**15.1 – Multiple Regression Model and Terms (revisited)**

The form of the multiple regression model for mean function is given by

Also we typically we assume . The are terms in the model and remember the terms are functions of the predictors. When the predictors are nominal/ordinal the information contained in them are represented by dummy variables and transforming them will not be an issue, with the possible exception of adding an interaction term.

If consider a regression where all of the predictors are continuous then the multiple regression model could be viewed as:

Estimating a p-dimensional function without imposing some structure on it will be a very difficult/impossible task. However, if we impose the restriction that the effects of the predictors are additive, then we have the mean function below which is called an ***Additive Model***.

Note that this model is essentially the same as the formulation of the multiple linear regression model using terms, that is if we think of the functions as terms (based on the predictors .

**Example 1:** would be the same as having a term .

**Example 2:** If is a cubic polynomial in the predictor ,

we could add the following terms based on the predictor to the model

The intercept can be incorporated into the usual intercept term so we don’t need to specify a separate term for it.

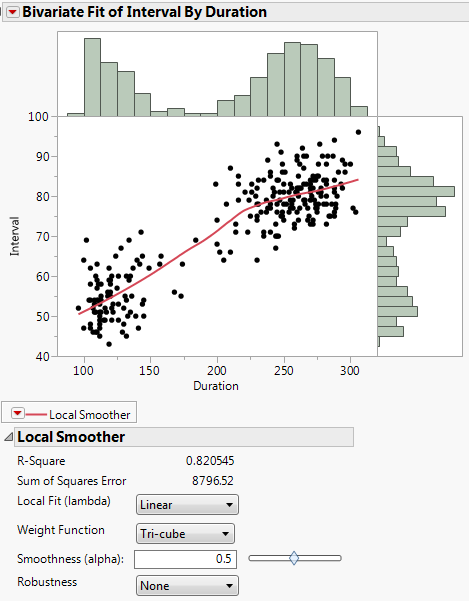
**Example 15.1 – Old Faithful Geyser Data (Datafile: Old Faithful.JMP)**



Quote from park rangers from Yellowstone National Park:   
  
Old Faithful erupts every 35-120 minutes for 1.5-5 minutes to a height of 90-184 feet. The rangers say that 90% of their predictions are within +/- 10 minutes. The time to the next eruption is predicted using the duration of the current eruption. The longer the eruption lasts, the longer the interval until the next eruption. For instance, a 2 minute eruption results in an interval of about 50 minutes whereas a 4.5 minute eruption results in an interval of about 85 minutes. It is not possible to predict more than one eruption in advance.

**Goal**: Develop a regression model to predict or explain the waiting time/interval (min.) until the next eruption of the Old Faithful Geyser given the duration (min.) of the current eruption.   
  
The variables are:

* Duration – duration of the current eruption (in minutes)
* Interval – waiting time until the next eruption (in minutes)



The smooth added scatterplot can be thought of fitting the model   
or in terms of the observed data:

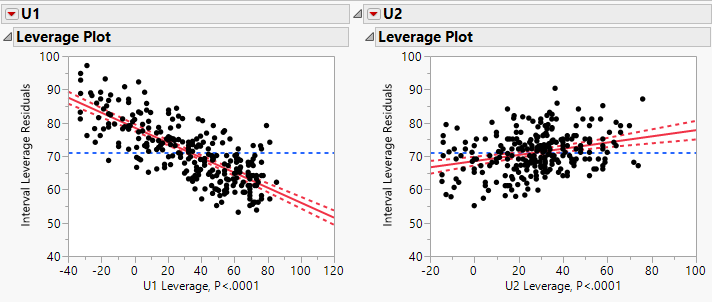
The smooth in the scatterplot is a nonparametric estimate of , i.e. .

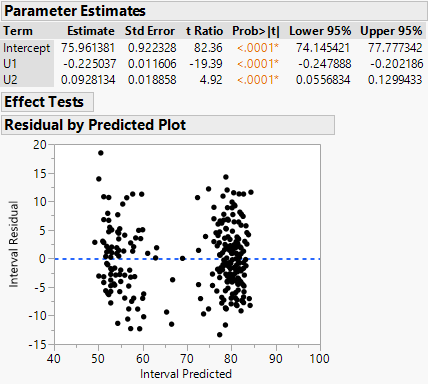
As the smooth suggests the mean function is piece-wise linear with a change occurring around Duration = 215 minutes, we might consider the following terms which give the so called ***broken stick*** function.

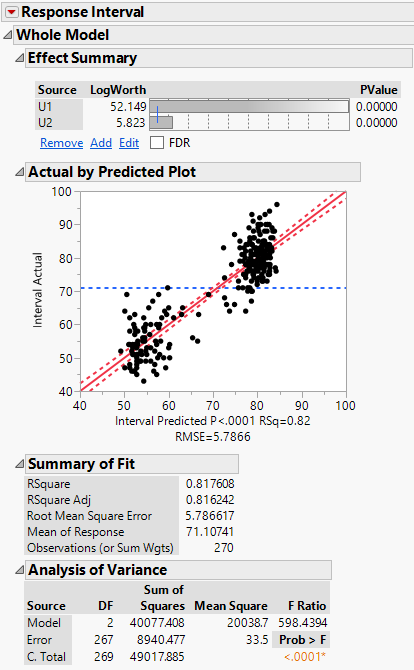
and

Here seems like a reasonable choice.

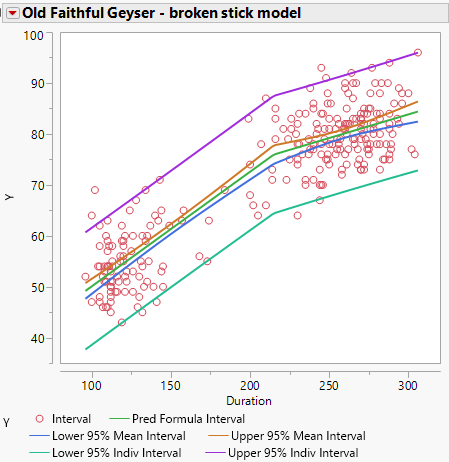
A summary of the broken stick model fit to these data is shown below.







Using **Graph > Overlay Plots** we can visualize fitted mean function, confidence interval for the E(Interval|Duration), and a prediction interval for an individual eruption.



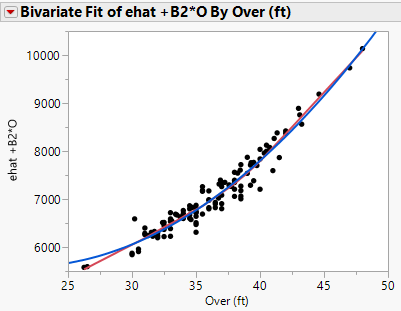
The previous example illustrates an important concept when choosing transformations/terms for predictors in developing a regression model. In this example we used a nonparametric smooth to visualize the functional form of a predictor in a regression model. By adding a smooth to the scatterplot of we were able to develop terms that would allow us to approximate this smooth. In simple linear regression, where we have a single predictor, a smooth added to the scatterplot of will always give an accurate visualization of the which can be used to develop a parametric model using the predictor .

In multiple regression we cannot simply examine scatterplots of for the purpose of developing multiple regression model because we also need to consider how the relate to one another. However, by considering the correct scatterplot we can use a smoother to get visual impression of , or using the transformation notation from the previous section, .

**15.2 – Component + Residual Plots (C+R Plots)**

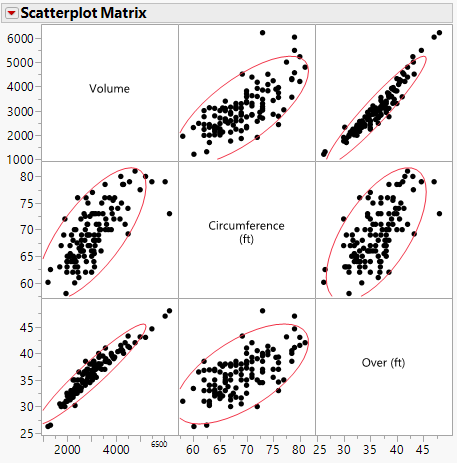
One of the first plots proposed for the purpose of visualizing the function form for a predictor in a multiple regression model is the Component + Residual plot (Larsen & McCleary (1972); Wood (1973)).

The C+R Plot is a plot of where the residuals and the estimated coefficient come from the current multiple regression model being considered. A smooth added to this plot will give a visual impression of the function form of that might improve upon the current model.

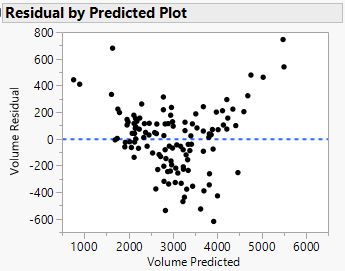


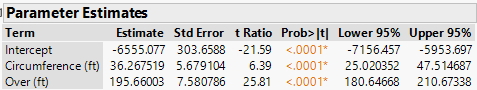
**Example 15.2 – Haystacks**

A scatterplot matrix of these data is shown below.



Fitting the model obtain the following parameter estimates and residual plot.



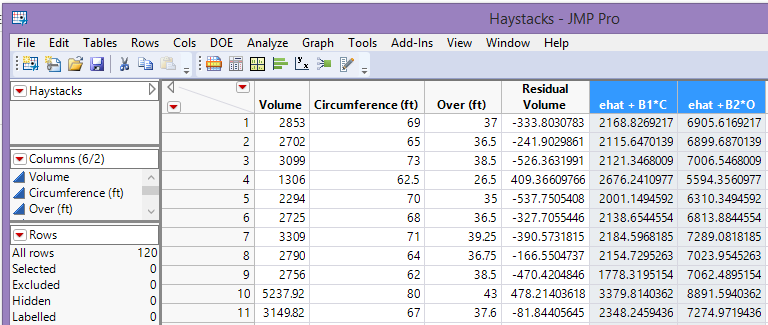
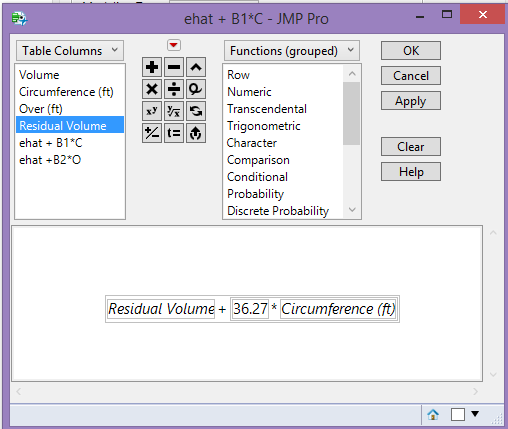


Clearly there is curvature in the residuals that needs

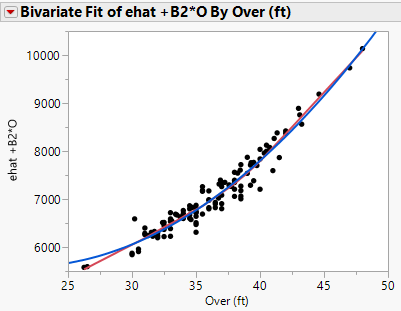
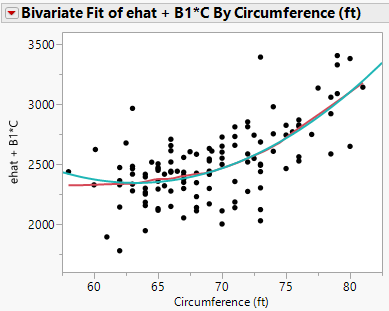
to be addressed. We could consider transforming

the response or by including terms in the mean   
function that address this curvature.

We first form for both C and O

C+R Plot for Circumference (C) C+R Plot for Over (O)



Both C+R plots have a smooth and quadratic polynomial added. The quadratic polynomials both match the smooth fairly well. So we might consider adding squared terms for C and O to the model.

A summary of the model with the squared terms added is shown below:

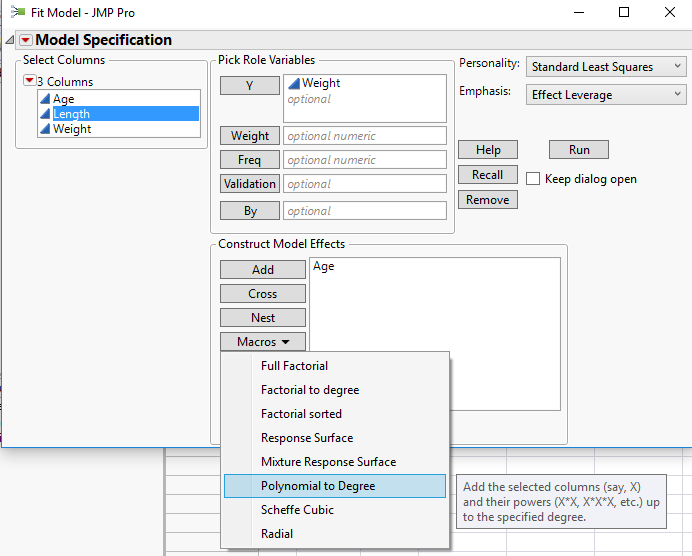


As we have seen in Section 13, squared terms can be highly correlated the linear terms, thus mean centering the squared terms is recommended, i.e.

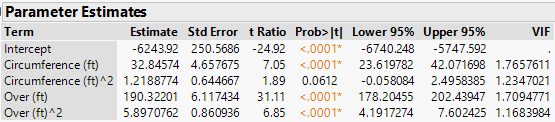
 JMP formula for mean centered terms

and

We can alternatively use the Polynomial to Degree option in the Macro drop-down menu in the Fit Model dialog box.

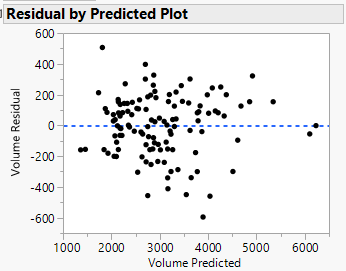


After mean centering the squared terms we have:



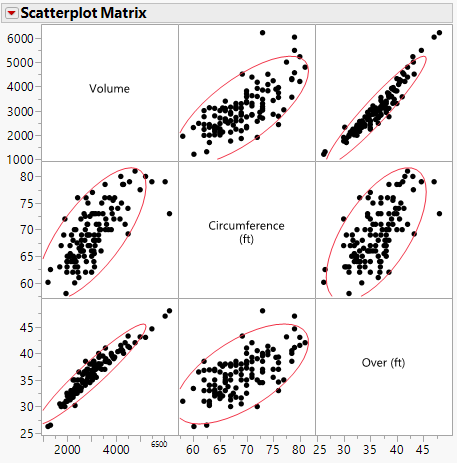
We could consider dropping the squared term for circumference as .

The residual plot from the model using only the mean centered squared term for Over is shown below.



C+R Plots will only give a proper visualization of the function form of when one of the following conditions holds:

1. The unadjusted relationship between is linear.
2. The relationships between all of the are linear.

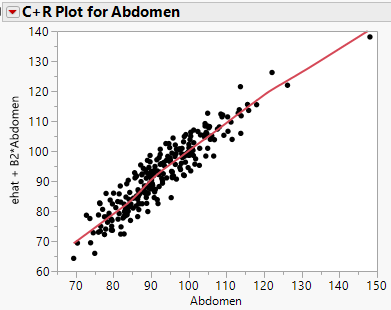


Comments:

When the predictors are not linearly related, we can try using the Bulging Rule to transform them to achieve linearity before examining C+R plots.

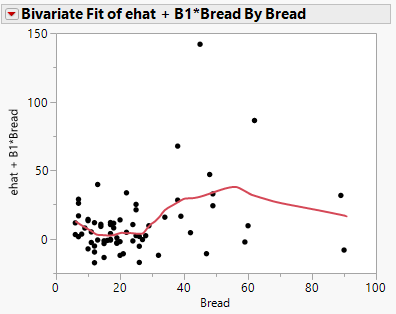
**Example 15.3 – Body Fat Study**

Below is the C+R Plot for the predictor Abdomen in a multiple regression model for % body fat. As the plot is clearly linear, no transformation is suggested by this plot.



**Example 15.4 – Big Mac Study**

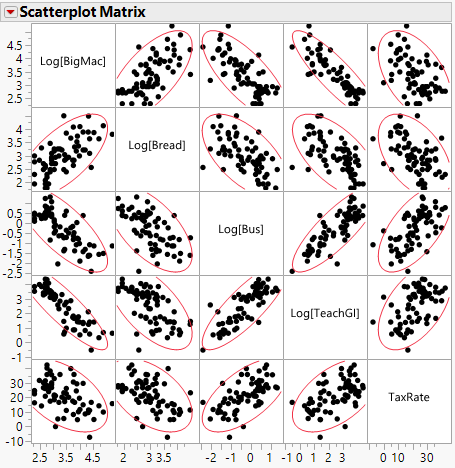
These data were considered in an example section 13. Consider the model

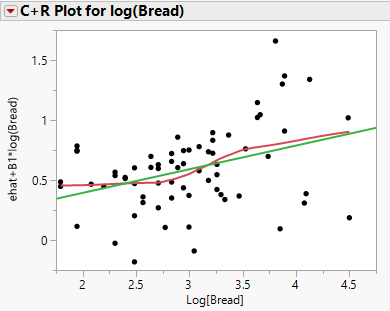
C+R Plot for Bread  


The functional form is NOT easily recognized from this plot, it almost looks cubic or quartic. We have seen previously that using the log transformation for Bread, Bus, & TeachGI and log transforming the response Big Mac produced a reasonable model.

fit to these data. Notice that several of the relationships amongst the predictors are nonlinear in the scatterplot matrix below.   
  


After taking the logarithms of the response and three of the predictors the relationships between predictors are linear as are the plots of .





After log transforming the response and the predictors Bread, Bus, and TeachGI the C+R plot for log(Bread) is reasonably linear.

C+R Plots belong in our regression toolbox but they are limited in their usefulness. We next consider a better graphical tool for visualizing the functional form of predictors in a multiple regression model.

**15.2 – CERES Plots (Combining Conditional Expectations and RESiduals Plots)**

When the relationships between the predictors () are not linear and we cannot, or decide not to, transform them to improve linearity then C+R plots may not give the correct impression of the functional forms of the . The nonlinear relationships amongst the predictors will “bleed” into the C+R plot and give a distorted view of the functional form of

In general we are interested in determining if the model

can be improved by modifying the model to include an unknown transformation of the predictor/term , , i.e. the model below is an improvement over the model above,

Typically at the start, all of the terms will be the predictors themselves, i.e. , but if we transform some the predictors to improve linearity/normality, as we did in the Big Mac example in the previous section, then this need not be the case.

To visualize the transformation for we can construct the CERES plot which is a scatterplot of

and then add smooth to the plot. We can then choose terms based on that match the smooth, e.g. we might use or we might add squared and cubed terms if the smooth appears to a cubic polynomial in . Also if the smooth appears linear then NO transformation is needed.

The key in this process is what model we use to obtain **.** The choice of the model used to obtain this component depends on how relates to the other predictors/terms in the model. We can break this down into different cases which are outlined below.

As stated above, in order to estimate we need to consider the relationship between and the other predictors/terms in the model .

**Case 1** – 🡨 This says is unrelated to all of the other   
 terms in the model, i.e. or .

If Case 1 applies, which it rarely does unless we are working with a designed experiment (e.g. wool data), then fitting the model without the predictor/term to be transformed included (without ) and plotting the residuals from this fit vs. will give a visual impression of the transformation .

**Case 2** - 🡨 This says is linearly related to all other .

If Case 2 applies we estimate the component by fitting the model.

This is called CERES with ***linear augmentation***.

**Case 3** - 🡨 This says the relationship between   
 and is quadratic for **at least one** .

If Case 3 applies we estimate the component by fitting the model.

This is called CERES with ***quadratic augmentation*.**

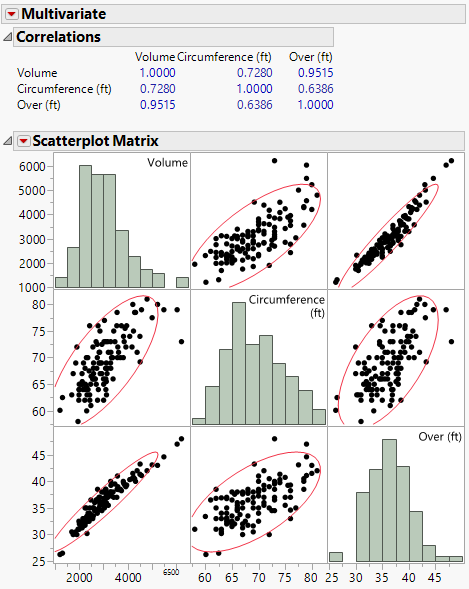
**Case 4** - 🡨 This says the conditional distribution of given   
 is some other nonlinear function for some .   
 This function itself would need to be estimated by   
 smoothing a scatterplot of vs. .

In Case 4 assuming we need to do that for all of the other terms in the model then we estimate by fitting the model,

This is called CERES with ***smooth augmentation***.

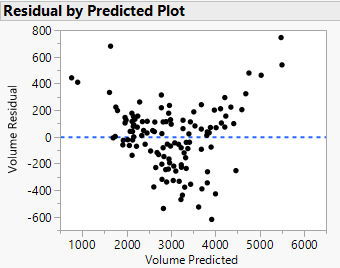
**Example 15.5 – Haystacks**

Below is a scatterplot matrix for these data.

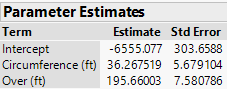
  
As the relationship between Circumference (C) and Over (O) is linear we are in Case 2. As Over has the strongest relationship with the response () we will begin by transforming Over first.

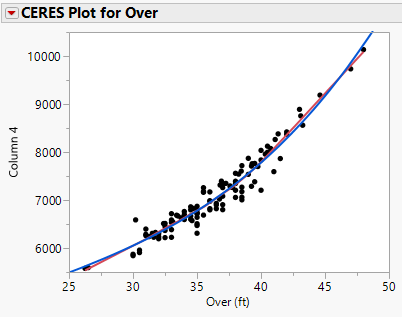
Comments:

Residuals from



We first fit the model and obtain the following.



Thus and we will subtract this quantity from the response and plot this vs. Over.  


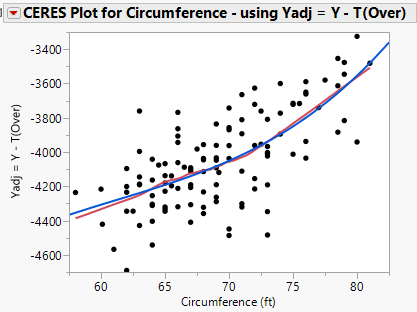
A cubic polynomial matches the smooth very well. Thus we could add terms or the mean centered versions of these. We could also simply add which is consistent with earlier geometric considerations of the haystacks.

Radius = and thus,

V =

So is sensible term to add to the model.

We could reverse the roles and consider a CERES plot for Circumference (C) or when transforming multiple predictors we can plot vs. C to get a visual impression of . We can do this easily by saving the fitted values from either fit in the scatterplot above and subtracting them from the response to form .



Again a cubic polynomial again matches the smooth for Circumference. Thus we could add terms or the mean centered versions of these. We could also simply add which is consistent with earlier geometric considerations of the haystacks.

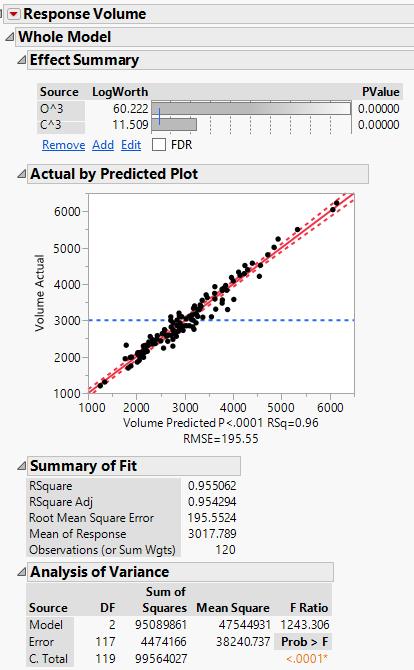
Radius = and thus,

V =

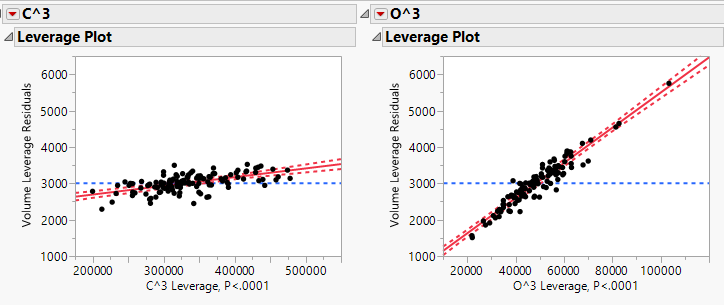
So is sensible term to add to the model.

We now fit the model using terms suggested by the CERES plots, i.e.

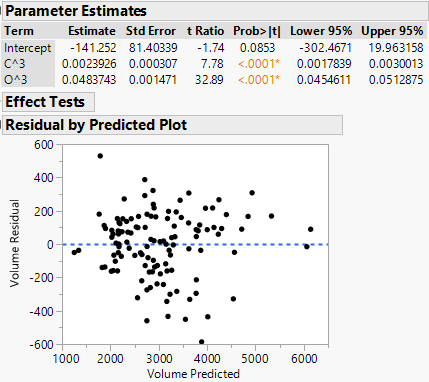
This model is summarized on the following page.



AVPs

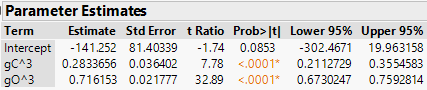


Parameter Estimates and Residual Plot



We can incorporate the geometric considerations by multiplying the term by and the term by . Thus we will fit the model below.

The parameter estimates are:



The estimated coefficient nearly sum to 1, thus estimated volume from this regression model is essentially a weighted average of the two volume forumulae from the geometric considerations of the haystacks.

Constructing CERES plots in JMP is **very tedious**, especially if the number of terms/predictors is large! The car package in R has a function for constructing CERES plots for all of the predictors/terms in the model along with a plethora of functions for performing other regression related tasks we have covered.

> Haystacks = read.table(file.choose(),header=T,sep=",")

> names(Haystacks)

[1] "Volume" "Circum" "Over"   
  
> head(Haystacks)

Volume Circum Over

1 2853 69.0 37.0

2 2702 65.0 36.5

3 3099 73.0 38.5

4 1306 62.5 26.5

5 2294 70.0 35.0

6 2725 68.0 36.5

> hay.lm1 = lm(Volume~.,data=Haystacks)

> summary(hay.lm1)

Call:

lm(formula = Volume ~ ., data = Haystacks)

Residuals:

Min 1Q Median 3Q Max

-619.61 -141.14 26.43 126.34 744.46

Coefficients:

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

(Intercept) -6555.077 303.659 -21.587 < 2e-16 \*\*\*

Circum 36.268 5.679 6.386 3.57e-09 \*\*\*

Over 195.660 7.581 25.810 < 2e-16 \*\*\*

---

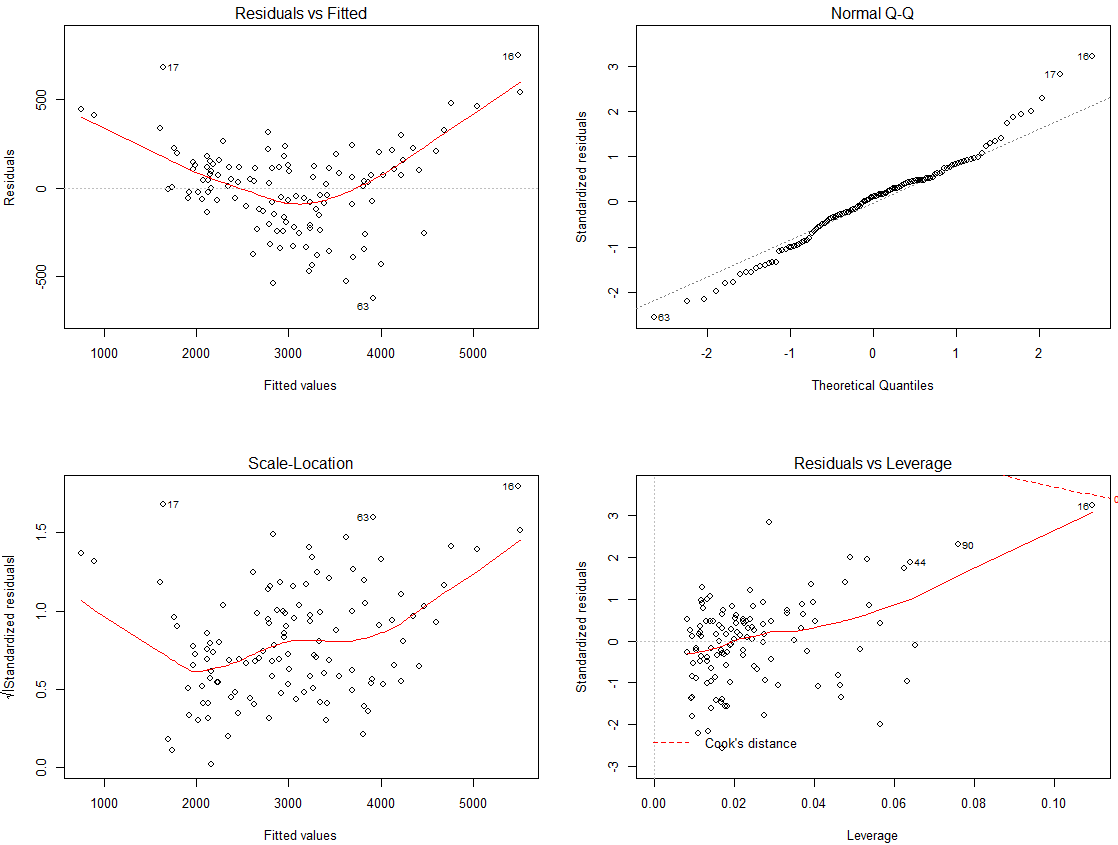
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 244.4 on 117 degrees of freedom

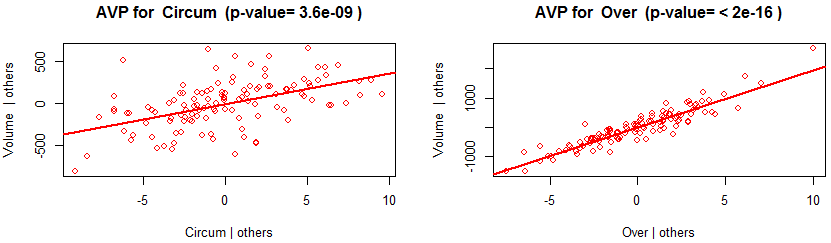
Multiple R-squared: 0.9298, Adjusted R-squared: 0.9286

F-statistic: 774.6 on 2 and 117 DF, p-value: < 2.2e-16

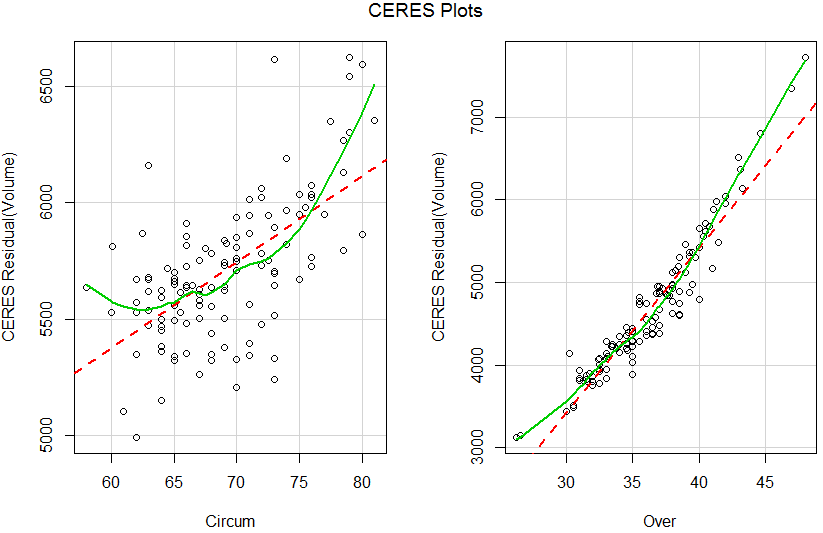
> plot(hay.lm1)



> AVPs(hay.lm1)



> ceresPlots(hay.lm1)



Both plots indicate the need to address curvature and given the lowess smooths, polynomial terms of degrees 2 or 3 would be appropriate. From previous geometric considerations we will add cubic terms based on each predictor.

> Over3 = Over^3

> Circum3 = Circum^3

> hay.lm3 = lm(Volume ~ Circum3 + Over3, data = Haystacks)

> summary(hay.lm)

Call:

lm(formula = Volume ~ Circum3 + Over3, data = Haystacks)

Coefficients:

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

(Intercept) -1.413e+02 8.140e+01 -1.735 0.0853 .

Circum3 2.393e-03 3.074e-04 7.784 3.1e-12 \*\*\*

Over3 4.837e-02 1.471e-03 32.886 < 2e-16 \*\*\*

---

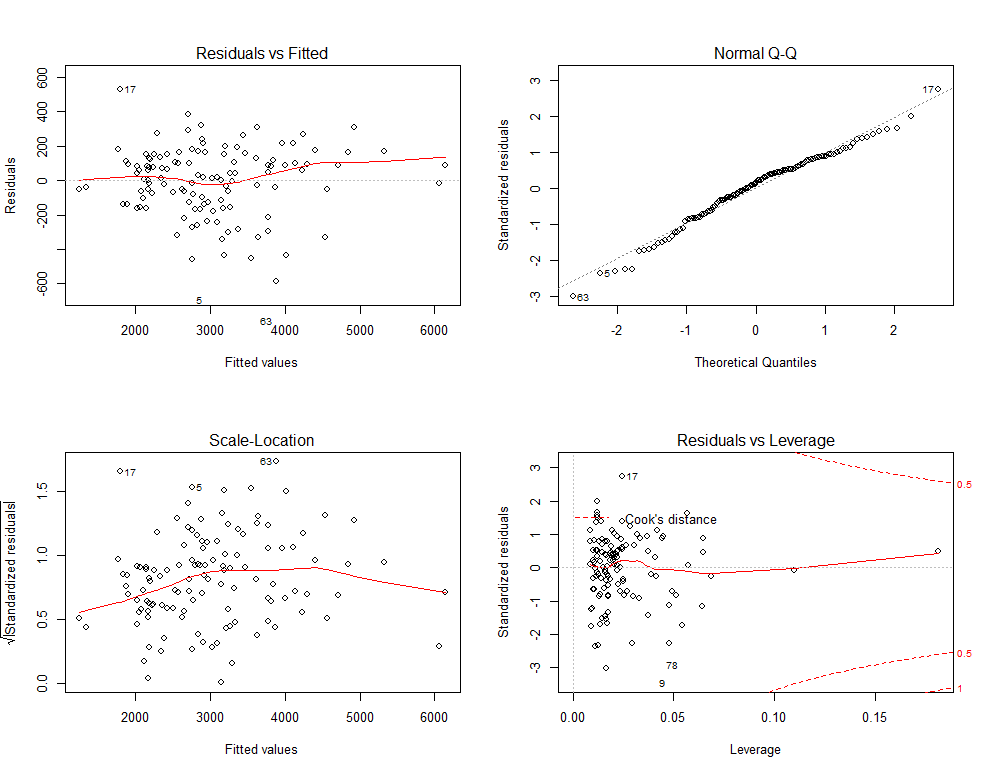
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 195.6 on 117 degrees of freedom

Multiple R-squared: 0.9551, Adjusted R-squared: 0.9543

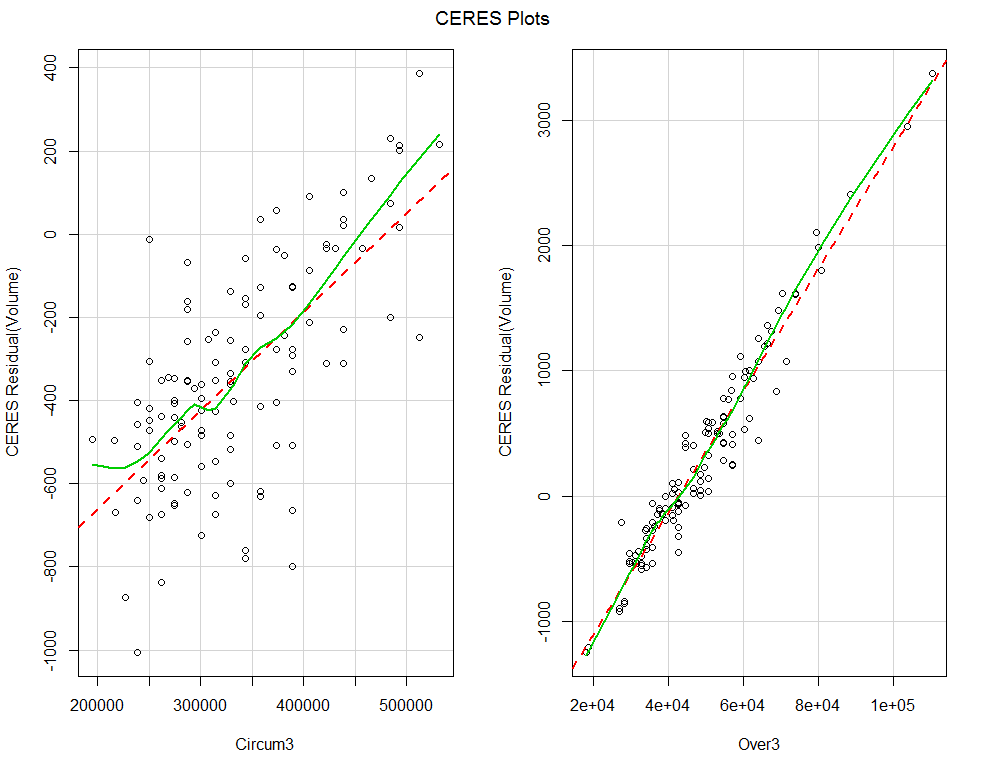
F-statistic: 1243 on 2 and 117 DF, p-value: < 2.2e-16

> plot(hay.lm)



We can again CERES plots to confirm that we have addressed the curvature in each term.

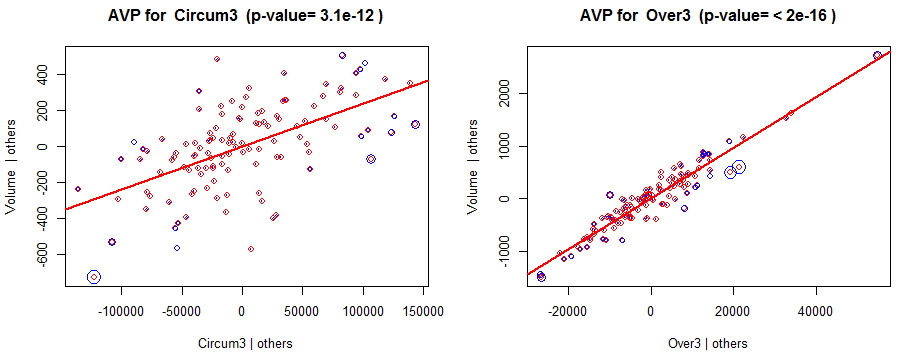
> ceresPlots(hay.lm)



The cubic terms appear to have addressed the curvature as the smooths added to the CERES plots are both approximately linear.

We continue with this example by examining some more tools for fitting, examining, and assessing adequacy of regression models in R. Again we will be using functions in the car library and in my Regression.RData R workspace.

Functions in the Regression.RData workspace

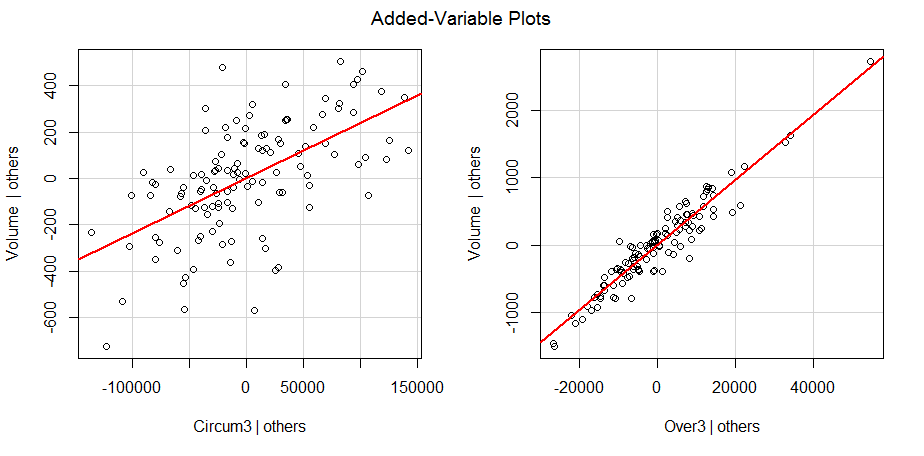
> Diagplot3(hay.lm,dfbet=T) 🡨 added variable plots with bubbles   


Variance Inflation Factor Table

Variable VIF Rsquared

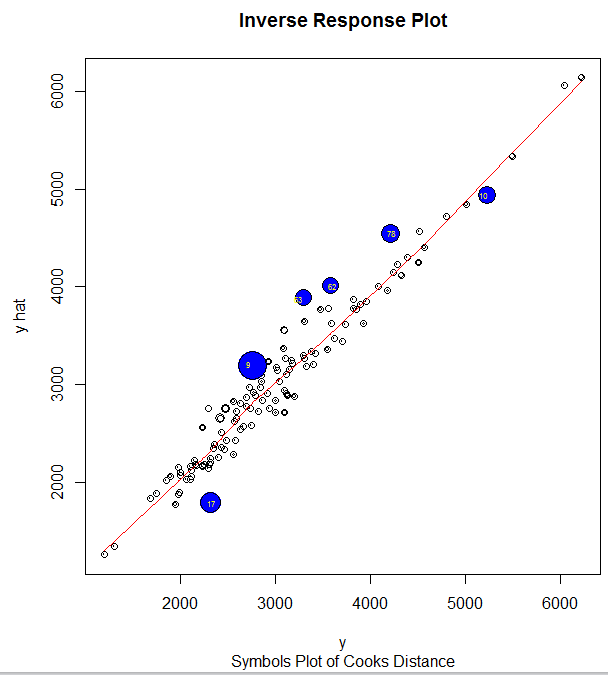
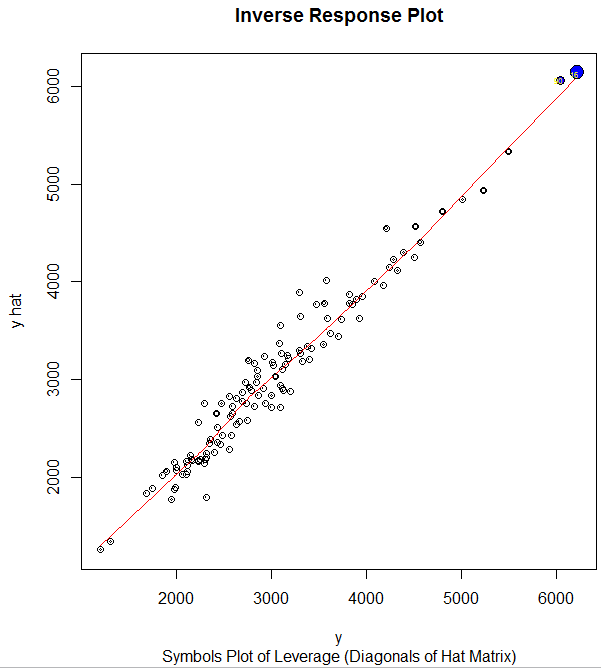
Circum3 Circum3 1.692145 0.4090341

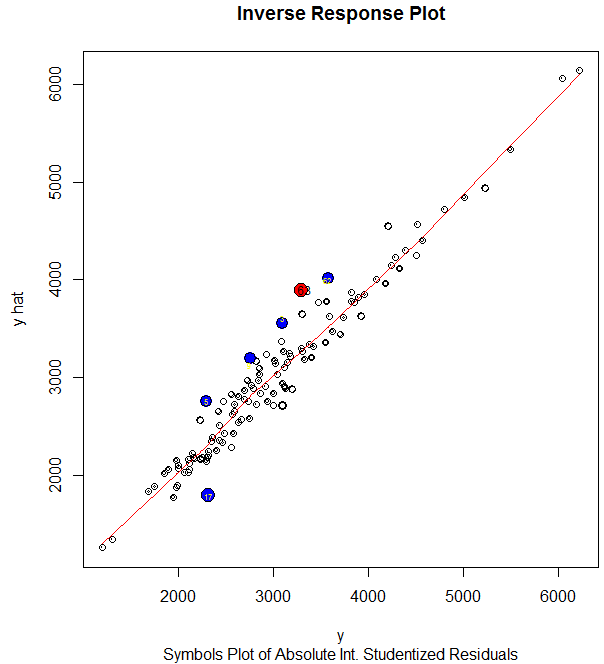
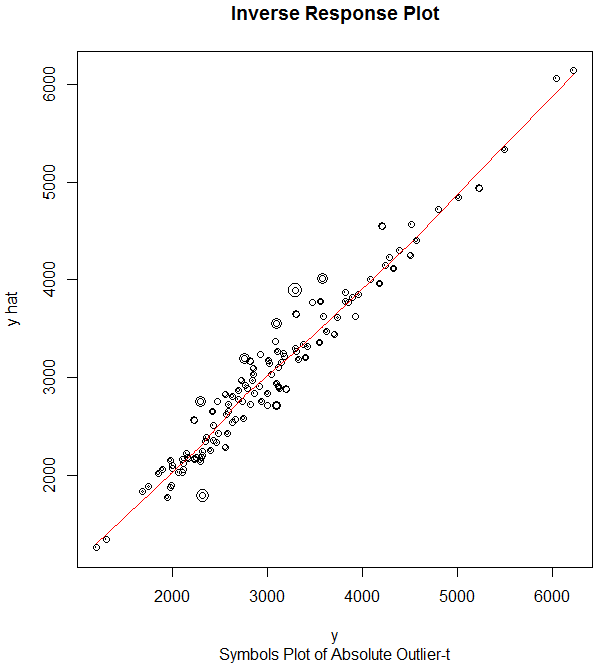
Over3 Over3 1.692145 0.4090341

> avPlots(hay.lm) 🡨 added variable plots from car library  
  


> MLRdiag(hay.lm) 🡨 Inverse Fitted Value (Response) Plot with Case Diagnostics

cases exceeding size adjusted cutoffs = blue, absolute cutoffs = red

Bonferroni Corrected Outlier t-test

> outlierTest(hay.lm)

No Studentized residuals with Bonferonni p < 0.05

Largest |rstudent|:

rstudent unadjusted p-value Bonferonni p

63 -3.134327 0.0021819 0.26183

**Big Example in R: Upper Ozone Concentration in Los Angeles Basin**

Ozone is one of the nasty constituents of photochemical smog, and its concentration is the standard indicator of the severity of such smog. If its level is high, a smog alert is called. Ozone is not emitted directly into the atmosphere. Rather, it is a product of chemical reactions that require solar radiation and emissions of primary pollutants from smoke stacks and automobile exhaust. When the ventilation of the atmosphere is low, the chemical reactions bring ozone to high levels. Low ventilation occurs when wind speeds are low and temperature is high because on hot, calm days in the summer the atmosphere cannot cleanse itself. The goal in the analysis of these data is to determine how ozone depends on the other variables.

A brief description of the variables and their abbreviations is given below:



* **upoz** – Upper ozone concentration (ppm)
* **day** – day of the year (1 – 330, where 1 = January 1st)
* **safb** – Sandburg Air Force Base temperature (Co)
* **inbh** – Inversion base height (ft.)
* **dagg** – Daggett pressure gradient (mmHg)
* **vis** – Visibility (miles)
* **v500** – Vandenburg 500 millibar height (m)
* **hum** – Humidity (%)
* **inbt** – Inversion base temperature (Fo)
* **wind** – Wind speed (mph)

We begin our analysis by examining a scatterplot matrix of the data. It appears that upper ozone concentration is related to several of the predictors, although not linearly. We also see that several of the variables are correlated with one another.

First read these data in from OzoneData.csv

> Ozone = read.table(file.choose(),header=T,sep=",")

> names(Ozone)

[1] "upoz" "day" "v500" "wind" "hum" "safb" "inbh" "dagg" "inbt" "vis"   
> attach(Ozone) 🡨 some of my functions require the data set to be attached.

> head(Ozone)

upoz day v500 wind hum safb inbh dagg inbt vis

1 3 3 5710 4 28 40 2693 -25 87 250

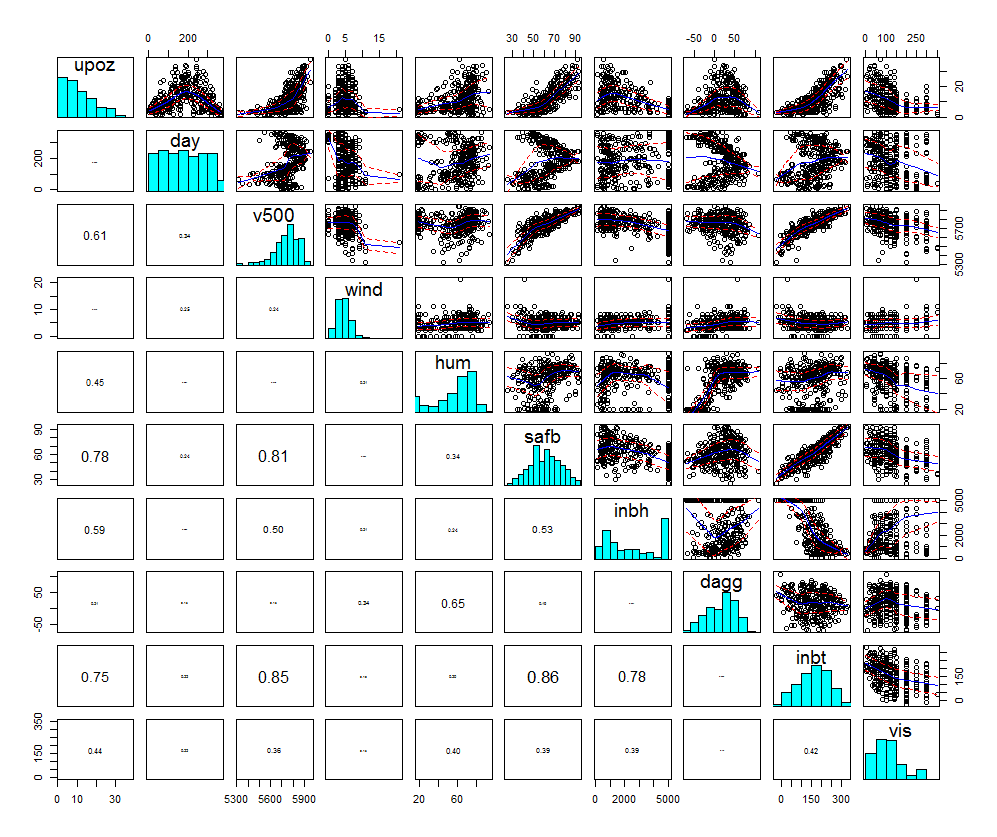
2 5 4 5700 3 37 45 590 -24 128 100

3 5 5 5760 3 51 54 1450 25 139 60

4 6 6 5720 4 69 35 1568 15 121 60

5 4 7 5790 6 19 45 2631 -33 123 100  
6 4 8 5790 3 25 55 554 -28 182 250

> pairs.plus(Ozone)

  
We begin by fitting an OLS multiple regression model using all available predictors.

> lm.oz1 = lm(upoz~.,data=Ozone)

> summary(lm.oz1)

Coefficients:

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

(Intercept) 18.3792938 29.5045242 0.623 0.53377

day -0.0088490 0.0027199 -3.253 0.00126 \*\*

v500 -0.0051340 0.0053950 -0.952 0.34200

wind -0.0198304 0.1238829 -0.160 0.87292

hum 0.0804923 0.0188345 4.274 2.54e-05 \*\*\*

safb 0.2743349 0.0497361 5.516 7.17e-08 \*\*\*

inbh -0.0002497 0.0002950 -0.846 0.39798

dagg -0.0036968 0.0112925 -0.327 0.74360

inbt 0.0292640 0.0136115 2.150 0.03231 \*

vis -0.0080742 0.0037565 -2.149 0.03235 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

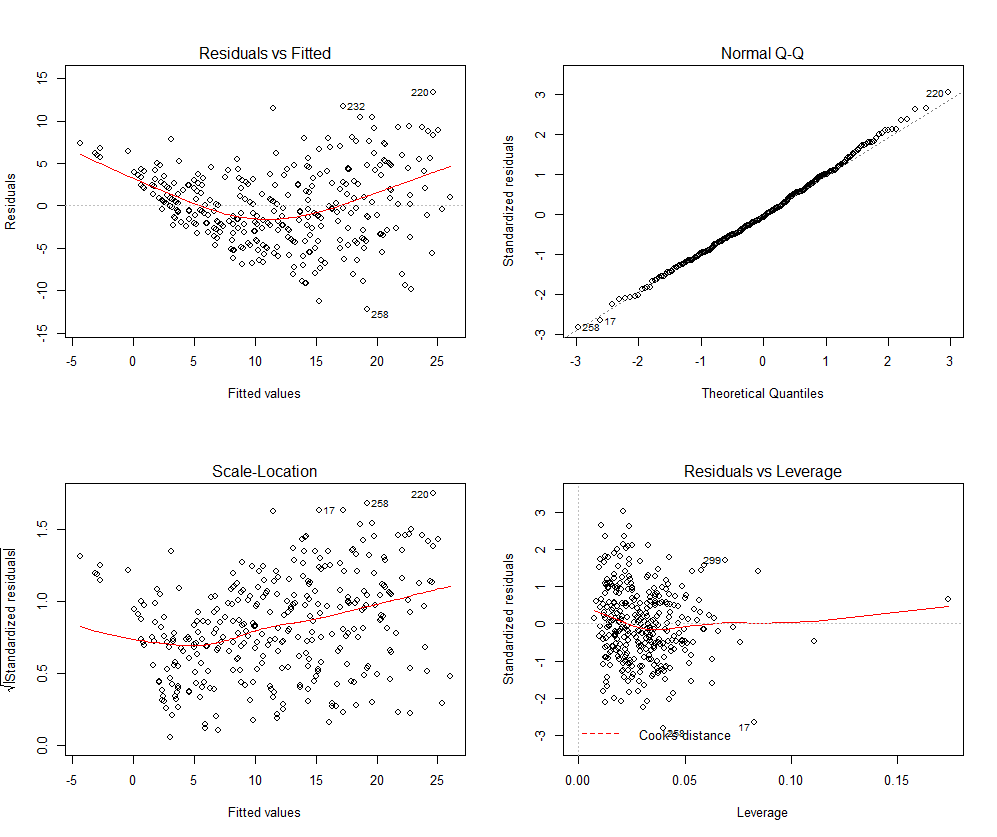
Residual standard error: 4.441 on 320 degrees of freedom

Multiple R-squared: 0.7011, Adjusted R-squared: 0.6927

F-statistic: 83.4 on 9 and 320 DF, p-value: < 2.2e-16

> par(mfrow=c(2,2)) 🡨Set up plotting region to allow for 2 row & 2 columns of plots

> plot(lm.oz1)



**Conduct Tukey’s Test for Nonadditivity**

> yhat = fitted(oz.lm)

> yhat2 = yhat^2

> temp = lm(upoz~.+yhat2,data=Ozone)

> summary(temp)

Call:

lm(formula = upoz ~ day + v500 + wind + hum + safb + inbh + dagg +

inbt + vis + yhat2, data = Ozone)

Residuals:

Min 1Q Median 3Q Max

-12.124 -2.307 -0.094 2.454 12.713

Coefficients:

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

(Intercept) -2.916e+01 2.753e+01 -1.059 0.290

day -1.865e-03 2.625e-03 -0.711 0.478

v500 6.296e-03 5.116e-03 1.231 0.219

wind -1.038e-01 1.134e-01 -0.916 0.361

hum 1.569e-02 1.894e-02 0.829 0.408

safb -1.318e-02 5.753e-02 -0.229 0.819

inbh -3.147e-04 2.691e-04 -1.169 0.243

dagg 5.332e-03 1.036e-02 0.515 0.607

inbt -6.227e-03 1.316e-02 -0.473 0.636

vis -1.850e-04 3.560e-03 -0.052 0.959

yhat2 3.936e-02 4.846e-03 8.122 1.02e-14 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

**Nonconstant Variance Tests (Score Test)**

> ncvTest(oz.lm1)

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 35.86603 Df = 1 p = 2.113617e-09

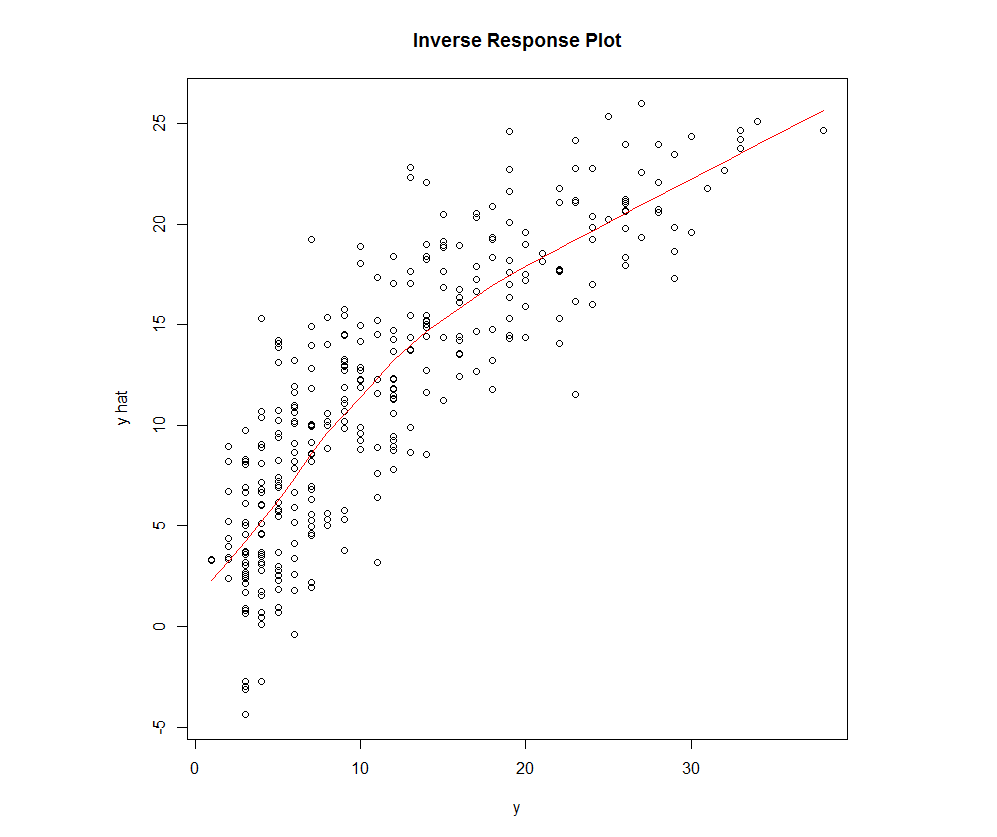
> ncvTest(oz.lm1,~v500+hum+dagg+inbt,data=Ozone)

Non-constant Variance Score Test

Variance formula: ~ v500 + hum + dagg + inbt

Chisquare = 48.00608 Df = 4 p = 9.410319e-10

**Inverse Fitted Value Plot**   
> invplot(oz.lm) 🡨 requires Ozone be attached



The inverse fitted value plot suggests a transformation of the response may be warranted to address the curvature and nonconstant variation.

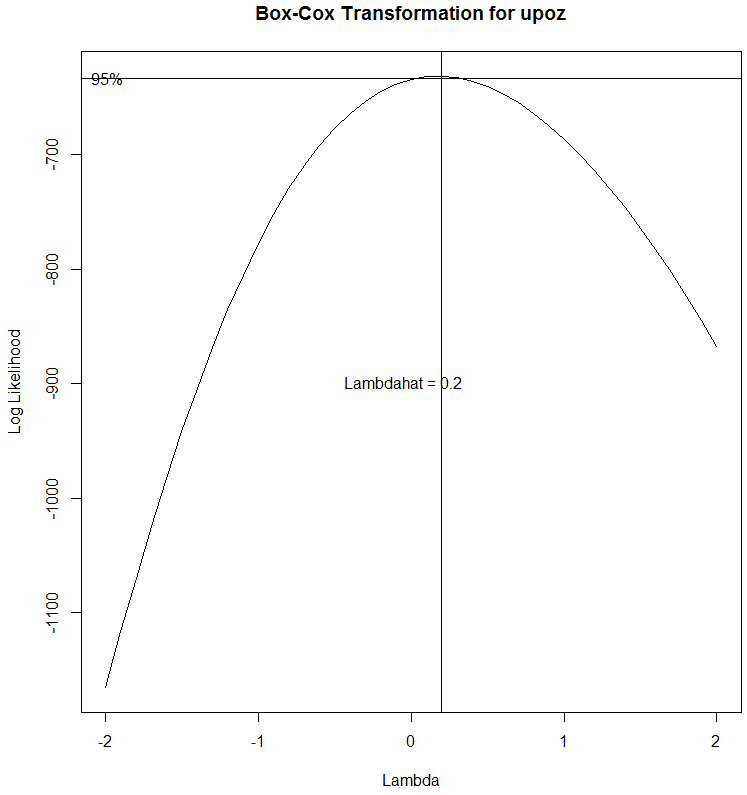
What should we use for ?

We could also consider using the Box-Cox Procedure for transforming the response. The functions BCtran in my regression library is function that plots the Box-Cox objective function and provides a visual CI for the optimal value.

The function powerTransform in the car library will give numeric output with the CI and tests whether and . These are both demonstrated on the following page for the upper ozone concentration ().

**Box-Cox Transformation**

> BCtran(upoz)



> summary(powerTransform(upoz))

bcPower Transformation to Normality

Est.Power Std.Err. Wald Lower Bound Wald Upper Bound

upoz 0.1785 0.0742 0.0331 0.3239

Likelihood ratio tests about transformation parameters

LRT df pval

LR test, lambda = (0) 5.922072 1 0.01495236

LR test, lambda = (1) 110.044096 1 0.00000000

We might consider using or or possibly , however neither are common transformations so we might also consider using the log of ozone instead. Furthermore there is no theoretical reason to use a cube or fourth root transformation, e.g. neither is a variance stabilizing transformation. The log transformation however is a variance stabilizing transformation and a variable that is normal in the log scale is said to have a *lognormal distribution.* I have not heard of *cube-* or *fourth-root normal distribution*.

**Model with**

**Comments:**

> oz.lm2 = lm(upoz^0.25~.,data=Ozone)

> summary(oz.lm2)

Coefficients:

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

(Intercept) 1.252e+00 1.125e+00 1.113 0.266543

day -4.026e-04 1.037e-04 -3.881 0.000126 \*\*\*

v500 -4.724e-05 2.057e-04 -0.230 0.818537

wind -1.617e-03 4.724e-03 -0.342 0.732353

hum 2.536e-03 7.183e-04 3.530 0.000476 \*\*\*

safb 1.240e-02 1.897e-03 6.535 2.51e-10 \*\*\*

inbh -3.074e-05 1.125e-05 -2.732 0.006649 \*\*

dagg 1.686e-04 4.306e-04 0.392 0.695643

inbt 4.782e-04 5.191e-04 0.921 0.357661

vis -3.569e-04 1.433e-04 -2.491 0.013240 \*

---

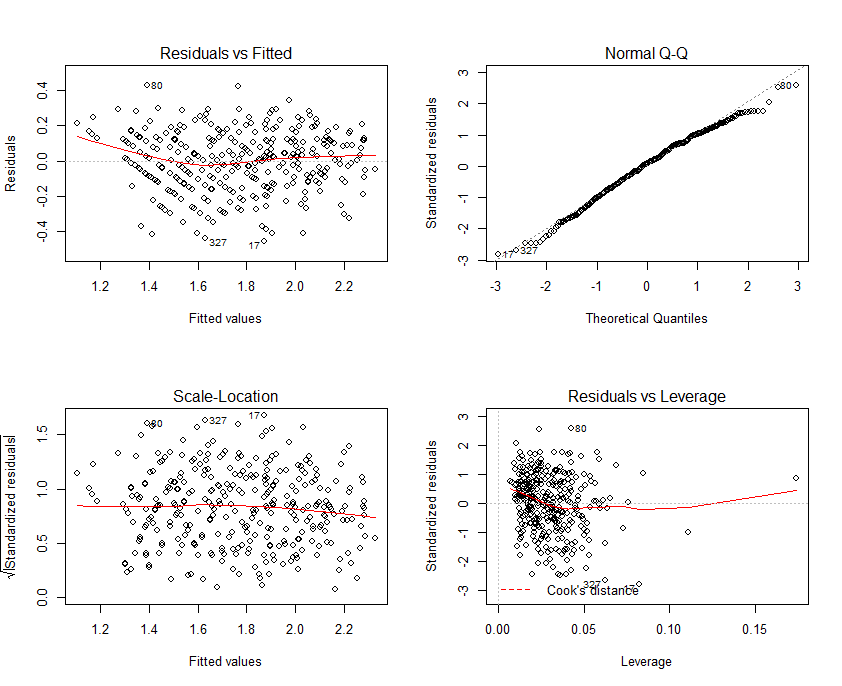
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

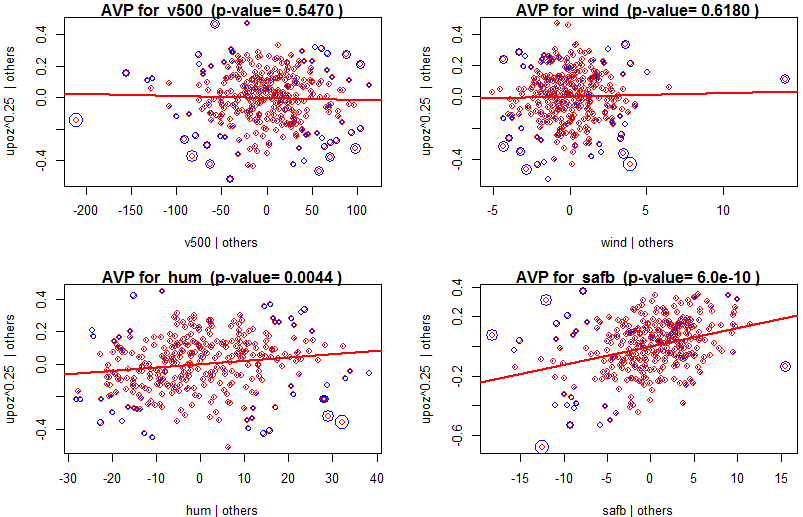
Residual standard error: 0.1694 on 320 degrees of freedom

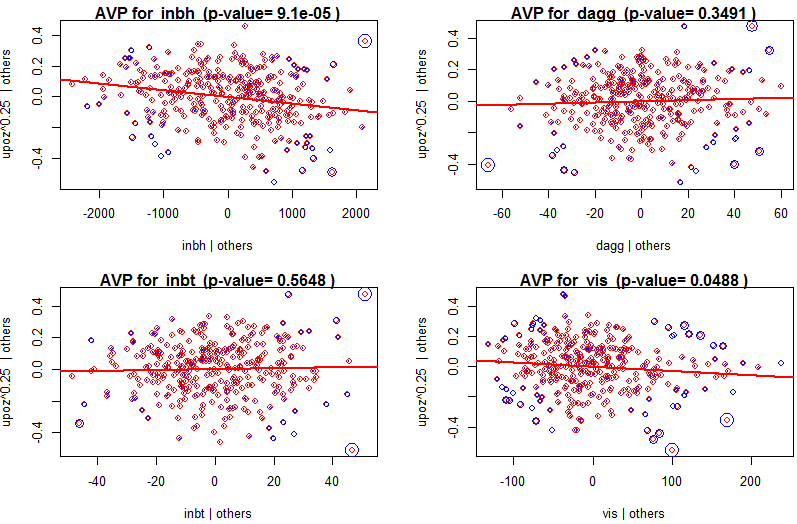
Multiple R-squared: 0.7324, Adjusted R-squared: 0.7248

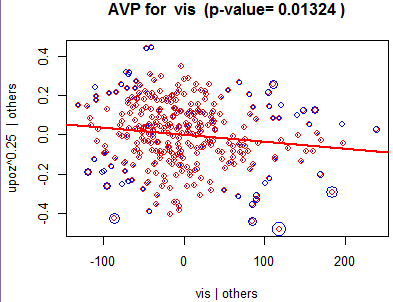
F-statistic: 97.3 on 9 and 320 DF, p-value: < 2.2e-16

> plot(oz.lm2)



**Added Variable Plots** – The function AVPs in my regression library will plot the added variable plots for the terms in the model and if you specify dfbet = T it will add bubbles around the points proportional to the DFBETAS statistic.> AVPs(oz.lm2,dfbet=T)





**Fit model using**> oz.lm3 = lm(log(upoz)~.,data=Ozone)

**Comments:**

Coefficients:

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

(Intercept) 4.015e-01 2.683e+00 0.150 0.88113

day -9.879e-04 2.473e-04 -3.994 8.07e-05 \*\*\*

v500 2.248e-05 4.906e-04 0.046 0.96348

wind -4.088e-03 1.127e-02 -0.363 0.71694

hum 5.023e-03 1.713e-03 2.933 0.00360 \*\*

safb 2.977e-02 4.523e-03 6.582 1.90e-10 \*\*\*

inbh -8.692e-05 2.683e-05 -3.240 0.00132 \*\*

dagg 5.096e-04 1.027e-03 0.496 0.62007

inbt 4.448e-04 1.238e-03 0.359 0.71956

vis -8.484e-04 3.416e-04 -2.484 0.01352 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

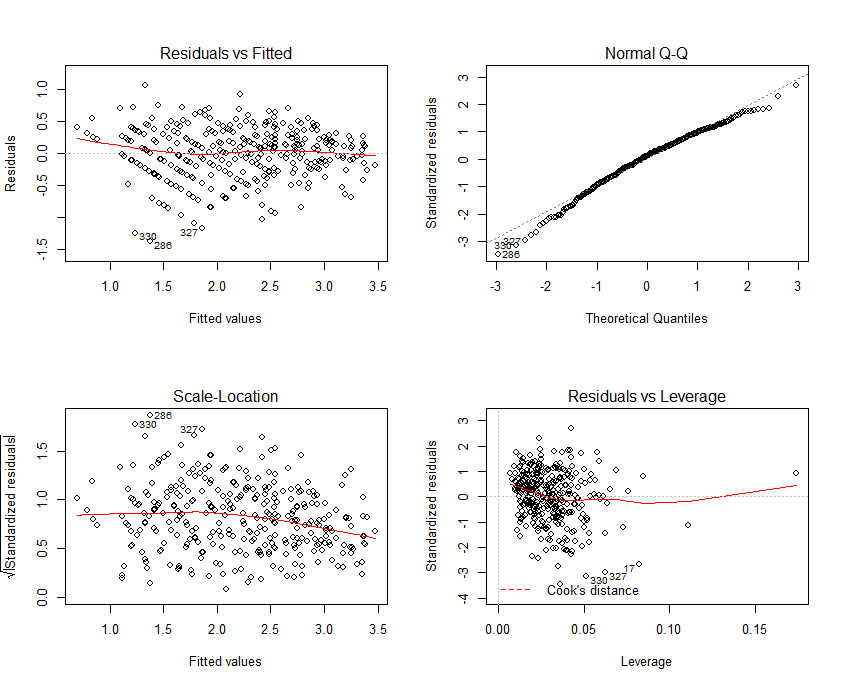
Residual standard error: 0.4039 on 320 degrees of freedom

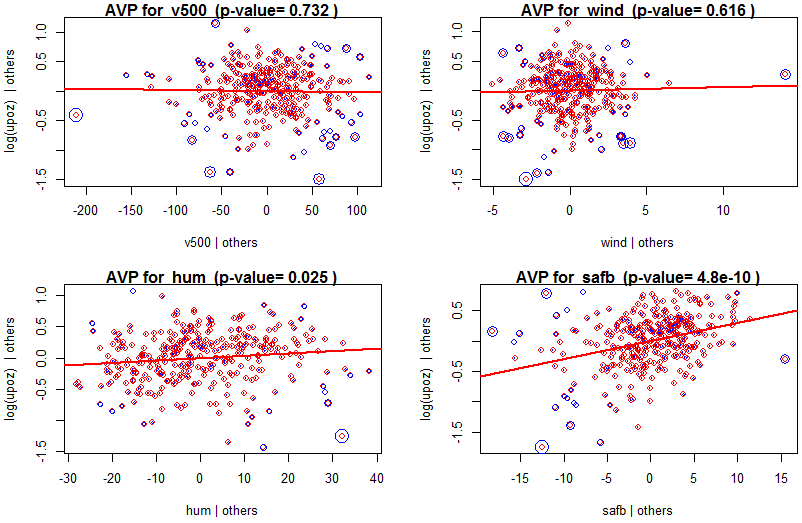
Multiple R-squared: 0.7167, Adjusted R-squared: 0.7088

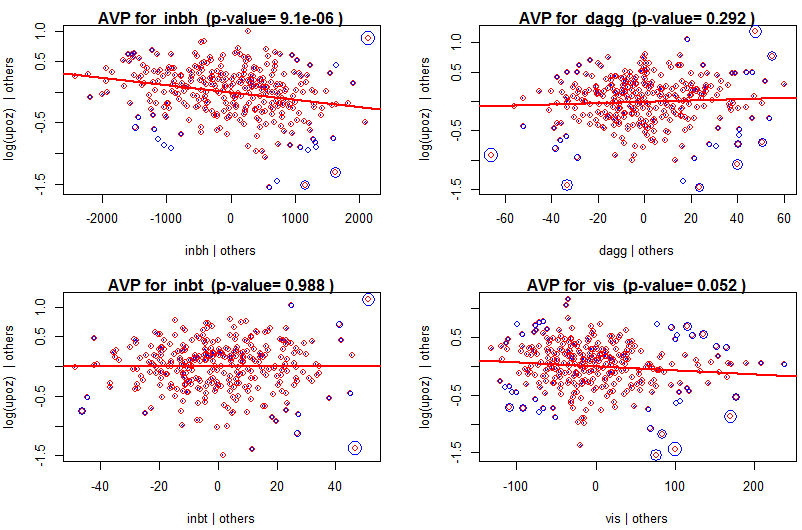
F-statistic: 89.96 on 9 and 320 DF, p-value: < 2.2e-16

> par(mfrow=c(2,2))

> plot(oz.lm3)

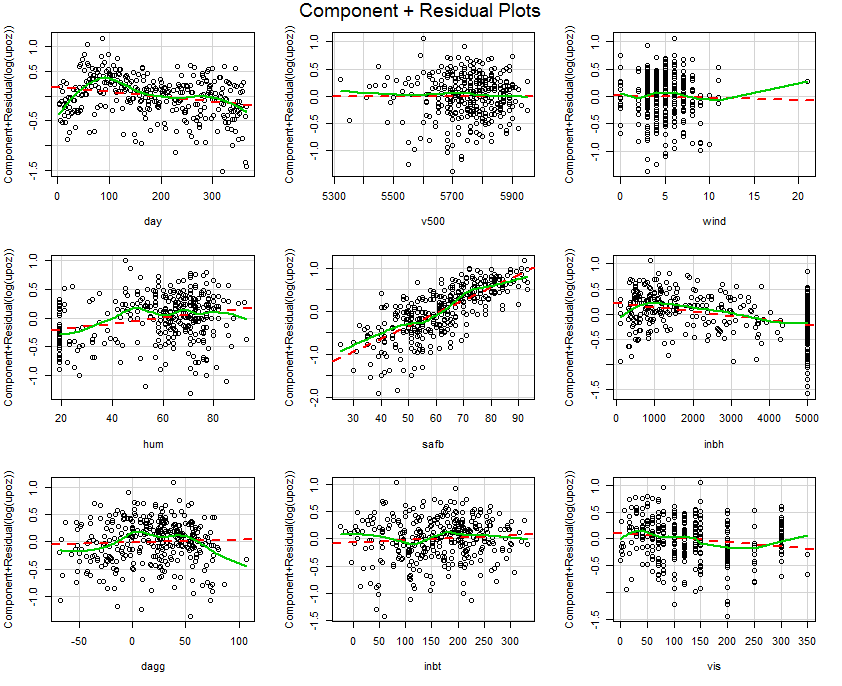


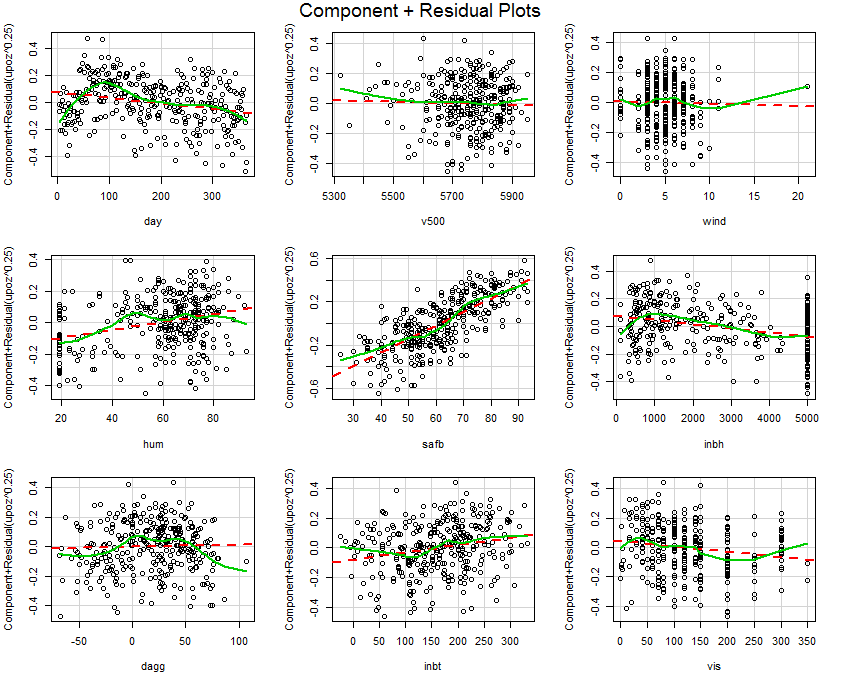
**AVPs** -AVPs(oz.lm3)  
 **C+R Plots**



To obtain component-plus-residual plots (or partial residual plots) in R we can use the function crPlots in the car package. We will examine C+R plots for the predictors in both of the models using as the response.

> crPlots(oz.lm2) > crPlots(oz.lm3)





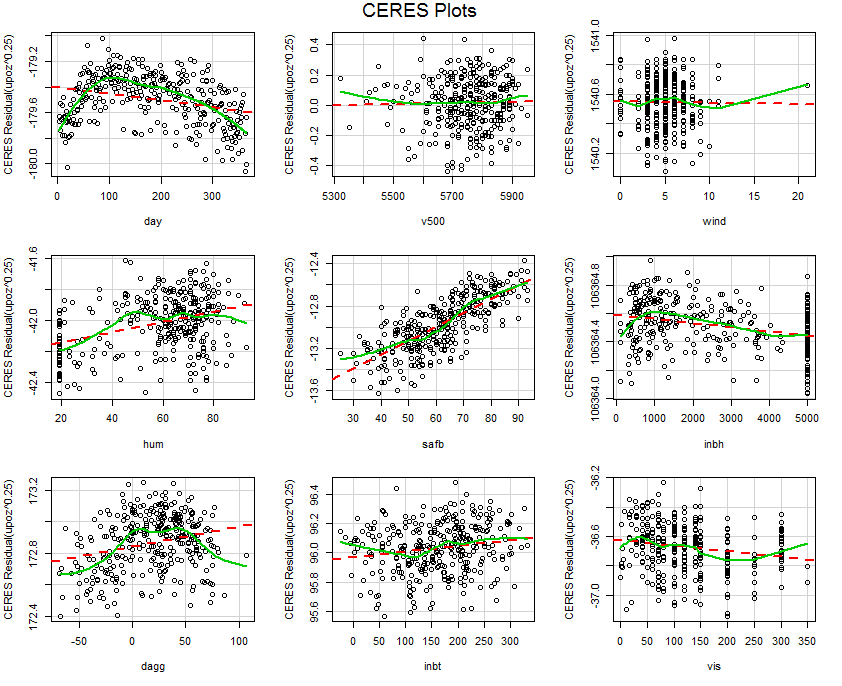
Do these plots suggest any terms to allow for curvature should be added to the model?

If so, for which predictors?

**CERES Plots**

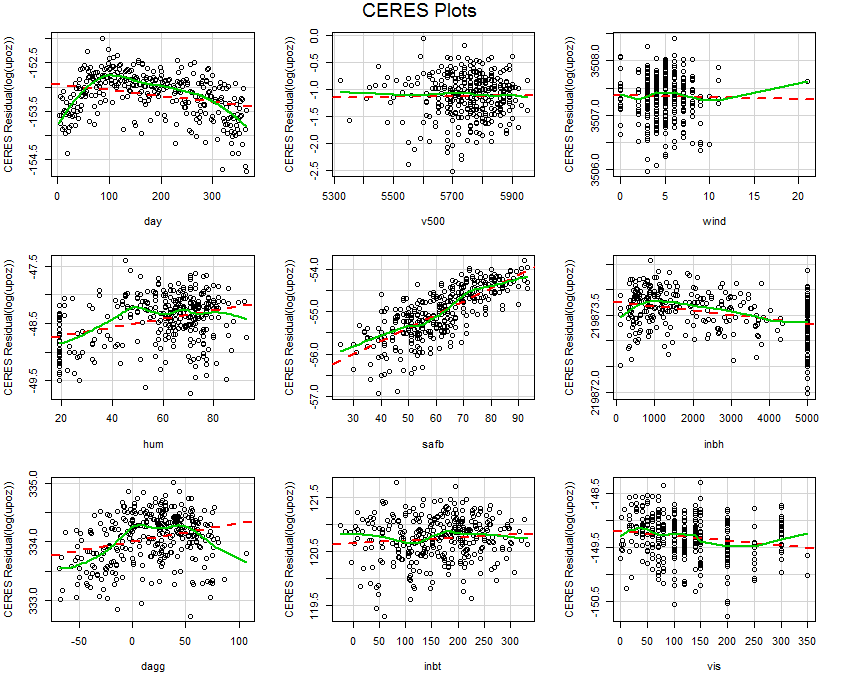
Finally to obtain CERES plots for all of the predictors (dummy terms and factors excluded) use the function ceresPlots in the car package. Again we consider plots for all of the predictors from both the and models.

> ceresPlots(oz.lm2)



These plots suggest some curvature in the following predictors:

> ceresPlots(oz.lm3)



Before transforming we may want to eliminate predictors that do not appear to be important in the current models – oz.lm2 and oz.lm3. Transforming terms that do not have strong adjusted relationships with the response does not make much sense, however if the curvature is strong in the adjusted relationship with the response, the tests for the terms/predictors entered linearly may not be significant.

Using **Backward Elimination** to reduce the models using the step function.

> oz.step2 = step(oz.lm2) 🡨 Perform backwards elimination and save resulting model in oz.step2

Start: AIC=-1162.13

upoz^0.25 ~ day + v500 + wind + hum + safb + inbh + dagg + inbt +

vis

Df Sum of Sq RSS AIC

- v500 1 0.00151 9.1802 -1164.1

- wind 1 0.00336 9.1820 -1164.0

- dagg 1 0.00440 9.1831 -1164.0

- inbt 1 0.02434 9.2030 -1163.3

<none> 9.1787 -1162.1

- vis 1 0.17800 9.3567 -1157.8

- inbh 1 0.21406 9.3927 -1156.5

- hum 1 0.35742 9.5361 -1151.5

- day 1 0.43208 9.6108 -1149.0

- safb 1 1.22495 10.4036 -1122.8

Step: AIC=-1164.07

upoz^0.25 ~ day + wind + hum + safb + inbh + dagg + inbt + vis

Df Sum of Sq RSS AIC

- wind 1 0.00263 9.1828 -1166.0

- dagg 1 0.00473 9.1849 -1165.9

- inbt 1 0.02395 9.2041 -1165.2

<none> 9.1802 -1164.1

- vis 1 0.17658 9.3568 -1159.8

- inbh 1 0.24684 9.4270 -1157.3

- hum 1 0.36871 9.5489 -1153.1

- day 1 0.44145 9.6216 -1150.6

- safb 1 1.26191 10.4421 -1123.6

. . . Etc . . .

Step: AIC=-1169.14

upoz^0.25 ~ day + hum + safb + inbh + vis

Df Sum of Sq RSS AIC

<none> 9.2064 -1169.14

- vis 1 0.1982 9.4046 -1164.11

- day 1 0.4668 9.6732 -1154.81

- hum 1 0.6373 9.8436 -1149.05

- inbh 1 1.0622 10.2686 -1135.10

- safb 1 7.8454 17.0517 -967.74

**Backward Elimination Model** (  
> summary(oz.step2)

Coefficients:

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

(Intercept) 9.910e-01 6.775e-02 14.628 < 2e-16 \*\*\*

day -3.829e-04 9.447e-05 -4.053 6.33e-05 \*\*\*

hum 2.490e-03 5.258e-04 4.736 3.27e-06 \*\*\*

safb 1.377e-02 8.288e-04 16.616 < 2e-16 \*\*\*

inbh -3.950e-05 6.460e-06 -6.114 2.79e-09 \*\*\*

vis -3.704e-04 1.402e-04 -2.641 0.00866 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1686 on 324 degrees of freedom

Multiple R-squared: 0.7316, Adjusted R-squared: 0.7274

F-statistic: 176.6 on 5 and 324 DF, p-value: < 2.2e-16

**Big F-Test for Comparing Nested Models**

> anova(oz.step2,oz.lm2)

Analysis of Variance Table

Model 1: upoz^0.25 ~ day + hum + safb + inbh + vis

Model 2: upoz^0.25 ~ day + v500 + wind + hum + safb + inbh + dagg + inbt +

vis

Res.Df RSS Df Sum of Sq F Pr(>F)

1 324 9.2064

2 320 9.1787 4 0.027674 0.2412 0.9149

**Backward Elimination Model**   
  
> oz.step3 = step(oz.lm3)  
> summary(oz.step3)

Call:

lm(formula = log(upoz) ~ day + hum + safb + inbh + vis, data = Ozone)

Residuals:

Min 1Q Median 3Q Max

-1.35533 -0.24427 0.04726 0.27241 1.09920

Coefficients:

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

(Intercept) 4.906e-01 1.614e-01 3.040 0.00256 \*\*

day -9.704e-04 2.251e-04 -4.311 2.16e-05 \*\*\*

hum 5.237e-03 1.253e-03 4.181 3.75e-05 \*\*\*

safb 3.151e-02 1.975e-03 15.958 < 2e-16 \*\*\*

inbh -9.466e-05 1.539e-05 -6.150 2.28e-09 \*\*\*

vis -8.694e-04 3.341e-04 -2.602 0.00969 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4016 on 324 degrees of freedom

Multiple R-squared: 0.7164, Adjusted R-squared: 0.712

F-statistic: 163.7 on 5 and 324 DF, p-value: < 2.2e-16

**Big F-Test Again**> anova(oz.step3,oz.lm3)

Analysis of Variance Table

Model 1: log(upoz) ~ day + hum + safb + inbh + vis

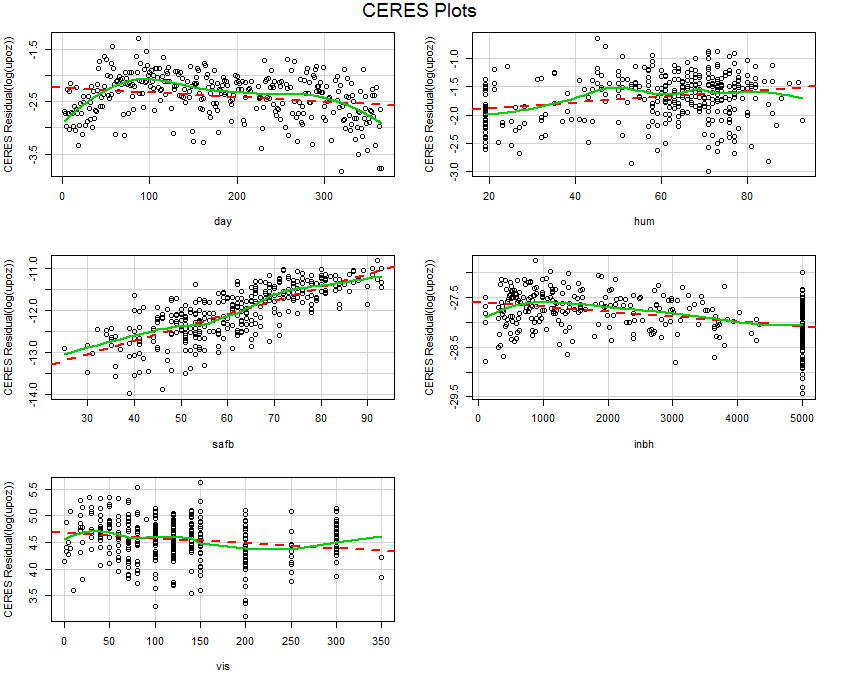
Model 2: log(upoz) ~ day + v500 + wind + hum + safb + inbh + dagg + inbt +

vis

Res.Df RSS Df Sum of Sq F Pr(>F)

1 324 52.258

2 320 52.193 4 0.065369 0.1002 0.9823

> ceresPlots(oz.step3)  


What terms would we use to capture the curvature exhibited in the CERES plots?

**Polynomial Regression Models**

> oz.poly = lm(log(upoz)~poly(day,3)+poly(hum,3)+poly(safb,2)+poly(inbh,3)+poly(vis,2))

> summary(oz.poly)

Coefficients:

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

(Intercept) 2.21297 0.02005 110.362 < 2e-16 \*\*\*

poly(day, 3)1 -1.42148 0.41104 -3.458 0.000618 \*\*\*

poly(day, 3)2 -3.39569 0.60695 -5.595 4.79e-08 \*\*\*

poly(day, 3)3 0.52938 0.40930 1.293 0.196829

poly(hum, 3)1 0.04364 0.49483 0.088 0.929773

poly(hum, 3)2 -0.99107 0.39462 -2.511 0.012523 \*

poly(hum, 3)3 0.10092 0.37677 0.268 0.788976

poly(safb, 2)1 6.36727 0.66914 9.516 < 2e-16 \*\*\*

poly(safb, 2)2 -0.34120 0.38294 -0.891 0.373608

poly(inbh, 3)1 -3.36160 0.48978 -6.864 3.56e-11 \*\*\*

poly(inbh, 3)2 -1.11331 0.39653 -2.808 0.005301 \*\*

poly(inbh, 3)3 0.86415 0.37636 2.296 0.022325 \*

poly(vis, 2)1 -1.84876 0.44941 -4.114 4.97e-05 \*\*\*

poly(vis, 2)2 1.51998 0.38730 3.925 0.000107 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3643 on 316 degrees of freedom

Multiple R-squared: 0.7724, Adjusted R-squared: 0.7631

F-statistic: 82.51 on 13 and 316 DF, p-value: < 2.2e-16

> vif(oz.poly)

GVIF Df GVIF^(1/(2\*Df))

poly(day, 3) 3.831530 3 1.250918

poly(hum, 3) 2.299601 3 1.148882

poly(safb, 2) 3.724661 2 1.389222

poly(inbh, 3) 2.272160 3 1.146586

poly(vis, 2) 1.693325 2 1.140736

As the squared terms for SAFB is not significant we will drop it.

> oz.poly = lm(log(upoz)~poly(day,3)+poly(hum,2)+safb+poly(inbh,3)+poly(vis,2))

> summary(oz.poly)

Coefficients:

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

(Intercept) 0.711720 0.158007 4.504 9.36e-06 \*\*\*

poly(day, 3)1 -1.390163 0.407536 -3.411 0.000730 \*\*\*

poly(day, 3)2 -3.346264 0.602713 -5.552 5.96e-08 \*\*\*

poly(day, 3)3 0.544262 0.407939 1.334 0.183102

poly(hum, 2)1 0.023733 0.493335 0.048 0.961660

poly(hum, 2)2 -0.961576 0.392633 -2.449 0.014863 \*

safb 0.024310 0.002538 9.578 < 2e-16 \*\*\*

poly(inbh, 3)1 -3.387331 0.487187 -6.953 2.04e-11 \*\*\*

poly(inbh, 3)2 -1.152110 0.385206 -2.991 0.002999 \*\*

poly(inbh, 3)3 0.816011 0.372021 2.193 0.028998 \*

poly(vis, 2)1 -1.846787 0.446857 -4.133 4.59e-05 \*\*\*

poly(vis, 2)2 1.498135 0.385010 3.891 0.000122 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3636 on 318 degrees of freedom

Multiple R-squared: 0.7718, Adjusted R-squared: 0.7639

F-statistic: 97.78 on 11 and 318 DF, p-value: < 2.2e-16

As the only the squared term for humidity is significant we can create a term and use it instead of the full quadratic.

> hum2 = hum^2

> oz.poly = lm(log(upoz)~poly(day,3)+hum2+safb+poly(inbh,3)+poly(vis,2))

> summary(oz.poly)

Coefficients:

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

(Intercept) 8.404e-01 1.692e-01 4.966 1.12e-06 \*\*\*

poly(day, 3)1 -1.445e+00 4.098e-01 -3.526 0.000483 \*\*\*

poly(day, 3)2 -3.814e+00 5.731e-01 -6.655 1.24e-10 \*\*\*

poly(day, 3)3 5.428e-01 4.108e-01 1.321 0.187350

hum2 -8.905e-06 1.316e-05 -0.677 0.499094

safb 2.277e-02 2.470e-03 9.219 < 2e-16 \*\*\*

poly(inbh, 3)1 -3.615e+00 4.809e-01 -7.516 5.77e-13 \*\*\*

poly(inbh, 3)2 -1.219e+00 3.869e-01 -3.152 0.001777 \*\*

poly(inbh, 3)3 6.825e-01 3.703e-01 1.843 0.066244 .

poly(vis, 2)1 -1.944e+00 4.481e-01 -4.338 1.93e-05 \*\*\*

poly(vis, 2)2 1.469e+00 3.876e-01 3.791 0.000179 \*\*\*

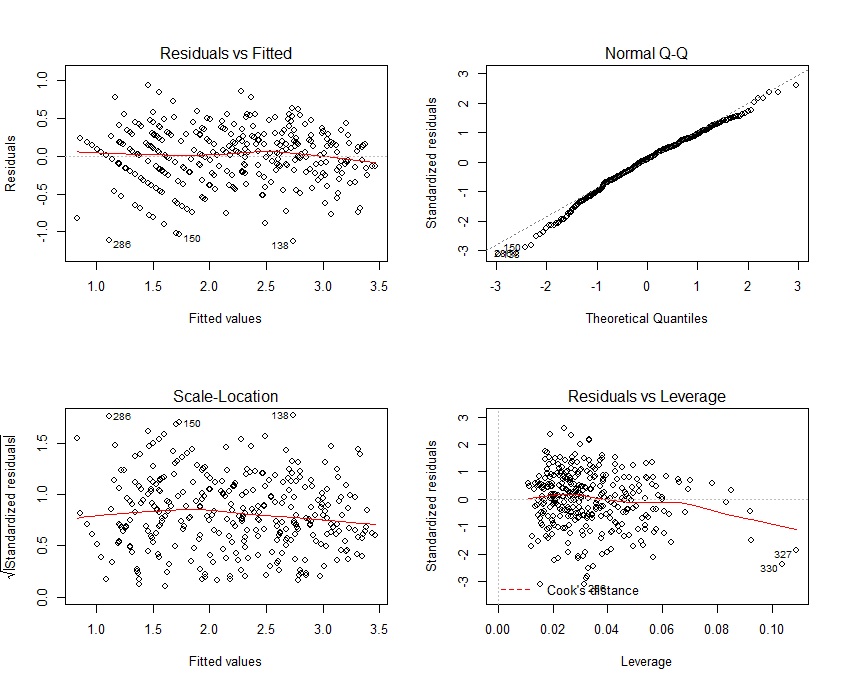
---

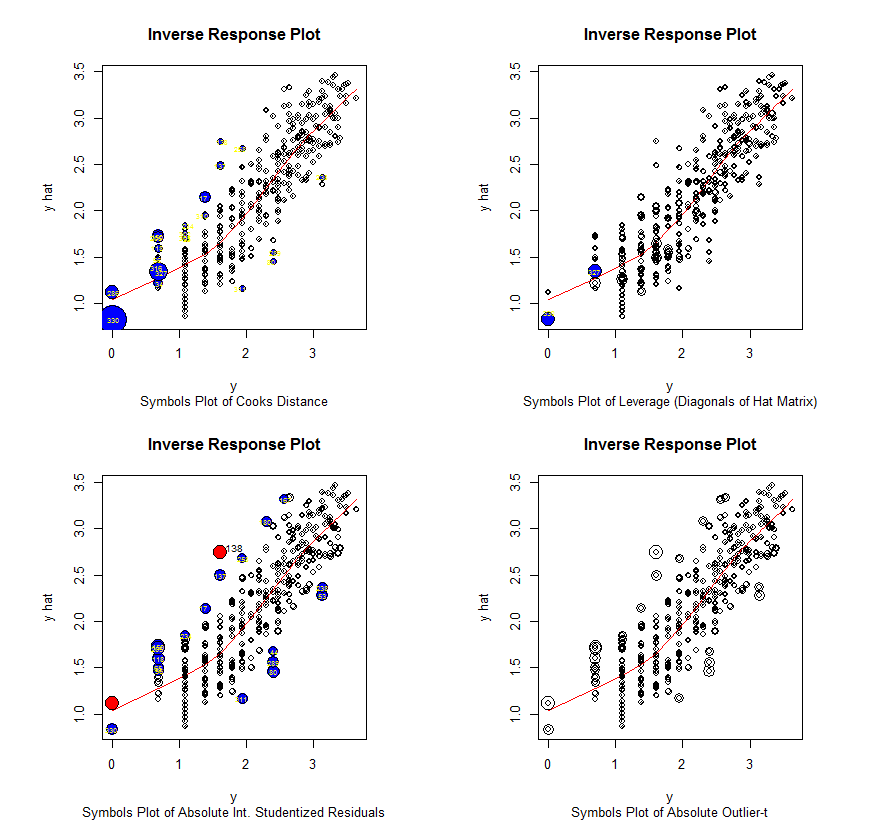
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3662 on 319 degrees of freedom

Multiple R-squared: 0.7678, Adjusted R-squared: 0.7606

F-statistic: 105.5 on 10 and 319 DF, p-value: < 2.2e-16

> plot(oz.poly)  
  
  
> MLRdiag(oz.poly)



Try model with

> oz.poly2 = update(oz.poly,upoz^0.25~.,data=Ozone)

> summary(oz.poly2)

Coefficients:

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

(Intercept) 1.118e+00 7.143e-02 15.658 < 2e-16 \*\*\*

poly(day, 3)1 -5.794e-01 1.729e-01 -3.350 0.000904 \*\*\*

poly(day, 3)2 -1.509e+00 2.419e-01 -6.239 1.40e-09 \*\*\*

poly(day, 3)3 3.022e-01 1.734e-01 1.743 0.082325 .

hum2 1.442e-06 5.554e-06 0.260 0.795352

safb 1.045e-02 1.042e-03 10.021 < 2e-16 \*\*\*

poly(inbh, 3)1 -1.478e+00 2.030e-01 -7.280 2.63e-12 \*\*\*

poly(inbh, 3)2 -3.768e-01 1.633e-01 -2.308 0.021640 \*

poly(inbh, 3)3 3.694e-01 1.563e-01 2.364 0.018677 \*

poly(vis, 2)1 -8.102e-01 1.891e-01 -4.284 2.44e-05 \*\*\*

poly(vis, 2)2 6.063e-01 1.636e-01 3.707 0.000248 \*\*\*

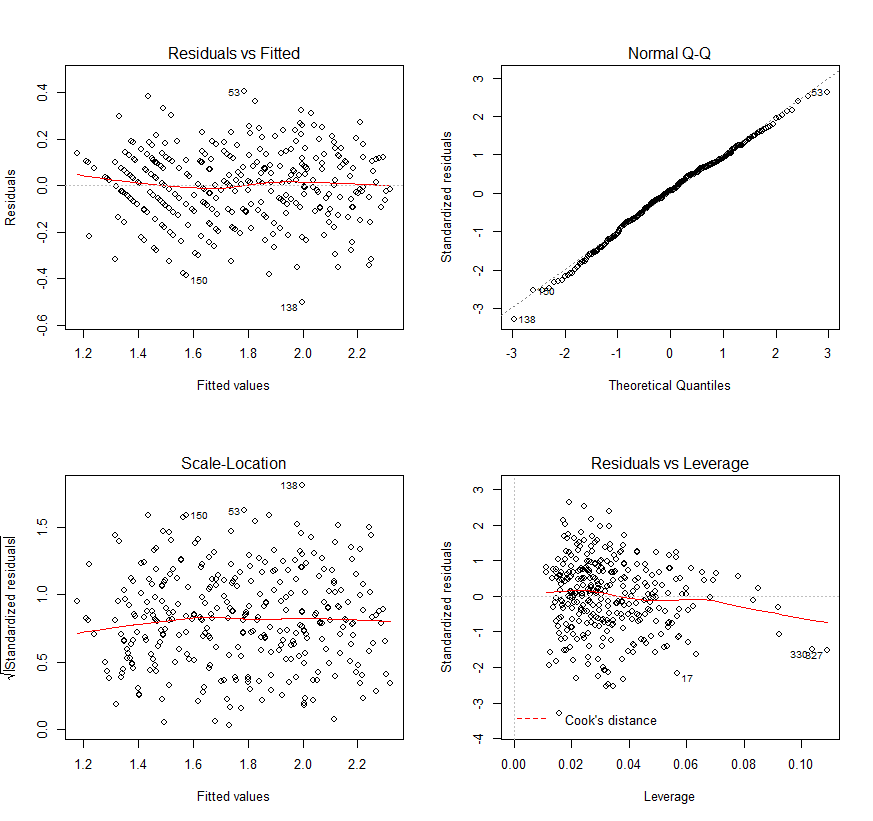
---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1545 on 319 degrees of freedom

Multiple R-squared: 0.7778, Adjusted R-squared: 0.7709

F-statistic: 111.7 on 10 and 319 DF, p-value: < 2.2e-16  
 **>** plot(oz.poly2)



> MLRdiag(oz.poly2)

