Fall 2019 Lab 10

11/14/2019

Contents

Question 8.06. Steroid level.

An endocrinologist was interested in exploring the relationship between the level of a steroid (Y) *and age* (X) *in healthy female subjects whose ages ranged from* 8 *to* 25 *years She collected a sample of* 27 *healthy females in this age range. >*

a. Fit regression model (8.2). Plot the fitted regression function and the data. Does the quadratic regression function appear to be a good fit here? Find R^2.

```
df806 = read.table("CH08PR06.txt", header=FALSE, sep="")
colNames =  c("Y", "X");
colnames(df806) = colNames
```

```
attach(df806)
x1 = X-mean(X);
model806.reg = lm(Y~x1+I(x1^2), data=df806)
summary(model806.reg)
```

```
##
## Call:
## lm(formula = Y ~ x1 + I(x1^2), data = df806)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5463 -2.5369  0.3868  2.1973  5.3020
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21.09416    0.91415  23.075  < 2e-16 ***
## x1           1.13736    0.11546   9.851 6.59e-10 ***
## I(x1^2)     -0.11840    0.02347  -5.045 3.71e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.153 on 24 degrees of freedom
## Multiple R-squared:  0.8143, Adjusted R-squared:  0.7989
## F-statistic: 52.63 on 2 and 24 DF,  p-value: 1.678e-09
```

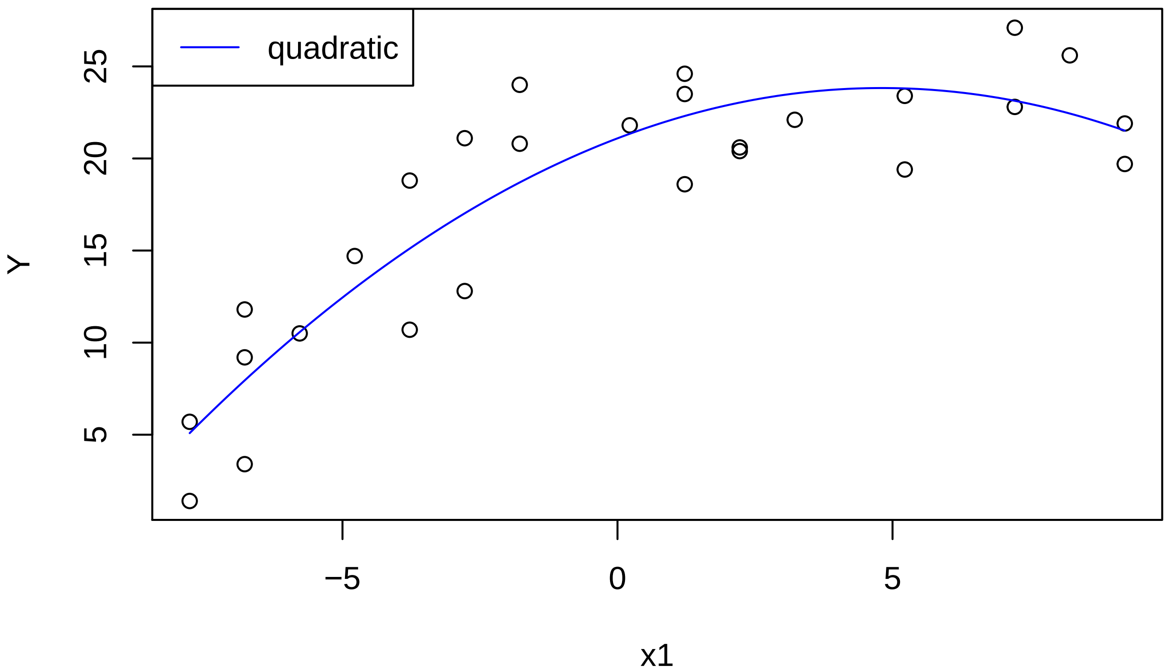
```
coeffs = summary(model806.reg)$coefficients
cat(sprintf("The regression model is Yhat = %f + %f*x1 + %f*x1^2\n", coeffs[1,1], coeffs[2,1], coeffs[3
```

```
## The regression model is Yhat = 21.094160 + 1.137357*x1 + -0.118401*x1^2
```

```
cat(sprintf("R^2: %f\n", summary(model806.reg)$r.squared))
```

```
## R^2: 0.814337
```

```
xnew = data.frame(x1 = seq(from = min(x1), to = max(x1), length.out = 200))
pred = predict.lm(model806.reg, newdata = xnew)
plot (x1, Y)
lines(pred~xnew$x1, col="blue")
legend("topleft", c("quadratic"), col = c("blue"), lty = 1)
```




```
cat("The quadratic regression function appears to be a good fit")
```

```
## The quadratic regression function appears to be a good fit
```

b. Test whether or not there is a regression relation; use $\alpha = .01$. State the alternatives, decision rule, and conclusion. What is the P-value of the test?

```
df_regression = 2
MSR = (anova(model806.reg)$"Sum Sq"[1] + anova(model806.reg)$"Sum Sq"[2])/df_regression
MSR
```

```
## [1] 523.1329
```

```
MSE = anova(model806.reg)$"Mean Sq"[3]
MSE
```

```
## [1] 9.9392
```

```
Fstat = MSR/MSE
Fstat
```

```
## [1] 52.6333
```

```r
alpha = 0.01
df_residual = anova(model806.reg)$Df[3]
df_residual
```

```
## [1] 24
```

```r
Fcritical = qf(1-alpha, df_regression, df_residual)
Fcritical
```

```
## [1] 5.613591
```

```r
cat("Since Fstat > Fcritical, we conclude Ha. not all betak's are 0\n")
```

```
## Since Fstat > Fcritical, we conclude Ha. not all betak's are 0
```

```r
cat("Regression relation is significant\n")
```

```
## Regression relation is significant
```

c. Obtain joint interval estimates for the mean steroid level of females aged 10, 15, and 20 respectively. Use the most efficient simultaneous estimation procedure and a 99 percent family confidence coefficient. Interpret your intervals.

```r
g = 3
alpha = 0.01
n = length(X)
B = qt(1-(alpha/(2*g)), n-3)
B
```

```
## [1] 3.258382
```

```r
W2 = 2*qf(0.99, 3, 24)
W = sqrt(W2)
W
```

```
## [1] 3.071824
```

```r
cat("Since W is less than B, we will use Working-Hoteling to estimate intervals\n")
```

```
## Since W is less than B, we will use Working-Hoteling to estimate intervals
```

```r
xnew = data.frame(x1 = c(10, 15, 20))
pred = predict.lm(model806.reg, newdata=xnew, se.fit=TRUE, interval="confidence", level=0.99)
pred$se.fit
```

```
##        1        2        3
## 1.894503 4.566935 8.500845
```

```r
cat(sprintf("For x1=10: %f <= E{Yh} <= %f\n", pred$fit[1]-W*pred$se.fit[1], pred$fit[1]+W*pred$se.fit[1
```

```
## For x1=10: 14.808030 <= E{Yh} <= 26.447187
```

```r
cat(sprintf("For x1=15: %f <= E{Yh} <= %f\n", pred$fit[2]-W*pred$se.fit[2], pred$fit[2]+W*pred$se.fit[2
```

```
## For x1=15: -2.514582 <= E{Yh} <= 25.543059
```

```r
cat(sprintf("For x1=20: %f <= E{Yh} <= %f\n", pred$fit[3]-W*pred$se.fit[3], pred$fit[3]+W*pred$se.fit[3
```

```
## For x1=20: -29.632290 <= E{Yh} <= 22.593904
```

d. Predict the steroid levels of females aged 15 using a 99 percent prediction interval. Interpret your interval.

```
xnew = data.frame(x1 = c(15))
pred = predict.lm(model806.reg, newdata=xnew, se.fit=TRUE, interval="prediction", level=0.99)
s_pred = sqrt(pred$residual.scale^2 + pred$se.fit^2)
s_pred
```

```
## [1] 5.549423
```

```
alpha = 0.01
n = length(X)
n
```

```
## [1] 27
```

```
tval = qt(1-alpha/2, n-3)
tval
```

```
## [1] 2.79694
```

```
cat(sprintf("For x1=15: Prediction interval is %f <= Yhnew <= %f\n", pred$fit[2], pred$fit[3]))
```

```
## For x1=15: Prediction interval is -4.007162 <= Yhnew <= 27.035640
```

e. Test whether the quadratic term can be dropped from the model; use $\alpha = .01$. State the alternatives, decision rule, and conclusion.

```
se_b11 = 0.02347
tstat = -0.1184/0.02347
tstat
```

```
## [1] -5.044738
```

```
tcritical = qt(0.995, 24)
tcritical
```

```
## [1] 2.79694
```

```
cat(sprintf("Since abs(tstat) > tcritical, we conclude Ha. The quadratic term is significant\n"))
```

```
## Since abs(tstat) > tcritical, we conclude Ha. The quadratic term is significant
```

f. Express the fitted regression function obtained in part (a) in terms of the original variable X.

```
b0_prime = summary(model806.reg)$coefficients[1,1] - summary(model806.reg)$coefficients[2,1]*mean(X) + s
b0_prime
```

```
## [1] -26.32541
```

```
b1_prime = summary(model806.reg)$coefficients[2,1] - 2*mean(X)*summary(model806.reg)$coefficients[3,1]
b1_prime
```

```
## [1] 4.873574
```

```
b11_prime = summary(model806.reg)$coefficients[3,1]
b11_prime
```

```
## [1] -0.1184012
```

```
cat(sprintf("The regression function in terms of original X is %f + %f*X + %f*X^2\n", b0_prime, b1_prim
```

```
## The regression function in terms of original X is -26.325413 + 4.873574*X + -0.118401*X^2
```

Question 9.12. Reference to Market share data set in Appendix C.3 and Problem 8.42.

a. Using only first-order terms for predictor variables, find the three best subset regression models according to the SBC_p criterion.

```
ms.df <- read.table("APPENC03.txt", header=FALSE, col.names = c("X1", "X2", "X3", "X4", "X5", "X6", "X7
ms.df$X8 = factor(ms.df$X8, ordered=FALSE)
#relevel the factor with reference as year=2000
ms.df$X8 = relevel(ms.df$X8, ref="2000")
#removing month (X7), as this variable is not of interest
```

```
attach(ms.df)
cat("X2 = market share, X3 = price, X4 = Nielsen, X5 = discount, \n X6 = package promo, X8 = year\n")
```

```
## X2 = market share, X3 = price, X4 = Nielsen, X5 = discount,
##  X6 = package promo, X8 = year
```

```
reg1 = regsubsets(X2 ~ X3 + X4 + X5 + X6 + X8,
  data = ms.df, nbest=3, nvmax = 5)
summary(reg1)
```

```
## Subset selection object
## Call: regsubsets.formula(X2 ~ X3 + X4 + X5 + X6 + X8, data = ms.df,
##     nbest = 3, nvmax = 5)
## 7 Variables  (and intercept)
##        Forced in Forced out
## X3         FALSE      FALSE
## X4         FALSE      FALSE
## X5         FALSE      FALSE
## X6         FALSE      FALSE
## X81999     FALSE      FALSE
## X82001     FALSE      FALSE
## X82002     FALSE      FALSE
## 3 subsets of each size up to 5
## Selection Algorithm: exhaustive
##          X3  X4  X5  X6  X81999 X82001 X82002
## 1  ( 1 ) " " " " "*" " " " "    " "    " "
## 1  ( 2 ) " " " " " " "*" " "    " "    " "
## 1  ( 3 ) "*" " " " " " " " "    " "    " "
## 2  ( 1 ) " " " " "*" "*" " "    " "    " "
## 2  ( 2 ) "*" " " "*" " " " "    " "    " "
## 2  ( 3 ) " " " " "*" " " " "    "*"    " "
## 3  ( 1 ) "*" " " "*" "*" " "    " "    " "
## 3  ( 2 ) " " " " "*" "*" " "    "*"    " "
## 3  ( 3 ) " " " " "*" " " " "    "*"    "*"
```

```
## 4  ( 1 ) "*" " " "*" "*" " "    "*"    " "
## 4  ( 2 ) " " " " "*" "*" " "    "*"    "*"
## 4  ( 3 ) "*" " " "*" "*" "*"    " "    " "
## 5  ( 1 ) "*" " " "*" "*" " "    "*"    "*"
## 5  ( 2 ) "*" " " "*" "*" "*"    "*"    " "
## 5  ( 3 ) "*" "*" "*" "*" " "    "*"    " "
```

```
res.sum = summary(reg1)

res.sum$bic
```

```
##  [1] -28.167234   3.652103   5.864500 -28.134234 -27.908142 -26.904567
##  [7] -29.798641 -27.394384 -26.118160 -28.166564 -27.551247 -26.669704
## [13] -25.651122 -24.712718 -24.617597
```

```
#top 3 indexes corresponding to lowest sbc/bic are 7, 1 and 10
order(res.sum$bic)
```

```
##  [1]  7  1 10  4  5 11  8  6 12  9 13 14 15  2  3
```

```
#The rows corresponding to index 7, 1 and 10 from summary(reg1):
cat("ANSWER\n")
```

```
## ANSWER
```

```
cat("X3(price), X5(discount) and X6(promo) has the lowest SBC value\n")
```

```
## X3(price), X5(discount) and X6(promo) has the lowest SBC value
```

```
cat("X5(discount)\n")
```

```
## X5(discount)
```

```
cat("X3,X5,X6,X8=2001\n")
```

```
## X3,X5,X6,X8=2001
```

b. Is your finding here in agreement with what you found in Problem 8.42(b) and (c)?

```
cat("The regression subset in problem 9.12a provides a good starting point to \n identify the important
```

```
## The regression subset in problem 9.12a provides a good starting point to
##  identify the important predictor variables
```

```
cat("The variables advertising index and year can be dropped based on results of\n there ssr contributi
```

```
## The variables advertising index and year can be dropped based on results of
##  there ssr contributions, and f-stat being less than fcritical
```

```
cat("The quadratic terms corresponding to quantitative variables price improves the model\n")
```

```
## The quadratic terms corresponding to quantitative variables price improves the model
```

Question 9.27. Reference to SENIC data set in Appendix C.1.

The primary objective of the Study on the Efficacy of Nosocomial Infection Control (SENIC Project) was to determine whether infection surveillance and control programs have reduced the rates of nosocomial (hospital-

acquired) infection in United States hospitals. This data set consists of a random sample of 113 hospitals selected from the original 338 hospitals surveyed. Each line of the dataset has an identification number and provides information on 11 variables for a single hospital. The data presented here are for the 1975-76 study period.

The regression model identified as best in Project 9.25 is to be validated by means of the validation data set consisting of cases 1-56.

a. Fit the regression model identified in Project 9.25 as best to the validation data set. Compare the estimated regression coefficients and their estimated standard deviations with those obtained in Project 9.25. Also compare the error mean squares and coefficients of multiple determination. Does the model fitted to the validation data set yield similar estimates as the model fitted to the model-building data set?

```r
senic.df <- read.table("APPENC01.txt", header=FALSE, col.names = c("id", "los", "age", "infection_risk"
senic.df[ ,c('msa', 'region')] <- list(NULL)
```

```r
#use cases 57-113 to build model
model.df = senic.df[57:113,]
attach(model.df)
model = regsubsets(log(los) ~ age + infection_risk + rcr + xray + nbds +adc + numnurses + facilities,
  data = model.df, nbest=3, nvmax = 5)
summary(model)
```

```
## Subset selection object
## Call: regsubsets.formula(log(los) ~ age + infection_risk + rcr + xray +
##     nbds + adc + numnurses + facilities, data = model.df, nbest = 3,
##     nvmax = 5)
## 8 Variables  (and intercept)
##                Forced in Forced out
## age                FALSE      FALSE
## infection_risk     FALSE      FALSE
## rcr                FALSE      FALSE
## xray               FALSE      FALSE
## nbds               FALSE      FALSE
## adc                FALSE      FALSE
## numnurses          FALSE      FALSE
## facilities         FALSE      FALSE
## 3 subsets of each size up to 5
## Selection Algorithm: exhaustive
##          age infection_risk rcr xray nbds adc numnurses facilities
## 1  ( 1 ) " " " "            " " " "  " "  "*" " "       " "
## 1  ( 2 ) " " " "            " " " "  "*"  " " " "       " "
## 1  ( 3 ) " " "*"            " " " "  " "  " " " "       " "
## 2  ( 1 ) " " " "            " " "*"  " "  "*" " "       " "
## 2  ( 2 ) "*" " "            " " " "  " "  "*" " "       " "
## 2  ( 3 ) " " " "            " " "*"  "*"  " " " "       " "
## 3  ( 1 ) "*" " "            " " "*"  " "  "*" " "       " "
## 3  ( 2 ) " " " "            " " "*"  " "  "*" "*"       " "
## 3  ( 3 ) " " " "            " " "*"  "*"  "*" " "       " "
## 4  ( 1 ) "*" " "            " " "*"  " "  "*" "*"       " "
## 4  ( 2 ) "*" " "            " " "*"  "*"  "*" " "       " "
## 4  ( 3 ) "*" " "            "*" "*"  " "  "*" " "       " "
## 5  ( 1 ) "*" " "            "*" "*"  " "  "*" "*"       " "
```

```
## 5  ( 2 ) "*" "*"            " " "*"  " "  "*" "*"       " "
## 5  ( 3 ) "*" " "            " " "*"  "*"  "*" "*"       " "
```

```r
res.sum = summary(model)

res.sum$cp
```

```
##  [1] 16.232900 20.605409 32.428980  7.479045  9.651600 10.924229  3.811204
##  [8]  6.175797  6.766933  3.863841  4.269604  4.656757  4.283919  4.449965
## [15]  4.907424
```

```r
#top 3 indexes corresponding to lowest cp are 7, 10 and 11
order(res.sum$cp)
```

```
##  [1]  7 10 11 13 14 12 15  8  9  4  5  6  1  2  3
```

```r
#we will pick the lowest cp as the best model
#this corresponds to variables X3(age), X6(xray ratio), and X10(adc-average daily census)
cat("Lowest cp corresponds to X3, X6 and X10\n")
```

```
## Lowest cp corresponds to X3, X6 and X10
```

```r
model925 = lm(log10(los)~age+xray+adc, data=model.df)
summary(model925)
```

```
##
## Call:
## lm(formula = log10(los) ~ age + xray + adc, data = model.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.11264 -0.03760  0.01283  0.03365  0.09347
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.104e-01  8.881e-02   6.873 7.22e-09 ***
## age         3.880e-03  1.627e-03   2.385  0.02069 *
## xray        1.175e-03  4.188e-04   2.805  0.00702 **
## adc         2.926e-04  4.558e-05   6.420 3.86e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05526 on 53 degrees of freedom
## Multiple R-squared:  0.5192, Adjusted R-squared:  0.4919
## F-statistic: 19.07 on 3 and 53 DF,  p-value: 1.614e-08
```

```r
val.df = senic.df[1:56,]
attach(val.df)
```

```
## The following objects are masked from model.df:
##
##     adc, age, facilities, id, infection_risk, los, nbds,
##     numnurses, rcr, xray
```

```r
model927 = lm(log10(los)~age+xray+adc, data=val.df)
summary(model927)
```

```
##
## Call:
```

```
## lm(formula = log10(los) ~ age + xray + adc, data = val.df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.135446 -0.045886 -0.003846  0.040176  0.217397
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.189e-01  1.248e-01   4.960 7.92e-06 ***
## age         3.994e-03  2.109e-03   1.894  0.06383 .
## xray        1.522e-03  4.372e-04   3.482  0.00102 **
## adc         1.568e-04  6.216e-05   2.522  0.01476 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06501 on 52 degrees of freedom
## Multiple R-squared:  0.2934, Adjusted R-squared:  0.2526
## F-statistic: 7.196 on 3 and 52 DF,  p-value: 0.0003955
```

```
#comparison table
model925_values = c(summary(model925)$coefficients[1,1], summary(model925)$coefficients[1,2], summary(mo

model927_values = c(summary(model927)$coefficients[1,1], summary(model927)$coefficients[1,2], summary(mo

answer.df = data.frame(model925_values, model927_values)
colnames(answer.df) = c("Model-building", "Validation")
rownames(answer.df) = c("b0", "s_b0", "b3", "s_b3", "b6", "s_b6", "b10", "s_b10", "MSE", "R^2")
answer.df
```

```
##       Model-building   Validation
## b0      6.104339e-01 6.188657e-01
## s_b0    8.881423e-02 1.247663e-01
## b3      3.880097e-03 3.993604e-03
## s_b3    1.626892e-03 2.108864e-03
## b6      1.174787e-03 1.522279e-03
## s_b6    4.188084e-04 4.372440e-04
## b10     2.926124e-04 1.567985e-04
## s_b10   4.557771e-05 6.216391e-05
## MSE     3.053276e-03 4.226859e-03
## R^2     5.191640e-01 2.933660e-01
```

```
cat("ANALYSIS")
```

```
## ANALYSIS
```

```
cat("The coefficient and standard error estimates are close between \n the 2 models for intercept, b3, a
```

```
## The coefficient and standard error estimates are close between
##  the 2 models for intercept, b3, and b6.
##
```

```
cat("The b10 and its standard error estimates are about ~2x off between the 2 models\n")
```

```
## The b10 and its standard error estimates are about ~2x off between the 2 models
```

```
cat("The MSE is 30% higher for validation model data set\n")
```

```
## The MSE is 30% higher for validation model data set
```

```r
cat("The R^2 value is ~40% lower for validation model data set suggesting a poor fit\n")
```

```
## The R^2 value is ~40% lower for validation model data set suggesting a poor fit
```

b. Calculate the mean squared prediction error in (9.20) and compare it to MSE obtained from the model-building data set. Is there evidence of a substantial bias problem in MSE here?

```r
#calculate MSPR
xnew = data.frame(val.df$age, val.df$xray, val.df$adc)
colnames(xnew) = c("age", "xray", "adc")
pred = predict.lm(model925, newdata=xnew)
resid = log10(val.df$los) - pred
MSPR = (sum(resid^2))/56
MSPR
```

```
## [1] 0.004611988
```

```r
cat("The MSPR for validation data set is about 50% higher compared \n to MSE of model building data set
```

```
## The MSPR for validation data set is about 50% higher compared
##  to MSE of model building data set
```

c. Combine the model-building and validation data sets and fit the selected regression model to the combined data. Are the estimated regression coefficients and their estimated standard deviations appreciably different from those for the model-building data set? Should you expect any differences in the estimates? Explain.

```r
attach(senic.df)
```

```
## The following objects are masked from val.df:
##
##     adc, age, facilities, id, infection_risk, los, nbds,
##     numnurses, rcr, xray

## The following objects are masked from model.df:
##
##     adc, age, facilities, id, infection_risk, los, nbds,
##     numnurses, rcr, xray
```

```r
model927c = lm(log10(los)~age+xray+adc, data=senic.df)
summary(model927c)
```

```
##
## Call:
## lm(formula = log10(los) ~ age + xray + adc, data = senic.df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.143590 -0.041768  0.000704  0.029141  0.225327
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.272e-01  7.383e-02   8.495 1.12e-13 ***
## age         3.525e-03  1.287e-03   2.738  0.00722 **
## xray        1.435e-03  2.968e-04   4.835 4.40e-06 ***
```

```
## adc         2.365e-04  3.743e-05   6.318 5.92e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06069 on 109 degrees of freedom
## Multiple R-squared:  0.3999, Adjusted R-squared:  0.3834
## F-statistic: 24.21 on 3 and 109 DF,  p-value: 4.392e-12
```

```
cat("For the combined data set\n")
```

```
## For the combined data set
```

```
cat(sprintf("The regression function is log10Y = %f + %f*X3 + %f*X6 + %f*X10\n", summary(model927c)$coe
summary(model927c)$coefficients[4,1]))
```

```
## The regression function is log10Y = 0.627178 + 0.003525*X3 + 0.001435*X6 + 0.000236*X10
```

```
cat(sprintf("Standard errors for s_b0=%f, s_b3=%f, s_b6=%f, s_b10=%f\n",
            summary(model927c)$coefficients[1,2], summary(model927c)$coefficients[2,2], summary(model92
```

```
## Standard errors for s_b0=0.073826, s_b3=0.001287, s_b6=0.000297, s_b10=0.000037
```

```
cat("ANALYSIS\n")
```

```
## ANALYSIS
```

```
cat("The standard errors for the full data set are lower than the model build data set\n")
```

```
## The standard errors for the full data set are lower than the model build data set
```

```
cat("This may be due to larger data sample and increased certainty in the model\n")
```

```
## This may be due to larger data sample and increased certainty in the model
```

```
cat("The coefficients for the full model (entries 1-113) appear\n to be close to the model building (en
```

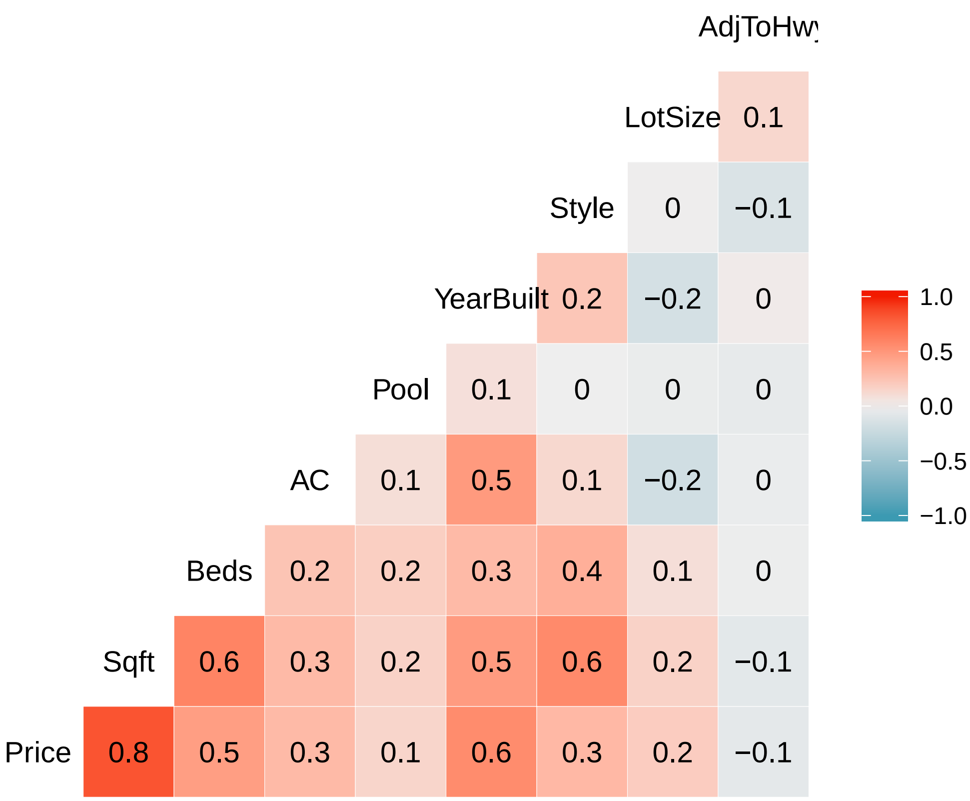
```
## The coefficients for the full model (entries 1-113) appear
##  to be close to the model building (entries 57-113)data set
```

Question 9.31. Refer to Real estate sales data set in Appendix C.7.

Residential sales that occurred during the year 2002 were available from a city in the midwest. Data on 522 arms-length transactions include sales price, style, finished square feet, number of bedrooms, pool, lot size, year built, air conditioning, and whether or not the lot is adjacent to a highway. The city tax assessor was interested in predicting sales price based on the demographic variable information given above. Select a random sample of 300 observations to use in the model-building data set. Develop a best subset model for predicting sales price. Justify your choice of model. Assess your model's ability to predict and discuss its use as a tool for predicting sales price.

```
dts4 <- read.table("APPENC07.txt", header=FALSE)
colnames(dts4) <- c(
  "ID",
  "Price",
  "Sqft",
  "Beds",
  "Baths",
  "AC",
```

```r
  "GarageSize",
  "Pool",
  "YearBuilt",
  "Quality",
  "Style",
  "LotSize",
  "AdjToHwy"
)

# only expected variables

set.seed(123)

dts4 <- dts4[,!(colnames(dts4) %in% c("ID","Quality","GarageSize","Baths"))]

tr <- sample(1:nrow(dts4))

dts4 <- dts4[sample(1:nrow(dts4)), ]
dts4train = dts4[tr[1:300],]
dts4test = dts4[tr[301:522],]

ggcorr(dts4train, label=TRUE)
```




```r
# obvious suspects - square footage, year used, style and bedrooms (usually you bucket year built into

mdl4 = lm(Price~Sqft,dts4train)
summary(mdl4)
```

```
##
```

```
## Call:
## lm(formula = Price ~ Sqft, data = dts4train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -181871  -36457   -6206   22510  387539
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -88665.96   14772.08  -6.002 5.64e-09 ***
## Sqft           161.04       6.32  25.481  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 76210 on 298 degrees of freedom
## Multiple R-squared:  0.6854, Adjusted R-squared:  0.6844
## F-statistic: 649.3 on 1 and 298 DF,  p-value: < 2.2e-16
```

```r
mdl4b = lm(Price~Sqft + YearBuilt,dts4train)
summary(mdl4b)
```

```
##
## Call:
## lm(formula = Price ~ Sqft + YearBuilt, data = dts4train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -171153  -32446   -6132   24646  378727
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.202e+06  5.111e+05  -6.264 1.31e-09 ***
## Sqft         1.416e+02  6.765e+00  20.935  < 2e-16 ***
## YearBuilt    1.606e+03  2.635e+02   6.093 3.43e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 71970 on 297 degrees of freedom
## Multiple R-squared:  0.7204, Adjusted R-squared:  0.7185
## F-statistic: 382.6 on 2 and 297 DF,  p-value: < 2.2e-16
```

```r
mdl4c = lm(Price~Sqft + YearBuilt + factor(Style),dts4train)
summary(mdl4c)
```

```
##
## Call:
## lm(formula = Price ~ Sqft + YearBuilt + factor(Style), data = dts4train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -167704  -32059   -4943   24772  352985
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -3.636e+06  5.436e+05  -6.689 1.16e-10 ***
```

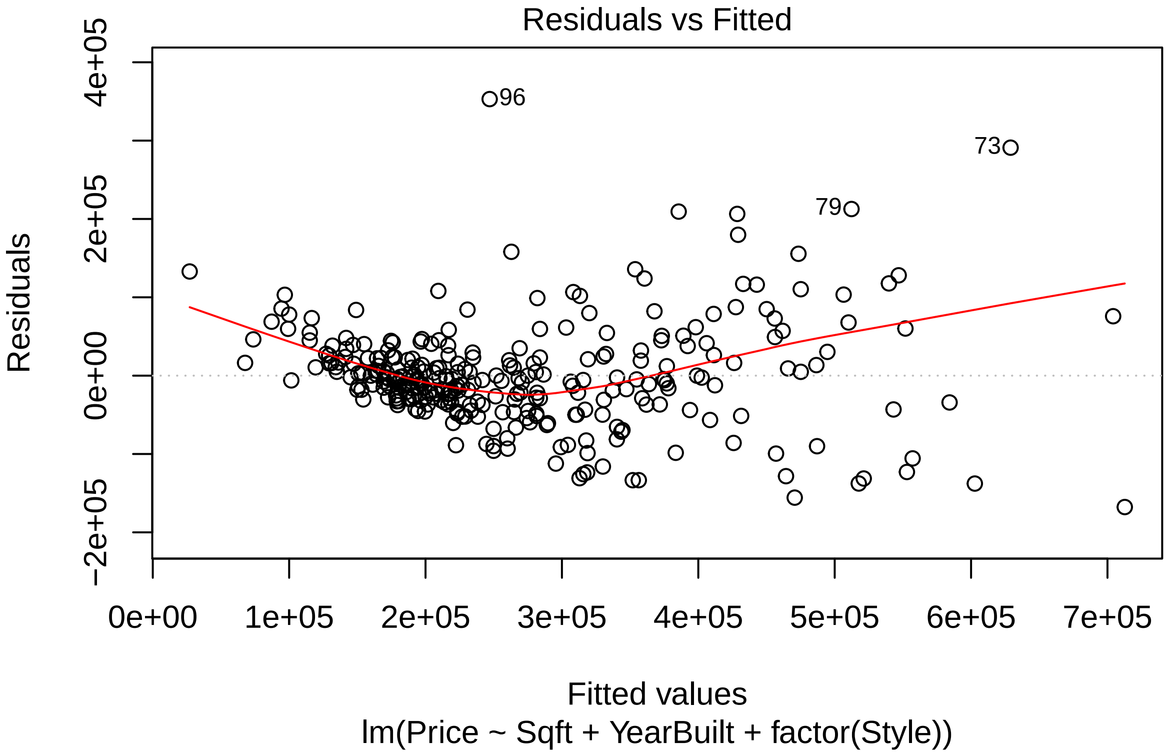
```
## Sqft             1.661e+02  7.964e+00  20.853  < 2e-16 ***
## YearBuilt        1.815e+03  2.808e+02   6.465 4.31e-10 ***
## factor(Style)2  -3.324e+04  1.366e+04  -2.434  0.01553 *
## factor(Style)3  -3.604e+04  1.269e+04  -2.840  0.00482 **
## factor(Style)4   1.156e+04  2.470e+04   0.468  0.64017
## factor(Style)5  -3.374e+04  2.110e+04  -1.599  0.11087
## factor(Style)6  -4.336e+04  2.415e+04  -1.796  0.07356 .
## factor(Style)7  -8.347e+04  1.217e+04  -6.856 4.29e-11 ***
## factor(Style)9  -1.152e+04  6.781e+04  -0.170  0.86525
## factor(Style)11 -9.577e+04  6.770e+04  -1.415  0.15821
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 67150 on 289 degrees of freedom
## Multiple R-squared:  0.7631, Adjusted R-squared:  0.755
## F-statistic: 93.12 on 10 and 289 DF,  p-value: < 2.2e-16
```

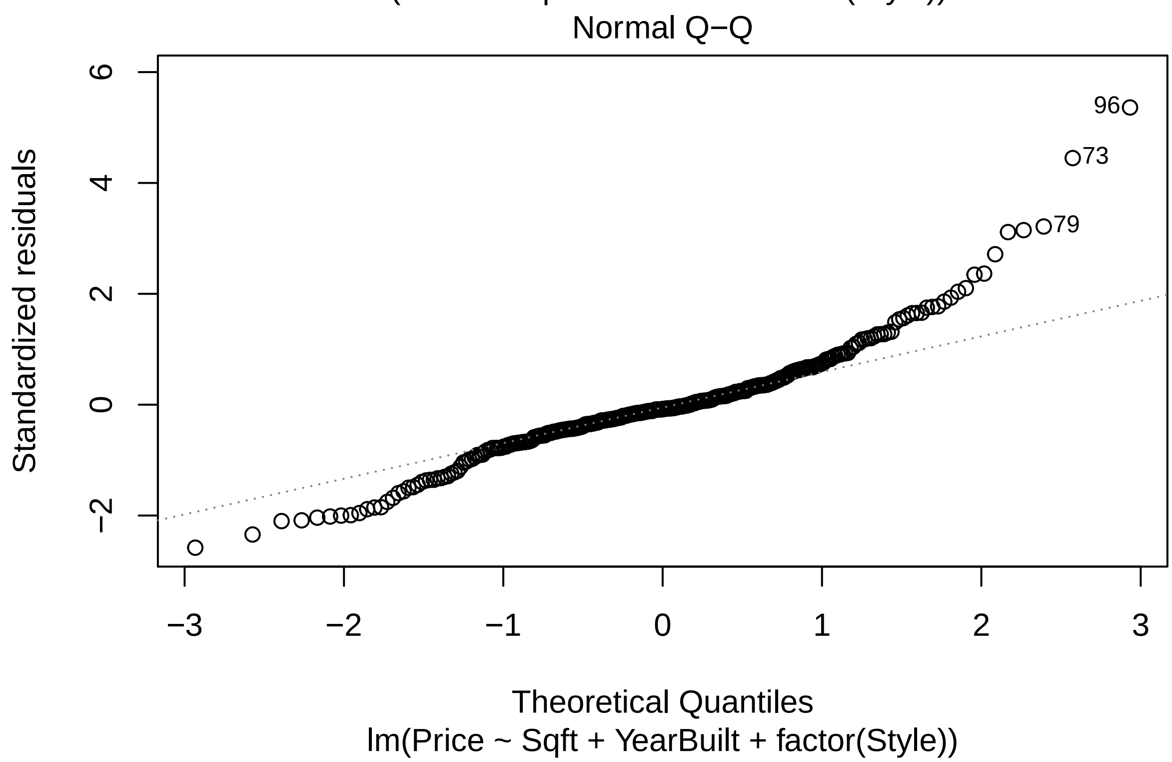
```r
# Increasing model is not beneficial, almost no increase in R-sq
summary(lm(Price~Sqft + YearBuilt + factor(Style) + AC + LotSize + Pool + Beds,dts4train))
```

```
##
## Call:
## lm(formula = Price ~ Sqft + YearBuilt + factor(Style) + AC +
##     LotSize + Pool + Beds, data = dts4train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -191773  -32346   -2665   26927  306433
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -4.131e+06  5.763e+05  -7.169 6.52e-12 ***
## Sqft             1.624e+02  9.425e+00  17.233  < 2e-16 ***
## YearBuilt        2.067e+03  2.976e+02   6.944 2.58e-11 ***
## factor(Style)2  -3.023e+04  1.417e+04  -2.134  0.03373 *
## factor(Style)3  -3.237e+04  1.268e+04  -2.553  0.01119 *
## factor(Style)4   1.210e+04  2.443e+04   0.495  0.62068
## factor(Style)5  -2.841e+04  2.109e+04  -1.347  0.17903
## factor(Style)6  -4.190e+04  2.402e+04  -1.745  0.08214 .
## factor(Style)7  -7.634e+04  1.229e+04  -6.214 1.83e-09 ***
## factor(Style)9  -1.442e+04  6.706e+04  -0.215  0.82993
## factor(Style)11 -1.112e+05  6.758e+04  -1.646  0.10095
## AC              -3.314e+03  1.139e+04  -0.291  0.77126
## LotSize          1.040e+00  3.366e-01   3.091  0.00219 **
## Pool             5.493e+03  1.671e+04   0.329  0.74266
## Beds            -4.998e+03  4.854e+03  -1.030  0.30404
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 66330 on 285 degrees of freedom
## Multiple R-squared:  0.7721, Adjusted R-squared:  0.7609
## F-statistic: 68.97 on 14 and 285 DF,  p-value: < 2.2e-16
```

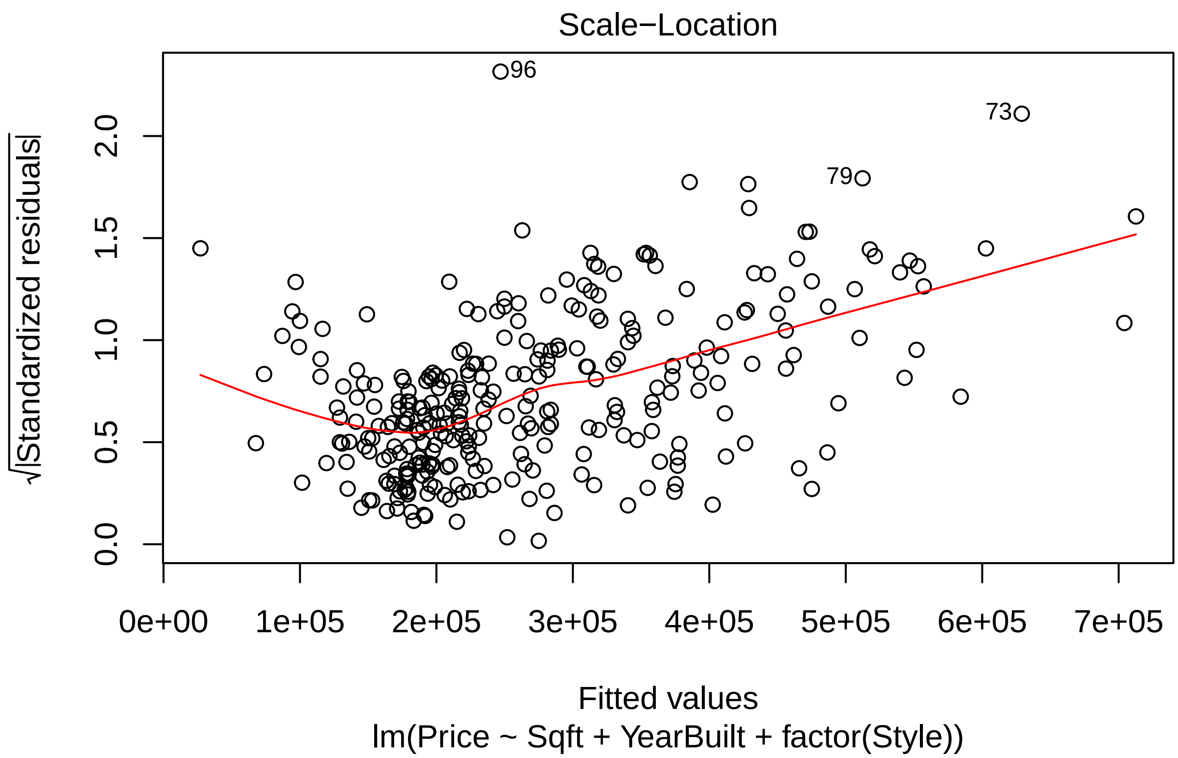
```r
#most likely the best model is mdl4c
plot(mdl4c)
```

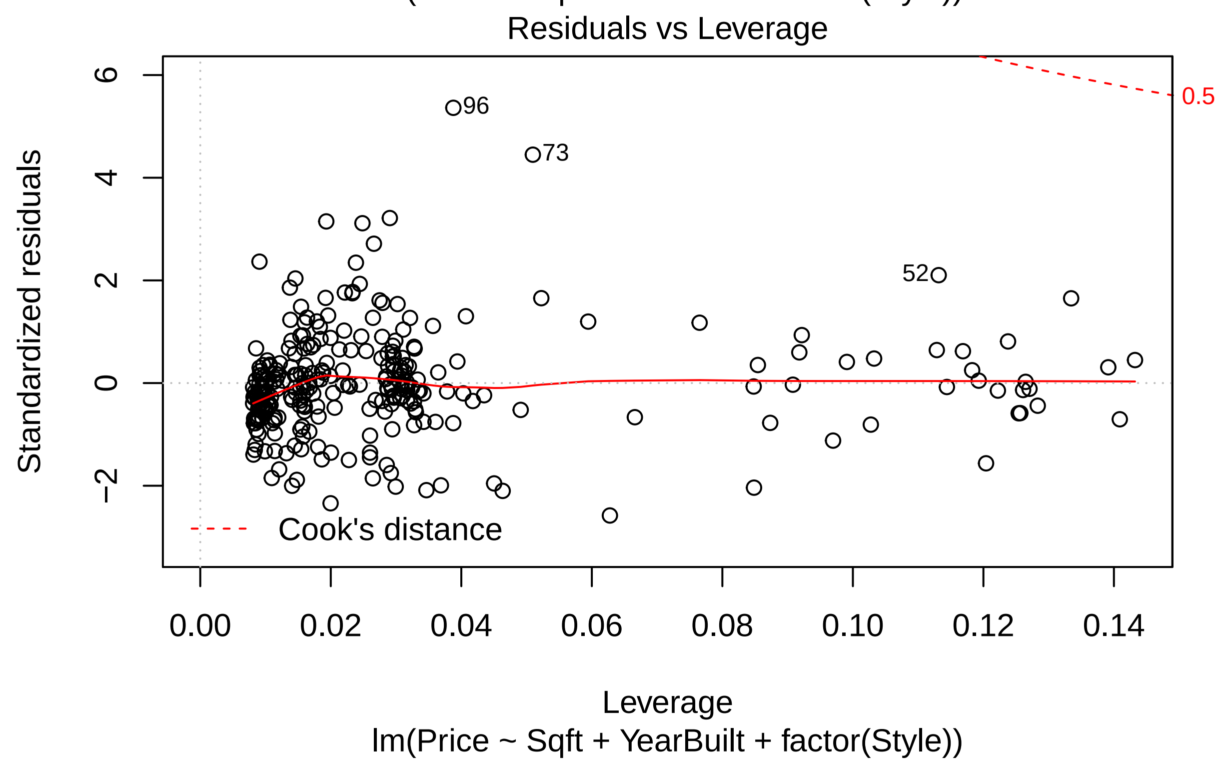
```
## Warning: not plotting observations with leverage one:
##   76, 261
```



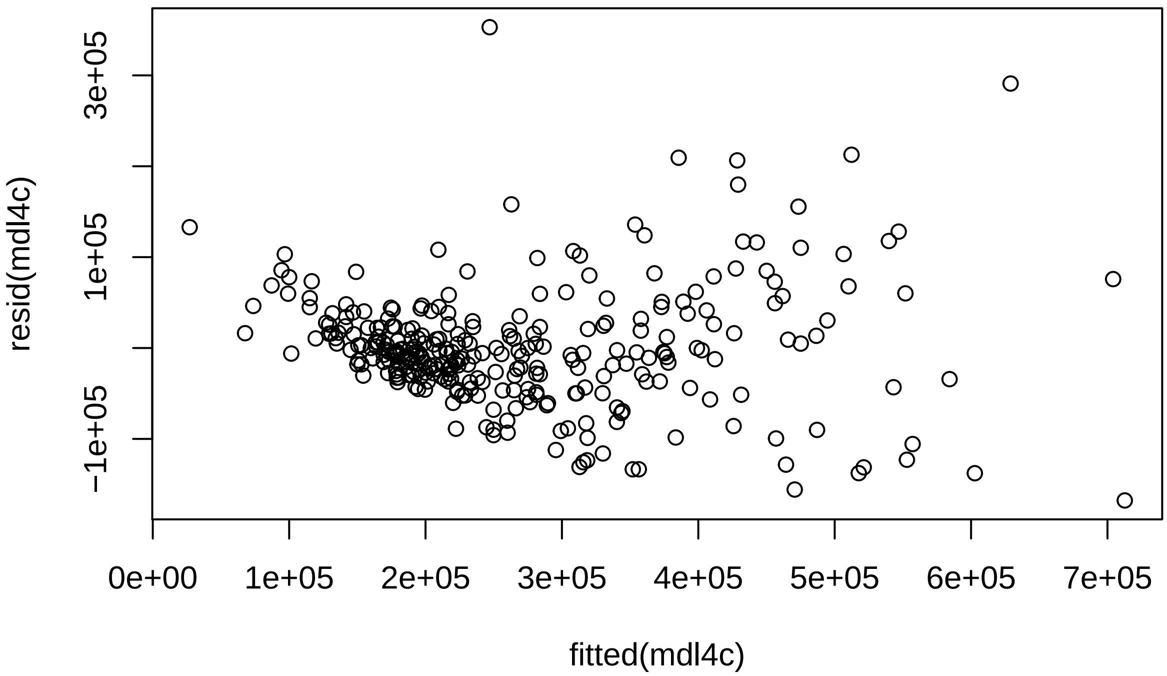



```
## Warning: not plotting observations with leverage one:
##   76, 261
```

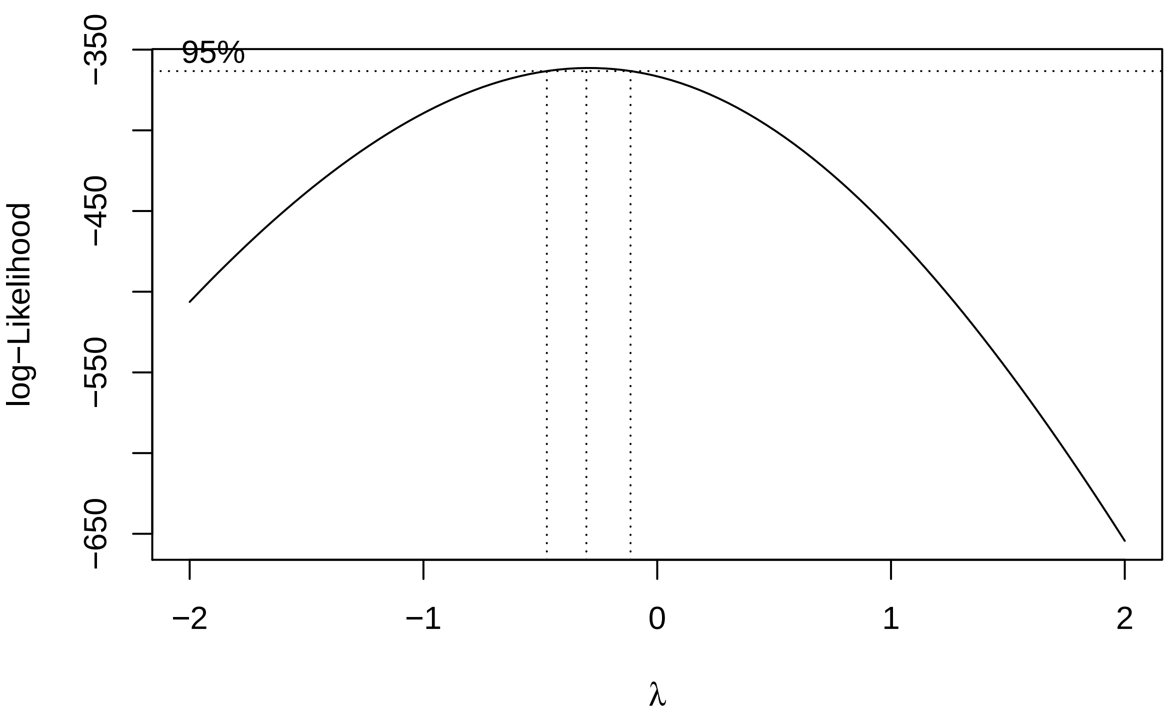





```
plot(fitted(mdl4c),resid(mdl4c))
```




```r
boxcox(mdl4c)
```



The model perform well for predicted home prices that are on a lower side. There are many outliers for the larger size houses.