NCSU ST 503 HW 6

Probems 8.5, 9.4, 9.5 9.6 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

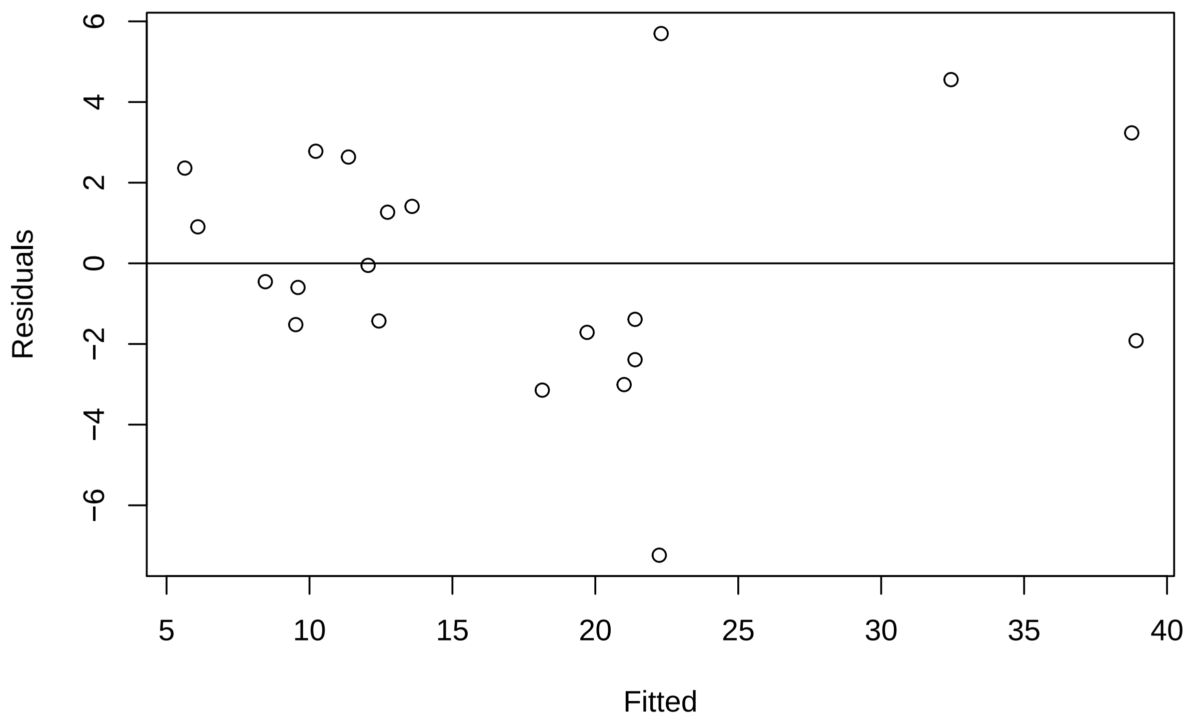
23 October, 2017

8.5 Comparing model fitting methods with the stackloss data

Using the stackloss data, fit a model with stack.loss as the response and the other three variables as predictors using the following methods:

(a) Least squares

```
##
## Call:
## lm(formula = stack.loss ~ ., data = stackloss)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2377 -1.7117 -0.4551  2.3614  5.6978
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -39.9197    11.8960  -3.356  0.00375 **
## Air.Flow      0.7156     0.1349   5.307  5.8e-05 ***
## Water.Temp    1.2953     0.3680   3.520  0.00263 **
## Acid.Conc.   -0.1521     0.1563  -0.973  0.34405
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.243 on 17 degrees of freedom
## Multiple R-squared:  0.9136, Adjusted R-squared:  0.8983
## F-statistic:  59.9 on 3 and 17 DF,  p-value: 3.016e-09
```



we see there may be an association with the variance of the residuals and the value of the response.

(b) Least absolute deviations

We use the quantreg::rq method for the L^1 regression. Its worth reading the details of the algorithmic methods for computing the fit here. Also worthy of note is that `quantrg::rq` provides a lasso option for sparse regression.

```
##
## Call: rq(formula = stack.loss ~ ., data = stackloss)
##
## tau: [1] 0.5
##
## Coefficients:
##             coefficients lower bd  upper bd
## (Intercept) -39.68986    -41.61973 -29.67754
## Air.Flow      0.83188      0.51278   1.14117
## Water.Temp    0.57391      0.32182   1.41090
## Acid.Conc.   -0.06087     -0.21348  -0.02891
```

(c) Huber method

We use the MASS::rlm() function to fit the model with the Huber loss.

```
##
## Call: rlm(formula = stack.loss ~ ., data = stackloss)
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.91753 -1.73127  0.06187  1.54306  6.50163
##
## Coefficients:
##             Value    Std. Error t value
## (Intercept) -41.0265   9.8073    -4.1832
## Air.Flow      0.8294   0.1112     7.4597
## Water.Temp    0.9261   0.3034     3.0524
## Acid.Conc.   -0.1278   0.1289    -0.9922
##
## Residual standard error: 2.441 on 17 degrees of freedom
```

```
##        21         4         3         1         2         5         6
## 0.3681411 0.5049409 0.7858871 1.0000000 1.0000000 1.0000000 1.0000000
##         7         8         9
## 1.0000000 1.0000000 1.0000000
```

We see that `21 4 and 3` have weights less than 1. We will investigate these points in our diagnostics later.

(d) Least trimmed squares Compare the results.

```
##   (Intercept)      Air.Flow    Water.Temp    Acid.Conc.
## -3.429167e+01  7.142857e-01  3.571429e-01  3.588783e-16
```

Now use diagnostic methods to detect any outliers or influential points. Remove these points and then use least squares. Compare the results.

Check Leverage

Table 1: High Leverage Data Elements

	Air.Flow	Water.Temp	Acid.Conc.	stack.loss
17	50	19	72	8

We've used the rule of thumb that points with a leverage greater than $\frac{2p}{n}$ should be looked at.

Check for outliers.

Table 2: Range of Studentized residuals

range.residuals.left	range.residuals.right
-3.33	2.052

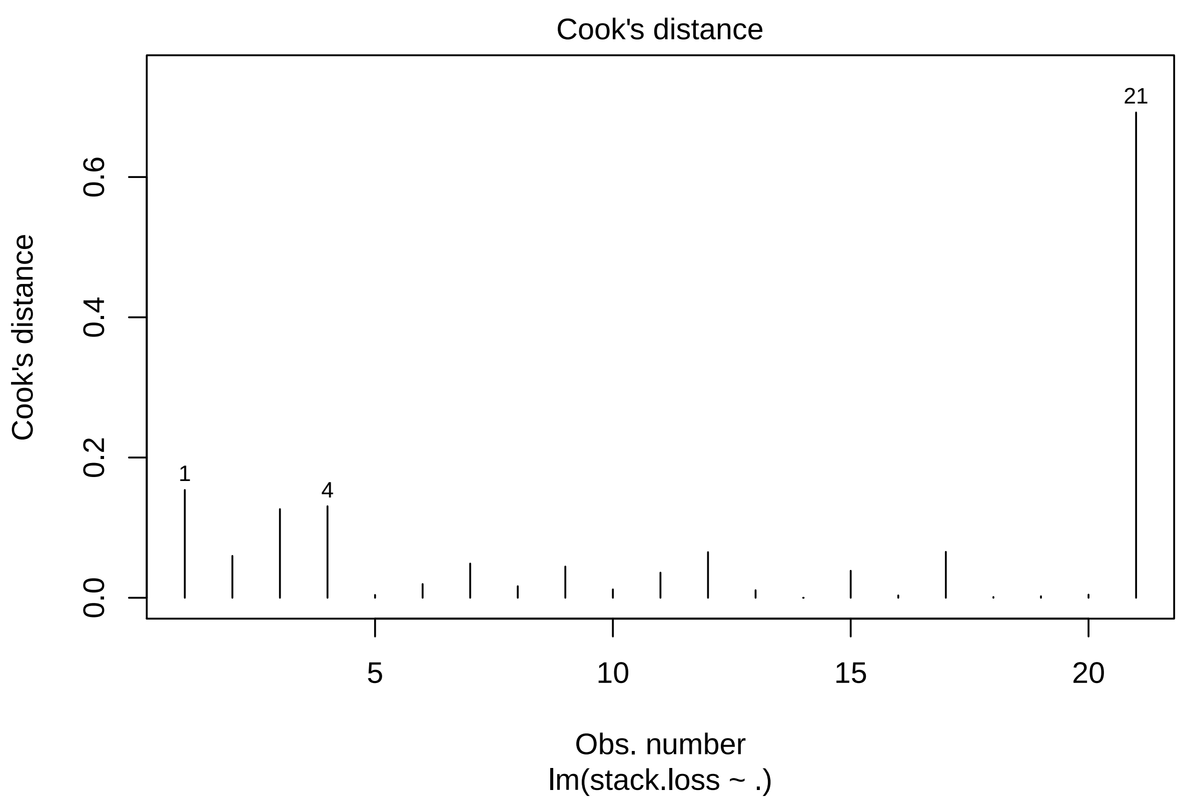
Table 3: Bonferroni corrected t-value

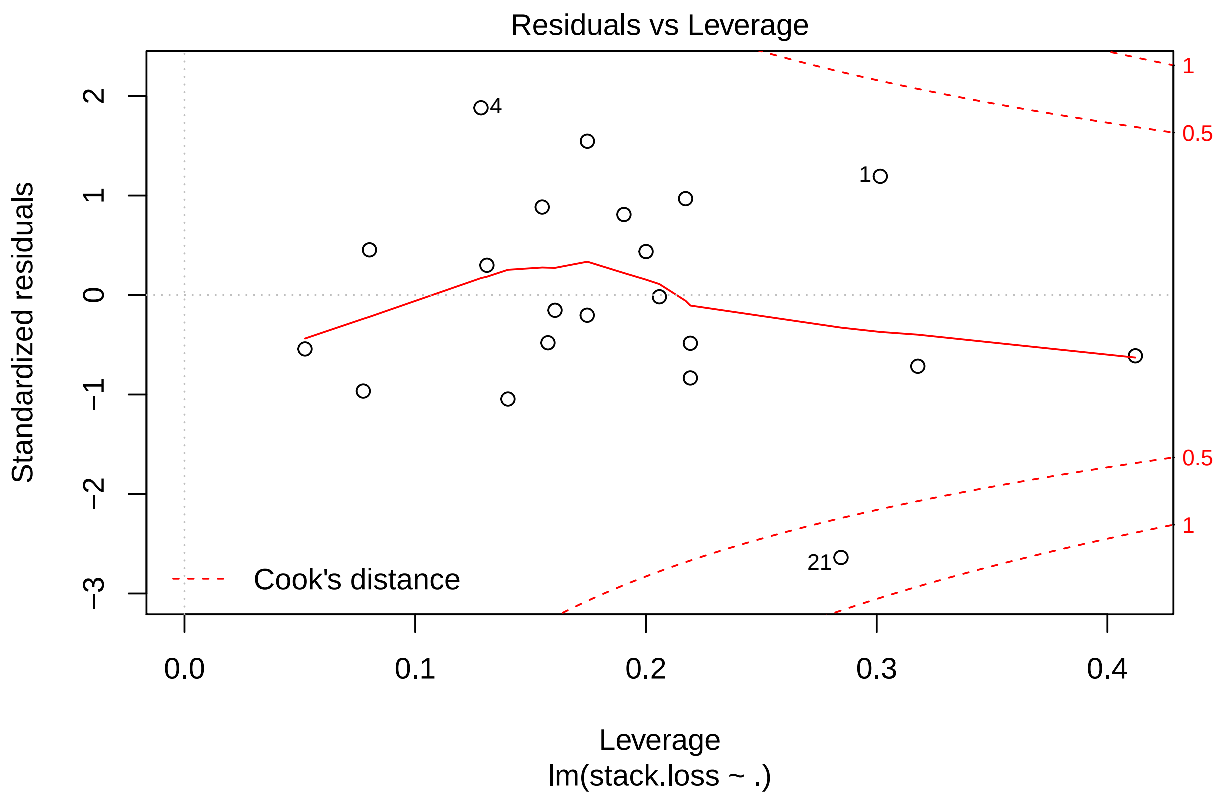
t.val.alpha
-3.604

Here we look for studentized residuals that fall outside the interval given by the Bonferroni corrected t-values.

Check for influential points.

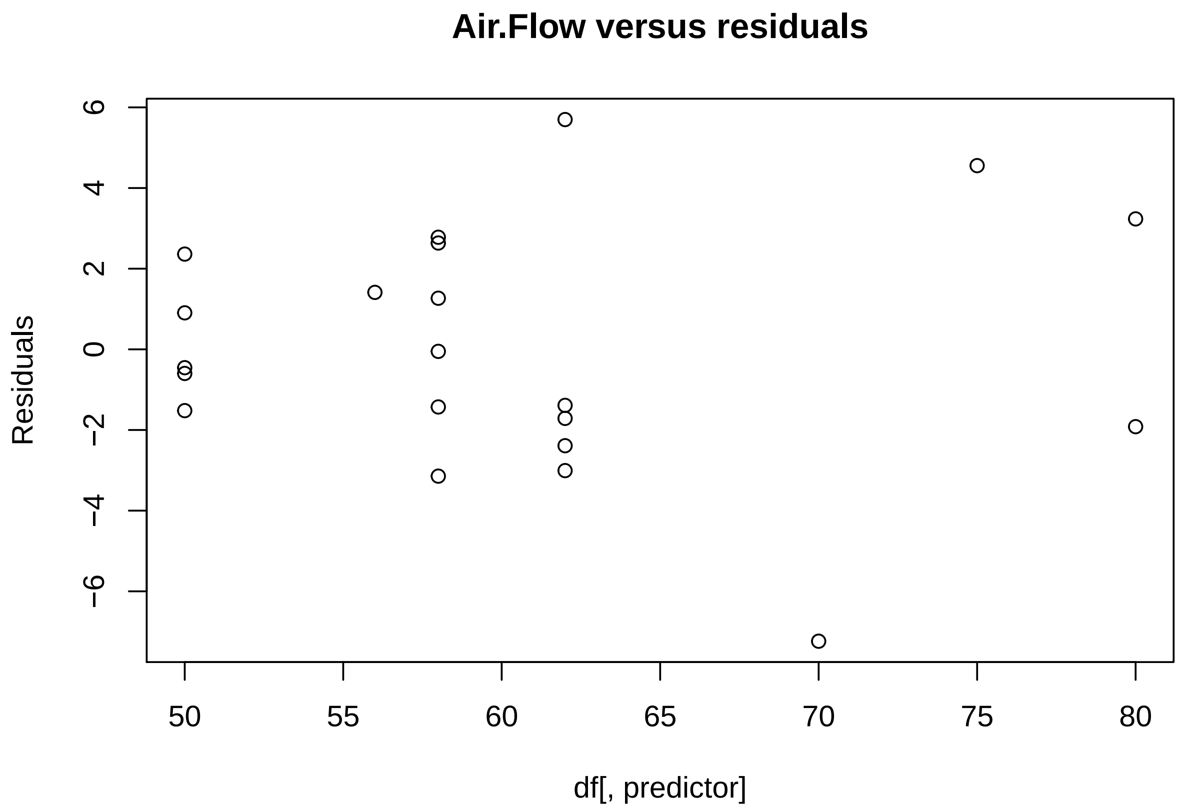
We plot the Cook's distances and the residual-leverage plot with level set contours of the Cook distance.

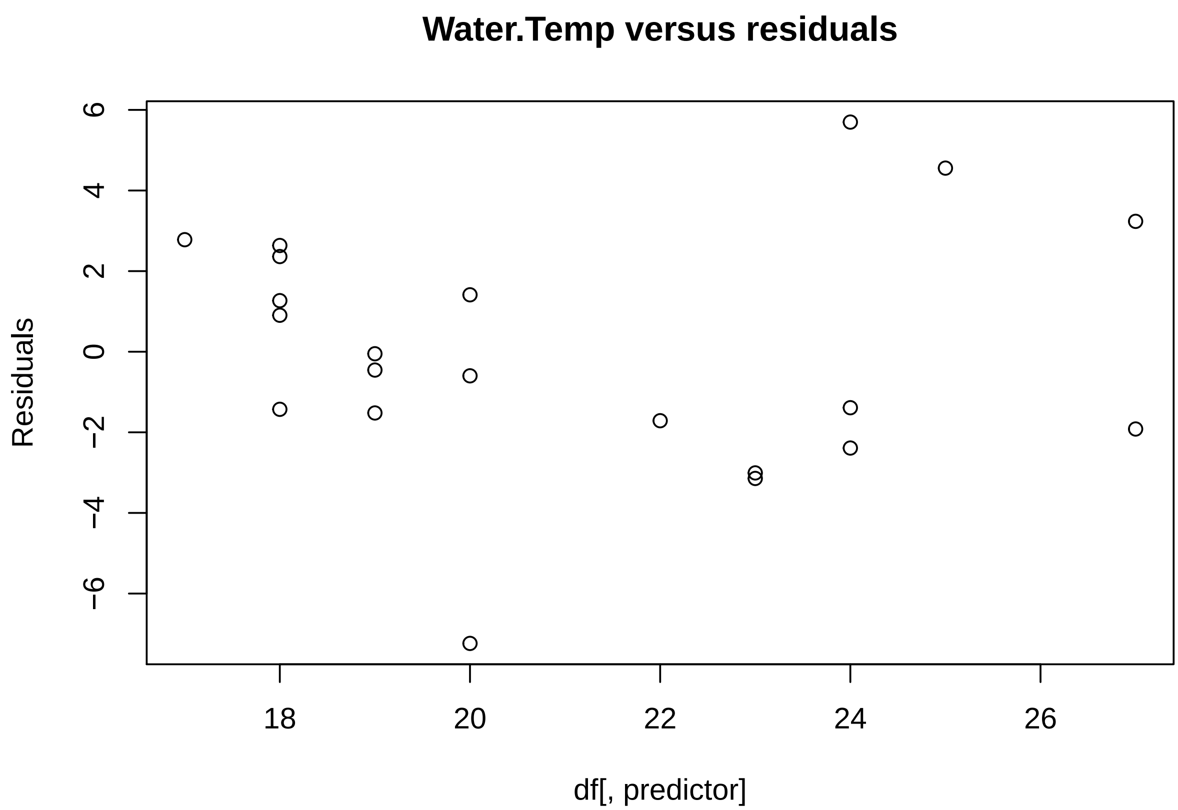


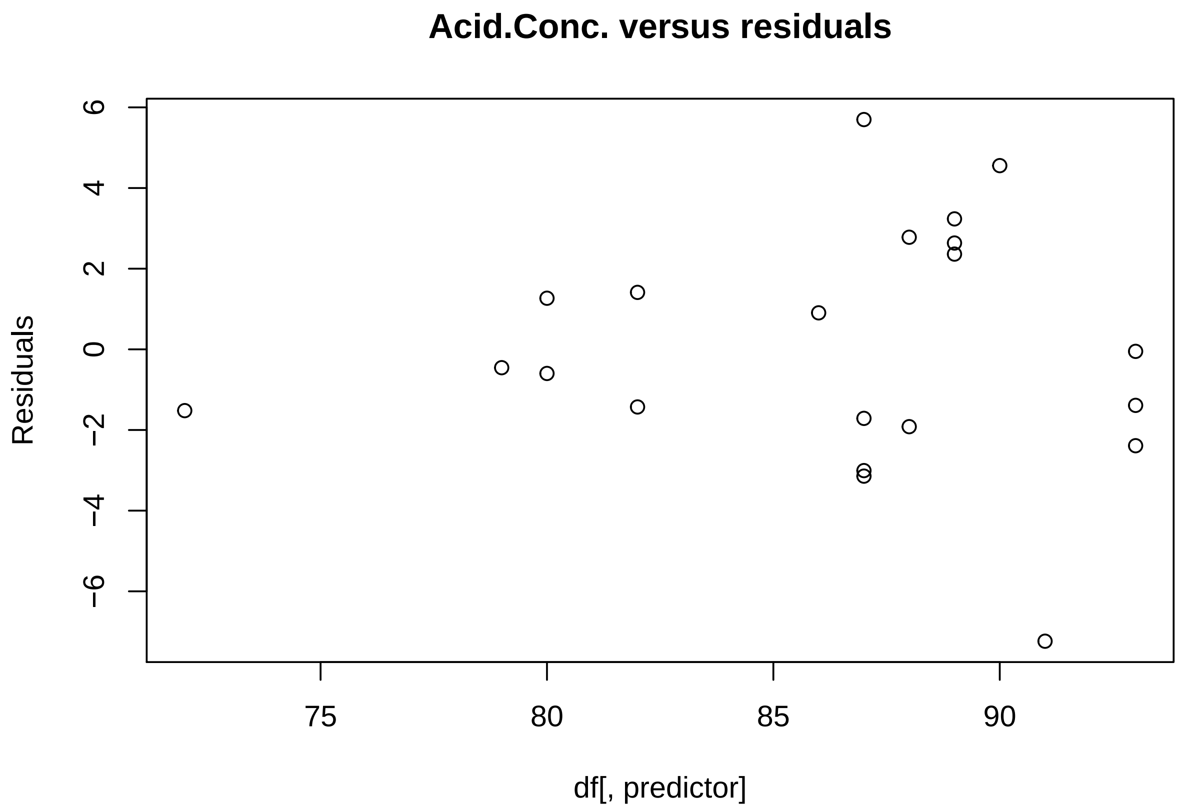


Check for structure in the model.

Plot residuals versus predictors

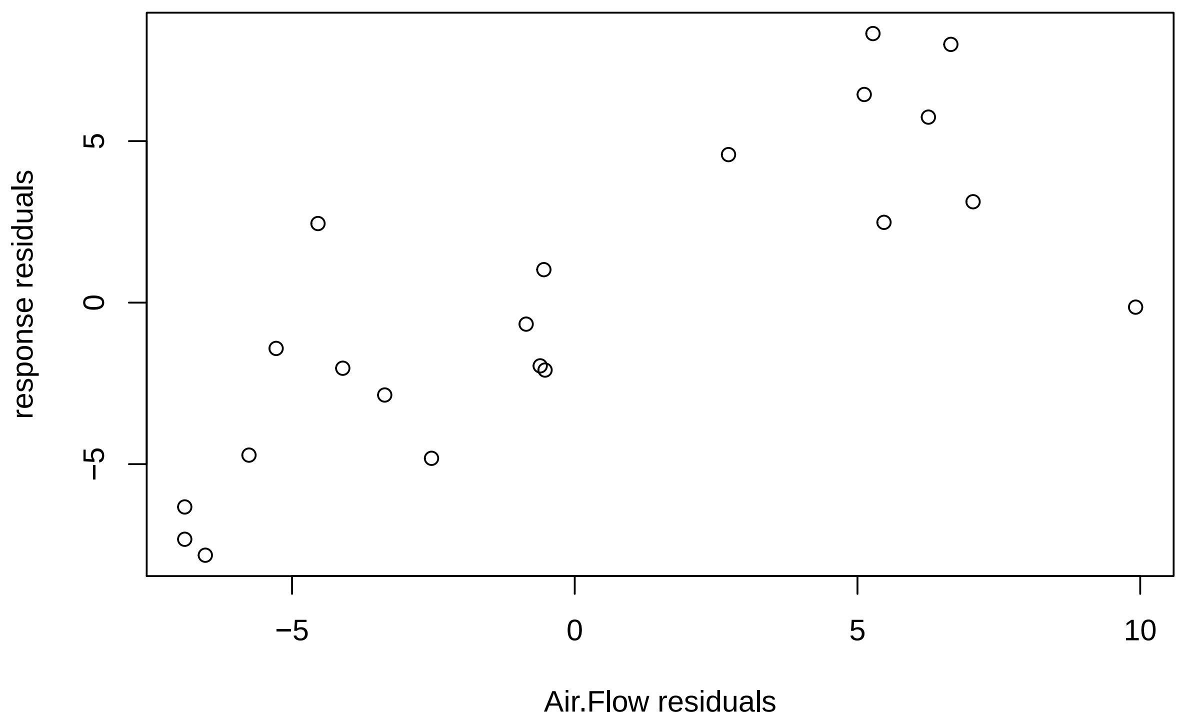

Air.Flow versus residuals
Residuals
6
4
2
0
−2
−4
−6
50
55
60
65
70
75
80
df[, predictor]


Water.Temp versus residuals
Residuals
6
4
2
0
−2
−4
−6
18
20
22
24
26
df[, predictor]

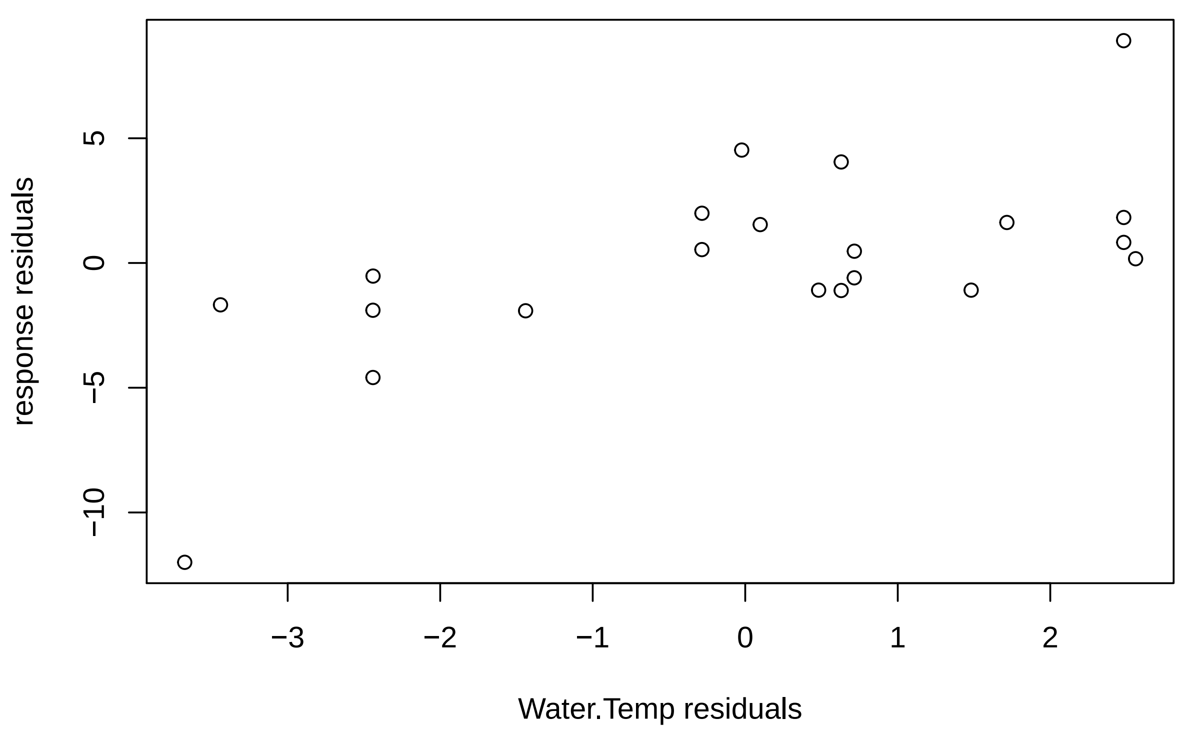


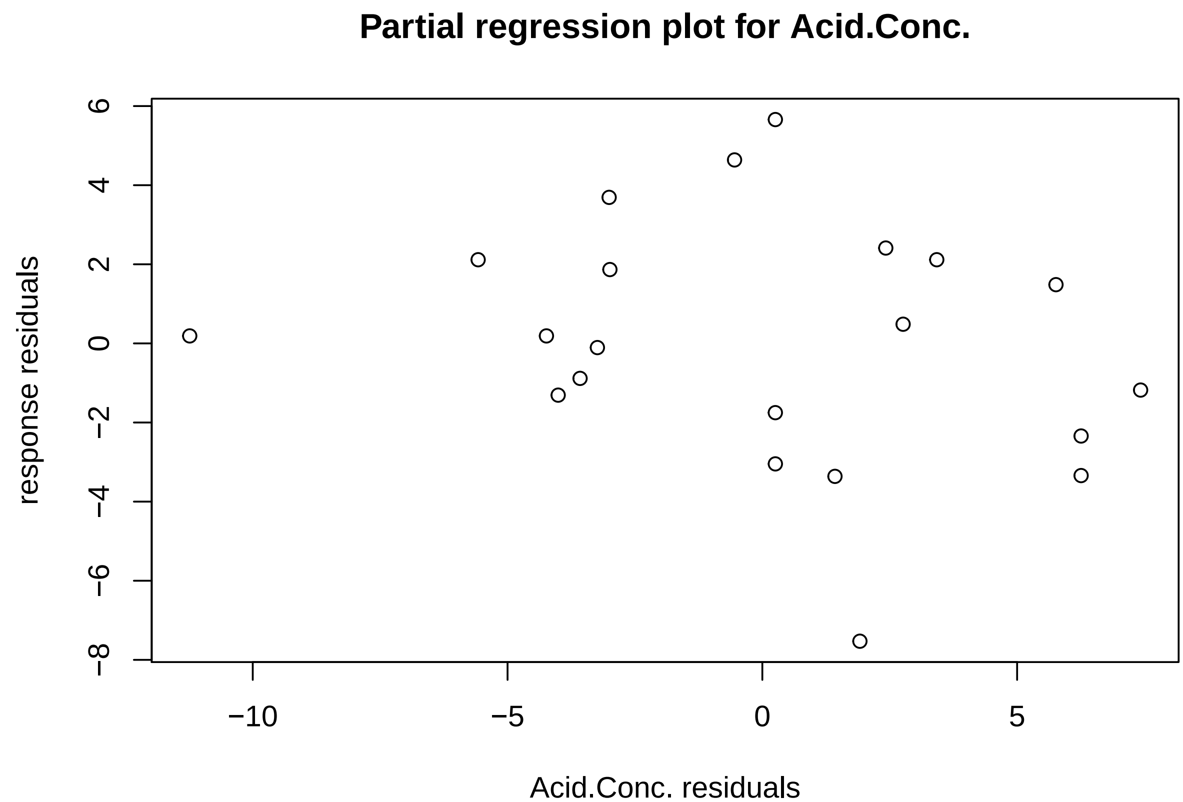
Perform partial regression

Partial regression plot for Air.Flow



Partial regression plot for Water.Temp



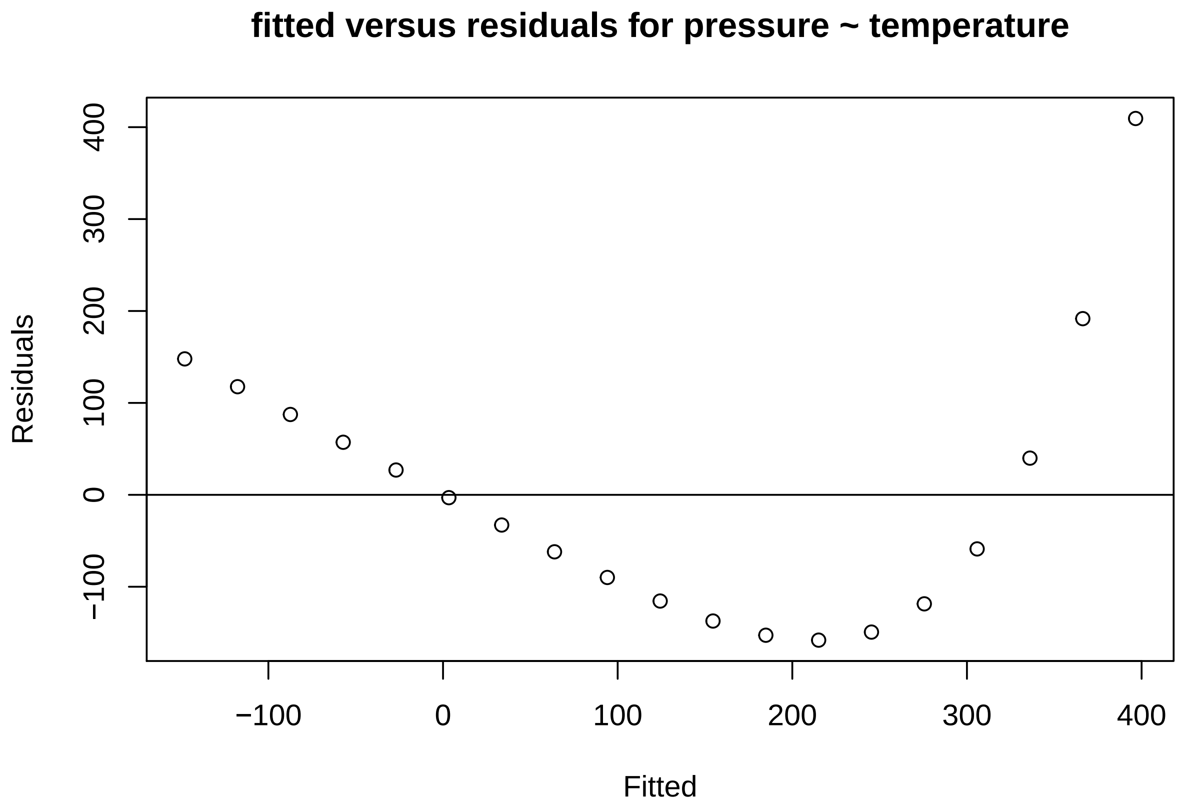


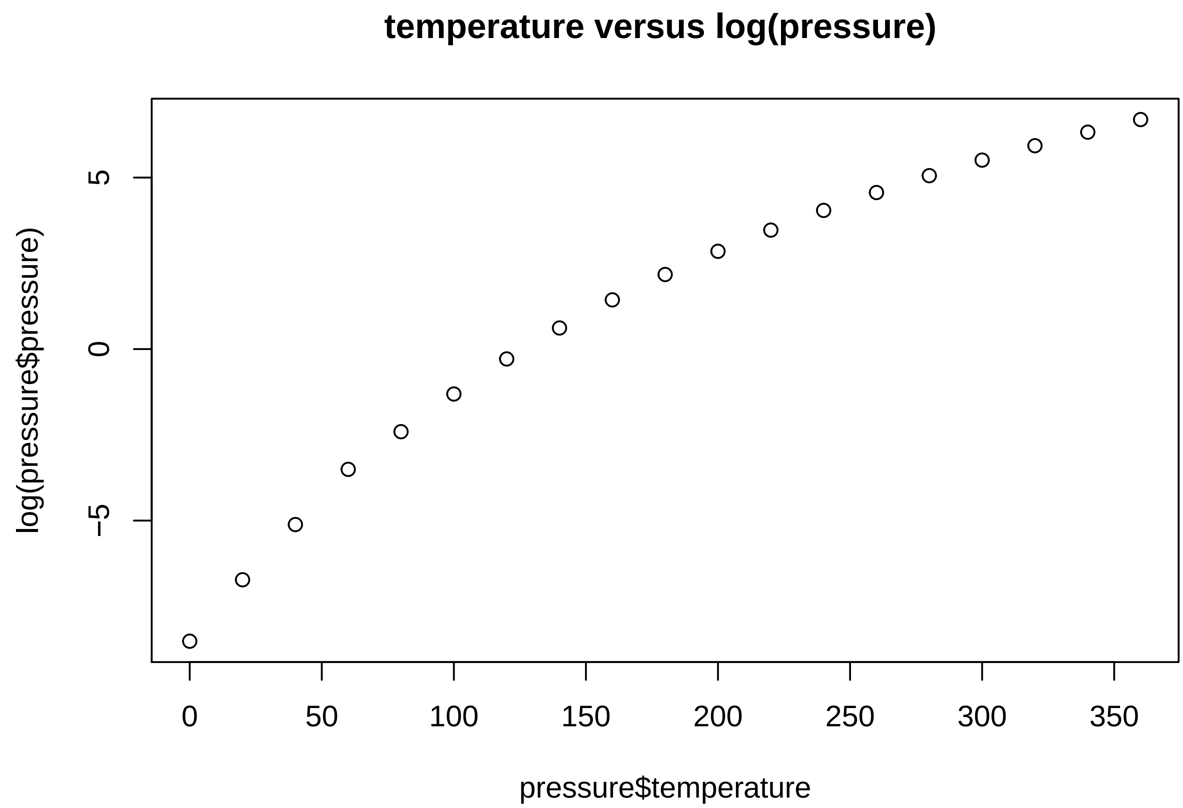
9.4 Using transformations in model of pressure data

Use the pressure data to fit a model with pressure as the response and temperature as the predictor using transformations to obtain a good fit.

```
##
## Call:
## lm(formula = pressure ~ ., data = pressure)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -158.08 -117.06  -32.84   72.30  409.43
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -147.8989    66.5529  -2.222 0.040124 *
## temperature    1.5124     0.3158   4.788 0.000171 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 150.8 on 17 degrees of freedom
## Multiple R-squared:  0.5742, Adjusted R-squared:  0.5492
## F-statistic: 22.93 on 1 and 17 DF,  p-value: 0.000171
```

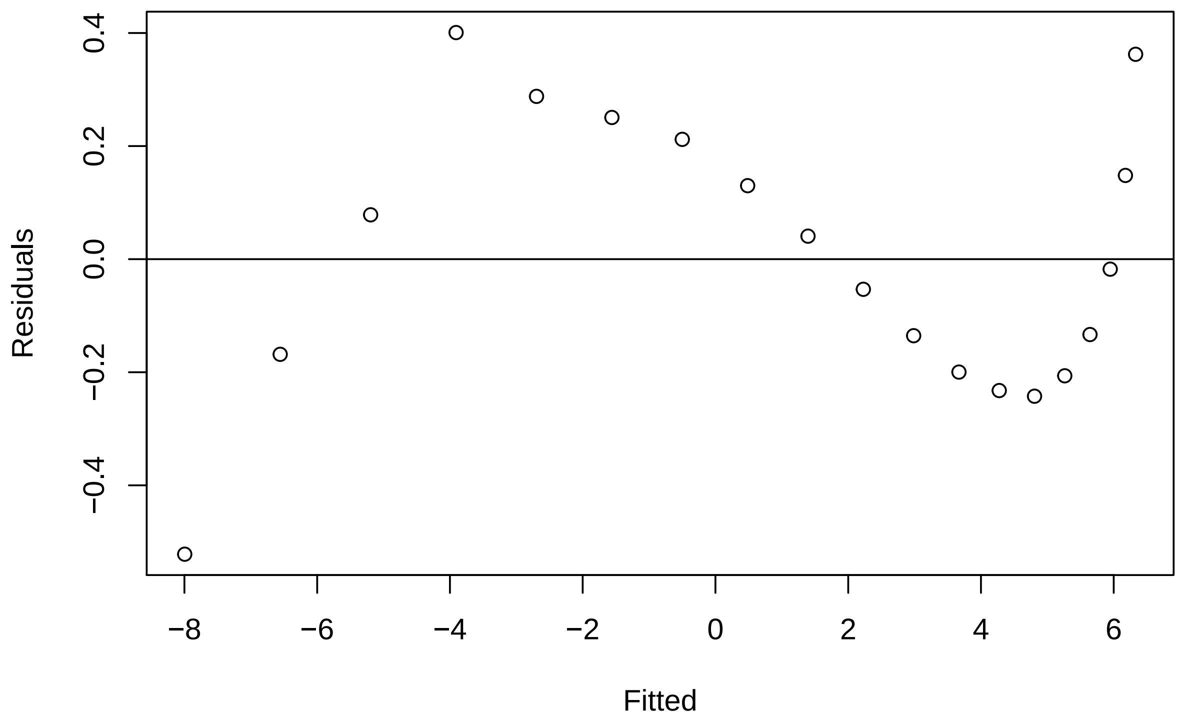




Based on the plots above we look into fitting a series of models of the form $log(pressure) \sim \sum b_i temperature^i$ We note this data looks highly regular, and appears to originate from a physical process. There's obviously some functional relationship between these variables. Knowing this may help us in our modelling. $PV = nRT$ is a good place to start! We also note that there are only 19 observations in this data set so we should not fit too many models or add too many predictors in looking for a good fit.

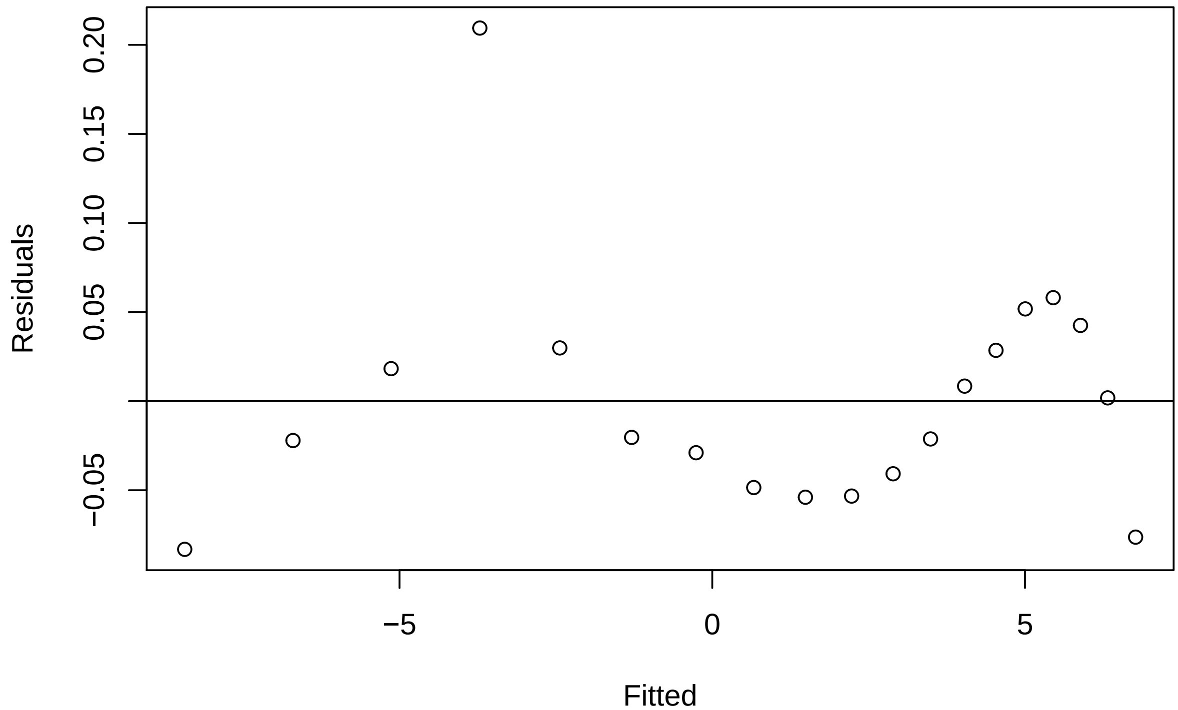
```
##
## Call:
## lm(formula = log(pressure) ~ temperature + I(temperature^2),
##     data = pressure)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5219 -0.1840 -0.0177  0.1800  0.4008
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -7.995e+00  1.603e-01  -49.87  < 2e-16 ***
## temperature       7.380e-02  2.065e-03   35.74  < 2e-16 ***
## I(temperature^2) -9.447e-05  5.536e-06  -17.07 1.09e-11 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2579 on 16 degrees of freedom
## Multiple R-squared:  0.9972, Adjusted R-squared:  0.9969
## F-statistic:  2859 on 2 and 16 DF,  p-value: < 2.2e-16
```




```
##
## Call:
## lm(formula = log(pressure) ~ temperature + I(temperature^2) +
##     I(temperature^3), data = pressure)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.08319 -0.04463 -0.02035  0.02919  0.20942
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -8.434e+00  5.518e-02 -152.83  < 2e-16 ***
## temperature       9.075e-02  1.364e-03   66.51  < 2e-16 ***
## I(temperature^2) -2.154e-04  8.951e-06  -24.07 2.13e-13 ***
```

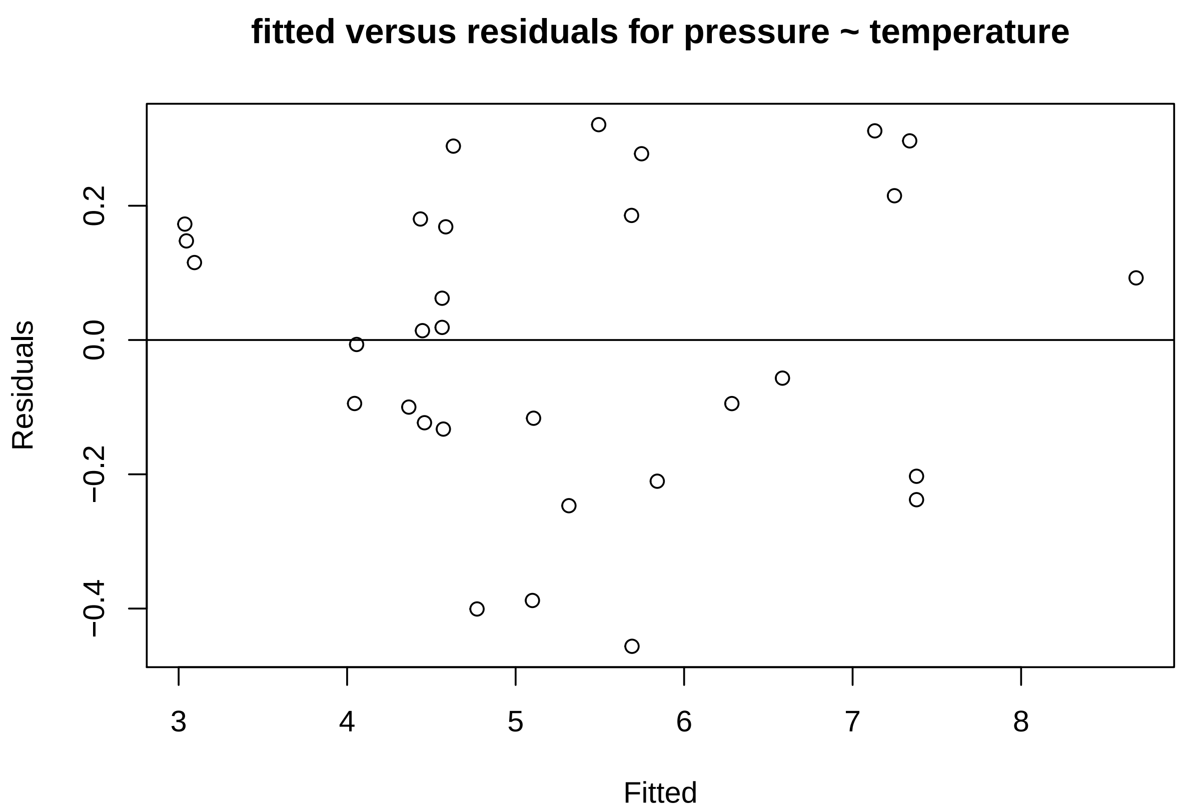
```
## I(temperature^3)  2.240e-07  1.632e-08   13.72 6.78e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07236 on 15 degrees of freedom
## Multiple R-squared:  0.9998, Adjusted R-squared:  0.9998
## F-statistic: 2.428e+04 on 3 and 15 DF,  p-value: < 2.2e-16
```

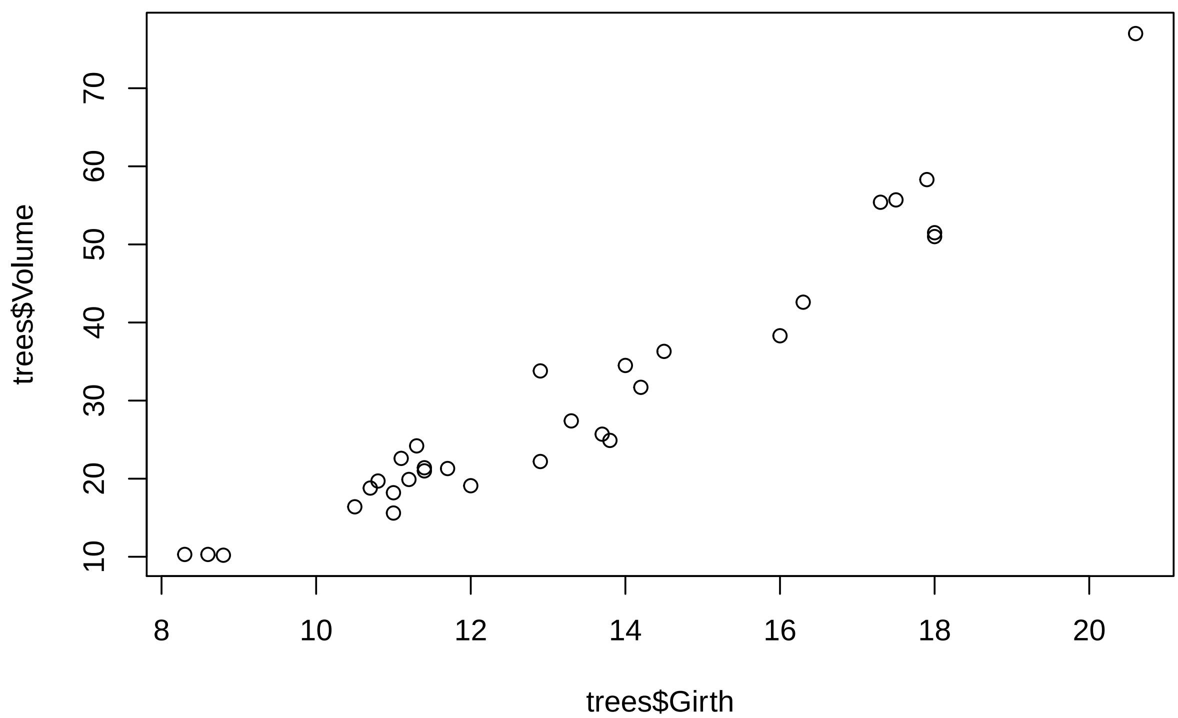


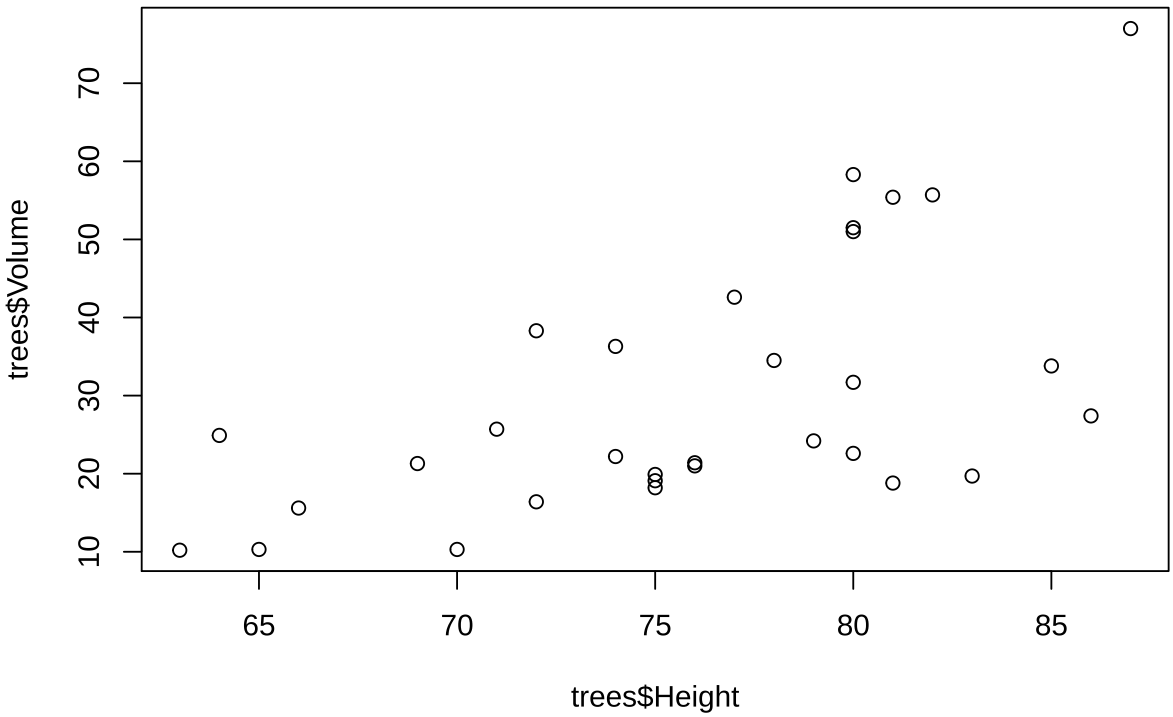
9.5 Use transformations to find a good model for volume in terms of girth and height using the trees data.

```
##
## Call:
## lm(formula = sqrt(Volume) ~ Girth + Height, data = trees)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4562 -0.1280  0.0139  0.1765  0.3208
##
## Coefficients:
```

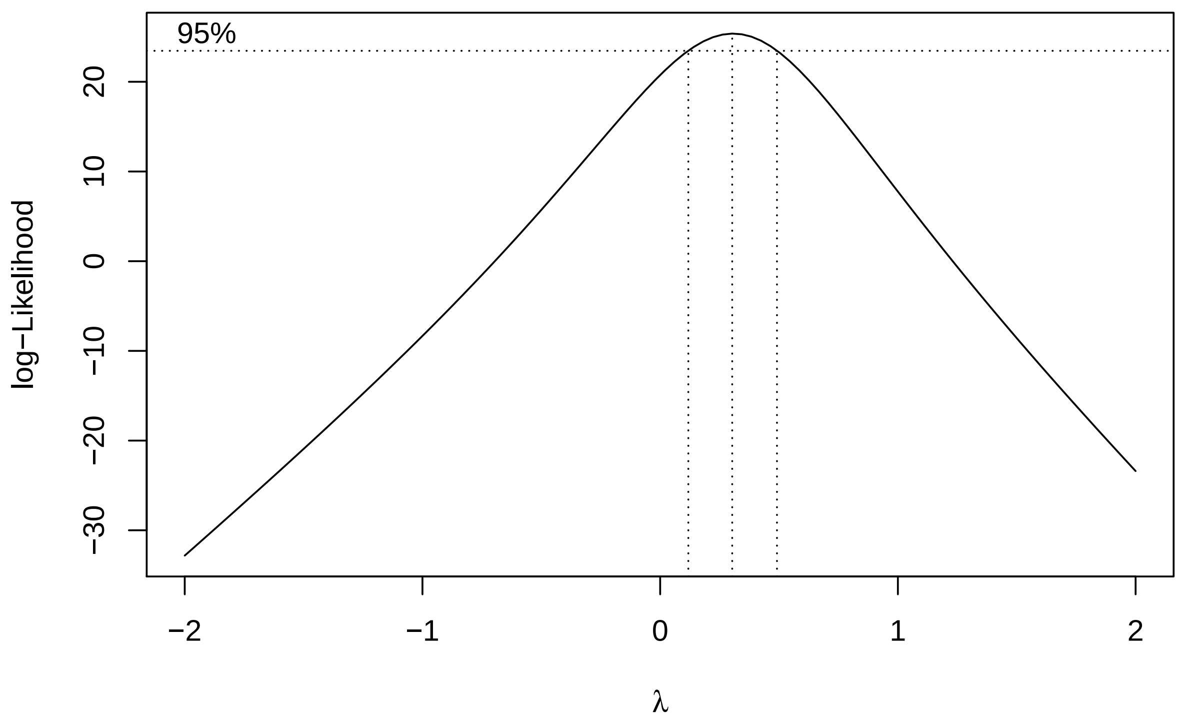
```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.769955   0.509414  -5.438 8.40e-06 ***
## Girth        0.404922   0.015584  25.983  < 2e-16 ***
## Height       0.035758   0.007675   4.659 7.05e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2289 on 28 degrees of freedom
## Multiple R-squared:  0.9757, Adjusted R-squared:  0.974
## F-statistic: 563.1 on 2 and 28 DF,  p-value: < 2.2e-16
```

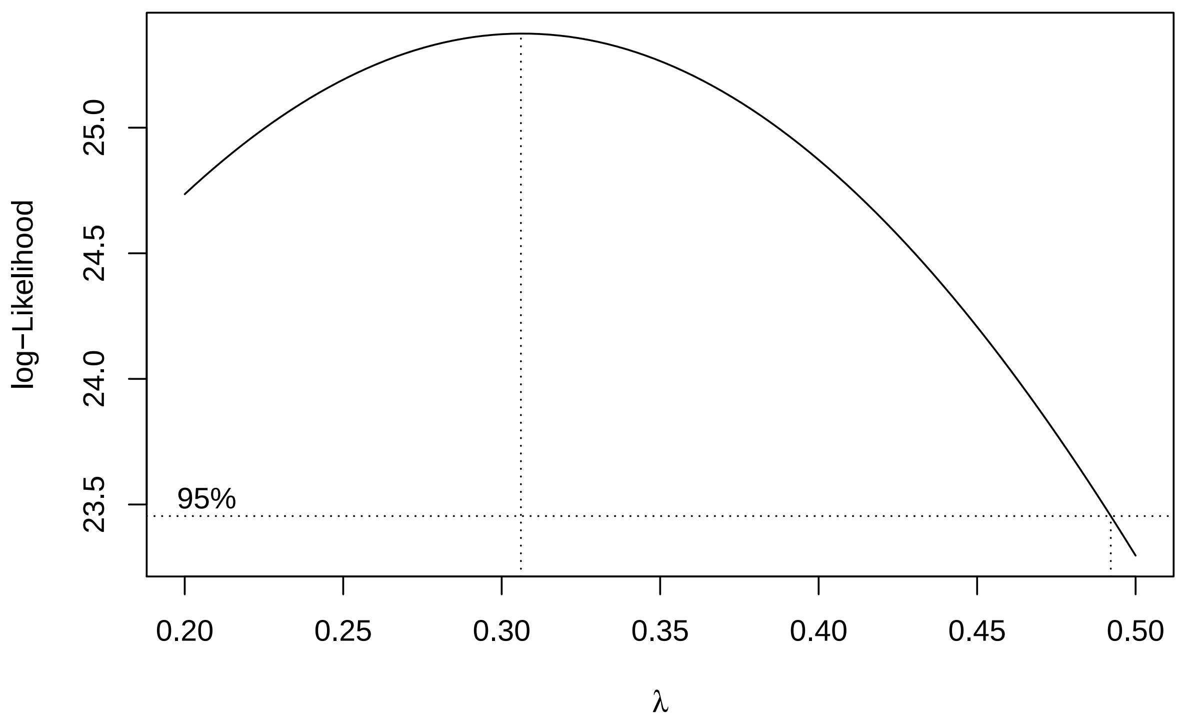



trees$Volume
10
20
30
40
50
60
70
8
10
12
14
16
18
20
trees$Girth



We chose a sqrt transformation of the response after seeing a quadratic relationship among fitted versus residuals. Now we use the Box-Cox method to validate our choice.


95%
log−Likelihood
20
10
0
−10
−20
−30
−2
−1
0
1
2
λ



We see the Box-Cox suggests a lambda of ~ 0.3

```
##
## Call:
## lm(formula = Volume^0.3 ~ Girth + Height, data = trees)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.126316 -0.042838 -0.003901  0.055497  0.109593
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.194613   0.148552   1.310    0.201
## Girth       0.121559   0.004545  26.748  < 2e-16 ***
## Height      0.011799   0.002238   5.272 1.32e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06676 on 28 degrees of freedom
## Multiple R-squared:  0.9775, Adjusted R-squared:  0.9759
## F-statistic: 609.1 on 2 and 28 DF,  p-value: < 2.2e-16
```

Indeed we do have a better fit as evidenced by the lower RSE.

9.6 Response surface for odor data

(a) Fit a second order response surface for the odor response using the other three variables as predictors. How many parameters does this model use and how many degrees of freedom are left?

There should be 3ˆ2 +1 parameters in this model.

```
##
## Call:
## lm(formula = odor ~ polym(temp, gas, pack, degree = 2), data = odor)
##
## Residuals:
##        1        2        3        4        5        6        7        8
## -20.6250  -6.8750   6.8750  20.6250  15.5000   1.7500  -1.7500 -15.5000
##        9       10       11       12       13       14       15
##   5.1250 -22.3750  22.3750  -5.1250  -0.3333  -4.3333   4.6667
##
## Coefficients:
##                                          Estimate Std. Error t value
## (Intercept)                                15.200      5.804   2.619
## polym(temp, gas, pack, degree = 2)1.0.0   -34.295     22.479  -1.526
## polym(temp, gas, pack, degree = 2)2.0.0    61.991     22.603   2.743
## polym(temp, gas, pack, degree = 2)0.1.0   -48.083     22.479  -2.139
## polym(temp, gas, pack, degree = 2)1.1.0    66.000     89.914   0.734
## polym(temp, gas, pack, degree = 2)0.2.0    92.423     22.603   4.089
## polym(temp, gas, pack, degree = 2)0.0.1   -60.458     22.479  -2.690
## polym(temp, gas, pack, degree = 2)1.0.1    12.000     89.914   0.133
## polym(temp, gas, pack, degree = 2)0.1.1   -14.000     89.914  -0.156
## polym(temp, gas, pack, degree = 2)0.0.2    11.754     22.603   0.520
##                                          Pr(>|t|)
## (Intercept)                               0.04716 *
## polym(temp, gas, pack, degree = 2)1.0.0   0.18761
## polym(temp, gas, pack, degree = 2)2.0.0   0.04067 *
## polym(temp, gas, pack, degree = 2)0.1.0   0.08542 .
## polym(temp, gas, pack, degree = 2)1.1.0   0.49588
## polym(temp, gas, pack, degree = 2)0.2.0   0.00946 **
## polym(temp, gas, pack, degree = 2)0.0.1   0.04332 *
## polym(temp, gas, pack, degree = 2)1.0.1   0.89903
## polym(temp, gas, pack, degree = 2)0.1.1   0.88236
## polym(temp, gas, pack, degree = 2)0.0.2   0.62524
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.48 on 5 degrees of freedom
## Multiple R-squared:  0.882,  Adjusted R-squared:  0.6696
## F-statistic: 4.152 on 9 and 5 DF,  p-value: 0.06569
```

As expected there are 9 predictors. There are VERY few degrees of freedom left. Any model we produce with this many predictors and so few degrees of freedom would be dubious.

(b) Fit a model for the same response but now excluding any interaction terms but including linear and quadratic terms in all three predictors. Compare this model to the previous one. Is this simplification justified?

```
##
## Call:
## lm(formula = odor ~ temp + gas + pack + I(temp^2) + I(gas^2) +
##     I(pack^2), data = odor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.625  -9.625  -1.375   4.021  28.875
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -30.667     10.840  -2.829   0.0222 *
## temp         -12.125      6.638  -1.827   0.1052
## gas          -17.000      6.638  -2.561   0.0336 *
## pack         -21.375      6.638  -3.220   0.0122 *
## I(temp^2)     32.083      9.771   3.284   0.0111 *
## I(gas^2)      47.833      9.771   4.896   0.0012 **
## I(pack^2)      6.083      9.771   0.623   0.5509
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.77 on 8 degrees of freedom
## Multiple R-squared:  0.8683, Adjusted R-squared:  0.7695
## F-statistic: 8.789 on 6 and 8 DF,  p-value: 0.003616
```

Based on the adjusted R^2 the simplification is justified.

(c) Use the previous model to determine the values of the predictors which result in the minimum predicted odor.

Table 4: Predictor values resulting in minimum fitted value

	odor	temp	gas	pack	yhat
13	-31	0	0	0	-30.67