Statistics for Engineers, 4C3/6C3

Assignment 6

Kevin Dunn, dunnkg@mcmaster.ca **Due date: 09 March 2011**

Assignment objectives

- To become more comfortable using R to fit, interpret and manipulate least squares models.
- The questions in this assignment are typical of the exploratory/learning type questions that will be in the take-home midterm.

Question 1 [1.5]

Use the mature cheddar cheese data set for this question.

1. Choose any x-variable, either `Acetic` acid concentration (already log-transformed), `H2S` concentration (already log-transformed), or `Lactic` acid concentration (in original units) and use this to predict the `Taste` variable in the data set. The `Taste` is a subjective measurement, presumably measured by a panel of tasters.

   Prove that you get the same linear model coefficients, R^2, S_E and confidence intervals whether or not you first mean center the x and y variables.

2. What is the level of correlation between each of the x-variables. Also show a scatterplot matrix to learn what this level of correlation looks like visually.
   - Report your correlations as a 3×3 matrix, where there should be 1.0's on the diagonal, and values between -1 and $+1$ on the off-diagonals.

3. Build a linear regression that uses all three x-variables to predict y.
   - Report the slope coefficient and confidence interval for each x-variable
   - Report the model's standard error. Has it decreased from the model in part 1?
   - Report the model's R^2 value. Has it decreased?

Solution

1. We used the acetic acid variable as x and derived the following two models to predict taste, y:
   - No mean centering of x and y: $y = -61.5 + 15.65x$
   - With mean centering of x and y: $y = 0 + 15.65x$

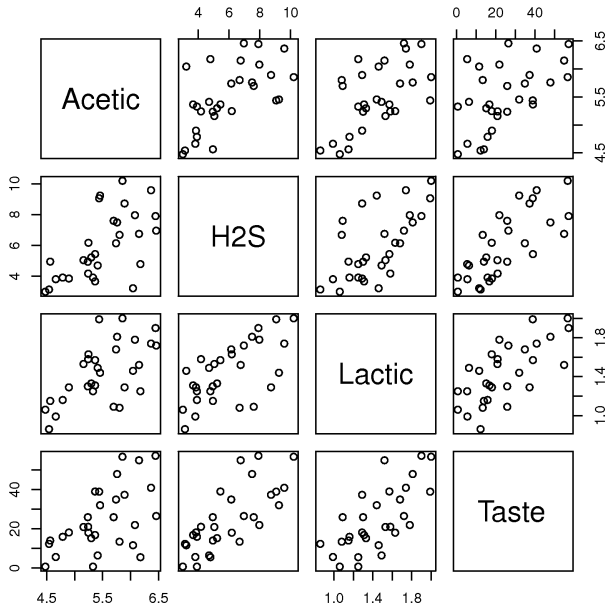
   These results were found from *both* models:
   - Residual standard error, S_E = 13.8 on 28 degrees of freedom
   - Multiple R-squared, R^2 = 0.30

- Confidence interval for the slope, b_a was: $6.4 \leq b_A \leq 24.9$.

Please see the R code at the end of this question.

If you had used x = `H2S`, then $S_E = 10.8$ and if used x = `Lactic`, then $S_E = 11.8$.

2. The visual level of correlation is shown in the first 3×3 plots below, while the relationship of each x to y is shown in the last row and column:



The numeric values for the correlation between the x-variables are:

$$\begin{bmatrix} 1.0 & 0.618 & 0.604 \\ 0.618 & 1.0 & 0.644 \\ 0.604 & 0.644 & 1.0 \end{bmatrix}$$

There is about a 60% correlation between each of the x-variables in this model, and in each case the correlation is positive.

3. A combined linear regression model is $y = -28.9 + 0.31x_A + 3.92x_S + 19.7x_L$ where x_A is the log of the acetic acid concentration, x_S is the log of the hydrogen sulphide concentration and x_L is the lactic acid concentration in the cheese. The confidence intervals for each coefficient are:

   - $-8.9 \leq b_A \leq 9.4$
   - $1.4 \leq b_S \leq 6.5$
   - $1.9 \leq b_A \leq 37$

   The R^2 value is 0.65 in the MLR, compared to the value of 0.30 in the single variable regression. The R^2 value will always decrease when adding a new variable to the model, even if that variable has little value to the regression model (yet another caution related to R^2).

   The MLR standard error is 10.13 on 26 degrees of freedom, a decrease of about 3 units from the individual regression in part 1; a small decrease given the y-variable's range of about 50 units.

Since each x-variable is about 60% correlated with the others, we can loosely interpret this by inferring that *either* `lactic`, *or* `acetic` *or* `H2S` could have been used in a single-variable regression. In fact, if you compare S_E values for the single-variable regressions, (13.8, 10.8 and 11.8), to the combined regression S_E of 10.13, there isn't much of a reduction in the MLR's standard error.

This interpretation can be quite profitable: it means that we get by with one only one x-variable to make a reasonable prediction of taste in the future, however, the other two measurements must be consistent. In other words we can pick lactic acid as our predictor of taste (it might be the cheapest of the 3 to measure). But a new cheese with high lactic acid, must also have high levels of `H2S` and `acetic` acid for this prediction to work. If those two, now unmeasured variables, had low levels, then the predicted taste may not be an accurate reflection of the true cheese's taste! We say "the correlation structure has been broken" for that new observation.

Other, advanced explanations:

Highly correlated x-variables are problematic in least squares, because the confidence intervals and slope coefficients are not independent anymore. This leads to the problem we see above: the acetic acid's effect is shown to be insignificant in the MLR, yet it was significant in the single-variable regression! Which model do we believe?

This resolution to this problem is simple: look at the raw data and see how correlated each of the x-variables are with each other. One of the shortcomings of least squares is that we must invert $\mathbf{X}'\mathbf{X}$. For highly correlated variables this matrix is unstable in that small changes in the data lead to large changes in the inversion. What we need is a method that handles correlation.

One quick, simple, but suboptimal way to deal with high correlation is to create a new variable, $x_{\text{avg}} = 0.33x_A + 0.33x_S + 0.33x_L$ that blends the 3 separate pieces of information into an average. Averages are always less noisy than the separate variables the make up the average. Then use this average in a single-variable regression. See the code below for an example.

```
cheese <- read.csv('http://datasets.connectmv.com/file/cheddar-cheese.csv')
summary(cheese)

# Proving that mean-centering has no effect on model parameters
x <- cheese$Acetic
y <- cheese$Taste
summary(lm(y ~ x))
confint(lm(y ~ x))

x.mc <- x - mean(x)
y.mc <- y - mean(y)
summary(lm(y.mc ~  x.mc))
confint(lm(y.mc ~  x.mc ))

# Correlation amount in the X's.  Also plot it
cor(cheese[,2:5])
bitmap('../images/cheese-data-correlation.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(cheese[,2:5])
dev.off()

# Linear regression that uses all three X's
model <- lm(cheese$Taste ~ cheese$Acetic + cheese$H2S + cheese$Lactic)
summary(model)
```

```
confint(model)

# Use an "average" x
x.avg <- 1/3*cheese$Acetic + 1/3*cheese$H2S + 1/3*cheese$Lactic
model.avg <- lm(cheese$Taste ~ x.avg)
summary(model.avg)
confint(model.avg)
```

Question 2 [2.5]

In this question we will revisit the bioreactor yield data set and fit a linear model with all x-variables to predict the yield.

1. Provide the interpretation for each coefficient in the model, and also comment on its confidence interval when interpreting it.
2. Compare the 3 slope coefficient values you just calculated, to those from the last assignment:
   - $\hat{y} = 102.5 - 0.69T$, where T is tank temperature
   - $\hat{y} = -20.3 + 0.016S$, where S is impeller speed
   - $\hat{y} = 54.9 - 16.7B$, where B is 1 if baffles are present and $B = 0$ with no baffles

   Explain why your coefficients do not match.
3. Are the residuals from the multiple linear regression model normally distributed?
4. In this part we are investigating the variance-covariance matrices used to calculate the linear model.
   (a) First center the x-variables and the y-variable that you used in the model.

   Note: feel free to use MATLAB, or any other tool to answer this question. If you are using R, then you will benefit from this page in the R tutorial. Also, read the help for the `model.matrix(...)` function to get the $\mathbf{X}$-matrix. Then read the help for the `sweep(...)` function, or more simply use the `scale(...)` function to do the mean-centering.

   (b) Show your calculated $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}^T\mathbf{y}$ variance-covariance matrices from the centered data.

   (c) Explain why the interpretation of covariances in $\mathbf{X}^T\mathbf{y}$ match the results from the full MLR model you calculated in part 1 of this question.

   (d) Calculate $\mathbf{b} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$ and show that it agrees with the estimates that R calculated (even though R fits an intercept term, while your $\mathbf{b}$ does not).
5. What would be the predicted yield for an experiment run without baffles, at 4000 rpm impeller speed, run at a reactor temperature of 90 °C?

Solution

1. The full linear model that relates bioreactor yield to 3 factors is:

$$y = 52.5 - 0.47x_T + 0.0087x_S - 9.1x_B$$

where x_T is the temperature value in °C, x_S is the speed in RPM and x_B is a coded variable, 0=no baffles and 1=with baffles.

- *Temperature effect*: $-0.74 < \beta_T < -0.21$, with $b_T = -0.47$ indicates that increasing the temperature by 1 °C will decrease the yield on average by 0.47 units, holding the speed and baffle effects constant. The confidence interval does not span zero, indicating this coefficient is significant. An ad-hoc way I sometimes use to gather the effect of a variables is to ask what is the effect over the entire range of temperature, ~ 40°C:
  - $\Delta y = -0.74 \times 40 = -29.6$ % decrease in yield
  - $\Delta y = -0.21 \times 40 = -8.4$ % decrease in yield

  A tighter confidence interval will have these two values even closer, but given the range of the y's in the data cover about 35% units, this temperature effect is important, and will have a noticeable effect at either end of the confidence interval.
- *Speed effect*: $0.34 < \beta_S < 17.0822$ with $b_S = 8.7$ per 1000 RPM: indicates that increase the RPM by 1000 units will increase the yield by about 8.7 units, holding the other factors constant. While the confidence interval does not span zero, it is quite wide.
- *Baffles effect*: $-15.9 < \beta_B < -2.29$ with $b_B = -9.1$ indicates the presence of baffles decreases yield on average by 9.1 units, holding the temperature and speed effects constant. The confidence interval does not span zero, indicating this coefficient is significant. It is an important effect to consider when wanting to change yield.

2. In the last assignment we considered the separate effects:
   - $\hat{y} = 102.5 - 0.69T$, where T is tank temperature
   - $\hat{y} = -20.3 + 0.016S$, where S is impeller speed
   - $\hat{y} = 54.9 - 16.7B$, where B is 1 if baffles are present and $B = 0$ with no baffles

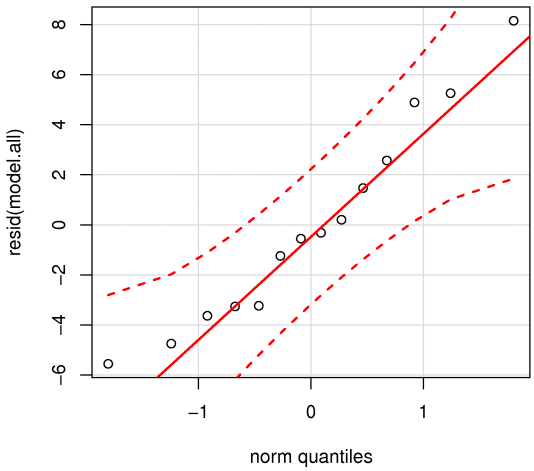
   The signs of the coefficients between MLR and OLS (ordinary least squares) are in agreement, but not the magnitudes. The problem is that when building the single-variable regression model we place all the other effects into the residuals. For example, a model considering only temperature, but ignoring speed and baffles is essentially saying:

$$y = b_0 + b_T x_T + e$$
$$y = b_0 + b_T x_T + (e' + b'_S x_S + b'_B x_B)$$

   i.e. we are lumping the effect of speed and baffles which we have omitted from the model, into the residuals, and we should see structure in our residuals due to these omitted effects.

   Since the objective function for least squares is to minimize the sum of squares of the residuals, the effect of speed and baffles can be "smeared" into the coefficient we are estimating, the b_T coefficient, and this is even more so when any of the x-variables are correlated with each other.

3. The residuals from the multiple linear regression model are normally distributed. This can be verified in the q-q plot below:



4. The $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}^T\mathbf{y}$ variance-covariance matrices from the centered data, where the order of the variables is: temperature, speed and then baffles:

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} 1911 & -9079 & 36.43 \\ -9079 & 1844000 & -1029 \\ 36.43 & -1029 & 3.43 \end{bmatrix}$$

$$\mathbf{X}^T\mathbf{y} = \begin{bmatrix} -1310 \\ 29690 \\ -57.3 \end{bmatrix}$$

The covariances show a negative relationship between temperature and yield (-1310), a positive relationship between speed and yield (29690) and a negative relationship between baffles and yield (-57.3). Unfortunately, covariances are unit-dependent, so we cannot interpret the relative magnitude of these values: i.e. it would be wrong to say that speed has a greater effect than temperature because its covariance magnitude is larger. If we had two x-variables with the same units, then we could compare them fairly, but not in this case where all 3 units are different.

We can calculate

$$\mathbf{b} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} = \begin{bmatrix} -0.471 \\ 0.0087 \\ -9.1 \end{bmatrix}$$

which agrees with the estimates that R calculated (even though R fits an intercept term, while we do not estimate an intercept).

5. The predicted yield yield for an experiment run without baffles, at 4000 rpm impeller speed, run at a reactor temperature of 90 °C would be 45%:

$$\hat{y} = 52.5 - 0.47x_T + 0.0087x_S - 9.1x_B$$
$$\hat{y} = 52.5 - 0.47(90) + 0.0087(4000) - 9.1(0) = \mathbf{45.0}$$

All the code for this question is given below:

```
bio <- read.csv('http://datasets.connectmv.com/file/bioreactor-yields.csv')
summary(bio)

# Temperature-Yield model
model.temp <- lm(bio$yield ~ bio$temperature)
summary(model.temp)

# Impeller speed-Yield model
model.speed <- lm(bio$yield ~ bio$speed)
summary(model.speed)

# Baffles-Yield model
model.baffles <- lm(bio$yield ~ bio$baffles)
summary(model.baffles)

# Model of everything
model.all <- lm(bio$yield ~ bio$temperature + bio$speed + bio$baffles)
summary(model.all)
confint(model.all)

# Residuals normally distributed? Yes
library(car)
bitmap('../images/bioreactor-residuals-qq-plot.png', type="png256",
         width=6, height=6, res=300, pointsize=14)
qqPlot(resid(model.all))
dev.off()

# Calculate X matrix and y vector
data <- model.matrix(model.all)
X <- data[,2:4]
y <- matrix(bio$yield)

# Center the data first
X <- scale(X, scale=FALSE)
y <- scale(y, scale=FALSE)

# Now calculate variance-covariance matrices
XTy <- t(X) %*% y
XTX <- t(X) %*% X
b <- solve(XTX) %*% XTy
# b agrees with R's calculation from ``model.all``
```

Question 3 [3]

In this question we will use the LDPE data which is data from a high-fidelity simulation of a low-density polyethylene reactor. LDPE reactors are very long, thin tubes. In this particular case the tube is divided in 2 zones, since the feed enters at the start of the tube, and some point further down the tube (start of the second zone). There is a temperature profile along the tube, with a certain maximum temperature somewhere along the length. The maximum temperature in zone 1, `Tmax1` is reached some fraction `z1` along the length; similarly in zone 2 with the `Tmax2` and `z2` variables.

We will build a linear model to predict the `SCB` variable, the short chain branching (per 1000 carbon atoms)

which is an important quality variable for this product. Note that the last 4 rows of data are known to be from abnormal process operation, when the process started to experience a problem. However, we will pretend we didn't know that when building the model, so keep them in for now.

1. Use only the following subset of x-variables: `Tmax1`, `Tmax2`, `z1` and `z2` and the y variable = `SCB`. Show the relationship between these 5 variables in a scatter plot matrix.

   Use this code to get you started (make sure you understand what it is doing):

   ```
   LDPE <- read.csv('http://datasets.connectmv.com/file/ldpe.csv')
   subdata <- data.frame(cbind(LDPE$Tmax1, LDPE$Tmax2, LDPE$z1, LDPE$z2, LDPE$SCB))
   colnames(subdata) <- c("Tmax1", "Tmax2", "z1", "z2", "SCB")
   ```

   Using bullet points, describe the nature of relationships between the 5 variables, and particularly the relationship to the y-variable.

2. Let's start with a linear model between `z2` and `SCB`. We will call this the `z2` model. Let's examine its residuals:

   (a) Are the residuals normally distributed?

   (b) What is the standard error of this model?

   (c) Are there any time-based trends in the residuals (the rows in the data are already in time-order)?

   (d) Use any other relevant plots of the predicted values, the residuals, the x-variable, as described in class, and diagnose the problem with this linear model.

   (e) What can be done to fix the problem? (You don't need to implement the fix yet).

3. Show a plot of the hat-values (leverage) from the `z2` model.

   (a) Add suitable horizontal cut-off lines to your hat-value plot.

   (b) Identify on your plot the observations that have large leverage on the model

   (c) Remove the high-leverage outliers and refit the model. Call this the `z2.updated` model

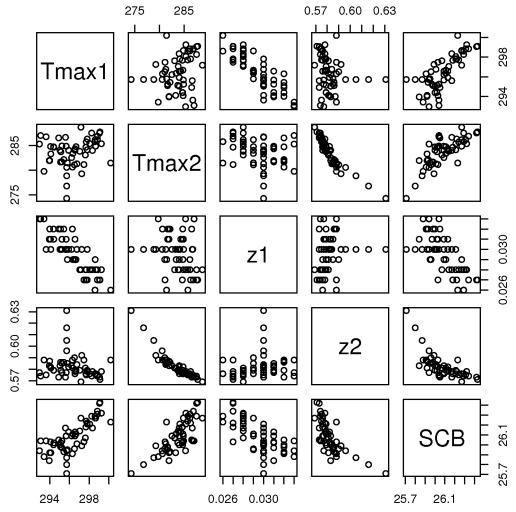
   (d) Show the updated hat-values and verify whether the problem has mostly gone away

   Note: see the R tutorial on how to rebuild a model by removing points

4. Use the `influenceIndexPlot(...)` function in the `car` library on both the `z2` model and the `z2.updated` model. Interpret what each plot is showing for the two models. You may ignore the *Bonferroni p-values* subplot.

Solution

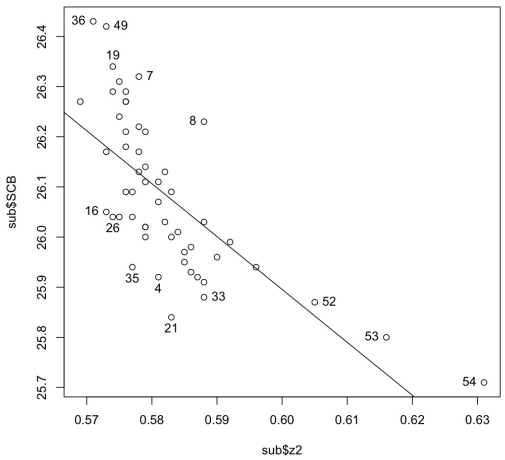
1. A scatter plot matrix of the 5 variables is



- `Tmax1` and `z1` show a strongish negative correlation
- `Tmax1` and `SCB` show a strong positive correlation
- `Tmax2` and `z2` have a really strong negative correlation, and the 4 outliers are very clearly revealed in almost any plot with `z2`
- `z1` and `SCB` have a negative correlation
- `Tmax2` and `SCB` have a negative correlation
- Very little relationship appears between `Tmax1` and `Tmax2`, which is expected, given how/where these 2 data variables are recorded.
- Similarly for `Tmax2` and `z2`.

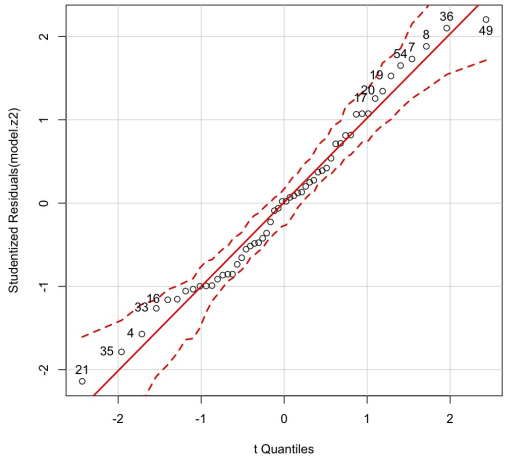
2. A linear model between `z2` and `SCB`: $\widehat{\text{SCB}} = 32.23 - 10.6 z_2$

   First start with a plot of the raw data with this regression line superimposed:



which helps when we look at the q-q plot of the Studentized residuals to see the positive and the

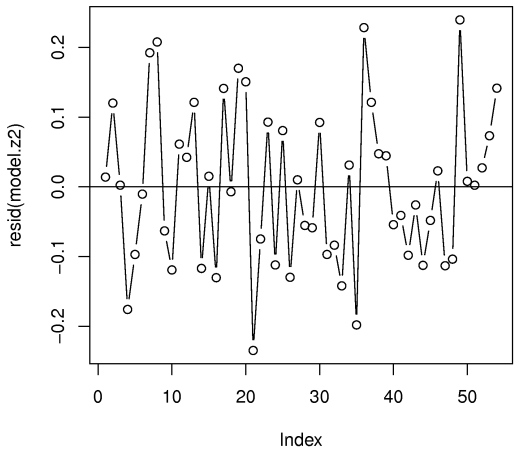
negative residuals:



(a) We notice there is no strong evidence of non-normality, however, we can see a trend in the tails on both sides (there are large positive residuals and large negative residuals). The identified points in the two plots help understand which points affect the residual tails.

(b) This model's standard error is $S_E = 0.114$, which should be compared to the range of the y-axis, 0.70 units, to get an idea whether this is large or small, so about 15% of the range. Given that a conservative estimate of the prediction interval is $\pm 2S_E$, or a total range of $4S_E$, this is quite large.

(c) The residuals in time-order

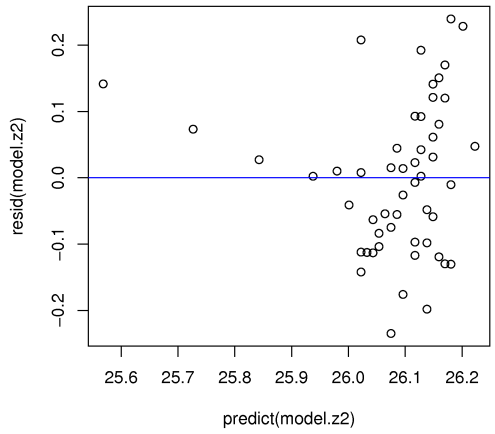


Show no consistent structure, however we do see the short upward trend in the last 4 points. The autocorrelation function (not shown here), shows there is no autocorrelation, i.e. the residuals appear independent.

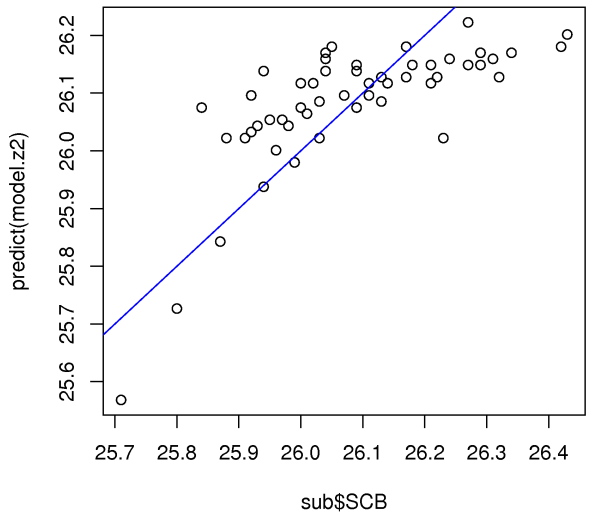
(d) Three plots that do show a problem with the linear model:

- *Predictions vs residuals*: definite structure in the residuals. We expect to see no structure,

but a definite trend, formed by the 4 points is noticeable, as well as a negative correlation at high predicted SCB.

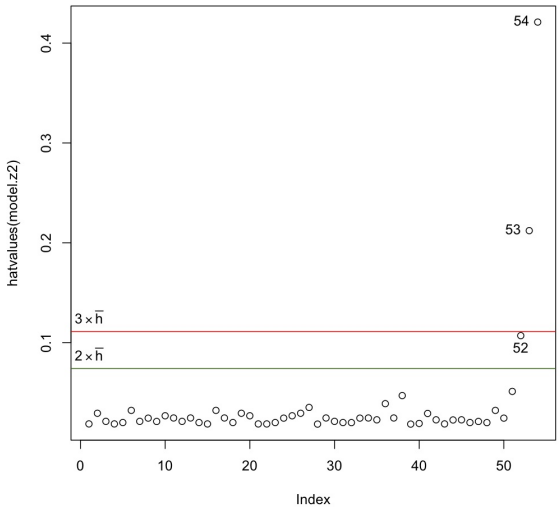


- x-variable vs residuals: definite structure in the residuals, which is similar to the above plot.
- Predicted vs measured y: we expect to see a strong trend about a 45° line (shown in blue). The strong departure from this line indicates there is a problem with the model



(e) We can consider removing the 4 points that strongly bias the observed *vs* predicted plot above.

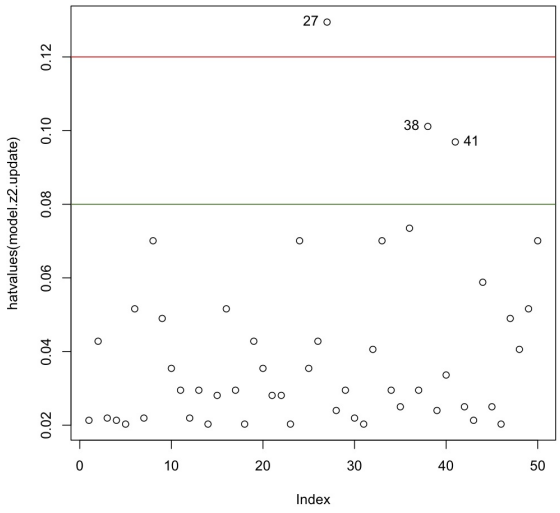
3. A plot of the hat-values (leverage) from the regression of SCB on z2 is:



with 2 and 3 times the average hat value shown for reference. Points 52, 53 and 54 have leverage that is excessive, confirming what we saw in the previous part of this question.

Once these points are removed, the model was rebuilt, and this time showed point 51 as an high-leverage outlier. This point was removed and the model rebuilt.

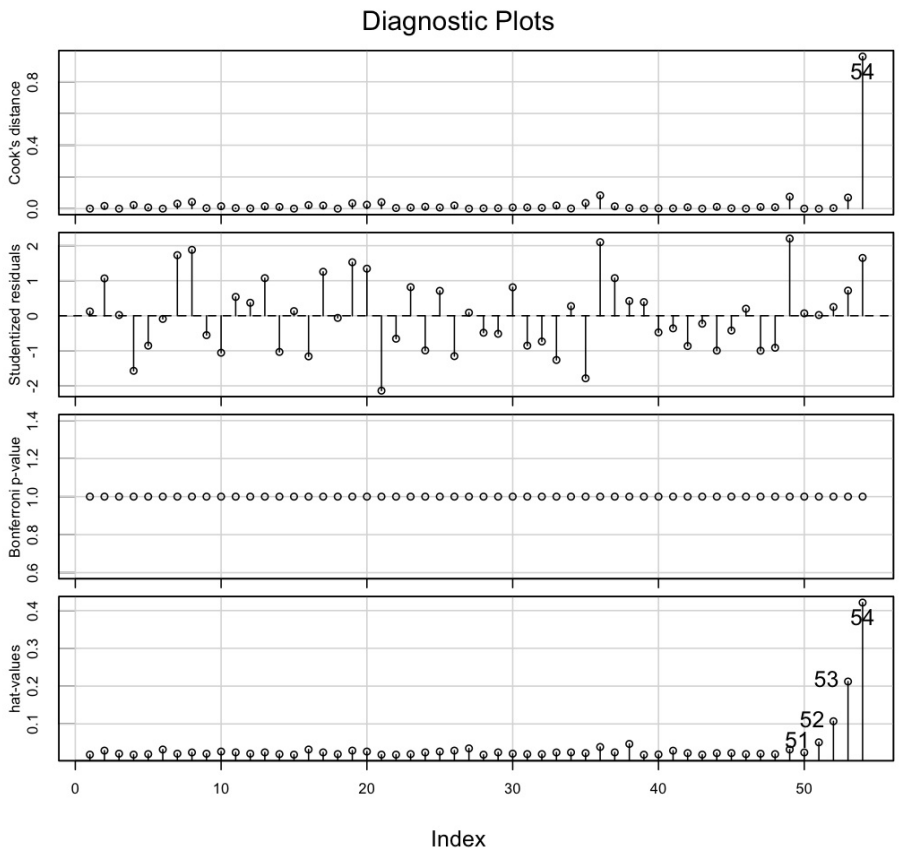
The hat values from this updated model are:



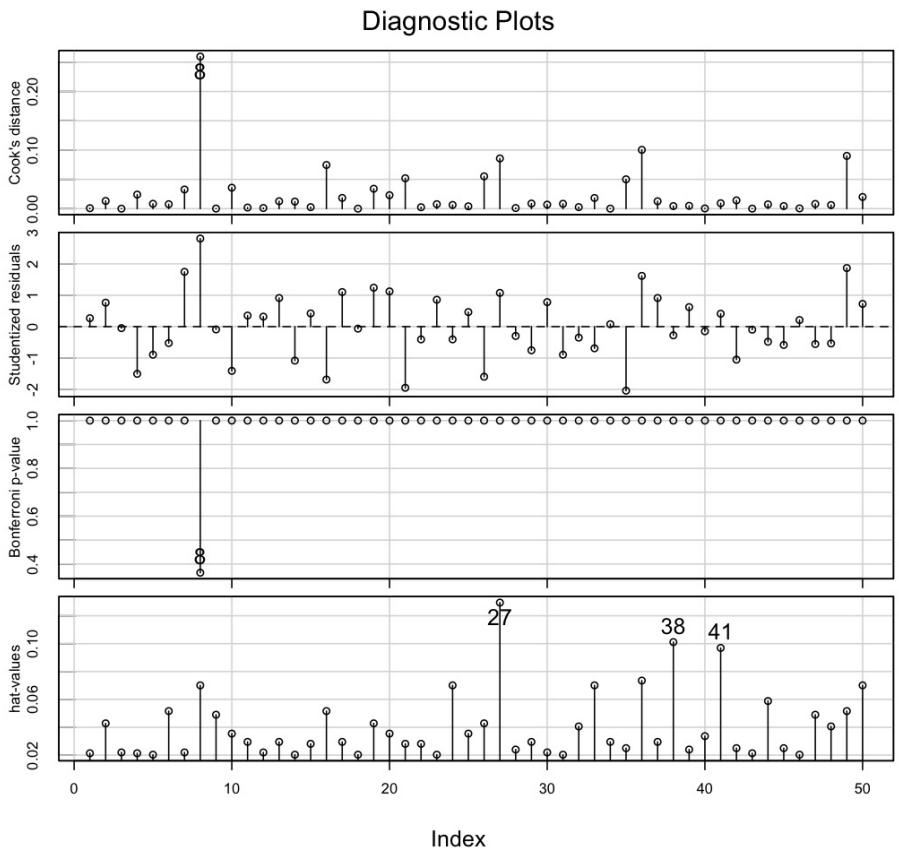
which is reasonable to stop at, since the problem has mostly gone away. If you keep omitting points, you will likely deplete all the data. At some point, especially when there is no obvious structure in the residuals, it is time to stop interrogating (i.e. investigating) and removing outliers.

> The updated model has a slightly improved standard error $S_E = 0.11$ and the least squares model fit (see the R code below) appears much more reasonable in the data.

4. The influence index plots for the model with all 54 points is shown first, followed by the influence index plot of the model with only the first 50 points.



The increasing leverage, as the abnormal process operation develops is clearly apparent. This leverage is not "bad" (i.e. influential) initially, because it is "in-line" with the regression slope. But by observation 54, there is significant deviation that observation 54 has high residuals distance, and therefore a combined high influence on the model (high Cook's D).



The updated model shows shows only point 8 as an influential observation, due to its moderate leverage and large residual. However, this point does not warrant removal, since it is just above the cut-off value of $4/(n-k) = 4/(50-2) = 0.083$ for Cook's distance.

The other large hat values don't have large Studentized residuals, so they are not influential on the model.

Notice how the residuals in the updated model are all a little smaller than in the initial model.

All the code for this question is given below:

```
LDPE <- read.csv('http://datasets.connectmv.com/file/LDPE.csv')
summary(LDPE)
N <- nrow(LDPE)

sub <- data.frame(cbind(LDPE$Tmax1, LDPE$Tmax2, LDPE$z1, LDPE$z2, LDPE$SCB))
colnames(sub) <- c("Tmax1", "Tmax2", "z1", "z2", "SCB")

bitmap('../images/ldpe-scatterplot-matrix.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(sub)
dev.off()

model.z2 <- lm(sub$SCB ~ sub$z2)
summary(model.z2)
```

```
# Plot raw data
bitmap('../images/ldpe-z2-SCB-raw-data.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(sub$z2, sub$SCB)
abline(model.z2)
identify(sub$z2, sub$SCB)
dev.off()

# Residuals normal? Yes, but have heavy tails
bitmap('../images/ldpe-z2-SCB-resids-qqplot.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
library(car)
qqPlot(model.z2, id.method="identify")
dev.off()

# Residual plots in time order: no problems detected
# Also plotted the acf(...): no problems there either
bitmap('../images/ldpe-z2-SCB-raw-resids-in-order.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(resid(model.z2), type='b')
abline(h=0)
dev.off()

acf(resid(model.z2))

# Predictions vs residuals: definite structure in the residuals!
bitmap('../images/ldpe-z2-SCB-predictions-vs-residuals.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(predict(model.z2), resid(model.z2))
abline(h=0, col="blue")
dev.off()

# x-data vs residuals: definite structure in the residuals!
plot(sub$Tmax2, resid(model.z2))
abline(h=0, col="blue")
identify(sub$z2, resid(model.z2))

# Predictions-vs-y
bitmap('../images/ldpe-z2-SCB-predictions-vs-actual.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(sub$SCB, predict(model.z2))
abline(a=0, b=1, col="blue")
identify(sub$SCB, predict(model.z2))
dev.off()

# Plot hatvalues
bitmap('../images/ldpe-z2-SCB-leverage.png', type="png256",
        width=6, height=6, res=300, pointsize=14)
plot(hatvalues(model.z2))
avg.hat <- 2/N
abline(h=2*avg.hat, col="darkgreen")
abline(h=3*avg.hat, col="red")
text(1, y=2*avg.hat, expression(2 %*% bar(h)), pos=3)
text(1, y=3*avg.hat, expression(3 %*% bar(h)), pos=3)
abline(h=hbar1*3, col="red", lwd=3, lty=2)
```

```
identify(hatvalues(model.z2))
dev.off()

# Remove observations (observation 51 was actually detected after
# the first iteration of removing 52, 53, and 54: high-leverage points)
build <- seq(1,N)
remove <- -c(51, 52, 53, 54)
model.z2.update <- lm(model.z2, subset=build[remove])

# Plot updated hatvalues
plot(hatvalues(model.z2.update))
N <- length(model.z2.update$residuals)
avg.hat <- 2/N
abline(h=2*avg.hat, col="darkgreen")
abline(h=3*avg.hat, col="red")
identify(hatvalues(model.z2.update))
# Observation 27 still has high leverage: but only 1 point

# Problem in the residuals gone? Yes
plot(predict(model.z2.update), resid(model.z2.update))
abline(h=0, col="blue")

# Does the least squares line fit the data better?
plot(sub$z2, sub$SCB)
abline(model.z2.update)

# Finally, show an influence plot
influencePlot(model.z2, id.method="identify")
influencePlot(model.z2.update, id.method="identify")

# Or the influence index plots
influenceIndexPlot(model.z2, id.method="identify")
influenceIndexPlot(model.z2.update, id.method="identify")


#-------- Use all variables in an MLR (not required for question)

model.all <- lm(sub$SCB ~ sub$z1 + sub$z2 + sub$Tmax1 + sub$Tmax2)
summary(model.all)
confint(model.all)
```

END