Concentration - x_mean
x_2 <- x_1^2

model.71 <- lm(Strength ~ x_1 + x_2)

summary(model.71)
```

Call: lm(formula = Strength ~ x_1 + x_2)

Residuals: Min 1Q Median 3Q Max -5.8503 -3.2482 -0.7267 4.1350 6.5506

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 45.29497 1.48287 30.55 1.29e-15 ***x_1 2.54634 0.25384 10.03 2.63e-08*** x_2 -0.63455 0.06179 -10.27 1.89e-08 *** — Signif. codes: 0 ‘***’ **0.001** ’’ *0.01* ’ 0.05 ‘.’ 0.1 ’ ’ 1

Residual standard error: 4.42 on 16 degrees of freedom Multiple R-squared: 0.9085, Adjusted R-squared: 0.8971 F-statistic: 79.43 on 2 and 16 DF, p-value: 4.912e-09

```
xtable(summary(model.71))
```

	Estimate	Std. Error	t value	Pr(>|t|)
	<dbl>	<dbl>	<dbl>	<dbl>

	Estimate <dbl>	Std. Error <dbl>	t value <dbl>	Pr(>|t|) <dbl>
(Intercept)	45.2949731	1.48287271	30.54542	1.291590e-15
x_1	2.5463440	0.25383896	10.03134	2.629797e-08
x_2	-0.6345492	0.06178832	-10.26973	1.894349e-08

3 rows

Create ANOVA table (by hand and with R); compare to Table 7.2 on p. 228

```
# create ANOVA table by hand
n <- length(Strength)
k <- 2                    # number of regressors
p <- k + 1                # number of parameters

X <- cbind(matrix(1,length(Strength),1),as.matrix(x_1),as.matrix(x_2))
y <- as.matrix(Strength)

# X'X matrix
xTx <- t(X) %*% X

# Beta Coefficients
beta_hat <- ginv(xTx,tol =  .Machine$double.eps) %*% t(X) %*% y

# eqn 3.24
SS_Reg <- t(beta_hat) %*% t(X) %*% y - (sum(y))^2/n

MS_Reg <- SS_Reg/k

# eqn 3.16
SS_Res <- t(y) %*% y - t(beta_hat) %*% t(X) %*% y

# eqn 3.17
MS_Res <- SS_Res/(n-p)

# eqn 3.24
SS_T <- t(y) %*%  y - (sum(y))^2/n

F_0 <- MS_Reg/MS_Res

# t-critical and p-value calculations
# for alpha = 0.05
alpha = 0.05

p_val_f = 1 - pf(F_0, k, n-p)

# create ANOVA table with R
xtable(anova(model.71))
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
	<int>	<dbl>	<dbl>	<dbl>	<dbl>
x_1	1	1043.4274	1043.42743	53.39985	1.758404e-06
x_2	1	2060.8195	2060.81954	105.46729	1.894349e-08
Residuals	16	312.6383	19.53989	NA	NA

3 rows

Anova table by hand is listed above. The anova table by hand (i.e., values) are listed below:

SS Regression: 3104.2469747

SS Residual: 312.6382884

SS Total: 3416.8852632

Regression Degrees of Freedom: 2

Residual Degrees of Freedom: 16

Total Degrees of Freedom: 18

Mean Square Regression: 1552.1234874

Mean Square Residual: 19.539893

$F_0 = 79.4335714$

P Value: 4.9124048×10^{-9}

Test significance of quadratic term using Partial F-test. Use alpha = 0.01 as the significance level. Compare to p. 229

```
# full model including the quadratic term
model.full <- lm(Strength ~ x_1 + x_2)

# reduced model excluding the quadratic term
model.reduced <- lm(Strength ~ x_1)

# anova -- comparision of reduced to full model
anova(model.reduced, model.full)
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	17	2373.4578	NA	NA	NA	NA
2	16	312.6383	1	2060.82	105.4673	1.894349e-08

2 rows

```
# F crit
alpha <- 0.01
df_SS_R <- anova(model.reduced, model.full)$'Df'[2]
df_SS_Res <- anova(model.reduced, model.full)$'Res.Df'[2]

F_crit <- qf(1-alpha,df_SS_R,df_SS_Res)
```

We're investigating the contribution of the quadratmodel.reduced, model.fullic term to the model. That is, we wish to test

$$H_0 : \beta_2 = 0, H_1 : \beta_2 \neq 0$$

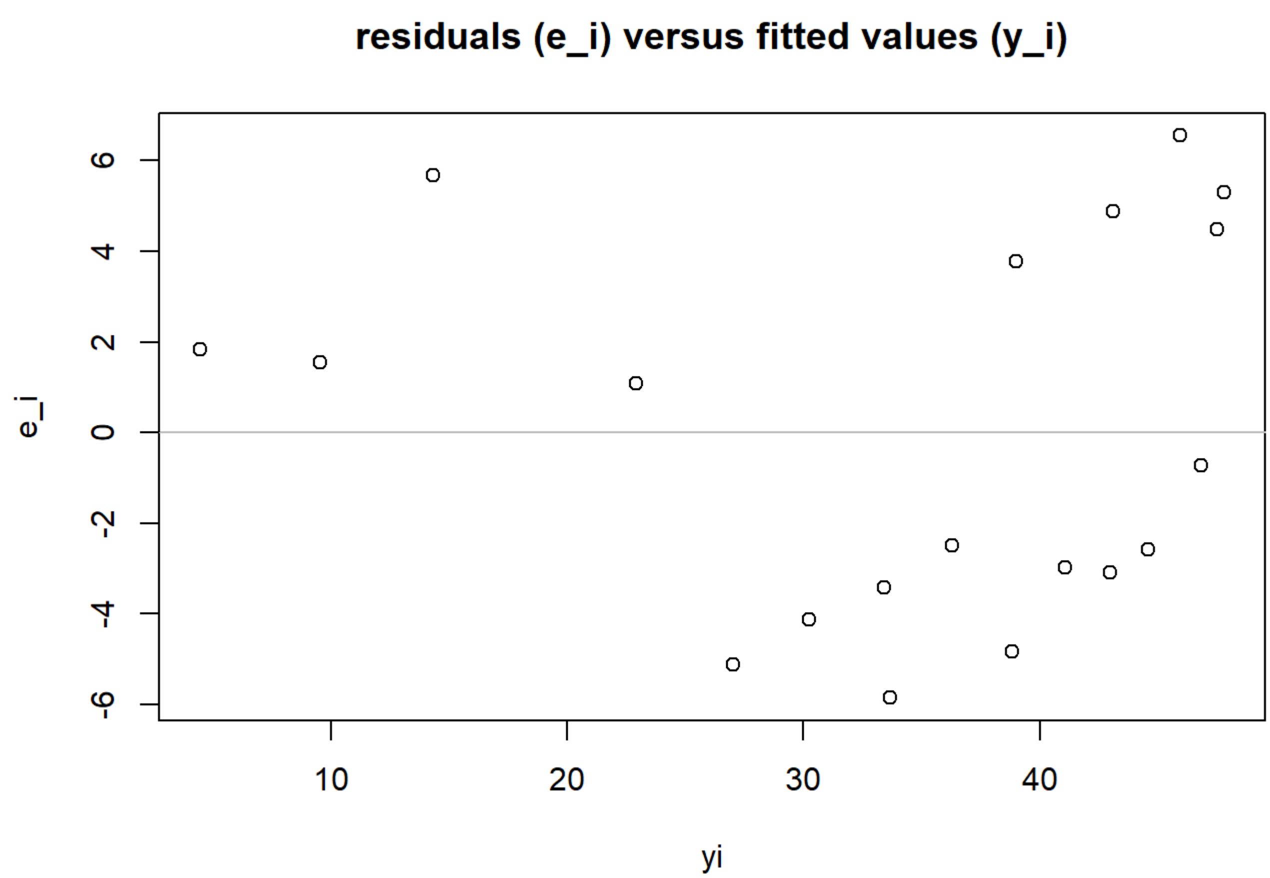
we have

$$F_0 = \frac{SS_R(\beta_2|\beta_1, \beta_0)/1}{MS_{Res}} = \frac{2060.819541/1}{19.539893} = 105.4672888$$

and since $F_{0.01,1,16} = 8.5309653$, we have $F_0 > F_{0.01,1,16}$ and as a result, we conclude that the quadratic term contributes significantly to the model.

Create residuals versus fits plot

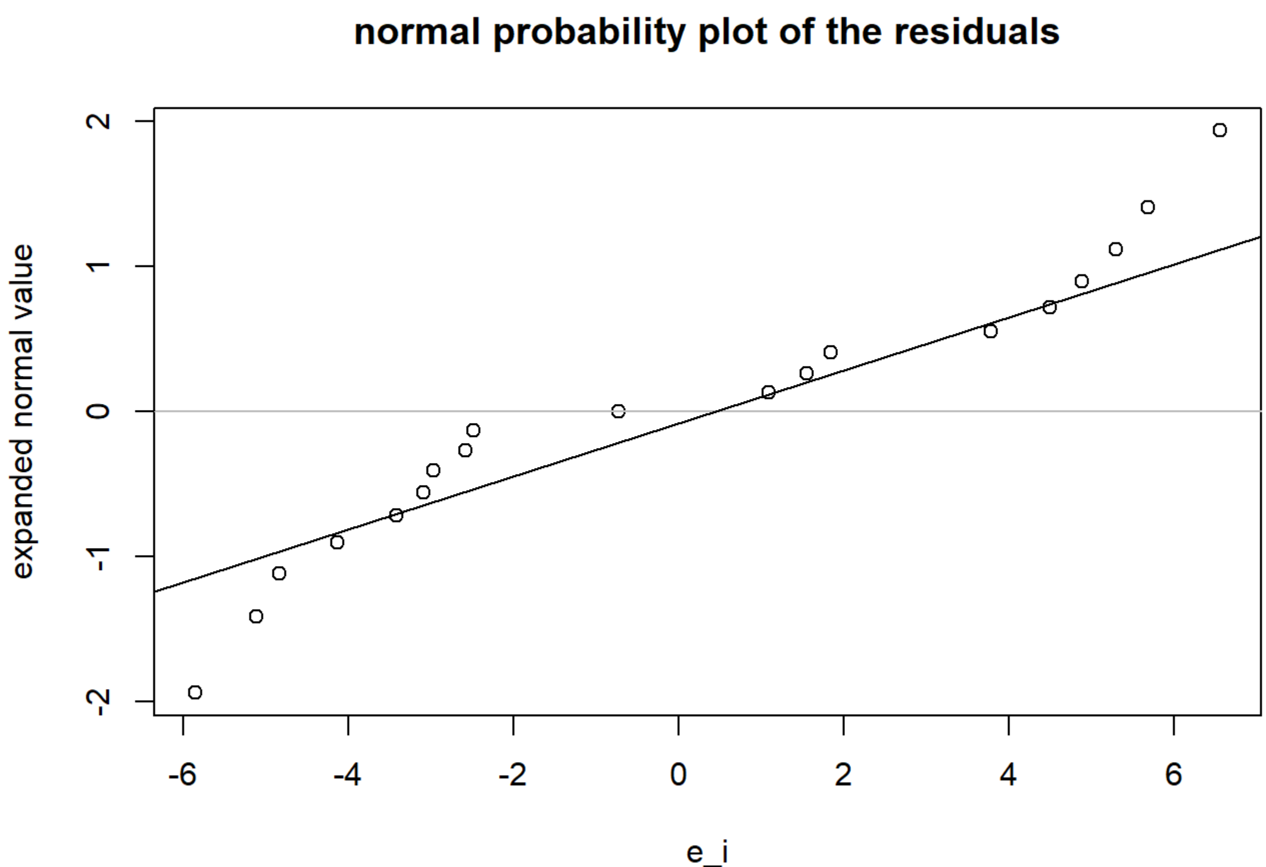
```
plot(model.71$fitted.values, model.71$residuals, main = "residuals (e_i) versus fitted values (ŷ
_i)", xlab = "ŷi", ylab = "e_i")
abline(0, 0, col = "gray")
```



This plot does not reveal any serious model inadequacy.

Create normal probability plot of residuals

```
# set datax = TRUE to resemble plot in book; default value is datax = FALSE
# however, y- and x-label won't change iaw datax = TRUE
qqnorm(model.71$residuals, datax = TRUE, main="normal probability plot of the residuals", ylab =
"e_i", xlab = "expanded normal value")
abline(0, 0, col = "gray")
qqline(model.71$residuals, datax = TRUE)
```



The normal probability plot of the residuals indicates that the error distribution has heavier tails than the normal.