Bruce Campell ST 503 HW 1

Problems 1,3 Chapter 2 Faraway, Julian J.. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

29 August, 2017

Tue Aug 29 20:48:58 2017

Problem 1.1

The dataset teengamb concerns a study of teenage gambling in Britain. Make a numerical and graphical summary of the data, commenting on any features that you find interesting. Limit the output you present to a quantity that a busy reader would find sufficient to get a basic understanding of the data.

This report was rendered in R Markdown with the option `echo=FALSE`. We assume a busy reader does not want to see the code.

Load and inspect the data.

When loading and inspecting the data we will note which variables are numeric, and which are strings, we'll also be on the lookout for variables that we may want to encode as factors. Here we note that gender is a candidate for such encoding.

```
##   sex status income verbal gamble
## 1   1     51   2.00      8    0.0
## 2   1     28   2.50      8    0.0
## 3   1     37   2.00      6    0.0
## 4   1     28   7.00      4    7.3
## 5   1     65   2.00      8   19.6
## 6   1     61   3.47      6    0.1
```

Check for missing data

Table 1: Number of missing elements in data set

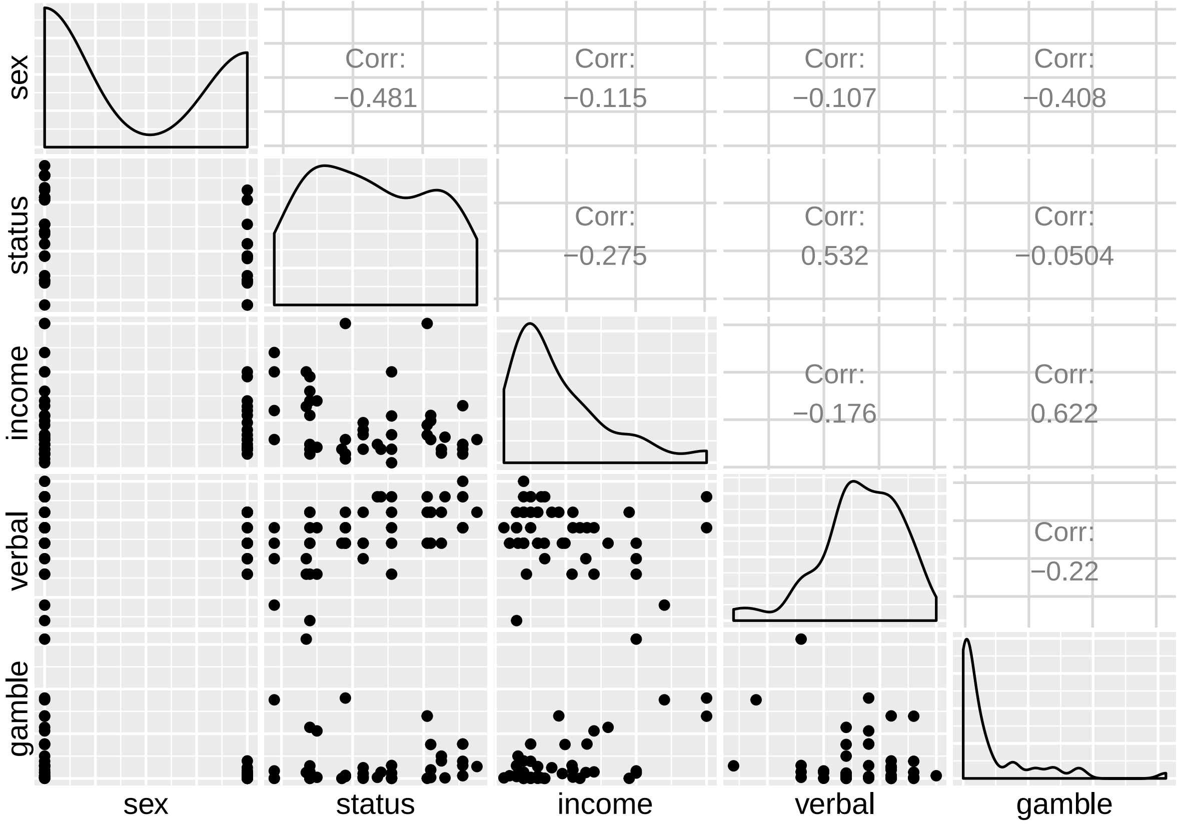
missing.count
0

Calculate summary statistics for the variables

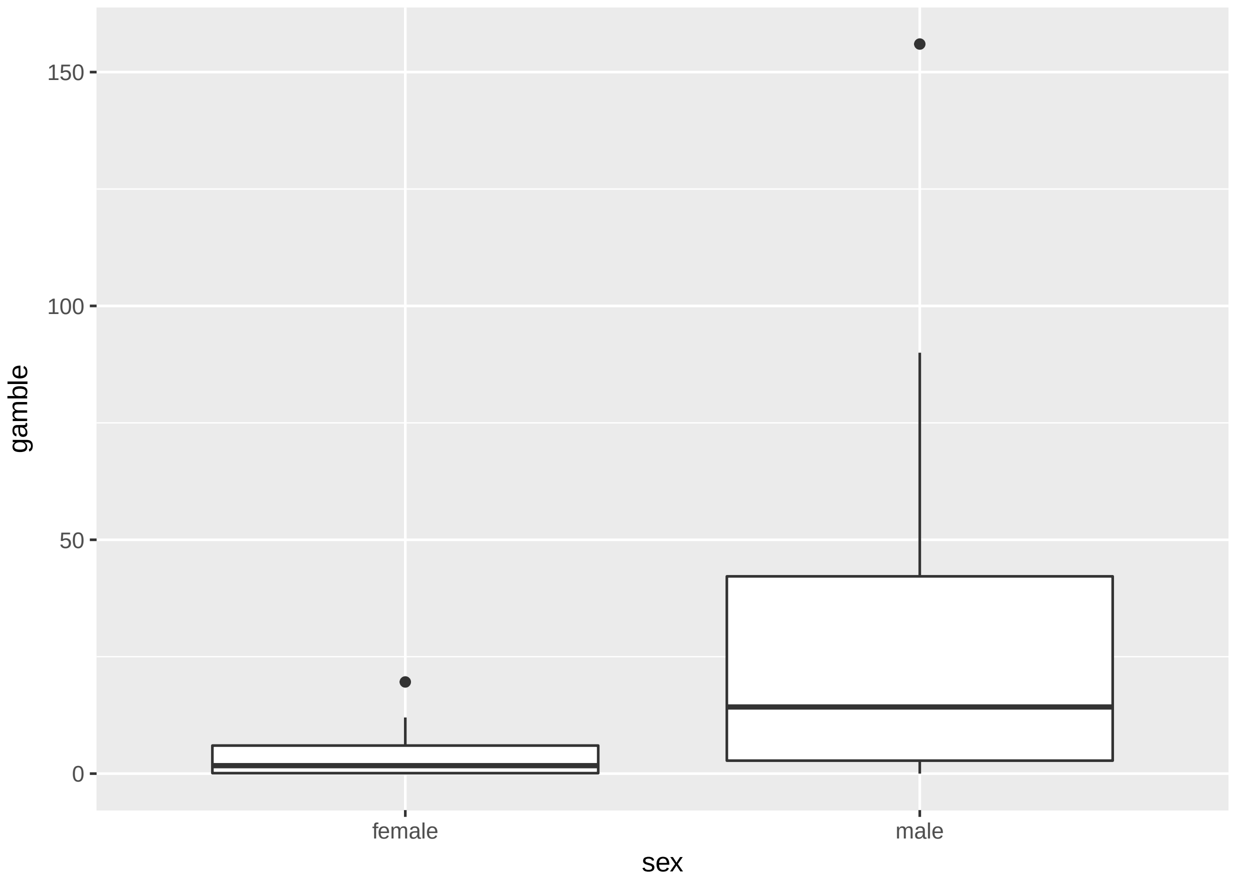
```
##       sex             status          income           verbal
##  Min.   :0.0000   Min.   :18.00   Min.   : 0.600   Min.   : 1.00
##  1st Qu.:0.0000   1st Qu.:28.00   1st Qu.: 2.000   1st Qu.: 6.00
##  Median :0.0000   Median :43.00   Median : 3.250   Median : 7.00
##  Mean   :0.4043   Mean   :45.23   Mean   : 4.642   Mean   : 6.66
##  3rd Qu.:1.0000   3rd Qu.:61.50   3rd Qu.: 6.210   3rd Qu.: 8.00
##  Max.   :1.0000   Max.   :75.00   Max.   :15.000   Max.   :10.00
##      gamble
##  Min.   :  0.0
##  1st Qu.:  1.1
##  Median :  6.0
##  Mean   : 19.3
##  3rd Qu.: 19.4
##  Max.   :156.0
```

Plot the features

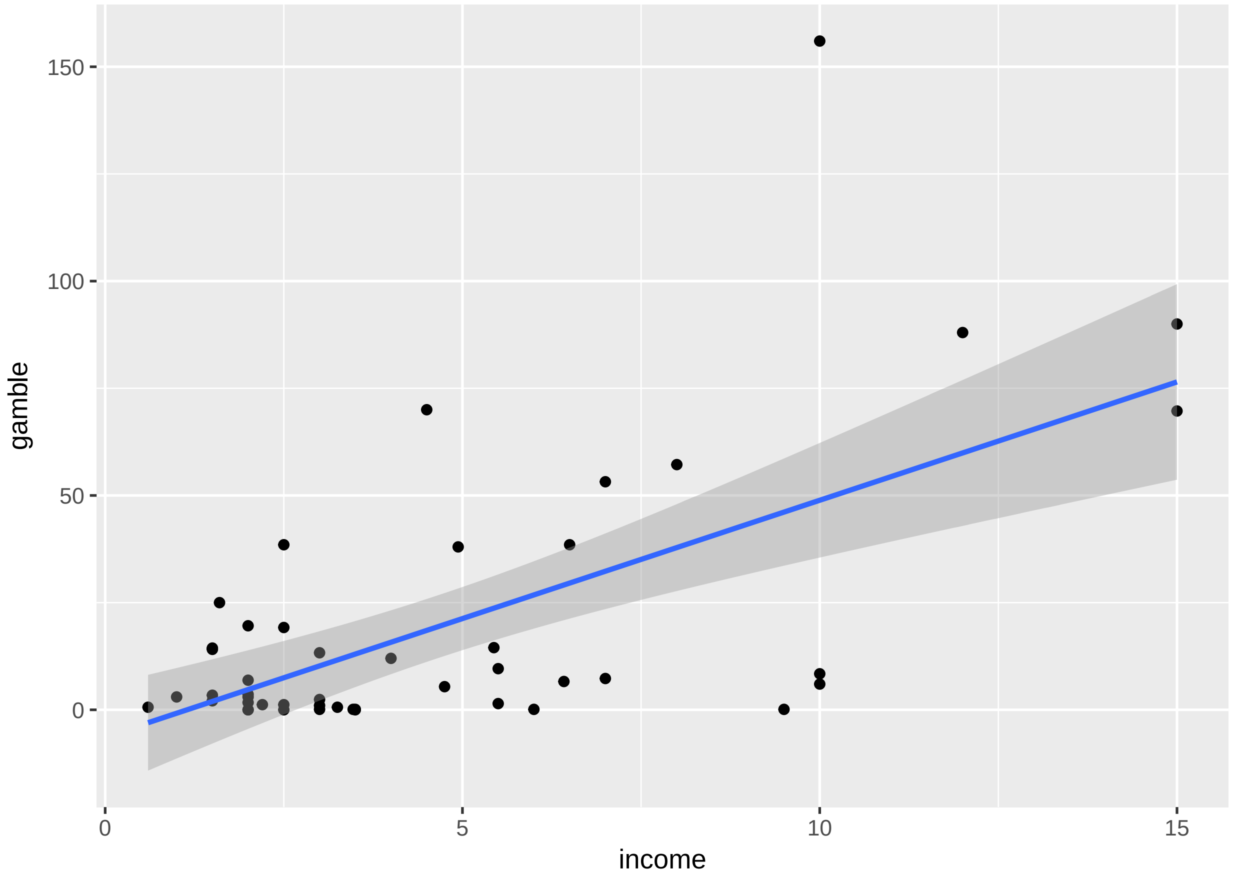
This data come from a study of teenage gambling in Britan. The response variable in this case is gamble, and the other variables in the data set are candidates for predictors in any modeling we do. When creating plots, we'll be interested in how the predictors relate to the response. We'll also be on the lookout for outliers.



We note that gender seems to vary with gamble. We can see this better in a box plot.



We also note that income seems to have an association with gambling.



We observe a data element with a large value of gamble. This needs to be noted and considered when we evaluate any models that we fit with this data.

Problem 1.3

The dataset prostate is from a study on 97 men with prostate cancer who were due to receive a radical prostatectomy. Make a numerical and graphical summary of the data as in the first question.

Load and inspect the data

```
##       lcavol lweight age      lbph svi      lcp gleason pgg45     lpsa
## 1 -0.5798185  2.7695  50 -1.386294   0 -1.38629       6     0 -0.43078
## 2 -0.9942523  3.3196  58 -1.386294   0 -1.38629       6     0 -0.16252
## 3 -0.5108256  2.6912  74 -1.386294   0 -1.38629       7    20 -0.16252
## 4 -1.2039728  3.2828  58 -1.386294   0 -1.38629       6     0 -0.16252
## 5  0.7514161  3.4324  62 -1.386294   0 -1.38629       6     0  0.37156
## 6 -1.0498221  3.2288  50 -1.386294   0 -1.38629       6     0  0.76547
```

We note that the documentation provides the following details the meaning of the features ;

- lcavol=log(cancer volume)
- lweight=log(prostate weight)
- age=age
- lbph=log(benign prostatic hyperplasia amount)

- svi=seminal vesicle invasion
- lcp=(capsular penetration)
- gleason=leason score
- pgg45=percentage Gleason scores 4 or 5
- lpsa=log(prostate specific antigen)

Check for missing data

Table 2: Number of missing elements in data set

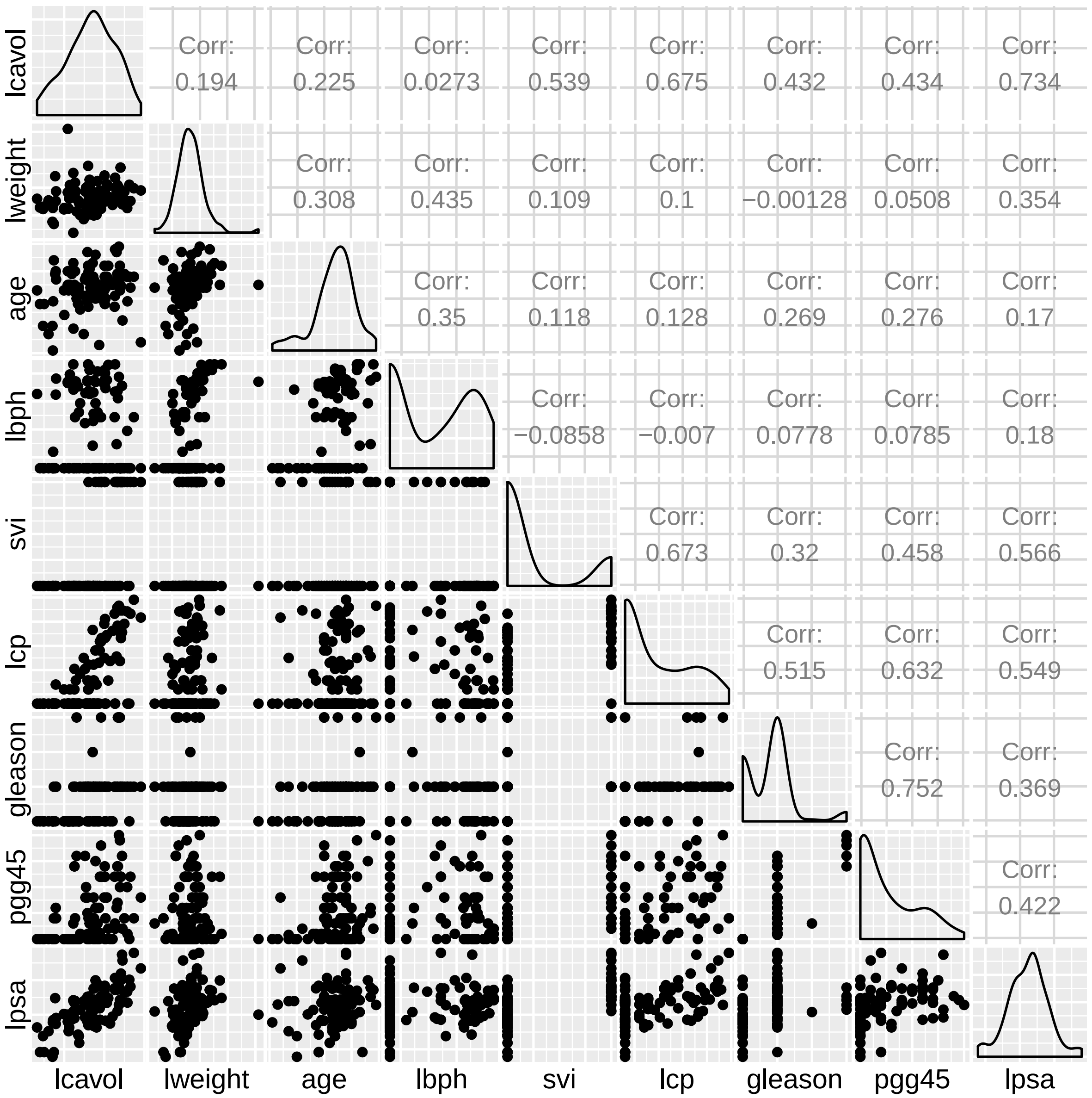
missing.count
0

We note that the gleason score might be a variable that is a candidate for encoding as a factor variable.

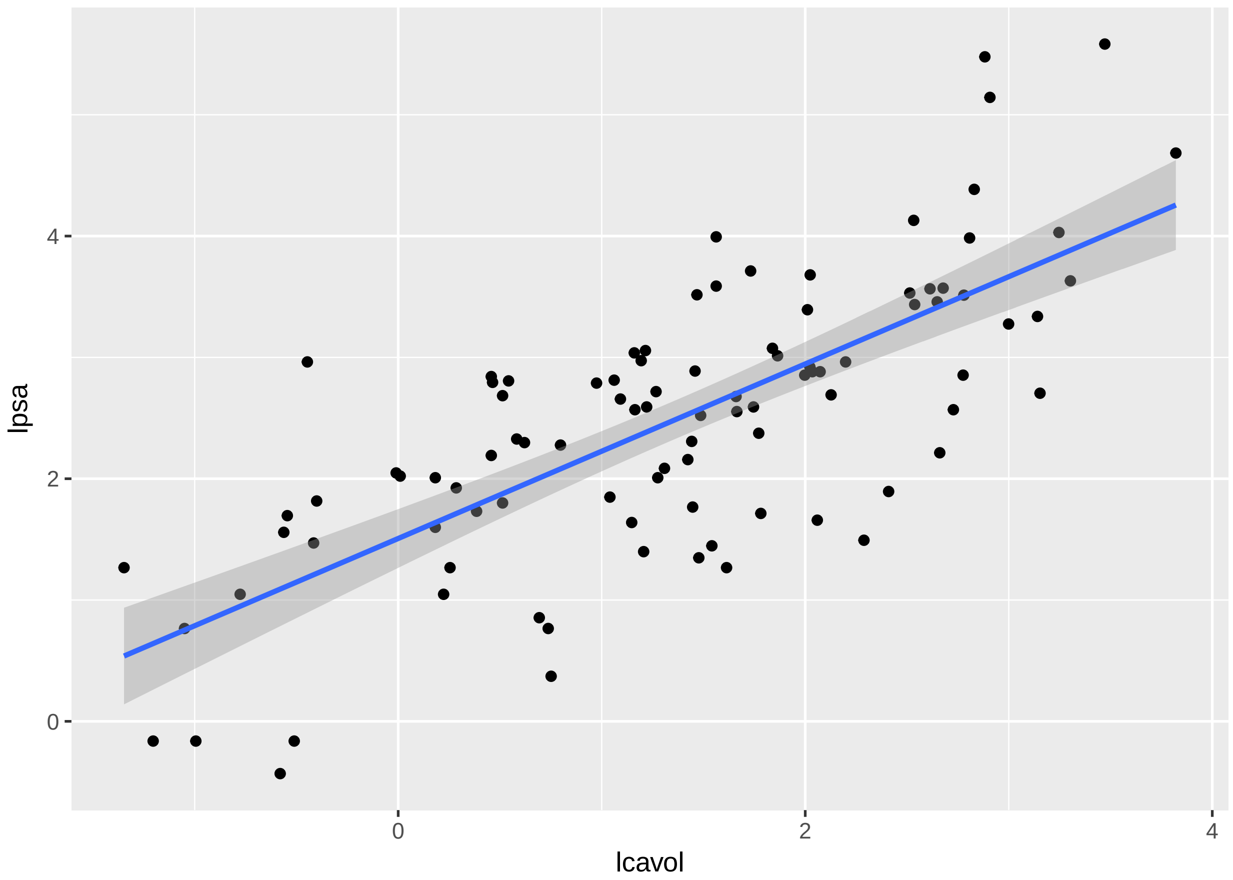
Summary statistics for the prostate data

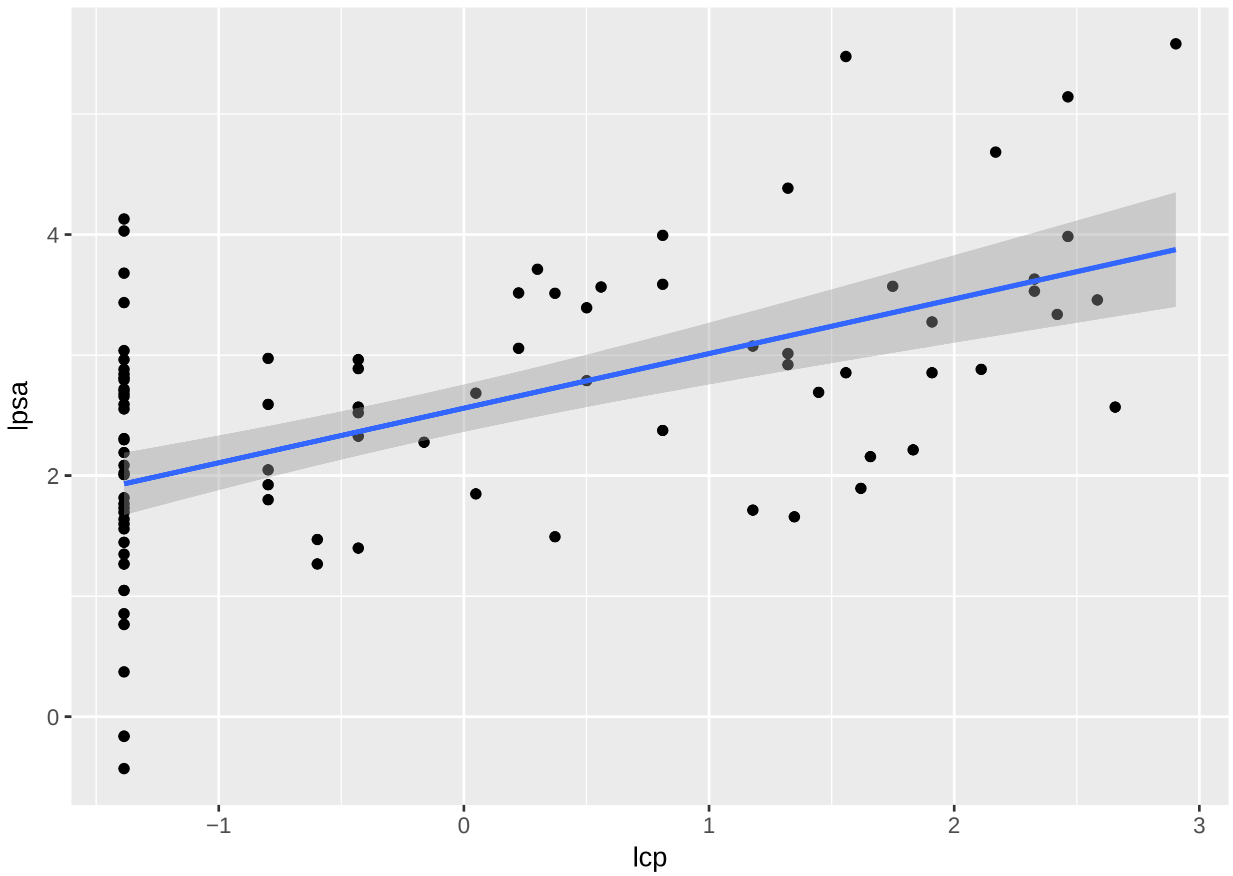
```
##      lcavol           lweight           age             lbph        
##  Min.   :-1.3471   Min.   :2.375   Min.   :41.00   Min.   :-1.3863  
##  1st Qu.: 0.5128   1st Qu.:3.376   1st Qu.:60.00   1st Qu.:-1.3863  
##  Median : 1.4469   Median :3.623   Median :65.00   Median : 0.3001  
##  Mean   : 1.3500   Mean   :3.653   Mean   :63.87   Mean   : 0.1004  
##  3rd Qu.: 2.1270   3rd Qu.:3.878   3rd Qu.:68.00   3rd Qu.: 1.5581  
##  Max.   : 3.8210   Max.   :6.108   Max.   :79.00   Max.   : 2.3263  
##       svi              lcp             gleason          pgg45       
##  Min.   :0.0000   Min.   :-1.3863   Min.   :6.000   Min.   :  0.00  
##  1st Qu.:0.0000   1st Qu.:-1.3863   1st Qu.:6.000   1st Qu.:  0.00  
##  Median :0.0000   Median :-0.7985   Median :7.000   Median : 15.00  
##  Mean   :0.2165   Mean   :-0.1794   Mean   :6.753   Mean   : 24.38  
##  3rd Qu.:0.0000   3rd Qu.: 1.1786   3rd Qu.:7.000   3rd Qu.: 40.00  
##  Max.   :1.0000   Max.   : 2.9042   Max.   :9.000   Max.   :100.00  
##       lpsa        
##  Min.   :-0.4308  
##  1st Qu.: 1.7317  
##  Median : 2.5915  
##  Mean   : 2.4784  
##  3rd Qu.: 3.0564  
##  Max.   : 5.5829  
```

Plot of the variables for the prostate data



We note a relationship between lcavol and lpsa, and between lcp and lpsa. We also note that there is a relationship between the gleason score and lpsa and pgg45.


lpsa
4
2
0
0
2
4
lcavol



We note that there are a number of lcp values at -1.39. We should follow up with the study authors to understand this better. It may be due to limitations or constraints in instrumentation that was used to make the measurements.

Bruce Campell ST 503 HW 2

Problems 1, 4, 7 Chapter 2 Faraway, Julian J. Linear Models with R, Second Edition. CRC Press.

Bruce Campbell

05 September, 2017

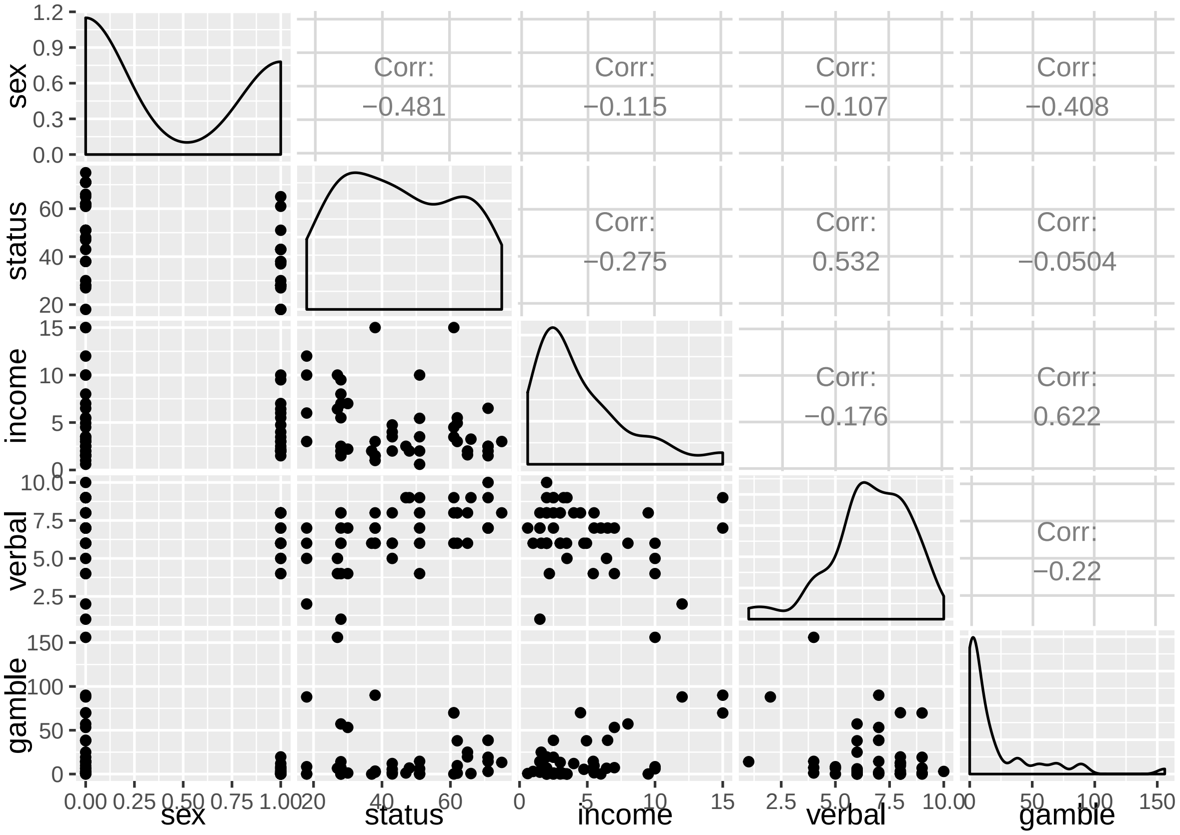
This document was rendered in Rmarkdown. Some of the code is not displayed. The markdown used to generate this is located on github at

https://github.com/brucebcampbell/applied-regression-with-R/blob/master/BruceCampbell_ST503_HW2_FarawayCh2_Pblms_1_4_7.Rmd

Problem 2.1

The dataset teengamb concerns a study of teenage gambling in Britain. Fit a regression model with the expenditure on gambling as the response and the sex, status, income and verbal score as predictors. Present the output.

- (a) What percentage of variation in the response is explained by these predictors?
- (b) Which observation has the largest (positive) residual? Give the case number.
- (c) Compute the mean and median of the residuals.
- (d) Compute the correlation of the residuals with the fitted values.
- (e) Compute the correlation of the residuals with the income.
- (f) For all other predictors held constant, what would be the difference in predicted expenditure on gambling for a male compared to a female?



```
##
## Call:
## lm(formula = gamble ~ sex + status + income + verbal, data = teengamb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.082 -11.320  -1.451   9.452  94.252
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.55565   17.19680   1.312   0.1968
## sex         -22.11833    8.21111  -2.694   0.0101 *
## status        0.05223    0.28111   0.186   0.8535
## income        4.96198    1.02539   4.839 1.79e-05 ***
## verbal       -2.95949    2.17215  -1.362   0.1803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06
```

(a) What percentage of variation in the response is explained by these predictors?

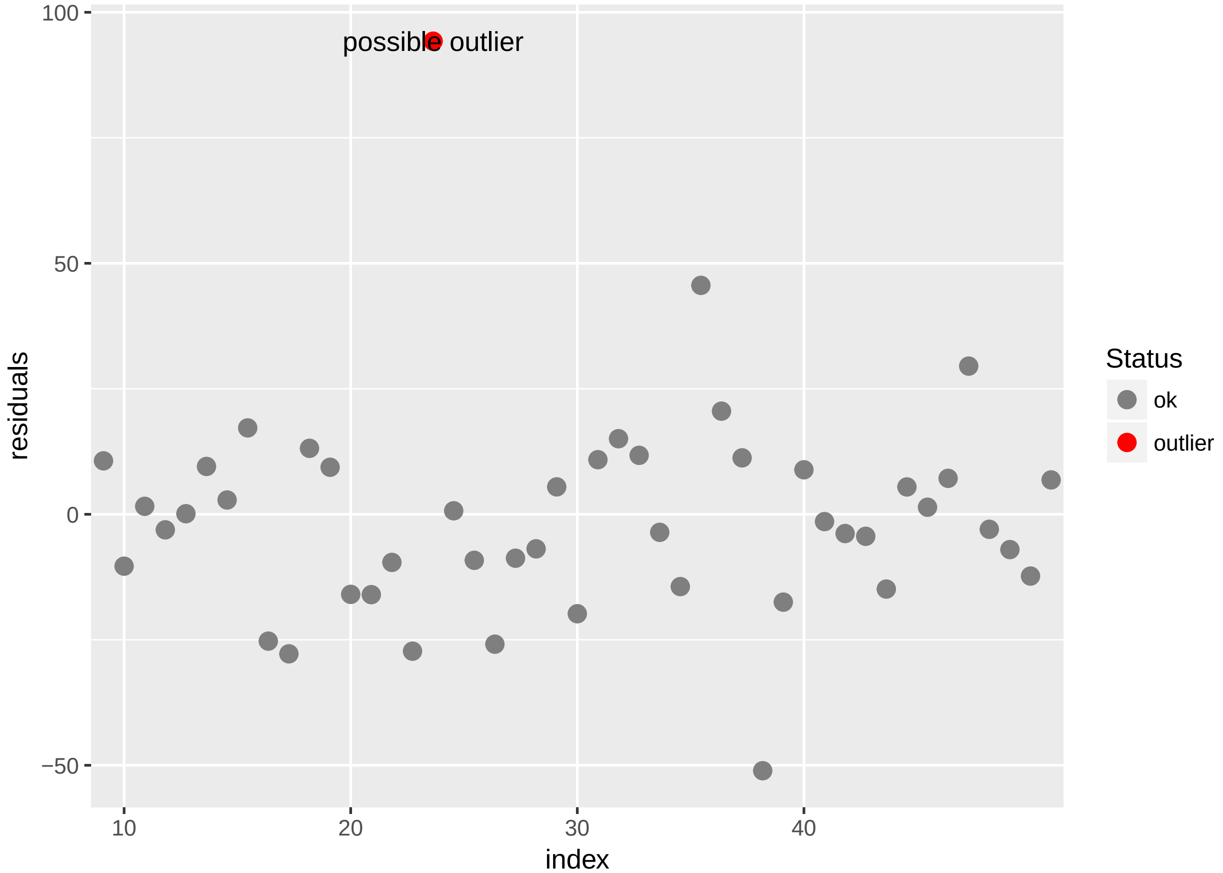
Here we calculate the proportion of explained and unexplained variance in the response that is given by the predictors in the mode we fit.

Table 1: Proportion of Variance Explained

var.explained.proportion
0.5267

(b) Which observation has the largest (positive) residual? Give the case number.

We're not sure if the question seeks the larges residual in absolute value or the largest of the positive residuals. We suspect that we're looking for the largest residual in absolute values since this may be an outlier that needs investigation, but we'll report both.



The largest residual occurs at index 24 of the dataframe. This is the associated case data.

Table 2: Potential outlier.

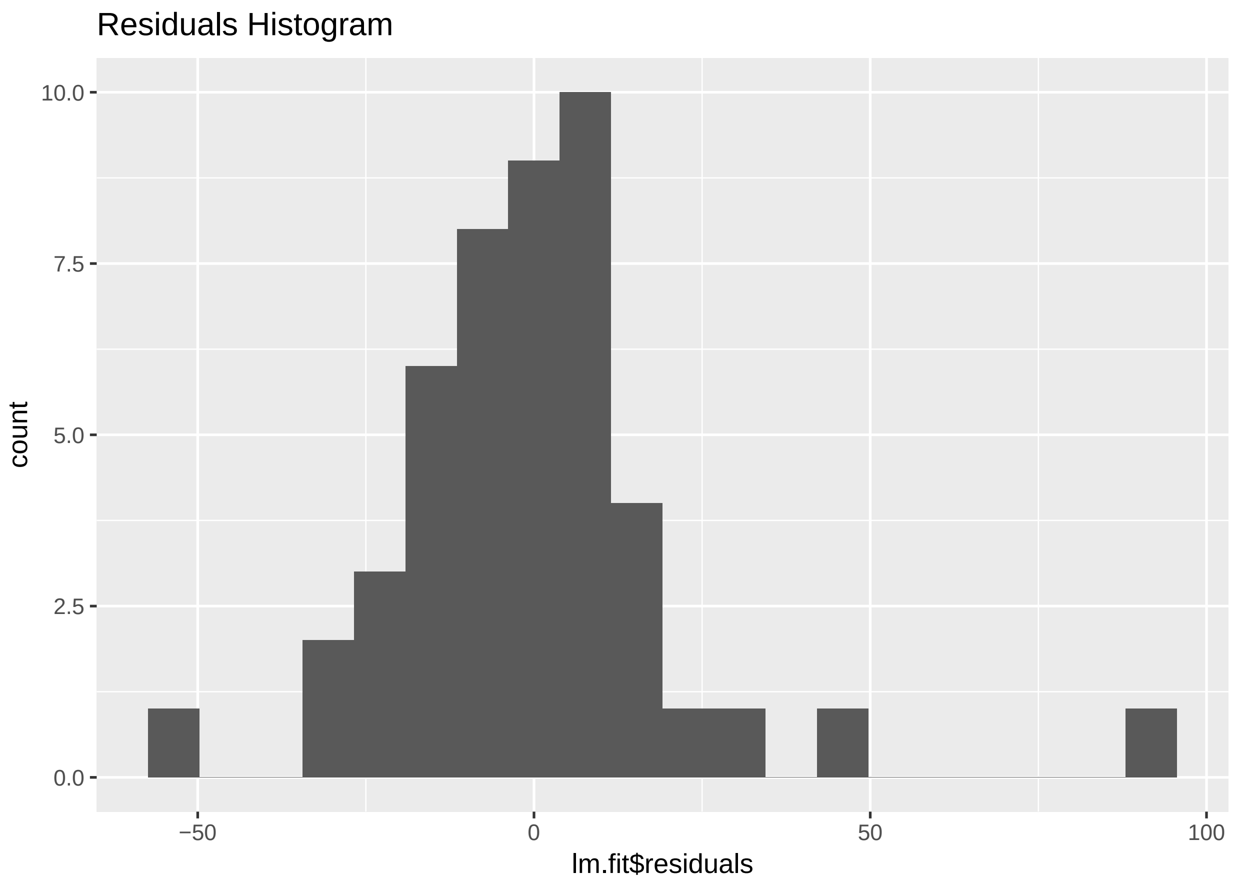
	sex	status	income	verbal	gamble
24	0	27	10	4	156

	sex	status	income	verbal	gamble

(c) Compute the mean and median of the residuals.

Table 3: mean and median of the residuals

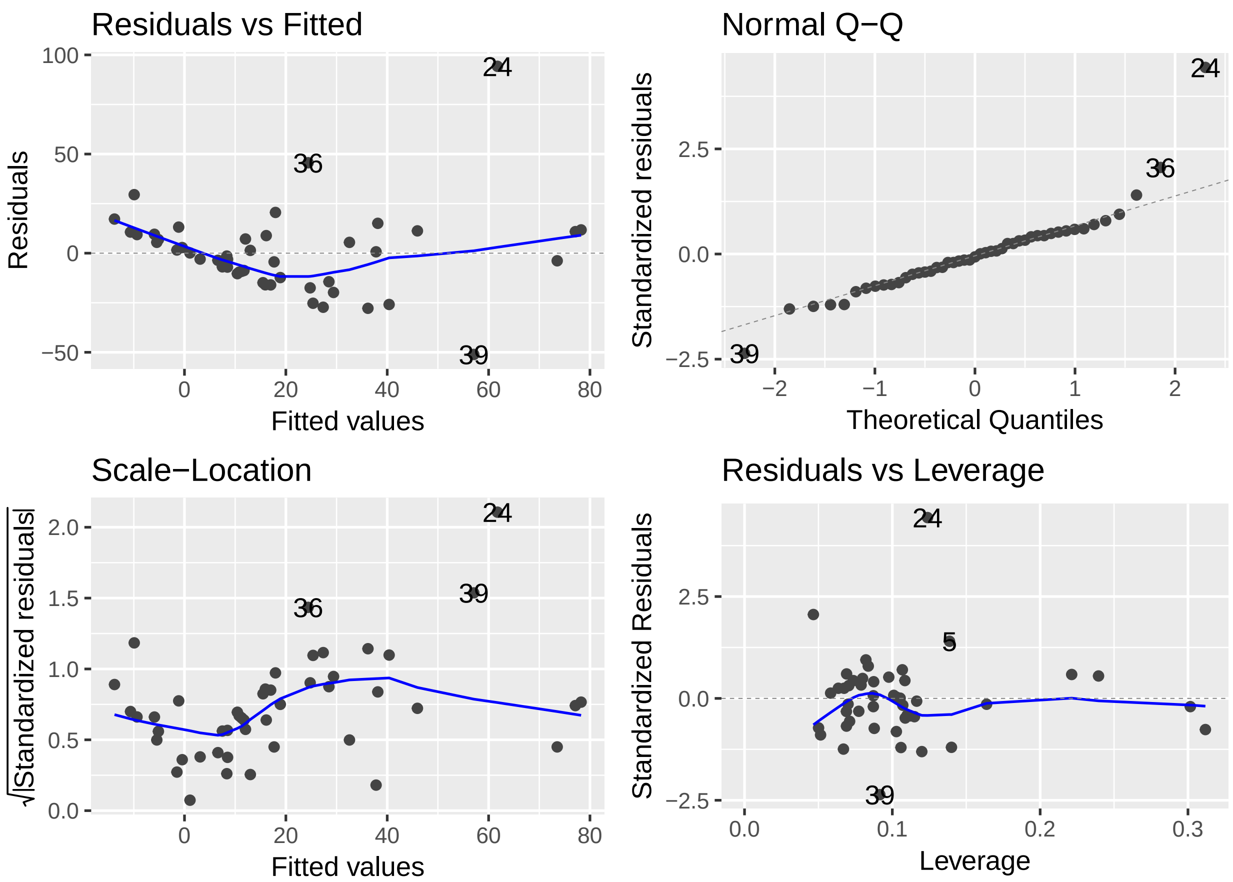
residuals.mean	residuals.median
-3.065e-17	-1.451



```
## [1] 21.68137
```

The mean residual is a very small number! We'd need to think through the implications of this - possibly it is an artifact of data that was generated.

Regression diagnostics are plotted below.

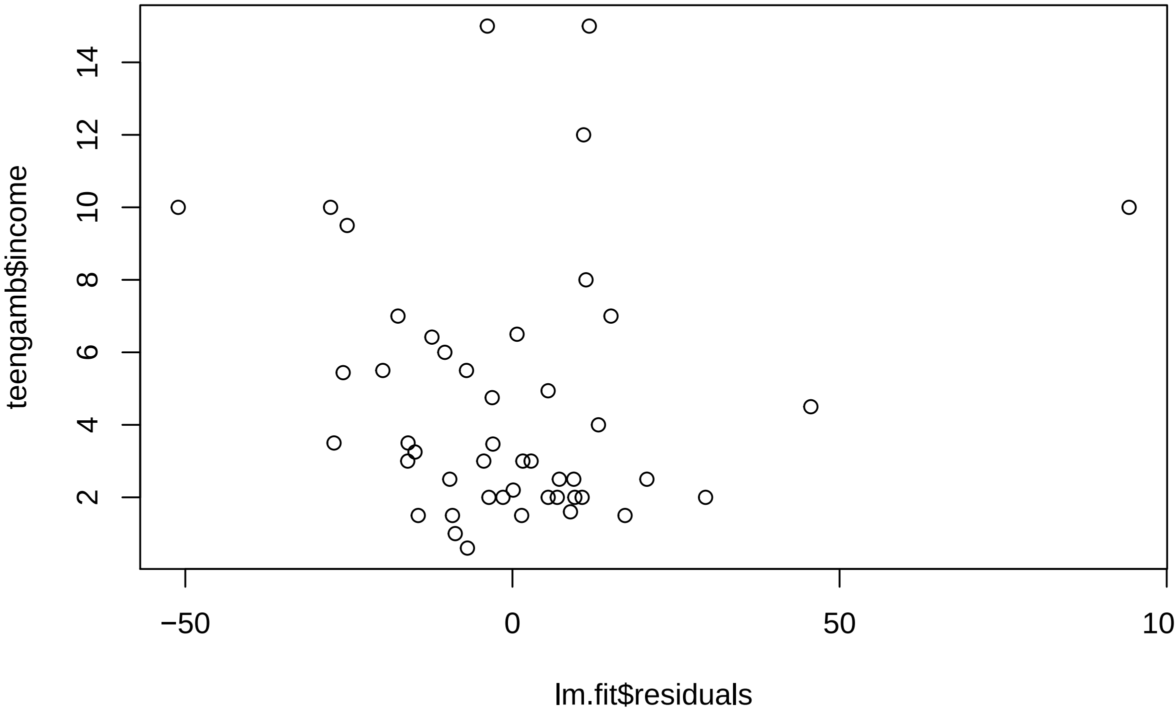


(d) Compute the correlation of the residuals with the fitted values.

corr.residuals.vs.fitted
-1.071e-16

(e) Compute the correlation of the residuals with the income.

corr.residuals.income
-7.242e-17



(f) For all other predictors held constant, what would be the difference in predicted expenditure on gambling for a male compared to a female?

This should be the value of the coefficient for gender. We need to be careful about the encoding and understanding whether this was treated as a factor in the regression. Querying the data `?teengamb` tells us that sex is encoded as so `0=male, 1=female`. Looking at the data frame teengamb we see that the class of the variable is integer and not a factor so we can now interpret the coefficient properly.

	gender.coefficient
sex	-22.12

This value represents the change in the response when there is a unit change in the predictor. In this case since female is encoded as `1` we can say that females have that much less gamble response (less because the coefficient is negative).

We can apply the model by hand to a element of the data set to see this in practice.

```
data.sample <- sample(nrow(teengamb), 1)
data.element <- teengamb[data.sample, ]
data.element$gamble <- NULL
```

```r
data.element <- as.matrix(cbind(intercept = 1, data.element))
beta.hat <- as.matrix(lm.fit$coefficients)

pander(data.frame(data.element), caption = "Data sample")
```

Table 7: Data sample

	intercept	sex	status	income	verbal
43	1	0	75	3	8

```r
response.orig <- (data.element) %*% beta.hat

# change the gender of our data element
data.element[1, 2] <- ifelse(data.element[1, 2] == 1, 0, 1)


pander(data.frame(data.element), caption = "Data sample with gender modified")
```

Table 8: Data sample with gender modified

	intercept	sex	status	income	verbal
43	1	1	75	3	8

```r
response.gendermod <- (data.element) %*% beta.hat

pander(data.frame(response.difference = (response.orig - response.gendermod)))
```

	response.difference
43	22.12

Problem 2.4

The dataset prostate comes from a study on 97 men with prostate cancer who were due to receive a radical prostatectomy. Fit a model with lpsa as the response and lcavol as the predictor. Record the residual standard error and the R^2. Now add lweight, svi, lbph, age, lcp, pgg45 and gleason to the model one at a time. For each model record the residual standard error and the R^2. Plot the trends in these two statistics.

Load data and fit the models

Fit lpsa ~ lcavol +lweight

```
##
## Call:
## lm(formula = lpsa ~ lcavol, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.67625 -0.41648  0.09859  0.50709  1.89673
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.50730    0.12194   12.36   <2e-16 ***
## lcavol       0.71932    0.06819   10.55   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7875 on 95 degrees of freedom
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5346
## F-statistic: 111.3 on 1 and 95 DF,  p-value: < 2.2e-16
```

```
##          term  estimate  std.error statistic      p.value
## 1 (Intercept) 1.5072979 0.12193682  12.36130 1.722234e-21
## 2      lcavol 0.7193201 0.06819288  10.54832 1.118616e-17
```

We will only print the models for $n = 2$ and $n = 8$ predictors.

Fit lpsa ~ lcavol +lweight

```
##          term   estimate  std.error  statistic      p.value
## 1 (Intercept) -0.3026179 0.56904195 -0.5318024 5.961175e-01
## 2      lcavol  0.6775253 0.06626223 10.2249086 6.120248e-17
## 3     lweight  0.5109495 0.15725697  3.2491371 1.606370e-03
```

Fit lpsa ~ lcavol +lweight + svi + lbph + age + lcp + pgg45+ gleason

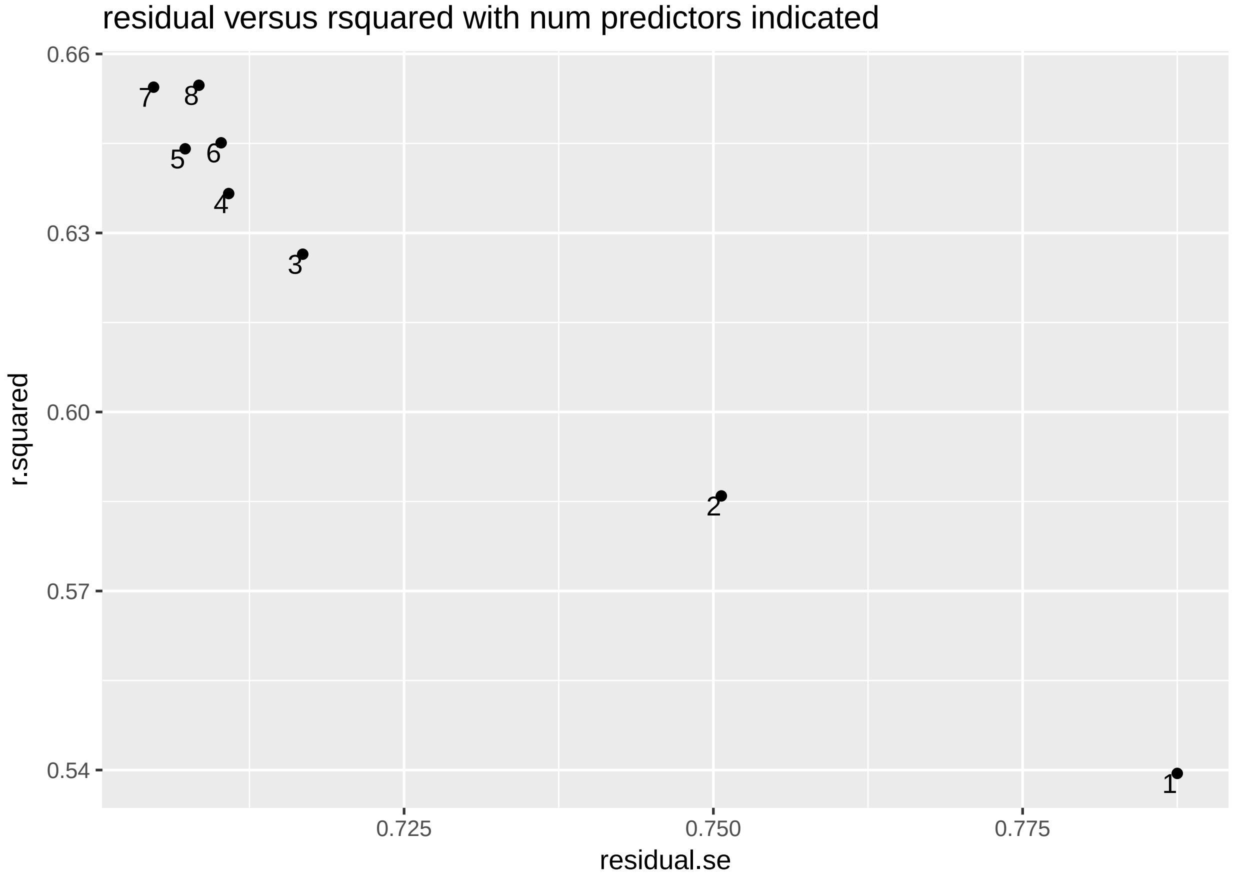
```
##          term     estimate   std.error  statistic      p.value
## 1 (Intercept)  0.669336698 1.296387471  0.5163091 6.069335e-01
## 2      lcavol  0.587021826 0.087920303  6.6767493 2.110698e-09
## 3     lweight  0.454467424 0.170012435  2.6731423 8.955363e-03
## 4         svi  0.766157326 0.244309148  3.1360157 2.328749e-03
## 5        lbph  0.107054031 0.058449214  1.8315735 7.039846e-02
## 6         age -0.019637176 0.011172725 -1.7575995 8.229321e-02
## 7         lcp -0.105474263 0.091013487 -1.1588861 2.496377e-01
## 8       pgg45  0.004525231 0.004421179  1.0235350 3.088604e-01
## 9     gleason  0.045141598 0.157464523  0.2866779 7.750328e-01
```

Present the model stats

Table 10: model statistics

num.predictors	r.squared	residual.se	model.string
8	0.6548	0.7084	lpsa ~ lcavol +lweight + svi + lbph + age + lcp + pgg45+ gleason
7	0.6544	0.7048	lpsa ~ lcavol +lweight + svi + lbph + age + lcp + pgg45
6	0.6451	0.7102	lpsa ~ lcavol +lweight + svi + lbph + age + lcp
5	0.6441	0.7073	lpsa ~ lcavol +lweight + svi + lbph + age
4	0.6366	0.7108	lpsa ~ lcavol +lweight + svi + lbph
3	0.6264	0.7168	lpsa ~ lcavol +lweight + svi
2	0.5859	0.7506	lpsa ~ lcavol +lweight
1	0.5394	0.7875	lpsa ~ lcavol

Plot SE versus R^2



We see that generally the proportion of variance explained by the model increases and the residual standard error decreases as the dimension of the model increases. The effect becomes less pronounced as we get to 6+ predictors. One could argue that inclusion of gleason to the model does not add much explanatory power. This may make empirical sense since the gleason score is assigned by a pathologist based on a stained tissue slide. It could be the case that this feature summaries or is a weak proxy for the biochemical variables.

Problem 2.7

An experiment was conducted to determine the effect of four factors on the resistivity of a semiconductor wafer. The data is found in wafer where each of the four factors is coded as - or + depending on whether the low or the high setting for that factor was used.

Fit the linear model $resist \sim x1 + x2 + x3 + x4$

```
## [1] "Inspect Data"
```

```
##   x1 x2 x3 x4 resist
## 1  -  -  -  -  193.4
## 2  +  -  -  -  247.6
## 3  -  +  -  -  168.2
## 4  +  +  -  -  205.0
## 5  -  -  +  -  303.4
## 6  +  -  +  -  339.9
```

```
## [1] "check the class of the columns"
```

```
## $x1
## [1] "factor"
##
## $x2
## [1] "factor"
##
## $x3
## [1] "factor"
##
## $x4
## [1] "factor"
##
## $resist
## [1] "numeric"
```

```
## [1] "Fi the model"
```

(a) Extract the X matrix using the model.matrix function. Examine this to determine how the low and high levels have been coded in the model.

```
##    (Intercept) x1+ x2+ x3+ x4+
## 1            1   0   0   0   0
## 2            1   1   0   0   0
## 3            1   0   1   0   0
## 4            1   1   1   0   0
## 5            1   0   0   1   0
## 6            1   1   0   1   0
## 7            1   0   1   1   0
## 8            1   1   1   1   0
## 9            1   0   0   0   1
## 10           1   1   0   0   1
## 11           1   0   1   0   1
## 12           1   1   1   0   1
## 13           1   0   0   1   1
## 14           1   1   0   1   1
## 15           1   0   1   1   1
## 16           1   1   1   1   1
## attr(,"assign")
## [1] 0 1 2 3 4
## attr(,"contrasts")
## attr(,"contrasts")$x1
## [1] "contr.treatment"
##
## attr(,"contrasts")$x2
## [1] "contr.treatment"
##
## attr(,"contrasts")$x3
## [1] "contr.treatment"
##
## attr(,"contrasts")$x4
## [1] "contr.treatment"
```

Now let's look at the data matrix to see how the factors are coded

x1	x2	x3	x4	resist
-	-	-	-	193.4
+	-	-	-	247.6
-	+	-	-	168.2
+	+	-	-	205
-	-	+	-	303.4
+	-	+	-	339.9
-	+	+	-	226.3
+	+	+	-	208.3
-	-	-	+	220
+	-	-	+	256.4
-	+	-	+	165.7
+	+	-	+	203.5
-	-	+	+	285
+	-	+	+	268
-	+	+	+	169.1
+	+	+	+	208.5

Comparing the model matrix to the original dataframe we see that low level − ⟶ 0 and high level + ⟶ 1

(b) Compute the correlation in the X matrix. Why are there some missing values in the matrix?

Table 12: Correlation of X

	X.Intercept.	x1.	x2.	x3.	x4.
(Intercept)	1	NA	NA	NA	NA
x1+	NA	1	0	0	0
x2+	NA	0	1	0	0
x3+	NA	0	0	1	0
x4+	NA	0	0	0	1

The correlation in the X matrix is the pairwise values of the column correlations. The correlation is the covariance divided by the sqaure root of the product of the two variances.

If X and Y are jointly distributed random variables and the variances and covariances of both X and Y exist and the variances are nonzero, then the correlation of X and Y , denoted by , is $\rho = Cov(X, Y)/\sqrt{Var(X)Var(Y)}$

In this case we're dealing with samples and calculating the sample correlations.

There are NaN's in the due to the intercept. The variance of this vector is 0 and when R attempts to divide by 0 in the calculation of $\rho_{i,j}$ the results is a NaN. Note that R sets the diagonal of the correlation to one - it does not calculate the value - that's why we see $\rho_{1,1} = 1$.

We noted that the $i =\neq j$ terms for $i, j > 1$ were zero - this cause concern and we wrote some test code to validate the entries. The values were verified to be correct.

(d) Refit the model without x4 and examine the regression coefficients and standard errors? What stayed the the same as the original fit and what changed?

Reduced Model

```
##
## Call:
## lm(formula = resist ~ x1 + x2 + x3, data = wafer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.137 -20.550   3.575  18.462  41.013
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   229.54      13.32  17.231 7.88e-10 ***
## x1+            25.76      13.32   1.934 0.077047 .
## x2+           -69.89      13.32  -5.246 0.000206 ***
## x3+            43.59      13.32   3.272 0.006677 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.64 on 12 degrees of freedom
## Multiple R-squared:  0.7777, Adjusted R-squared:  0.7221
## F-statistic: 13.99 on 3 and 12 DF,  p-value: 0.0003187
```

Full Model

```
##
## Call:
## lm(formula = resist ~ x1 + x2 + x3 + x4, data = wafer)
##
## Residuals:
```

```
##      Min      1Q  Median      3Q     Max 
## -43.381 -17.119   4.825  16.644  33.769 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   236.78      14.77  16.032 5.65e-09 ***
## x1+            25.76      13.21   1.950 0.077085 .  
## x2+           -69.89      13.21  -5.291 0.000256 ***
## x3+            43.59      13.21   3.300 0.007083 ** 
## x4+           -14.49      13.21  -1.097 0.296193    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 26.42 on 11 degrees of freedom
## Multiple R-squared:  0.7996, Adjusted R-squared:  0.7267 
## F-statistic: 10.97 on 4 and 11 DF,  p-value: 0.0007815
```

We note that the p value for X4 was not significant in the first model and that removing it resulted in a model where the explained variance is not significantly changes. We could do a LRT on the two models to further understand if adding X4 enhances the model.

(e) Explain how the change in the regression coefficients is related to the correlation matrix of X.

When the model matrix is orthogonal the covariance matrix of the sampling distribution of the regression parameters will be diagonal - when the error are iid $N(0, \sigma)$.

$$\hat{\beta} \sim N(\beta, \sigma(\mathbf{X}^\intercal\mathbf{X})^{-1})$$

This means the regression parameters are independent. That's why we did not see a change in the estimates of the coefficients for $X1\,X2$, $X3$ when we removed $X4$ from the model.

We can verify

$$(\mathbf{X}^\intercal\mathbf{X})^{-1}(\mathbf{X}^\intercal\mathbf{X}) = I$$

for our model matrix

```
##             (Intercept) x1+ x2+ x3+ x4+
## (Intercept)           1   0   0   0   0
## x1+                   0   1   0   0   0
## x2+                   0   0   1   0   0
## x3+                   0   0   0   1   0
## x4+                   0   0   0   0   1
```

Bruce Campbell NCSU ST-503 HW 3

Chapter 14 Problems 2,3,6,30,31 Rice, John A. Mathematical Statistics and Data Analysis, Cengage

Bruce Campbell

12 September, 2017

Problem 14.2

For the following data points Plot (x, y), fit and sketcha line $y = a + bx$ by the method of least squares, fit and sketch a line given by $x = c + dy$

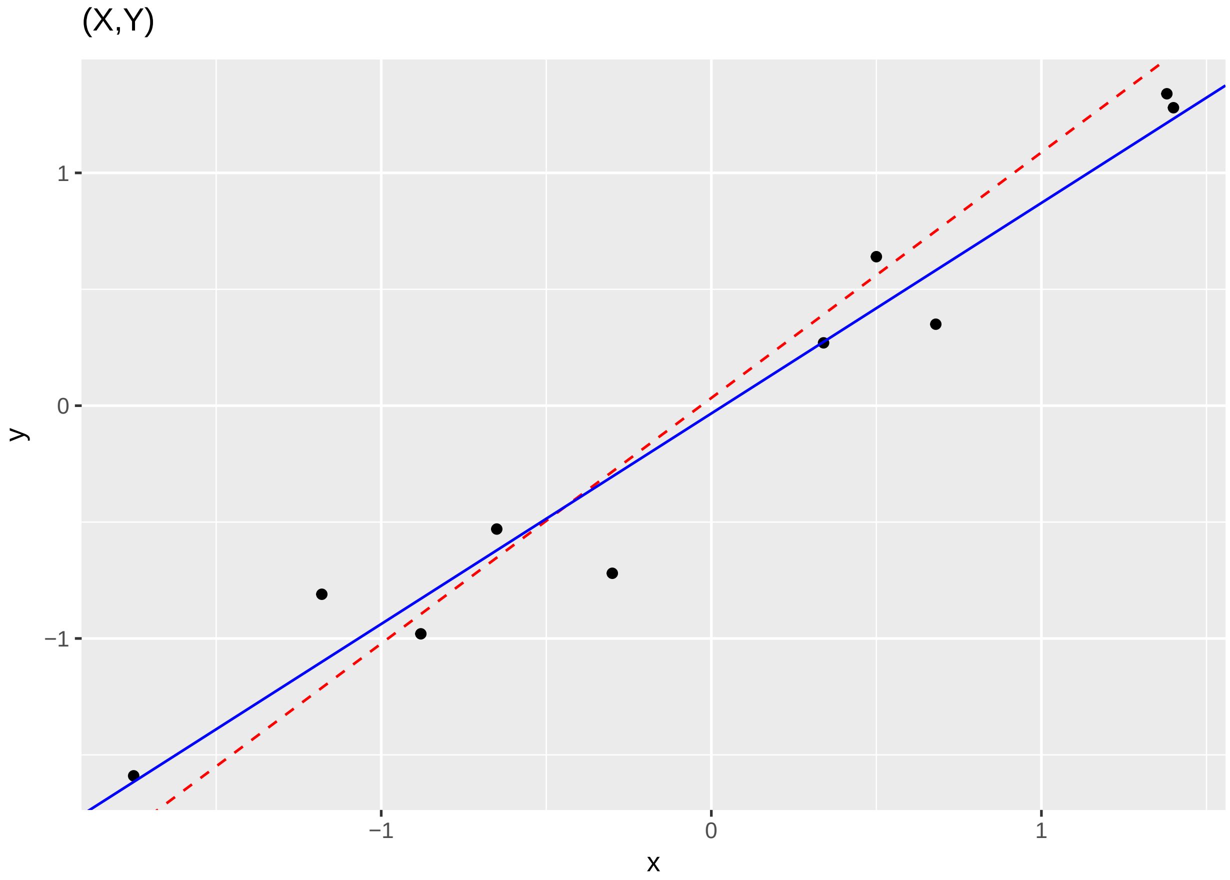


Table 1: My caption

x	.34	1.38	-.65	.68	1.40	-.88	-.30	-1.18	.50	-1.75
y	.27	1.34	-.53	.35	1.28	-.98	-.72	-.81	.64	-1.59

c. Are the lines in parts (a) and (b) the same? If not, why not?

The lines are not the same. Geometrically, we're minimizing errors in the y direction for the model $y \sim x$ and for the model $x \sim y$ we're minimizing errors in the x direction. Going further we can work out when we might see equality in the two regression lines.

Let's denote the two lines by $y = \beta_0 + \beta_1 x$ and $x = \beta_0' + \beta_1' y$ We know that $\bar{y} = \beta_0 + \beta_1 \bar{x}$ and $\bar{x} = \beta_0' + \beta_1' \bar{y}$ The point $(\bar{x}, \bar{y})$ is where the two regression lines above intersect. Now we also know that $(\beta_0, 0)$ is a point on $y = \beta_0 + \beta_1 x$ and that $(0, \beta_0')$ is a point on $x = \beta_0' + \beta_1' y$. The two lines will only be the same when $\beta_0 = \beta_0' = 0$. We can go to the derivation of the values of β_0, β_0' in the case of simple linear regression and ask about the conditions in which we will see $\beta_0 = \beta_0' = 0$. We won't show the algebra here, but if we did it right the data constraints for equality of the regression lines is

$$\frac{\bar{x}}{\bar{y}} = \frac{\hat{\sigma_x^2}}{\hat{\sigma_y^2}}$$

Problem 14.3

Show that when $y_i = \mu + \epsilon_i \ni e_i \;\; iid \;\; : \; E[\epsilon] = 0 \;, Var(\epsilon_i) = \sigma^2$ we have $\bar{y}$ is the least squares estimate for μ.

The LS errors we want to minimize are

$$S(\mu) = \sum (y_i - \mu)^2$$

Taking derivatives and setting equal to zero we have that

$$\frac{\partial S}{\partial \mu} = 0 = -2 \sum (y_i - \mu) \implies \sum y_i = n\mu$$

So $\hat{\mu} = \bar{y}$

Problem 14.5

Three objects are located on a line at points p1 < p2 < p3. These locations are not precisely known. A surveyor makes the following measurements: a. He stands at the origin and measures the three distances from there to p1, p2, p3. Let these measurements be denoted by Y1, Y2, Y3. b. He goes to p1 and measures the distances from there to p2 and p3. Let these measurements be denoted by Y4, Y5. c. He goes to p2 and measures the distance from there to p3. Denote this measurement by Y6.He thus makes six measurements in all, and they are all subject to error. In order to estimate the values p1, p2, p3, he decides to combine all the measurements by the method of least squares. Using matrix notation, explain clearly how the least squares estimates would be calculated (you don't have to do the actual calculations).

Rice, John A.. Mathematical Statistics and Data Analysis (Available 2010 Titles Enhanced Web Assign) (Page 592). Cengage Textbook. Kindle Edition.

The predictors are $X_i \in \{-1, 0, 1\}$ there will be three of them corresponding to the three objects. We'll be adding vectors to determine how the measurement was made. The coefficients are d_1, d_2, d_3 and these will denote the unknown distances. There is no intercept in this model.

The matrix equation that needs to be solve in this case is

$$\mathbf{Y} = \mathbf{X}\mathbf{d}$$

Where

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \\ Y_5 \\ Y_6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix}$$

The matrix solution is given by

$$\left(\mathbf{X}^\intercal \mathbf{X}\right)^{-1} \mathbf{X}^\intercal \mathbf{Y} = \hat{\mathbf{d}}$$

which is derived by solving the least squares problem for the model $\mathbf{Y} = \mathbf{X}\mathbf{d} + \boldsymbol{\epsilon}$ In practice the matrix equation is solved numerically using the QR matrix factorization.

Problem 14.30

Find $Var(\bar{X})$ where

$$\mathbf{X} = (X_1, \ldots, X_n) \ni Var(X_i) = \sigma \;\forall\; i$$

and

$$Cov(X_i, X_j) = \rho\, \sigma \;\forall\; i \neq j$$

The covariance matrix of the random vector X is given by

$$\Sigma_{X\,X} = \begin{pmatrix} \sigma^2 & \rho\sigma^2 & \cdots & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 & \cdots & \rho\sigma^2 \\ \vdots & \vdots & \ddots & \vdots \\ \rho\sigma^2 & \rho\sigma^2 & \cdots & \sigma^2 \end{pmatrix}$$

Now consider the mean as a vector.

$$Y = \frac{1}{n}\mathbf{1}^\intercal X = \bar{X} \in \mathbb{R}^1$$

$\Sigma_{Y\,Y}$ is given by a special case of the covarinace expression for affine transforms of random vectors

$$\mathbf{Y} = \mathbf{A}\mathbf{X} \;\; \mathbf{Z} = \mathbf{B}\mathbf{X} \implies \Sigma_{Y\,Z} = \mathbf{A}^\intercal \Sigma_{X\,X} \mathbf{B}$$

We have that

$$\Sigma_{Y\,Y} = \frac{1}{n}\mathbf{1}^\intercal\ \Sigma_{XX}\ \frac{1}{n}\mathbf{1} = \frac{1}{n}\mathbf{1}^\intercal \begin{pmatrix} \sigma^2 + (n-1)\rho\sigma^2 \\ \vdots \\ \sigma^2 + (n-1)\rho\sigma^2 \end{pmatrix} = \frac{\sigma^2}{n}(1 + (n-1)\rho)$$

Now by definition $\Sigma_{Y\,Y} = Var(\bar{X})$

Problem 14.31

Let

$$Z = \begin{pmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \end{pmatrix} \in \mathbb{R}^4$$

and $\Sigma_{Z\,Z} = \sigma^2 I$

$$U = Z_1 + Z_2 + Z_3 + Z_4$$

and

$$V = (Z_1 + Z_2) - (Z_3 + Z_4)$$

Find $Cov(U, V)$

First let's define U and V as linear forms $U = \mathbf{1}^\intercal Z$ and $V = a^\intercal Z\ :\ a = (1, 1, -1, -1)$

Then we have

$$Cov(U, V) = \Sigma_{U\,V} = \mathbf{1}^\intercal\,\Sigma_{Z\,Z}\,a = \mathbf{1}^\intercal \begin{pmatrix} \sigma^2 \\ \sigma^2 \\ -\sigma^2 \\ -\sigma^2 \end{pmatrix} = 0$$

We've use the result about the cross covariance of 2 affine transforms of a random vector;

$$\mathbf{Y} = \mathbf{AX}\ \ \mathbf{Z} = \mathbf{BX} \implies \Sigma_{Y\,Z} = \mathbf{A}^\intercal \Sigma_{X\,X} \mathbf{B}$$

We don't need it and the book doesn't state this but we note that adding a constant to $\mathbf{Y}$ or $\mathbf{Z}$ does not change the cross covariance.

NCSU ST 503 HW 4

Probems 3.2, 3.4, 3.5, 3.6, 4.2 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

18 September, 2017

Problem 3.2

Thirty samples of cheddar cheese were analyzed for their content of acetic acid, hydrogen sulfide and lactic acid. Each sample was tasted and scored by a panel of judges and the average taste score produced. Use the cheddar dataset

(a) Fit a regression model with taste as the response and the three chemical contents as predictors. Identify the predictors that are statistically significant at the 5% level.

```
lm.fit <- lm(taste ~ Acetic + H2S + Lactic, data = cheddar)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-28.8767696	19.735418	-1.4631952	0.1553991
Acetic	0.3277413	4.459757	0.0734886	0.9419798
H2S	3.9118411	1.248430	3.1334077	0.0042471
Lactic	19.6705434	8.629055	2.2795710	0.0310795

r.squared
0.6517747

We see that *H2S* and *Lactic* are significant to the 5% level.

(b) Acetic and H2S are measured on a log scale. Fit a linear model where all three predictors are measured on their original scale. Identify the predictors that are statistically significant at the 5% level for this model.

To undo the log transform we need the base - this is not specified in the help section for the data set. Since we're dealing with chemical concentration data, and based on part e) we will assume that *Acetic* and *H2S* are measured on a Log_e scale.

```r
lm.fit.exp <- lm(taste ~ I(exp(1)^Acetic) + I(exp(1)^H2S) + Lactic, data = cheddar)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-18.9727153	11.2680492	-1.683762	0.1041981
I(exp(1)^Acetic)	0.0189056	0.0156227	1.210135	0.2371145
I(exp(1)^H2S)	0.0007668	0.0004188	1.831110	0.0785679
Lactic	25.0073579	9.0621214	2.759548	0.0104624

rsquared
0.575407

We see that now only *Lactic* is significant at the 5% level. *H2S* is significant at 10%. We thought this could be due to numerical issues in the QR - to test that out we took the transformed data set, standardize it and fit that.

For comparison on the effect of scaling we also fit the scaled model without the inverse log transform. The scaled inverse log transformed model had *H2S* and *Lactic* significant to the 5% level.

(c) Can we use an F-test to compare these two models? Explain. Which model provides a better fit to the data? Explain your reasoning.

We can not use an F-test to compare these models since they are not nested. The model fit in *ln* scale is a better fit to the data based on the R^2 criteria.

(d) If H2S is increased 0.01 for the model used in (a), what change in the taste would be expected?

For the model fit in part a) we saw that $\beta_{H2S} = 3.9118$ this means that keeping all other variables constant and increasing *H2S* by 0.01 increases taste by 0.039118. We can verify this is the case numerically on an example data element from the training set.

```r
data.sample <- sample(nrow(cheddar), 1)
data.element <- cheddar[data.sample, ]
data.element$taste <- NULL
data.element <- as.matrix(cbind(intercept = 1, data.element))
beta.hat <- as.matrix(lm.fit$coefficients)
pander(data.frame(data.element), caption = "Data sample")
```

Table 5: Data sample

	intercept	Acetic	H2S	Lactic
14	1	5.236	4.942	1.3

```
response.orig <- (data.element) %*% beta.hat
# change the of our data element H2S by +0.01
data.element[1, 3] <- data.element[1, 3] + 0.01
pander(data.frame(data.element), caption = "Data sample data element H2S increased by +(
```

Table 6: Data sample data element H2S increased by +0.01

	intercept	Acetic	H2S	Lactic
14	1	5.236	4.952	1.3

```
response.mod <- (data.element) %*% beta.hat
pander(data.frame(response.difference = (response.mod - response.orig)))
```

	response.difference
14	0.03912

(e) What is the percentage change in H2S on the original scale corresponding to an additive increase of 0.01 on the (natural) log scale?

Let our log concentration be α then e^{α} is our concentration in the original scale. A δ change in the log scale H2S results in a concentration of $e^{\alpha+\delta}$

The percent change is

$$(\frac{e^{\alpha+\delta}-e^{\alpha}}{e^{\alpha}}) * 100\% = (e^{\delta}-1) * 100\%$$

In our case $\delta = 0.01$ and the percent change is `101.0050167`

Problem 3.3

Using the teengamb data, fit a model with gamble as the response and the other variables as predictors.

(a) Which variables are statistically significant at the 5% level?

```r
lm.fit <- lm(gamble ~ sex + status + income + verbal, data = teengamb)
```

term	estimate	std.error	statistic	p.value
(Intercept)	22.5556506	17.1968034	1.3116188	0.1967736
sex	-22.1183301	8.2111145	-2.6937062	0.0101118
status	0.0522338	0.2811115	0.1858118	0.8534869
income	4.9619792	1.0253923	4.8391032	0.0000179
verbal	-2.9594935	2.1721503	-1.3624718	0.1803109

We see that *gender* and *income* are both significant at the 5% level.

(b) What interpretation should be given to the coefficient for sex?

The variable *sex* is encoded $0 = male, 1 = female$ and the coefficient for it $\beta_{sex} = -22.118$. This means that when all the other variables are held constant and the gender changes from male to female that there will be a -22.118 change in *gamble*.

(c) Fit a model with just income as a predictor and use an F-test to compare it to the full model.

```r
lm.fit.income <- lm(gamble ~ income, data = teengamb)
```

The reduced model $gamble \sim income$

term	estimate	std.error	statistic	p.value
(Intercept)	-6.324559	6.029874	-1.048871	0.2998383
income	5.520485	1.035772	5.329824	0.0000030

Results of the F-test

res.df	rss	df	sumsq	statistic	p.value
45	28008.59	NA	NA	NA	NA

res.df	rss	df	sumsq	statistic	p.value
42	21623.77	3	6384.821	4.133761	0.0117721

Based on the p-value of the F-statistic we do have enough evidence to reject the null hypothesis that the models are equivalent in the variance explained via the RSS statistic. We claim that the full model is better based on the RSS criteria.

Problem 3.4

We are using the sat data for this problem.

(a) Fit a model with total sat score as the response and expend, ratio and salary as predictors. Test the hypothesis that $\beta_{salary} = 0$. Test the hypothesis that $\beta_{salary} = \beta_{ratio} = \beta_{expend} = 0$. Do any of these predictors have an effect on the response?

```r
lm.fit <- lm(total ~ expend + ratio + salary, data = sat)
tidy(lm.fit)
```

term	estimate	std.error	statistic	p.value
(Intercept)	1069.234168	110.924940	9.6392585	0.0000000
expend	16.468866	22.049899	0.7468907	0.4589302
ratio	6.330267	6.542052	0.9676272	0.3382908
salary	-8.822632	4.696794	-1.8784372	0.0666677

We see that salary is significant at the $\alpha = 10\%$ level.

```r
lm.fit.reduced <- lm(total ~ expend + ratio, data = sat)
anova(lm.fit.reduced, lm.fit)
```

Res.Df	RSS	Df	Sum of Sq	F	
47	233442.9	NA	NA	NA	
46	216811.9	1	16631.01	3.528526	
We see th	at the F-st	atist	ic has a p-v	alue of $0.	0667$ - this is the same as the p-value for the t

Test $H_0 : \beta_{salary} = \beta_{ratio} = \beta_{expend} = 0$

```r
lm.fit.null <- lm(total ~ 1, data = sat)

anova(lm.fit.null, lm.fit)
```

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
49	274307.7	NA	NA	NA	NA
46	216811.9	3	57495.74	4.066203	0.0120861

Based on the F-statistic we have enough evidence to reject the null hypothesis that all coefficients are zero. We claim at least one predictor has an effect on the response.

(b) Now add takers to the model. Test the hypothesis that $\beta_{takers} = 0$. Compare this model to the previous one using an F-test. Demonstrate that the F-test and t-test here are equivalent.

Fit the model $total \sim expend + ratio + salary + takers$

```r
lm.fit <- lm(total ~ expend + ratio + salary + takers, data = sat)
tidy(lm.fit)
```

term	estimate	std.error	statistic	p.value
(Intercept)	1045.971536	52.869760	19.7839283	0.0000000
expend	4.462594	10.546528	0.4231339	0.6742130
ratio	-3.624232	3.215418	-1.1271418	0.2656570
salary	1.637917	2.387248	0.6861110	0.4961632
takers	-2.904481	0.231260	-12.5593745	0.0000000

Fir the model $total \sim expend + ratio + salary$ and perform the F-test.

```r
lm.fit.reduced <- lm(total ~ expend + ratio + salary, data = sat)
anova(lm.fit.reduced, lm.fit)
```

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
46	216811.9	NA	NA	NA	NA
45	48123.9	1	168688	157.7379	0

Just as above we see that the F-statistic for the reduced model has a p-value that is the same as the p-value for the t-statistic given above for the coefficient β_{takers}

Problem 3.5 R^2 and the F-test

Find a formula relating R 2 and the F-test for the regression.

Let Ω be the parameter space for a model in p dimensions and ω be the parameter space for a model in q dimensions.

$$R^2_\Omega = 1 - \frac{RSS_\Omega}{TSS}$$
$$R^2_\omega = 1 - \frac{RSS_\omega}{TSS}$$

Solving for TSS in the first case we have that $TSS = \frac{RSS_\Omega}{(1-R^2_\Omega)}$ and putting this into the the expression for R^2_ω

$$R^2_\omega = 1 - \frac{RSS_\omega}{RSS_\Omega}(1 - R^2_\Omega) \implies$$

$$\frac{RSS_\omega}{RSS_\Omega}R^2_\Omega - R^2_\omega = \frac{RSS_\omega - RSS_\Omega}{RSS_\Omega} \implies$$

$$(\frac{RSS_\omega}{RSS_\Omega}R^2_\Omega - R^2_\omega)\frac{dF_\Omega}{df_\Omega - df_\omega} = \frac{(RSS_\omega - RSS_\Omega)/(df_\Omega - df_\omega)}{RSS_\Omega/df_\Omega} \sim F_{df_\Omega - df_\omega, n-df_\Omega}$$

Another way to think about R^2 is that it says something about a model in Ω versus the null model $y \sim \beta_0$. $TSS = \sum(y_i - \bar{y})^2$ in this case will be RSS_{ω_0} - the sum of square residuals for the null model. In this case $df_\omega = 1$ and we can manipulate the expression for $R^2 = 1 - \frac{RSS_\Omega}{RSS_{\omega_0}}$ directly to get

$$1 - R^2 = \frac{RSS_\Omega}{RSS_{\omega_0}} \implies 1 - \frac{RSS_\Omega}{RSS_{\omega_0}} = \frac{R^2}{(1-R^2)}$$

and that *for the sace of comparing a full model against the null model* we have

$$\frac{R^2}{(1-R^2)}\frac{p}{p-1} = \frac{(RSS_{\omega_0} - RSS_\Omega)/(p-1)}{RSS_\Omega/p} \sim F_{p-1,n-p}$$

Problem 3.6 MBA Students

Thirty-nine MBA students were asked about happiness and how this related to their income and social life. The data are found in faraway::happy.

Note, pay attention to warnings in R! GGally has a happy data set as well.

Fit a regression model with happy as the response and the other four variables as predictors.

```
##
## Call:
## lm(formula = happy ~ money + sex + love + work, data = happy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7186 -0.5779 -0.1172  0.6340  2.0651
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.072081   0.852543  -0.085   0.9331
## money        0.009578   0.005213   1.837   0.0749 .
## sex         -0.149008   0.418525  -0.356   0.7240
## love         1.919279   0.295451   6.496 1.97e-07 ***
## work         0.476079   0.199389   2.388   0.0227 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.058 on 34 degrees of freedom
## Multiple R-squared:  0.7102, Adjusted R-squared:  0.6761
## F-statistic: 20.83 on 4 and 34 DF,  p-value: 9.364e-09
```

(a) Which predictors were statistically significant at the 1% level?

We see that money and love are significant at the 1% level.

(b) Use the table function to produce a numerical summary of the response. What assumption used to perform the t-tests seems questionable in light of this summary?

```
table(happy$happy)
```

2	3	4	5	6	7	8	9	10
1	1	4	5	2	8	14	3	1

```
# hist(happy$happy) plot(lm.fit)
```

Wow, I have NO idea what this question is asking me. The assumptions we make are; * Linear relationship * Multivariate normality * No or little multicollinearity * No auto-correlation *

Homoscedasticity - the variance around the regression line is the same for all values of the predictors

All of these assumptions involve elements beyond the distribution of the measured responses $ { y_i } $.

(c) Use the permutation procedure described in Section 3.3 to test the significance of the money predictor.

```
# summary(lm.fit)$coef[2,]# Est, sterr,t-stat,pval for the
nreps <- 4000
tstats <- numeric(nreps)
for (i in 1:nreps) {
    lm.resample.money <- lm(happy ~ sample(money) + sex + love + work, data = happy)
    tstats[i] <- summary(lm.resample.money)$coef[2, 3]  #Get the tstatistic for this re
}
simulated.pvalue <- mean(abs(tstats) > abs(summary(lm.fit)$coef[2, ]))  #Calculate the 

pander(data.frame(simulated.pvalue = simulated.pvalue))
```

simulated.pvalue
0.7432

WE see that the simulated pvalue based on resampling the *money* predictor is very close the value we got from performing the t-test on β_{money}

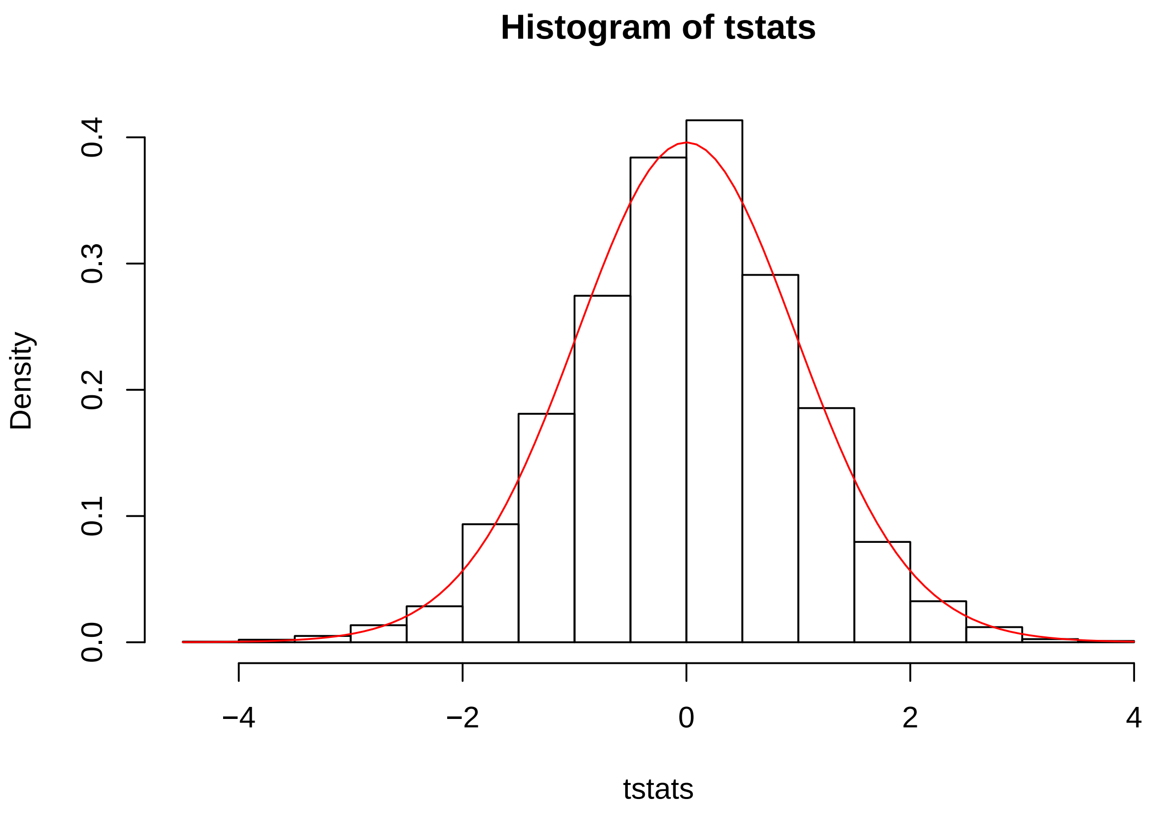
```
pvalue.money <- summary(lm.fit)$coef[2, 4]
pander(data.frame(pvalue.money = pvalue.money))
```

pvalue.money
0.07491

(d) AND (e) Plot a histgram of the permutation t-statistics. Overlay an appropriate t-density

```
hist(tstats, 30, freq = FALSE)
grid <- seq(-4, 4, length = 300)
n <- nrow(happy)
p <- 4 + 1  # Four predictors plus the intercept
df <- n - p
hist(tstats, 30, freq = FALSE)
```

```
curve(dt(x, df = df), add = TRUE, col = "red")
```



(f) Use the bootstrap procedure from Section 3.6 to compute 90% and 95% confidence intervals for *money*. Does zero fall within these confidence intervals? Are these results consistent with previous tests?

```
nb <- 4000
coefmat <- matrix(NA, nb, 5)
resids <- residuals(lm.fit)
preds <- fitted(lm.fit)
for (i in 1:nb) {
    booty <- preds + sample(resids, rep = TRUE)
    bmod <- update(lm.fit, booty ~ .)
    coefmat[i, ] <- coef(bmod)
}
colnames(coefmat) <- c("Intercept", "money", "sex", "love", "work")
coefmat <- data.frame(coefmat)
apply(coefmat, 2, function(x) quantile(x, c(0.05, 0.95)))
```

	Intercept	money	sex	love	work
5%	-1.353335	0.0013456	-0.7961964	1.457376	0.1663225
95%	1.262513	0.0178614	0.4625778	2.375133	0.7850518

We see that for a significance of $\alpha = 10\%$ that we have enough evidence to reject the null hypothesis that the coefficient for *money* is zero. This is the same result for the permutation test and for the t-test that is performed as part of R's `lm.summary` routine. Now we look at the 95 confidence interval.

```
apply(coefmat, 2, function(x) quantile(x, c(0.025, 0.975)))
```

	Intercept	money	sex	love	work
2.5%	-1.605365	-0.0001803	-0.9232030	1.388539	0.1053054
97.5%	1.492445	0.0196490	0.5813499	2.480985	0.8416759

Since 0 is in the interval for *money*; at a significance of $\alpha = 5\%$, with the data at hand, we do *not* have enough evidence to reject the null hypothesis that the coefficient for *money* is zero in the linear model $happy \sim money + sex + love + work$

Problem 4.2 - prediction with the teengamb data set.

Using the teengamb data, fit a model with gamble as the response and the other variables as predictors.

```
rm(list = ls())
data(teengamb, package = "faraway")
lm.fit <- lm(gamble ~ ., data = teengamb)
```

(a) Predict the amount that a male with average (given these data) status, income and verbal score would gamble along with an appropriate 95% CI.

```
x <- model.matrix(lm.fit)
x0 <- apply(x, 2, mean)
# The question asks for a male and here we set that value
x0["sex"] <- 0
# predict(lm.fit,new=data.frame(t(x0)))
pi <- predict(lm.fit, new = data.frame(t(x0)), interval = "confidence", level = 0.95)
pi
```

fit	lwr	upr
28.24252	18.78277	37.70227

```
pander(data.frame(pi.width = pi[3] - pi[2]), caption = "Confidence interval width")
```

Table 22: Confidence interval width

pi.width
18.92

(b) Repeat the prediction for a male with maximal values (for this data) of status, income and verbal score. Which CI is wider and why is this result expected?

```
x <- model.matrix(lm.fit)
x0 <- apply(x, 2, max)
# The question asks for a male and here we set that value
x0["sex"] <- 0
# predict(lm.fit,new=data.frame(t(x0)))
pi <- predict(lm.fit, new = data.frame(t(x0)), interval = "confidence", level = 0.95)
pi
```

fit	lwr	upr
71.30794	42.23237	100.3835

```
pander(data.frame(pi.width = pi[3] - pi[2]), caption = "Confidence interval width")
```

Table 24: Confidence interval width

pi.width
58.15

(c) Fit a model with sqrt(gamble) as the response but with the same predictors. Now predict the response and give a 95% prediction interval for the individual in (a). Take care to give your answer in the original units of the response.

```
lm.fit <- lm(sqrt(gamble) ~ ., data = teengamb)
x <- model.matrix(lm.fit)
x0 <- apply(x, 2, mean)
x0["sex"] <- 0
pi <- predict(lm.fit, new = data.frame(t(x0)), interval = "confidence", level = 0.95)
```

```
# ORIG
pi.orig <- c(pi[1]^2, pi[2]^2, pi[3]^2)
pi.orig
```

```
## [1] 16.39864 10.11670 24.19037
```

```
pander(data.frame(pi.width = pi.orig[3] - pi.orig[2]), caption = "Confidence interval wi
```

Table 25: Confidence interval width

pi.width
14.07

The square root transform is known to stabilize variance and we see that in the smaller prediction interval.

(d) Repeat the prediction for the model in (c) for a female with status=20, income=1, verbal = 10. Comment on the credibility of the result.

```
x0["sex"] <- 1
x0["status"] <- 20
x0["income"] <- 1
x0["verbal"] <- 10

pi <- predict(lm.fit, new = data.frame(t(x0)), interval = "confidence", level = 0.95)
# ORIG
pi
```

fit	lwr	upr
-2.08648	-4.445938	0.272978

```
pi.orig <- c(-pi[1]^2, pi[2]^2, pi[3]^2)
pi.orig
```

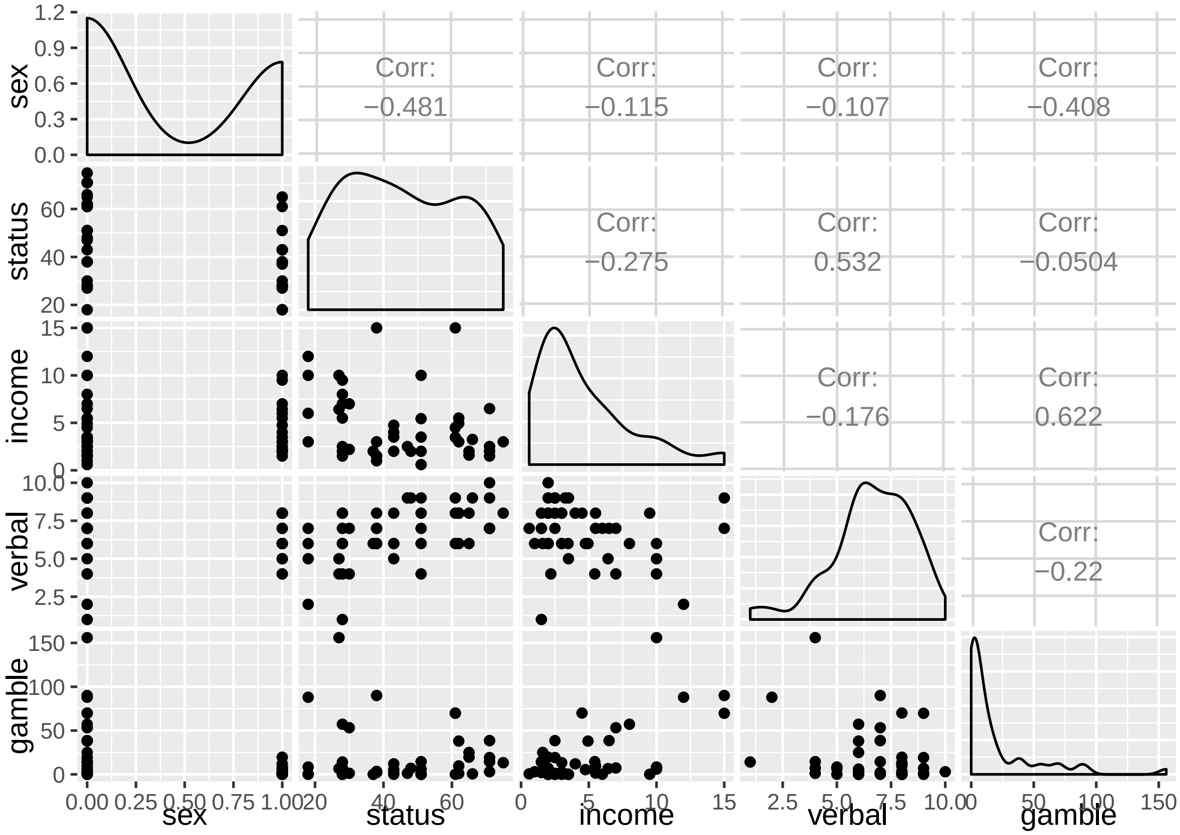
```
## [1] -4.35339768 19.76636002  0.07451699
```

```
pander(data.frame(pi.width = pi.orig[3] - pi.orig[2]), caption = "Confidence interval wi
```

Table 27: Confidence interval width

pi.width
-19.69

```r
ggpairs(teengamb)
```



```r
lm.fit <- lm(gamble ~ ., data = teengamb)
pi <- predict(lm.fit, new = data.frame(t(x0)), interval = "confidence", level = 0.95)
pi
```

fit	lwr	upr
-23.15096	-48.84003	2.538117

NCSU ST 503 HW 6

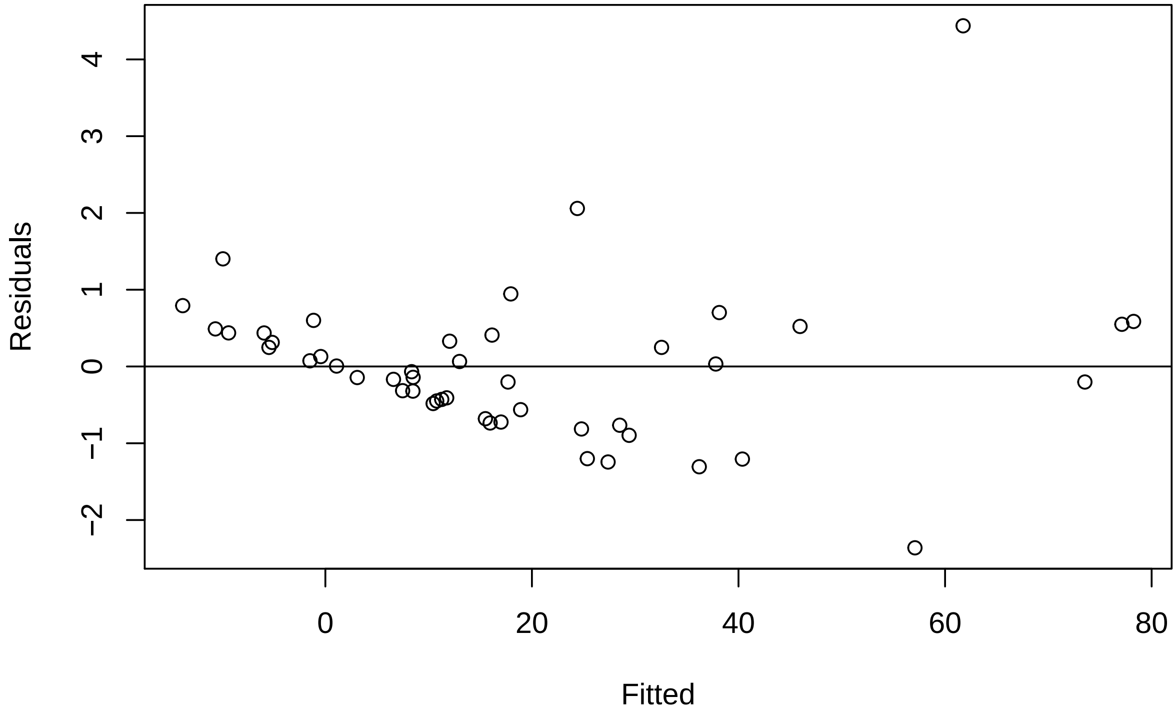
Probems 6.2,6.3,6.4,6.5,6.8 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

26 September, 2017

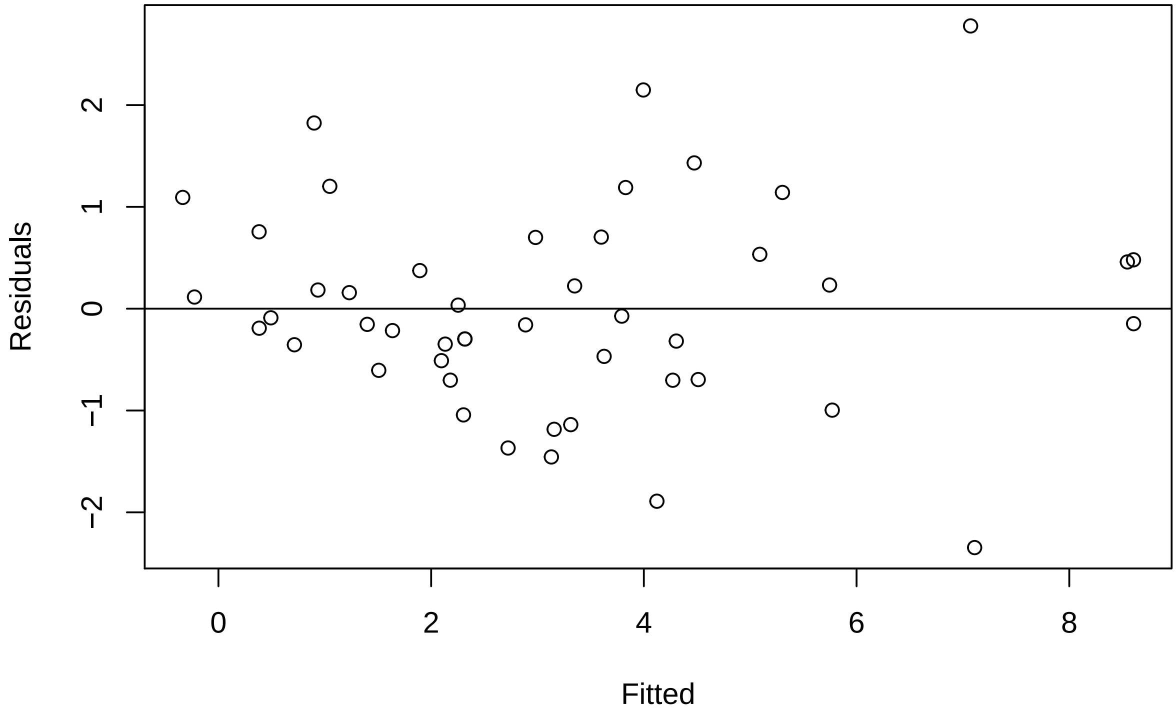
6.2 Using the teengamb dataset, fit a model with gamble as the response and the other variables as predictors.

(a) Check the constant variance assumption for the errors.

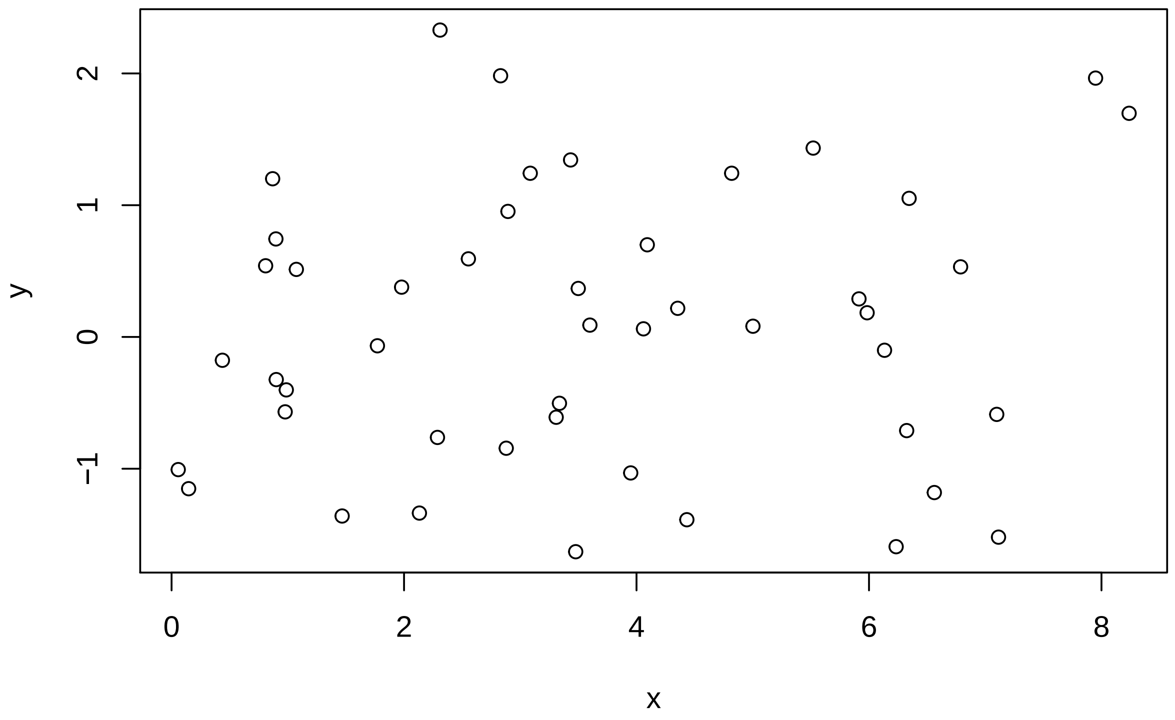


To check the assumption of constant variance we plot fitted values against the standardized residuals - looking for any structure in the distribution of values about the theoretical mean value line $E[\epsilon] = 0$. There appears to be structure and heteroskedasticity in the plot.

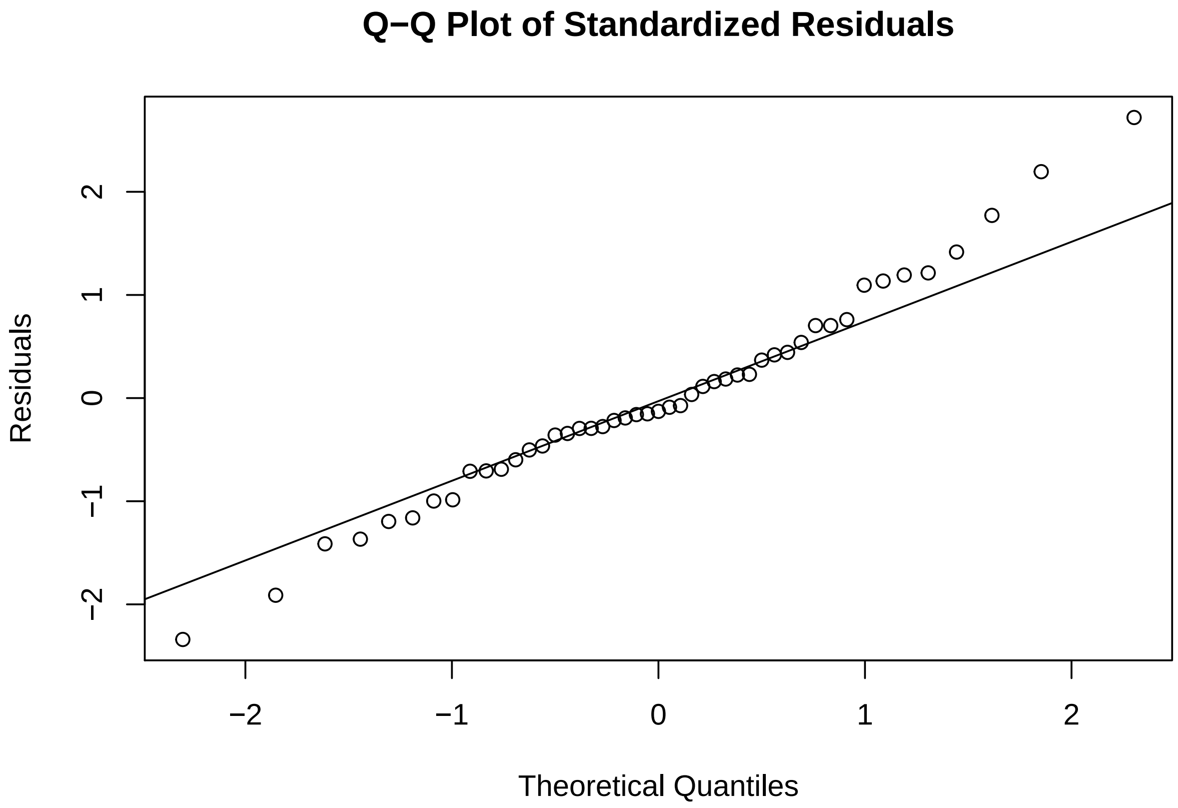
Below we plot the fitted values against the residuals for a model where the response has been transformed with the square root function. We see less structure and a more evenly distributed variance. We do see even with the transformed response evidence that the variance is not constant.



For reference we plot below what constant $N(0,1)$ error over the same range of the response would look like for the same number of data points. We ran this a number of times to get a good idea of what constant variance looks like with this number of points. It's helpful to calibrate this way when evaluating whether variance is constant for a small and medium data sets.


y
2
1
0
−1
0
2
4
6
8
x

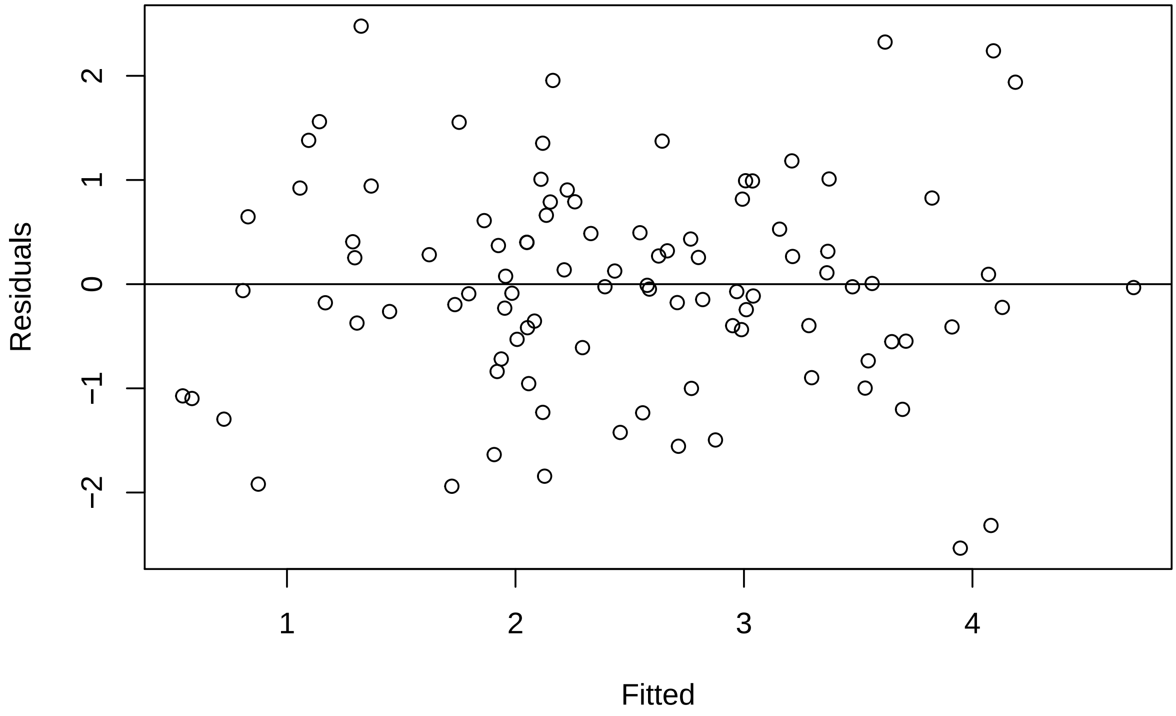
(b) Check the normality assumption.



We see clear evidence of long tails in the distribution of the residuals.

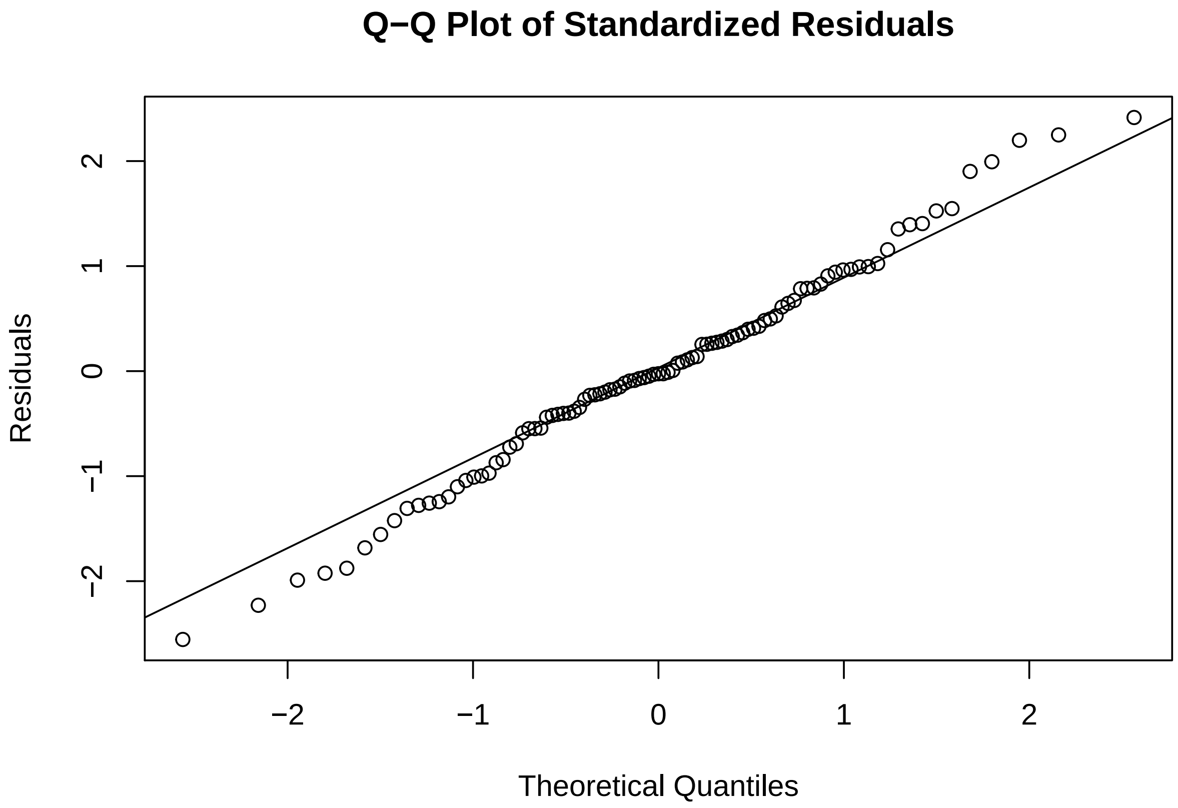
6.3 For the prostate data, fit a model with lpsa as the response and the other variables as predictors.

(a) Check the constant variance assumption for the errors.



The variance of the standardized residuals appears constant over the range of the fitted values. We're comfortable claiming homoskedasticity of residuals for this data set.

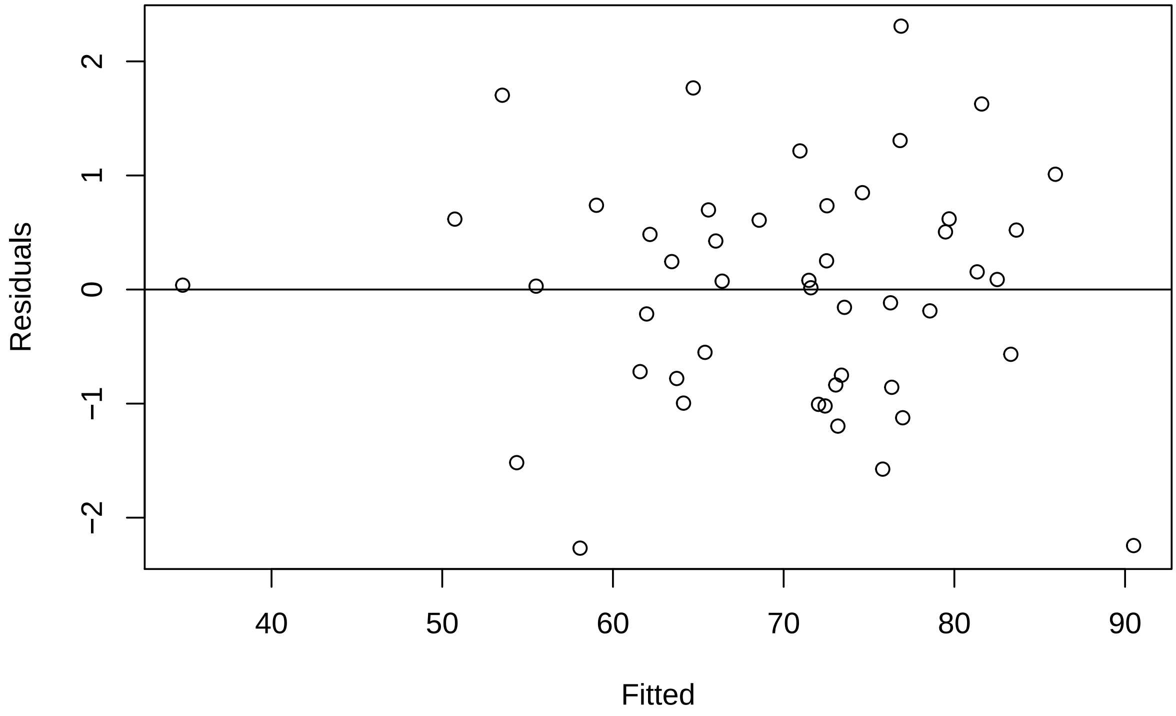
(b) Check the normality assumption.



The studentized residuals appear to be slightly long tailed.

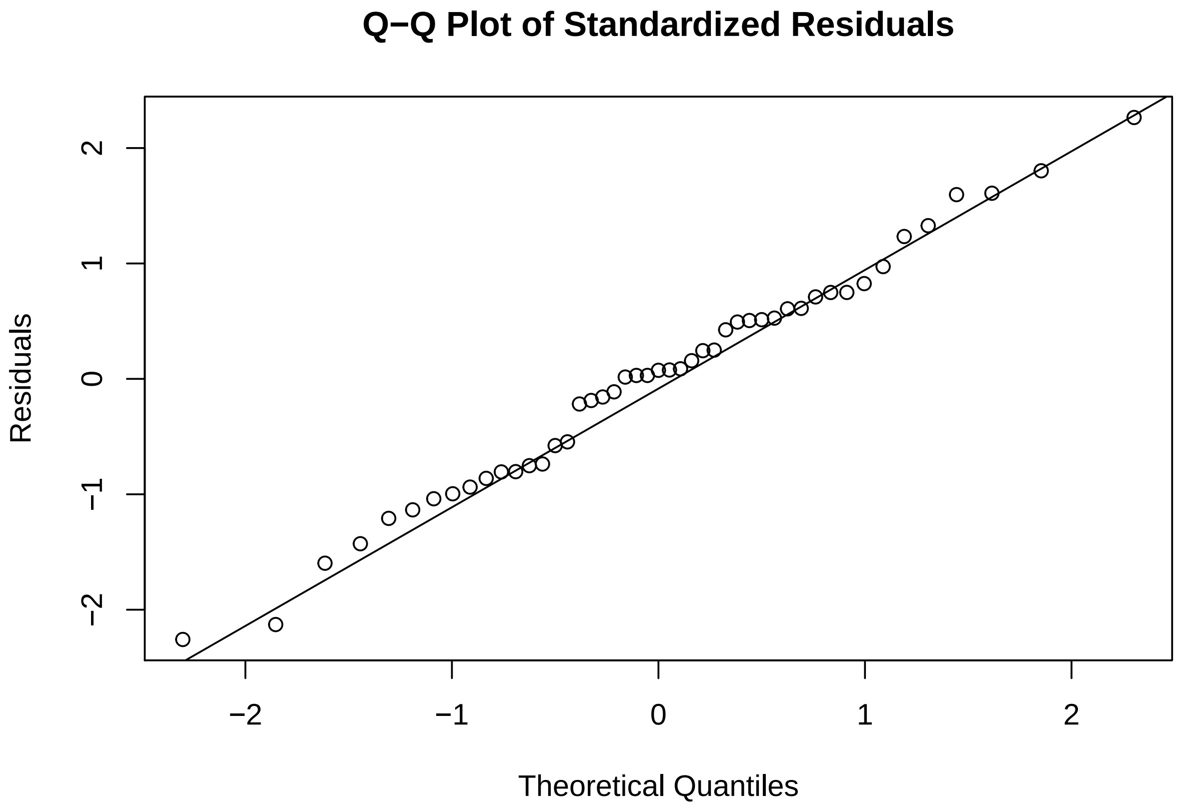
6.4 For the swiss data, fit a model with Fertility as the response and the other variables as predictors.

(a) Check the constant variance assumption for the errors.



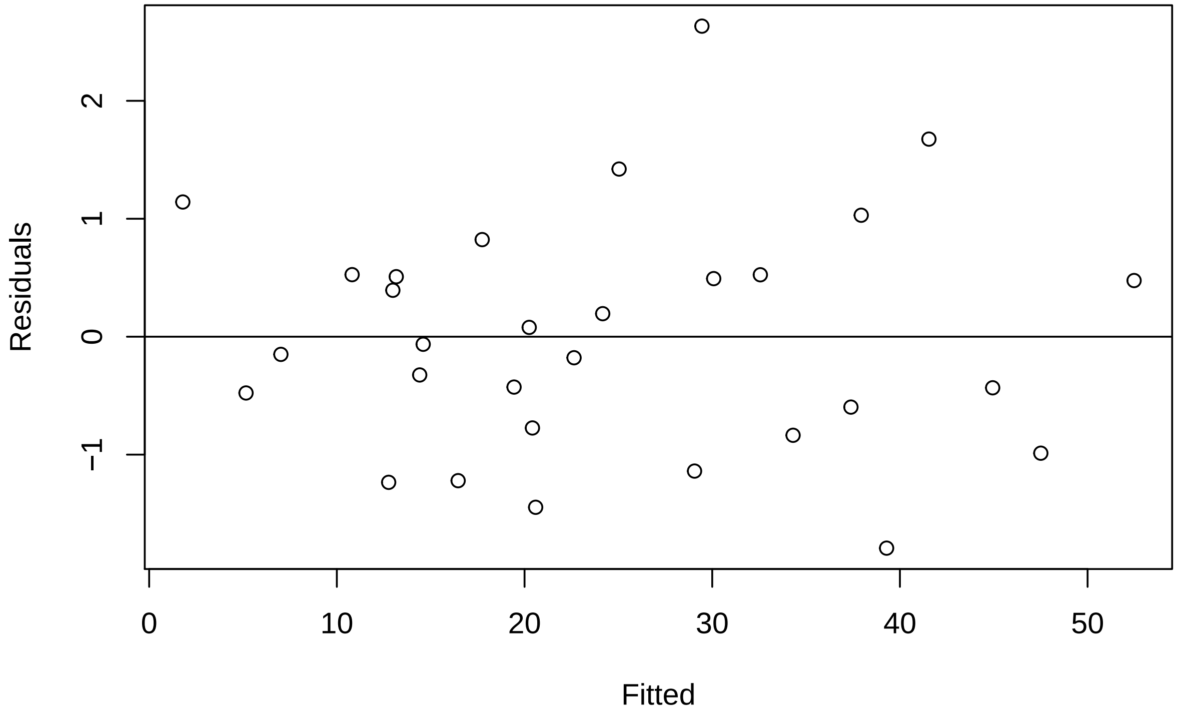
The variance of the standardized residuals appears constant over the range of the fitted values. We're comfortable claiming homoskedasticity of residuals for this data set.

(b) Check the normality assumption.



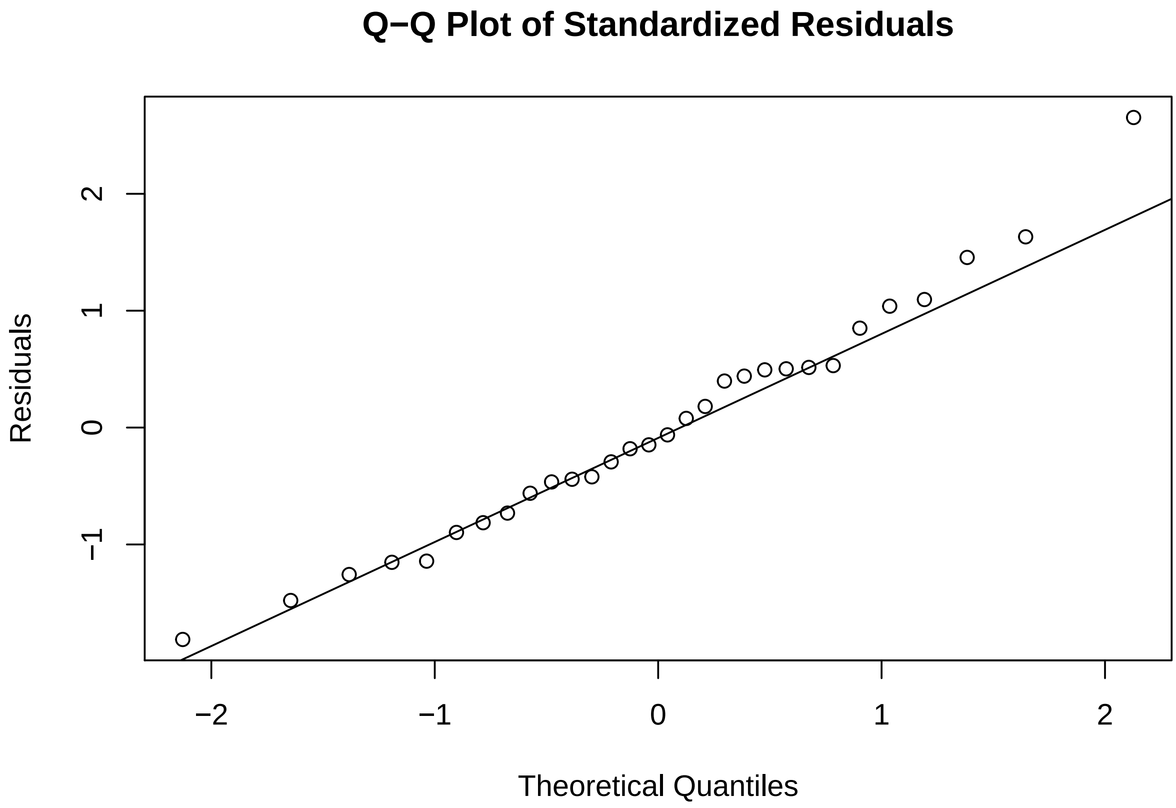
6.5 Using the cheddar data, fit a model with taste as the response and the other three variables as predictors.

(a) Check the constant variance assumption for the errors.



The variance of the residuals appears constant over the range of the fitted values. We're comfortable claiming homoskedasticity of residuals for this data set.

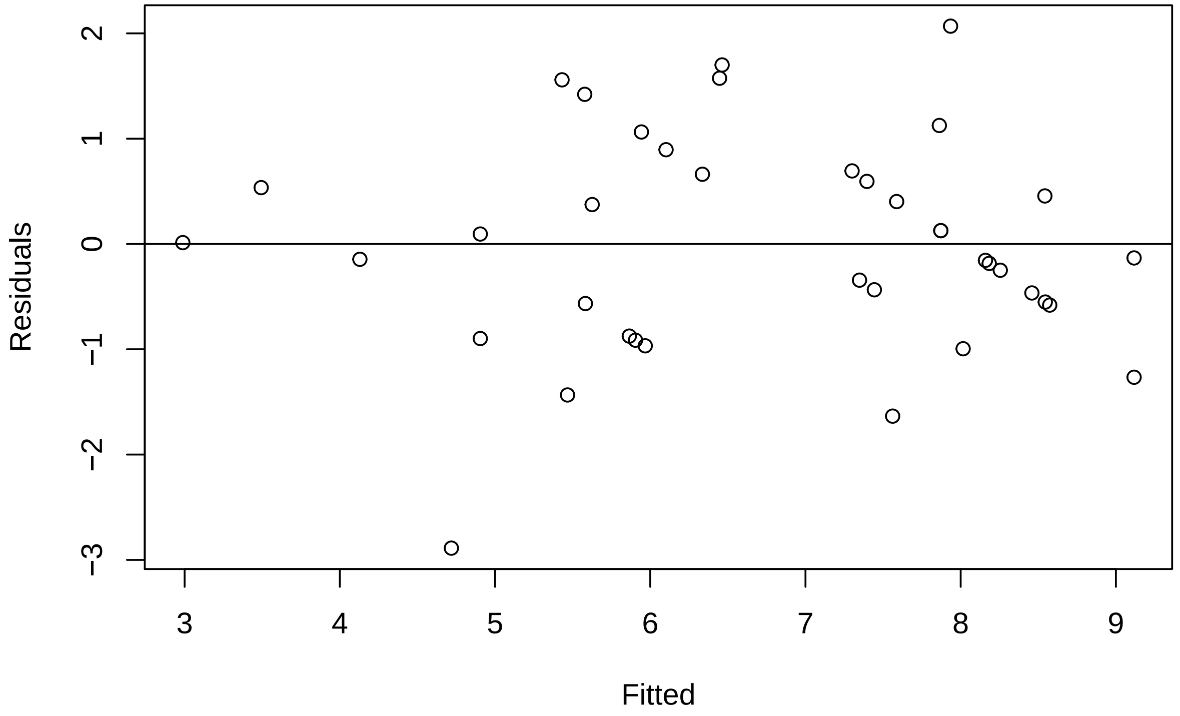
(b) Check the normality assumption.



The standardized residuals appear to be normally distributed.

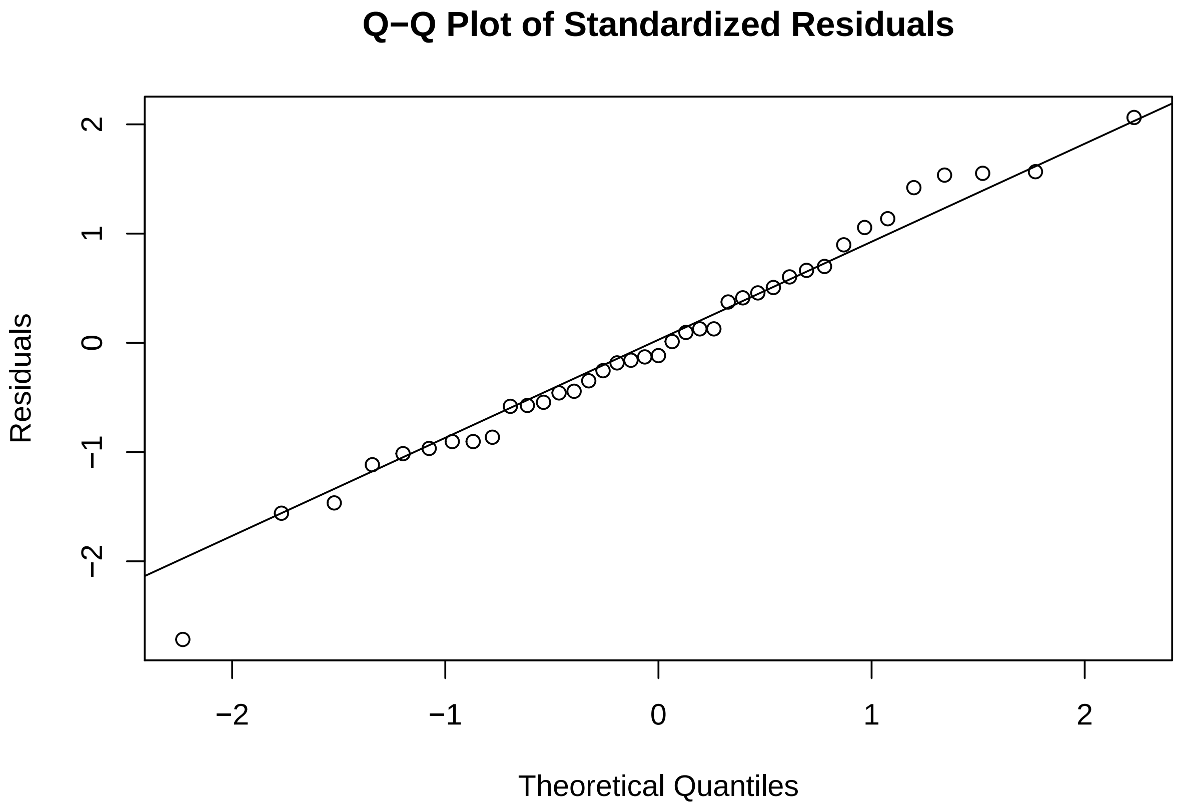
6.6 Using the happy data, fit a model with happy as the response and the other four variables as predictors.

(a) Check the constant variance assumption for the errors.



We see structure in the plot of the residuals. There is serial correlation in the residuals - which is a red flag for our model. The variance of the residuals appears slightly lower over the low end of the range of the response, and higher at the high end of the response. This is difficult to judge though since there are only a few points at the low end of the range of the response.

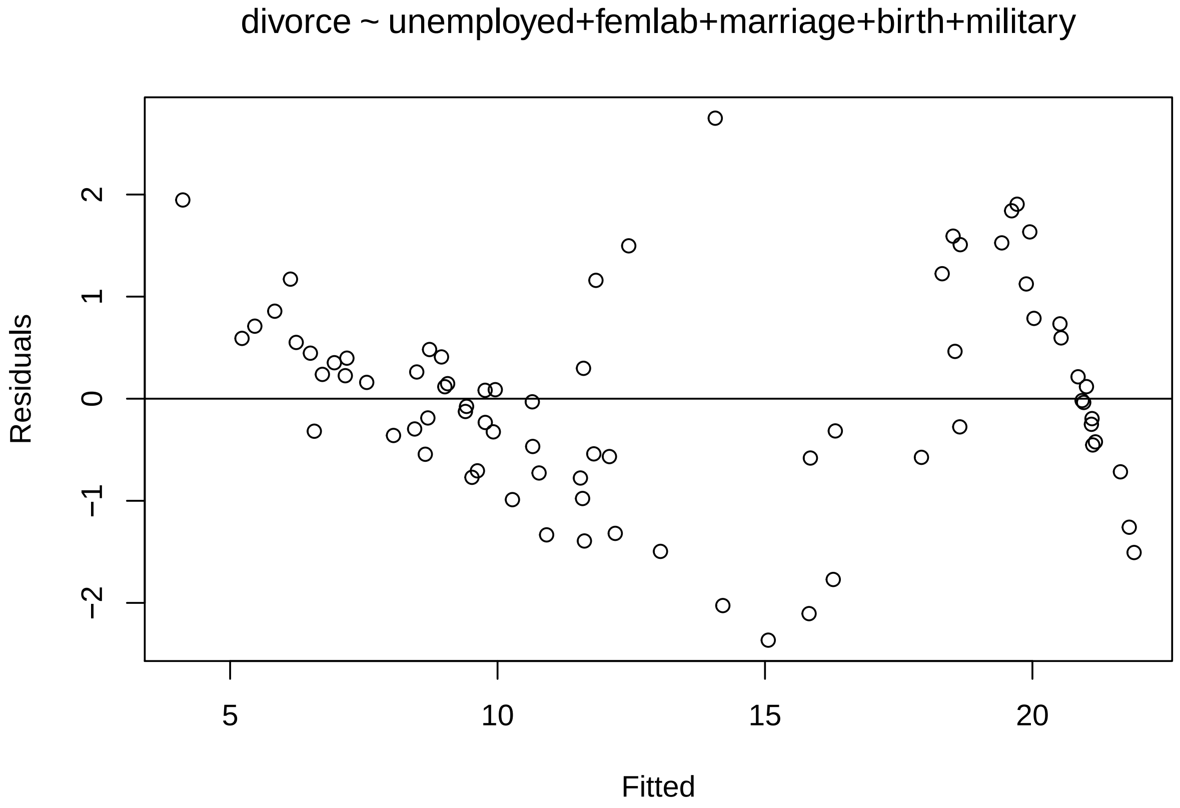
(b) Check the normality assumption.



The standardized residuals appear to be normally distributed. This is interesting in light of the results from part a).

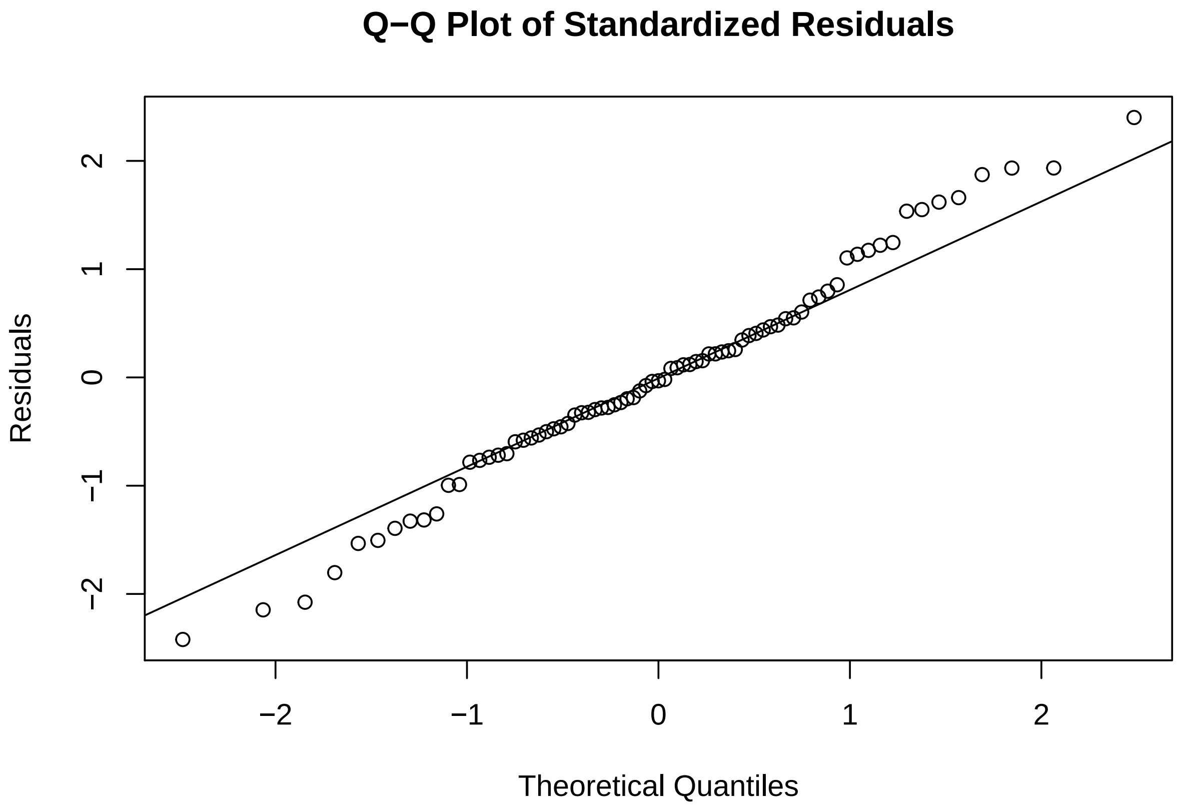
6.8 For the divusa data, fit a model with divorce as the response and the other variables, except year as predictors.

(a) Check the constant variance assumption for the errors.



We see clear structure and serial correlation in the residuals. We may want to plot the response against some of the predictors to look for which ones may be candidates for polynomial terms in the model.

(b) Check the normality assumption.



There is some evidence for mild long tail behavior in the residuals.

(c) Check for large leverage points.

Table 1: High Leverage Data Elements

	year	divorce	unemployed	femlab	marriage	birth	military
13	1932	6.1	23.6	24.46	56	81.7	1.96
14	1933	6.1	24.9	24.89	61.3	76.3	1.94
24	1943	11	1.9	35.7	83	94.3	66.15
25	1944	12	1.2	36.3	76.5	88.8	82.75
26	1945	14.4	1.9	35.8	83.6	85.9	86.64
27	1946	17.9	3.9	30.8	118.1	101.9	21.43

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

(d) Check for outliers.

Table 2: Range of Studentized residuals

range.residuals.left	range.residuals.right
-2.447	2.886

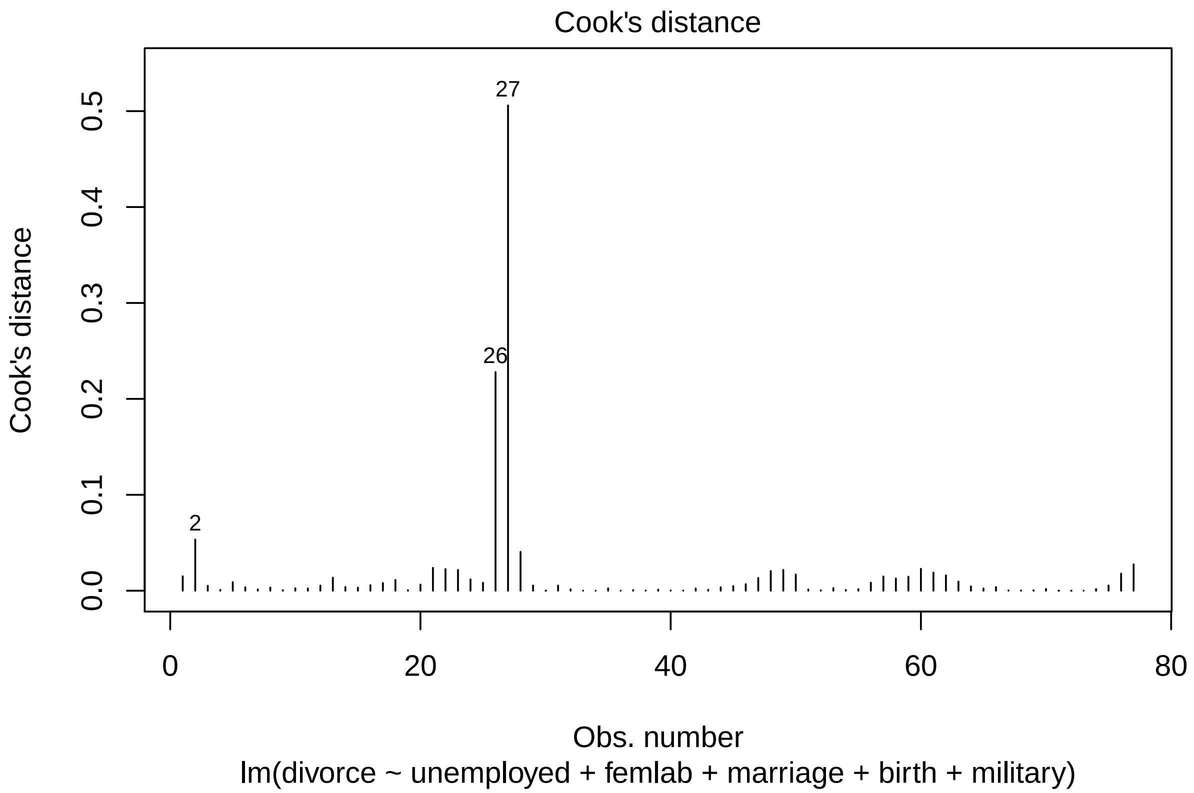
Table 3: Bonferroni corrected t-value

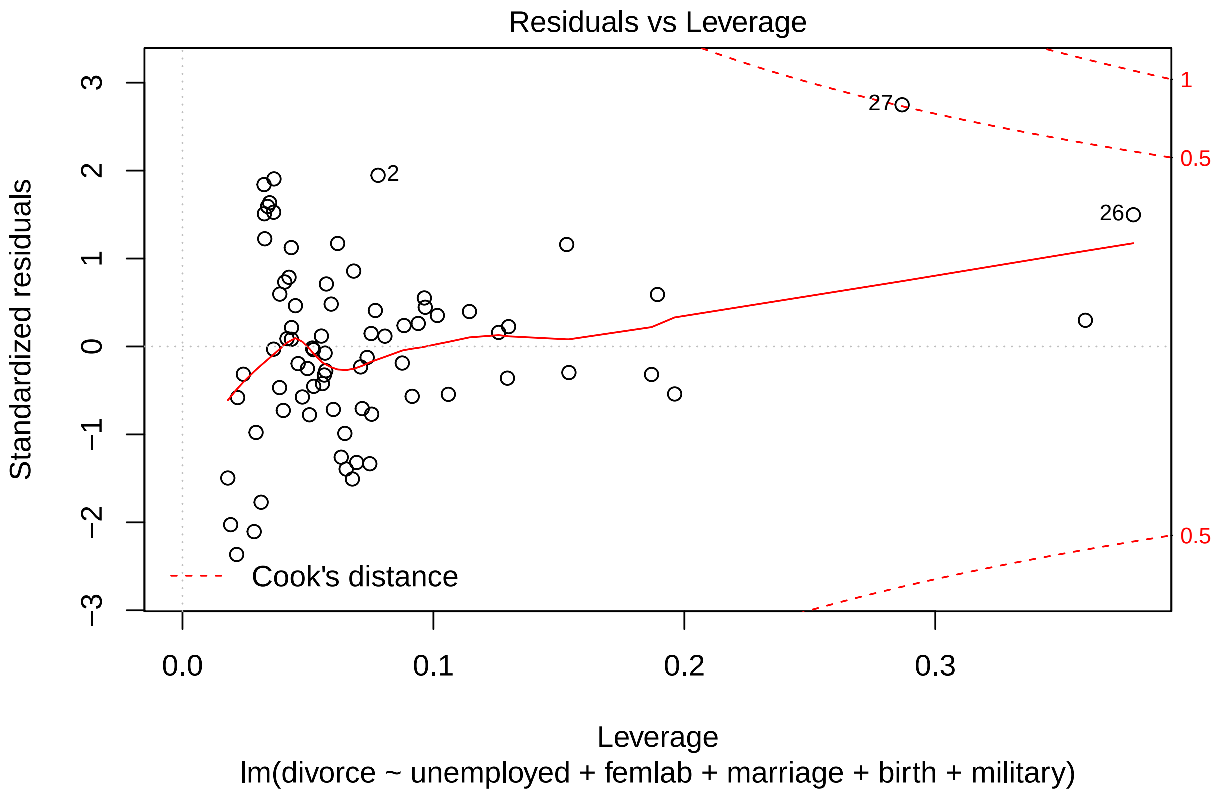
t.val.alpha
-3.57

Since none of the studentized residuals fall outside the interval given by the Bonferroni corrected t-values we claim there are no outliers in the dataset.

(e) Check for influential points.

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


Cook's distance
Cook's distance
0.0
0.1
0.2
0.3
0.4
0.5
27
26
2
0
20
40
60
80
Obs. number
lm(divorce ~ unemployed + femlab + marriage + birth + military)



We see two clear high leverage points - elements 26 and 27. A third is labelled by R, but the leverage doesn't seem very large. The book does not discuss a criteria for selecting influential points from the Cook distances.

Some guidelines for selecting influential points; * points with a Cook distance more than three times the mean Cook distance
* points with a Cook distance greater than 4/n * points with a cook distance greater than 1

Here we select points with a Cook distance more than three times the mean Cook distance.

Table 4: Mean Cook distance

mean.cooks.distance
0.01712

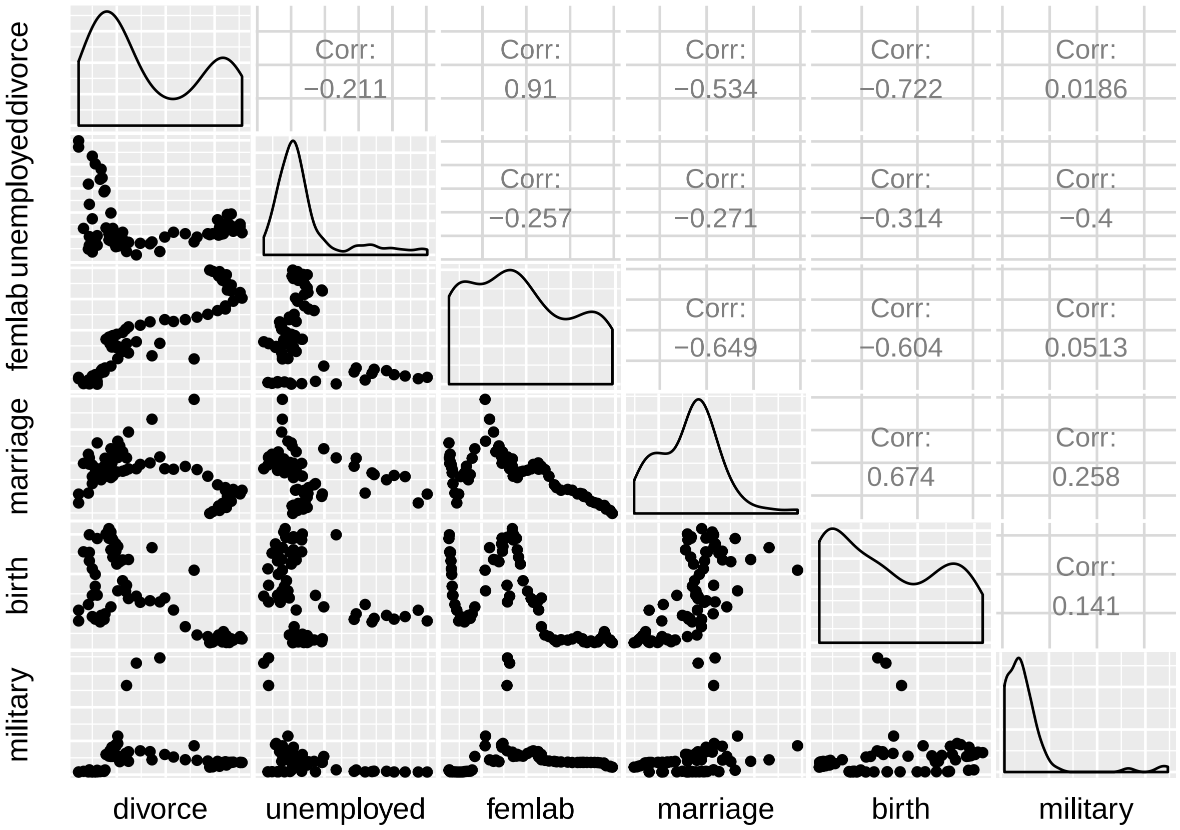
Table 5: Points with Cook distance greater than three times the mean Cook distance.

	cook.distance
2	0.05337
26	0.2279

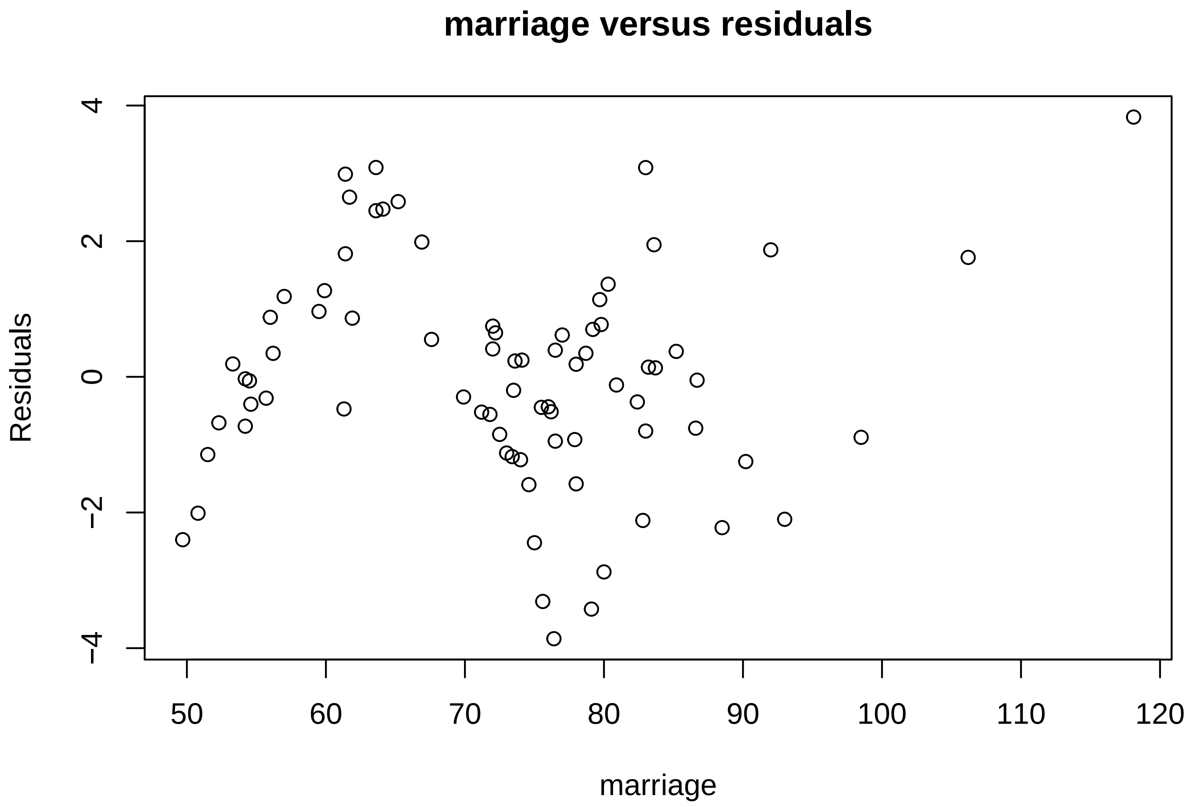
	cook.distance
27	0.5059

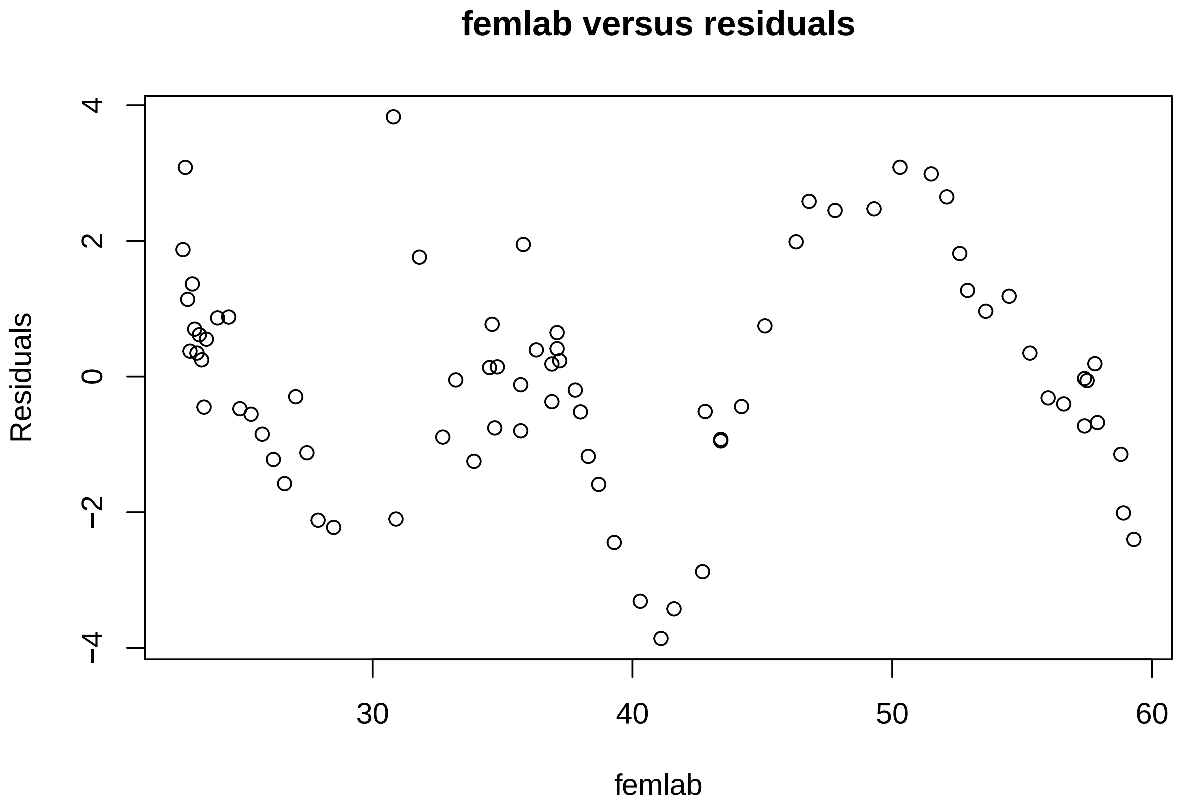
(f) Check for structure in the model.

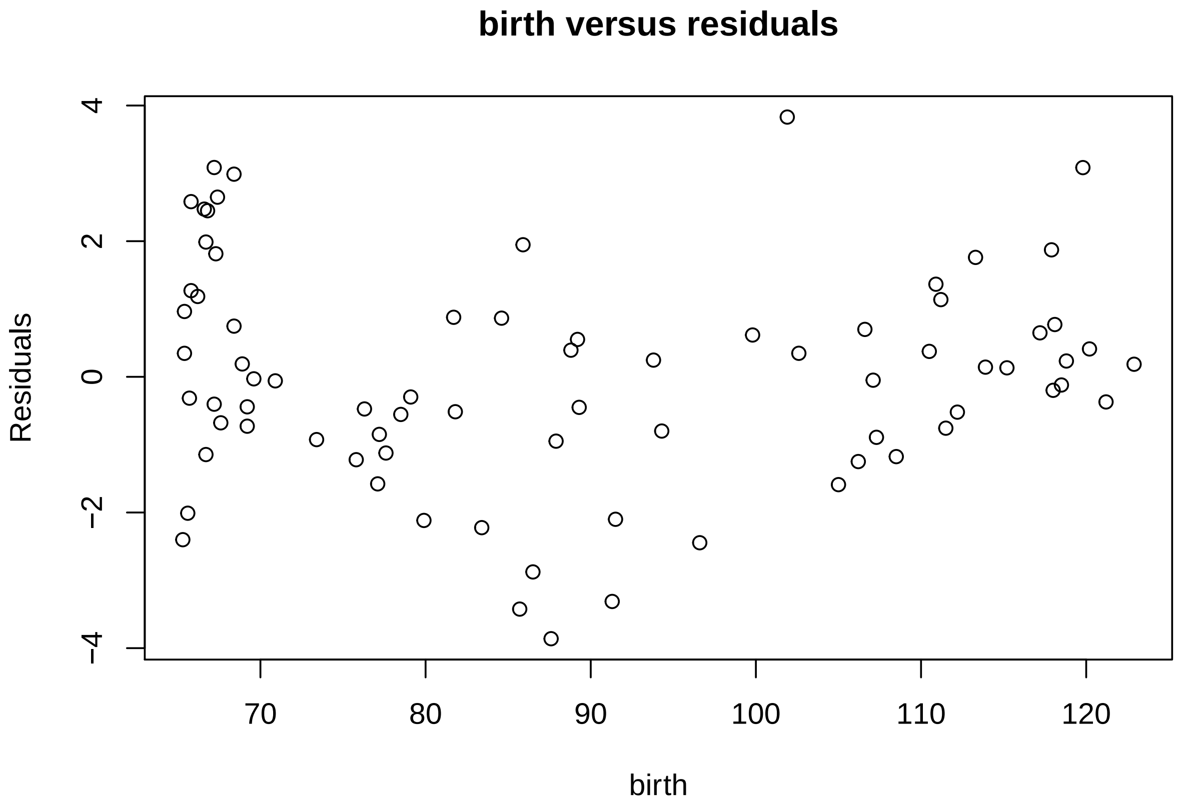
We saw evidence for additional structure not accounted for by the model. First a plot of the variables may help guide the next steps.



We plotted residuals against all the predictors and found that $femlab$ and $marriage$ had the most structure. These are the likely candidates for including additional terms in the regression. It's apparent that a third order polynomial would be appropriate. We plot $birth$ versus residuals because we found out later that adding in polynomial terms for that reduced the structure we saw in the residuals versus fitted plot.

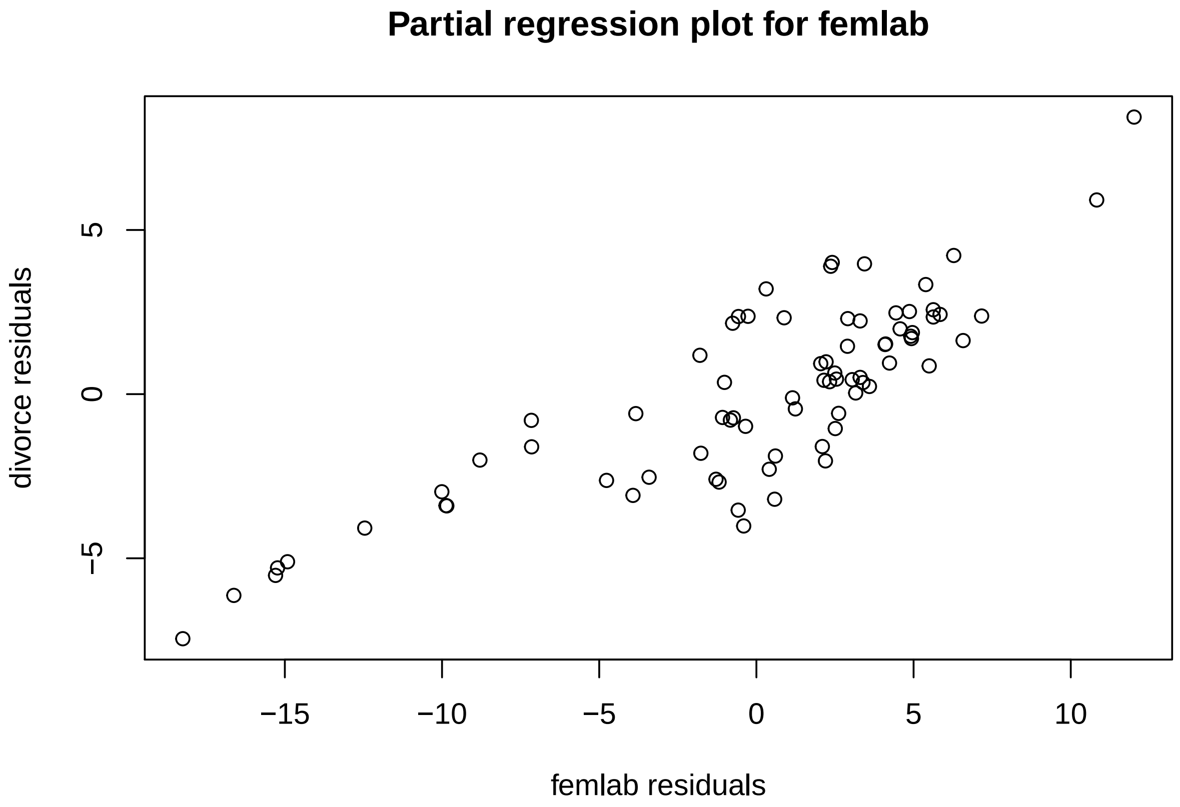

marriage versus residuals
Residuals
4
2
0
−2
−4
50
60
70
80
90
100
110
120
marriage


femlab versus residuals
Residuals
femlab
4
2
0
−2
−4
30
40
50
60

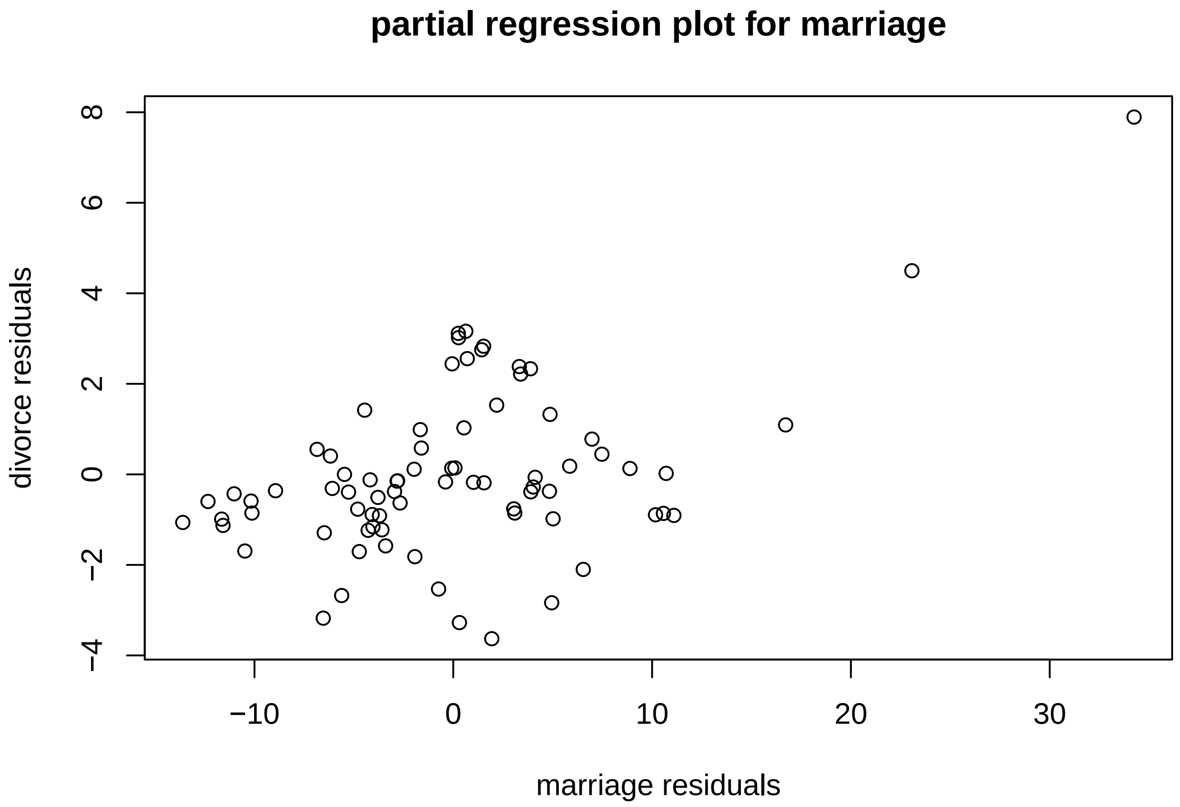


Before we try to remove the unexplained structure let's investigate the partial regression / added variable plot for these variables.

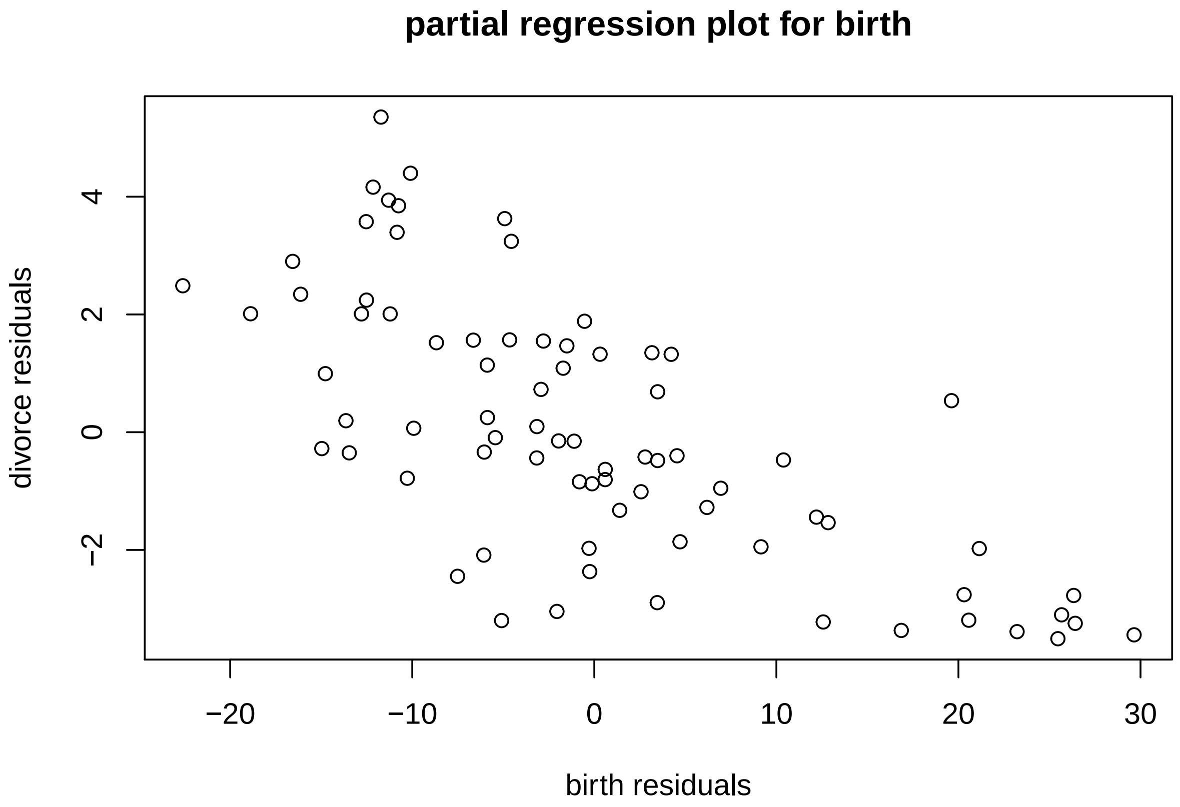
This is the partial regression plot for $femlab$



This is the partial regression plot for *marriage*

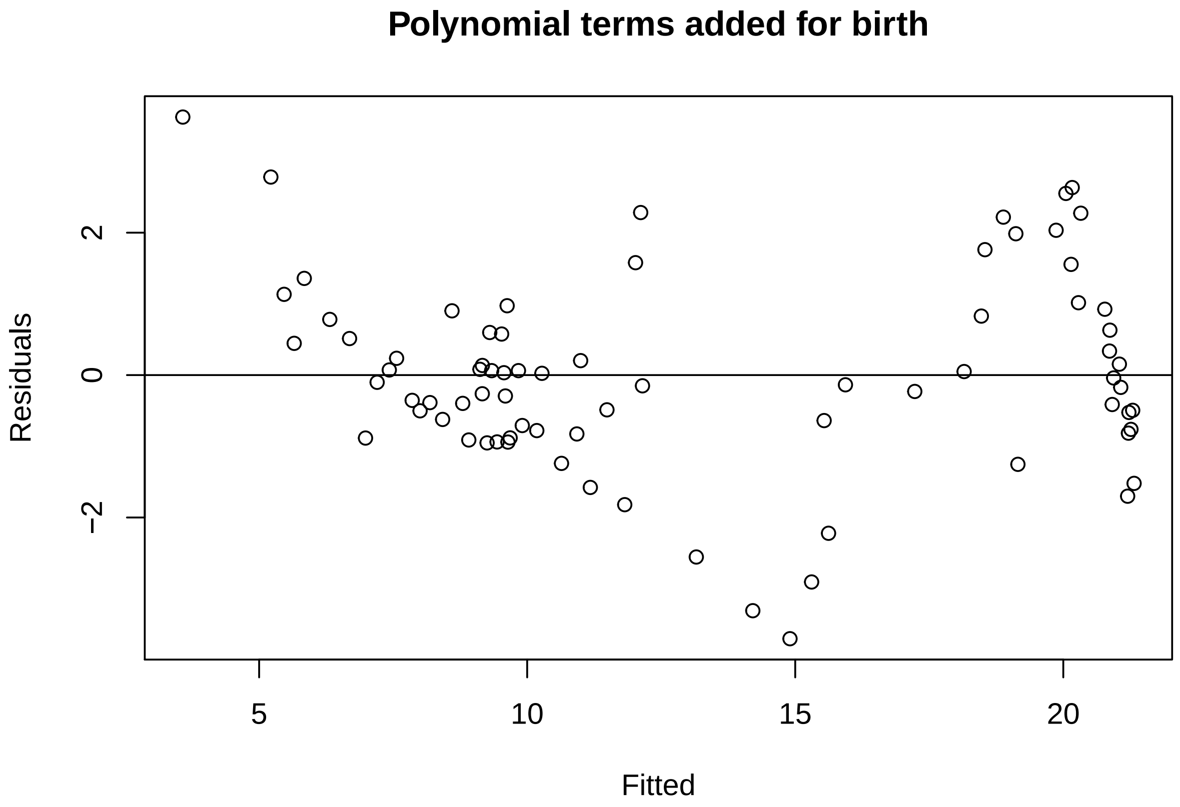


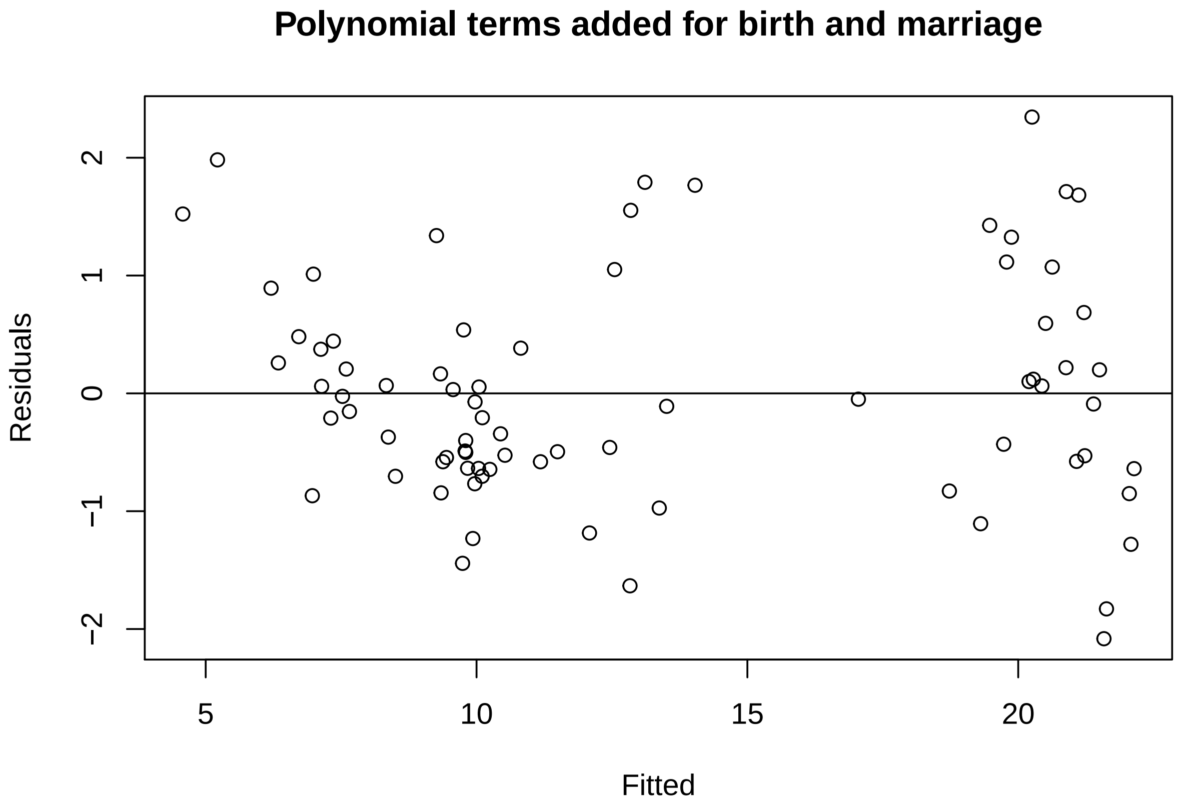
This is the partial regression plot for *birth*



I'm not sure why we don't see non-linearity in these plots. I'll return to the theory behind this and investigate - hopefully before the homework is due! for now let's see if introduction of polynomial terms reduces the structure in the residuals versus fitted plot.

We tried adding in polynomial terms for marriage and femlab. It was not until we added polynomial terms for birth and marriage that the structure in the residuals was reduced. The residuals versus fitted for the models with polynomial terms


Polynomial terms added for birth
Residuals
2
0
−2
5
10
15
20
Fitted


Polynomial terms added for birth and marriage
Residuals
2
1
0
−1
−2
5
10
15
20
Fitted

NCSU ST 503 HW 6

Probems 7.4, 7.6, 7.8 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

10 October, 2017

7.4 longley data analyis

```
##                  Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  -3.4823e+03  8.9042e+02 -3.9108 0.0035604
## GNP.deflator  1.5062e-02  8.4915e-02  0.1774 0.8631408
## GNP          -3.5819e-02  3.3491e-02 -1.0695 0.3126811
## Unemployed   -2.0202e-02  4.8840e-03 -4.1364 0.0025351
## Armed.Forces -1.0332e-02  2.1427e-03 -4.8220 0.0009444
## Population   -5.1104e-02  2.2607e-01 -0.2261 0.8262118
## Year          1.8292e+00  4.5548e-01  4.0159 0.0030368
##
## n = 16, p = 7, Residual SE = 0.30485, R-Squared = 1
```

(a) Compute and comment on the condition numbers.

We were not asked to do this, but it's interesting - so we give it a try.

Table 1: eigenvalues

ev.1	ev.2	ev.3	ev.4	ev.5	ev.6
66652993	209073	105355	18040	24.56	2.015

Table 2: condition numbers

rcond.1	rcond.2	rcond.3	rcond.4	rcond.5	rcond.6
1	17.86	25.15	60.78	1647	5751

We see a high condition number κ, and note that $\sqrt{\frac{\lambda_1}{\lambda_i}} > 30$ for $i = 4, 5$ as well where λ_i denotes the sorted eigenvalues.

(b) Compute and comment on the correlations between the predictors.

Table 3: Correlation (continued below)

	corr.mat.GNP.deflator	corr.mat.GNP	corr.mat.Unemployed
GNP.deflator	1	0.99	0.62
GNP	0.99	1	0.6
Unemployed	0.62	0.6	1
Armed.Forces	0.46	0.45	-0.18
Population	0.98	0.99	0.69
Year	0.99	1	0.67

	corr.mat.Armed.Forces	corr.mat.Population	corr.mat.Year
GNP.deflator	0.46	0.98	0.99
GNP	0.45	0.99	1
Unemployed	-0.18	0.69	0.67
Armed.Forces	1	0.36	0.42
Population	0.36	1	0.99
Year	0.42	0.99	1

We see a significant amount of correlation between the predictors.

(c) Compute the variance inflation factors.

Table 5: Variance Inflation Factors (continued below)

VIF.GNP.deflator	VIF.GNP	VIF.Unemployed	VIF.Armed.Forces
135.5	1789	33.62	3.589

VIF.Population	VIF.Year
399.2	759

The variance inflation factor for all but the Armed.Forces predictor is large.

We look at a reduced model below. This was iteratively defined by removing predictors with high condition numbers and VIF factors. We note that the R^2 is comparable to the original model.

```
##                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)  30.3934125  1.8642850 16.3030 1.494e-09
## GNP.deflator  0.3980758  0.0309950 12.8432 2.261e-08
## Unemployed   -0.0107250  0.0032205 -3.3303  0.005995
## Armed.Forces -0.0081653  0.0038294 -2.1322  0.054337
##
## n = 16, p = 4, Residual SE = 0.67763, R-Squared = 0.97
```

Table 7: eigenvalues

ev.1	ev.2	ev.3
2975066	112877	1589

Table 8: condition numbers

rcond.1	rcond.2	rcond.3
1	5.134	43.28

	GNP.deflator	Unemployed	Armed.Forces
GNP.deflator	1.00	0.62	0.46
Unemployed	0.62	1.00	-0.18
Armed.Forces	0.46	-0.18	1.00

Table 10: Variance Inflation Factors

VIF.GNP.deflator	VIF.Unemployed	VIF.Armed.Forces
3.655	2.959	2.32

7.6 cheddar dataset analysis

Using the cheddar data, fit a linear model with taste as the response and the other three variables as predictors.

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -28.87677   19.73542 -1.4632 0.155399
## Acetic        0.32774    4.45976  0.0735 0.941980
## H2S           3.91184    1.24843  3.1334 0.004247
## Lactic       19.67054    8.62905  2.2796 0.031079
##
## n = 30, p = 4, Residual SE = 10.13071, R-Squared = 0.65
```

(a) Is the predictor Lactic statistically significant in this model?

We see that lactic is statistically significant at a level of $\alpha = 0.05$ with a p-value of 0.031

(b) Give the R command to extract the p-value for the test of $\beta_{lactic} = 0$. Hint: look at faraway::sumary()$coef.

After some trial and error we got the command below.

```r
summary(lm.fit)$coefficients[4, 4]
```

```
## [1] 0.03107948
```

We really do not like to index model parameters by the numerical index. If this were production code we'd look for a way to use the predictor name directly. StackOverflow provided us the hint for the code below.

```r
coef(summary(lm.fit))["Lactic", "Pr(>|t|)"]
```

```
## [1] 0.03107948
```

(c) Add normally distributed errors to Lactic with mean zero and standard deviation 0.01 and refit the model. Now what is the p-value for the previous test?

```
## [1] 0.03014723
```

(d) Repeat this same calculation of adding errors to Lactic 1000 times within for loop. Save the p-values into a vector. Report on the average p-value. Does this much measurement error make a qualitative difference to the conclusions?

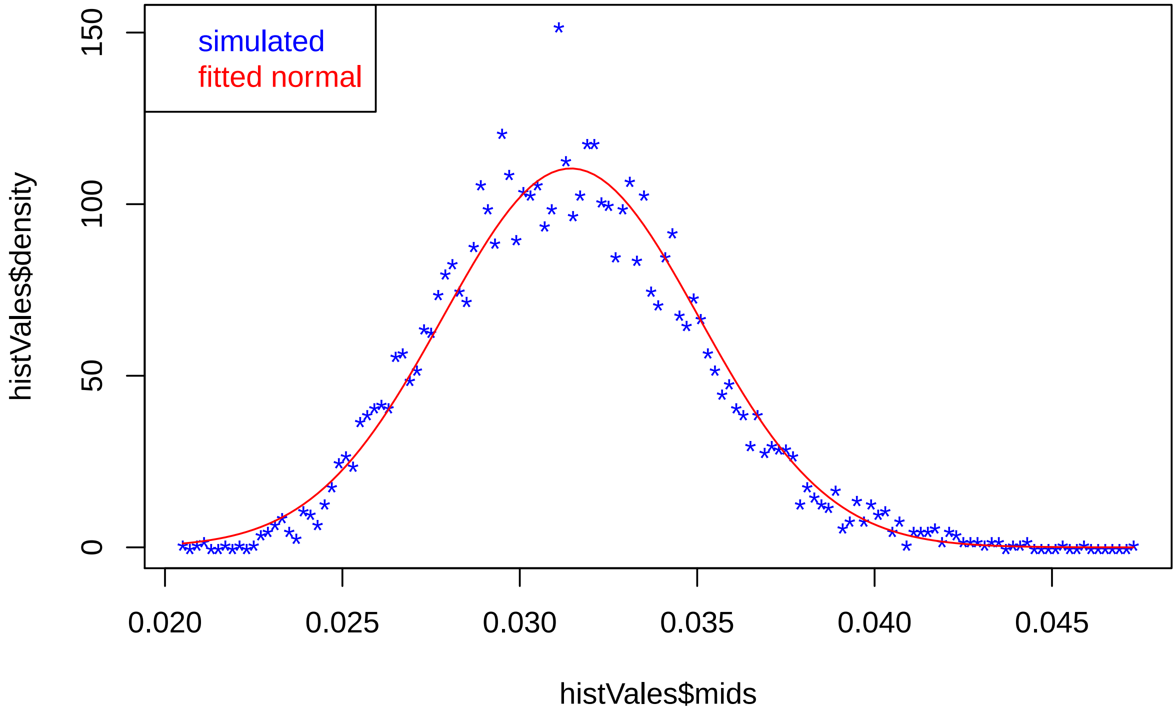


Table 11: Mean and sd of Lactic pvalues from simulation

mean.bval	sd.bval
0.03144	0.003614

We see that the p-values are not dramatically affected by the addition of noise. Above we have plotted the empirical distribution of the p-values and a normal with the same mean and standard deviation.

(e) Repeat the previous question but with a standard deviation of 0.1. Does this much measurement error make an important difference?

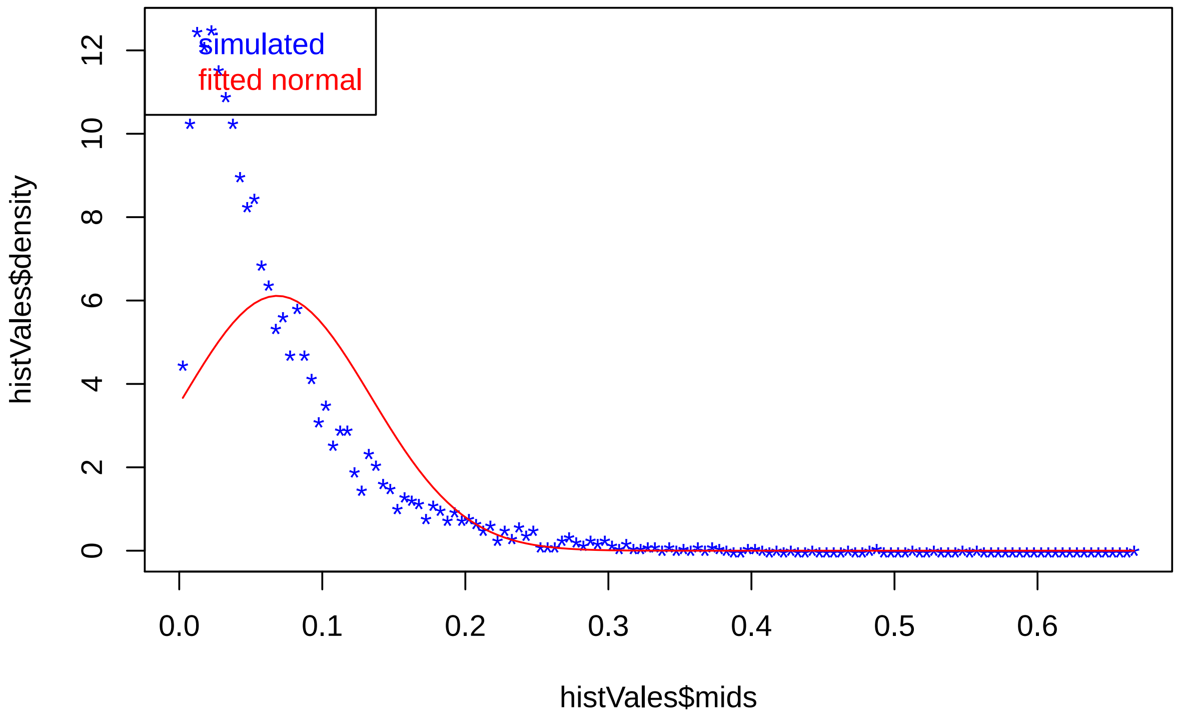


Table 12: Mean and sd of Lactic pvalues from simulation

mean.bval	sd.bval
0.06852	0.06527

We see that the p-value is significantly affected at this level of additional noise in the predictor.

7.8 fat data analysis

Use the fat data, fitting the model described in Section 4.2.

```
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) -15.292549  16.069921 -0.9516  0.342252
## age           0.056786   0.029965  1.8951  0.059290
## weight       -0.080310   0.049581 -1.6198  0.106602
## height      -0.064600   0.088930 -0.7264  0.468299
## neck        -0.437541   0.215334 -2.0319  0.043273
## chest       -0.023603   0.091839 -0.2570  0.797396
## abdom        0.885429   0.080077 11.0572 < 2.2e-16
## hip         -0.198419   0.135156 -1.4681  0.143406
## thigh        0.231895   0.133718  1.7342  0.084175
## knee        -0.011677   0.224143 -0.0521  0.958496
## ankle        0.163536   0.205143  0.7972  0.426142
## biceps       0.152799   0.158513  0.9640  0.336048
## forearm      0.430489   0.184452  2.3339  0.020436
## wrist       -1.476537   0.495519 -2.9798  0.003183
##
## n = 252, p = 14, Residual SE = 3.98797, R-Squared = 0.75
```

(a) Compute the condition numbers and variance inflation factors. Comment on the degree of collinearity observed in the data.

Table 13: eigenvalues (continued below)

ev.1	ev.2	ev.3	ev.4	ev.5	ev.6	ev.7	ev.8	ev.9	ev.10
19592555	64185	30597	5704	2804	1935	1030	637.7	528.1	431.8

ev.11	ev.12	ev.13
376.4	272.4	63.45

Table 15: condition numbers (continued below)

rcond.1	rcond.2	rcond.3	rcond.4	rcond.5	rcond.6	rcond.7	rcond.8
1	17.47	25.3	58.61	83.59	100.6	137.9	175.3

rcond.9	rcond.10	rcond.11	rcond.12	rcond.13
192.6	213	228.2	268.2	555.7

We note a high condition number for the model matrix, and a number of the individual predictors have a large value of $\frac{\lambda_1}{\lambda_i}$

Table 17: Variance Inflation Factors (continued below)

VIF.age	VIF.weight	VIF.height	VIF.neck	VIF.chest	VIF.abdom	VIF.hip
2.25	33.51	1.675	4.324	9.461	11.77	14.8

VIF.thigh	VIF.knee	VIF.ankle	VIF.biceps	VIF.forearm	VIF.wrist
7.778	4.612	1.908	3.62	2.192	3.378

We see weight and abdom - and marginally chest - have VIF values indicating colinearity with other predictors.

(b) Cases 39 and 42 are unusual. Refit the model without these two cases and recompute the collinearity diagnostics. Comment on the differences observed from the full data fit.

```
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept)  2.622509  21.654912  0.1211  0.903711
## age          0.065827   0.029806  2.2085  0.028168
## weight      -0.015960   0.062198 -0.2566  0.797713
## height      -0.223547   0.177083 -1.2624  0.208057
## neck        -0.359262   0.217585 -1.6511  0.100041
## chest       -0.110593   0.100290 -1.1027  0.271267
## abdom        0.839876   0.084675  9.9188 < 2.2e-16
## hip         -0.153126   0.135132 -1.1332  0.258298
## thigh        0.174473   0.136035  1.2826  0.200902
## knee        -0.069409   0.227540 -0.3050  0.760603
## ankle        0.174260   0.203676  0.8556  0.393100
## biceps       0.151605   0.157859  0.9604  0.337846
## forearm      0.268272   0.191700  1.3994  0.162995
## wrist       -1.642878   0.493838 -3.3268  0.001019
##
## n = 250, p = 14, Residual SE = 3.94247, R-Squared = 0.75
```

We see fewer significant predictors in the fit without the outliers. This makes intuitive sense - if the outlier(s) is(are) related to abnormal values of one of the predictors then it possible that predictor will have undue influence on the fit through the outliers. Removing the outliers eliminates the influence and the significance. Neck and thigh are predictors that we'd consider for this effect, and indeed inspecting the data confirms that is the case for one (39) of the redacted data points. The other outlier (42) has a very small value for the height predictor. Although it is not significant in the model fit with the redacted data, the p-value is half that for the mode with the outliers.

(c) Fit a model with brozek as the response and just age, weight and height as predictors. Compute the collinearity diagnostics and compare to the full data fit.

Table 19: eigenvalues

ev.1	ev.2	ev.3
9824566	51002	15672

Table 20: condition numbers

rcond.1	rcond.2	rcond.3
1	13.88	25.04

Table 21: Variance Inflation Factors

VIF.age	VIF.weight	VIF.height
1.083	1.381	1.47

We see the colinearity diagnostics all indicate that there is no linear association among the predictors.

(d) Compute a 95% prediction interval for brozek for the median values of age, weight and height.

Table 22: Median Value of Predictors

median.age	median.weight	median.height
43	176.1	70

Table 23: 95% Prediction Interval For Univariate Median of Predictors

fit	lwr	upr
18.49	8.648	28.33

Table 24: 95% Prediction Interval Width For Univariate Median of Predictors

pi.width
19.68

(e) Compute a 95% prediction interval for brozek for age=40, weight=200 and height=73. How does the interval compare to the previous prediction?

Table 25: 95% Prediction Interval For (age=40, weight=200 and height=73)

fit	lwr	upr
20.18	10.32	30.05

Table 26: 95% Prediction Interval Width For(age=40, weight=200 and height=73)

pi.width
19.73

This interval does not differ in width from the interval calculated from the median predictor values.

(f) Compute a 95% prediction interval for brozek for age=40, weight=130 and height=73. Are the values of predictors unusual? Comment on how the interval compares to the previous two answers.

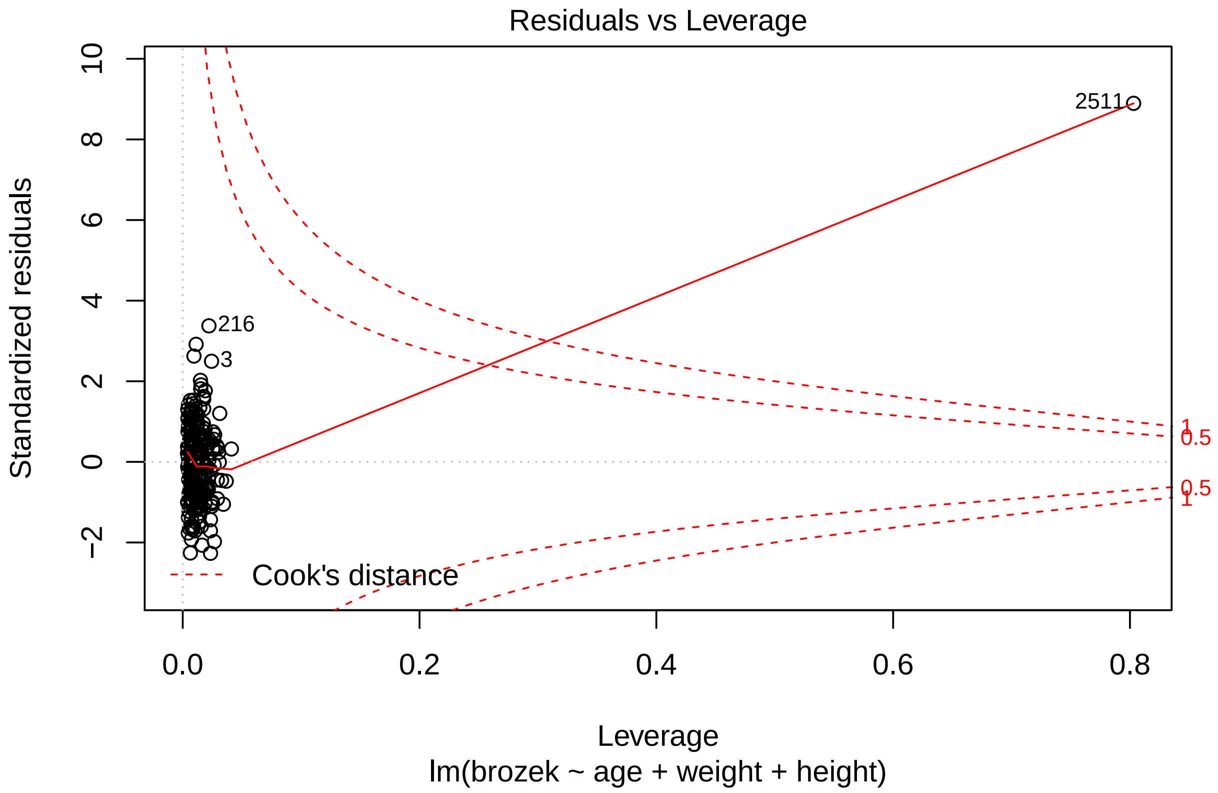
Table 27: 95% Prediction Interval For (age=40, weight=200 and height=73)

fit	lwr	upr
3.72	-6.282	13.72

Table 28: 95% Prediction Interval Width For(age=40, weight=200 and height=73)

pi.width
20

The prediction interval with is larger for this example, and the predicted body fat is a very low value. Due to the weight, this data points is likely a high leverage points. We can add it to the training set and see.



Indeed - the added point (251) is a high leverage point in the model with

NCSU ST 503 HW 7

Probems 8.1, 8.6, 8.8 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

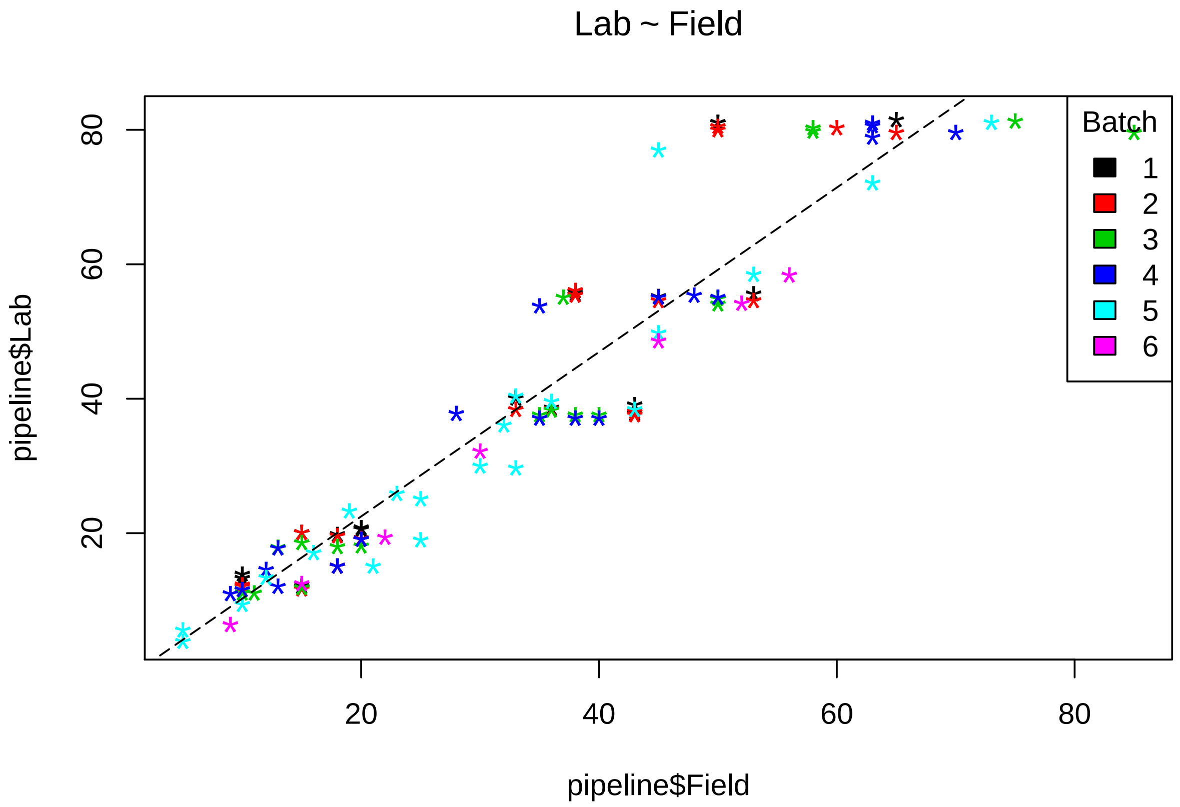
17 October, 2017

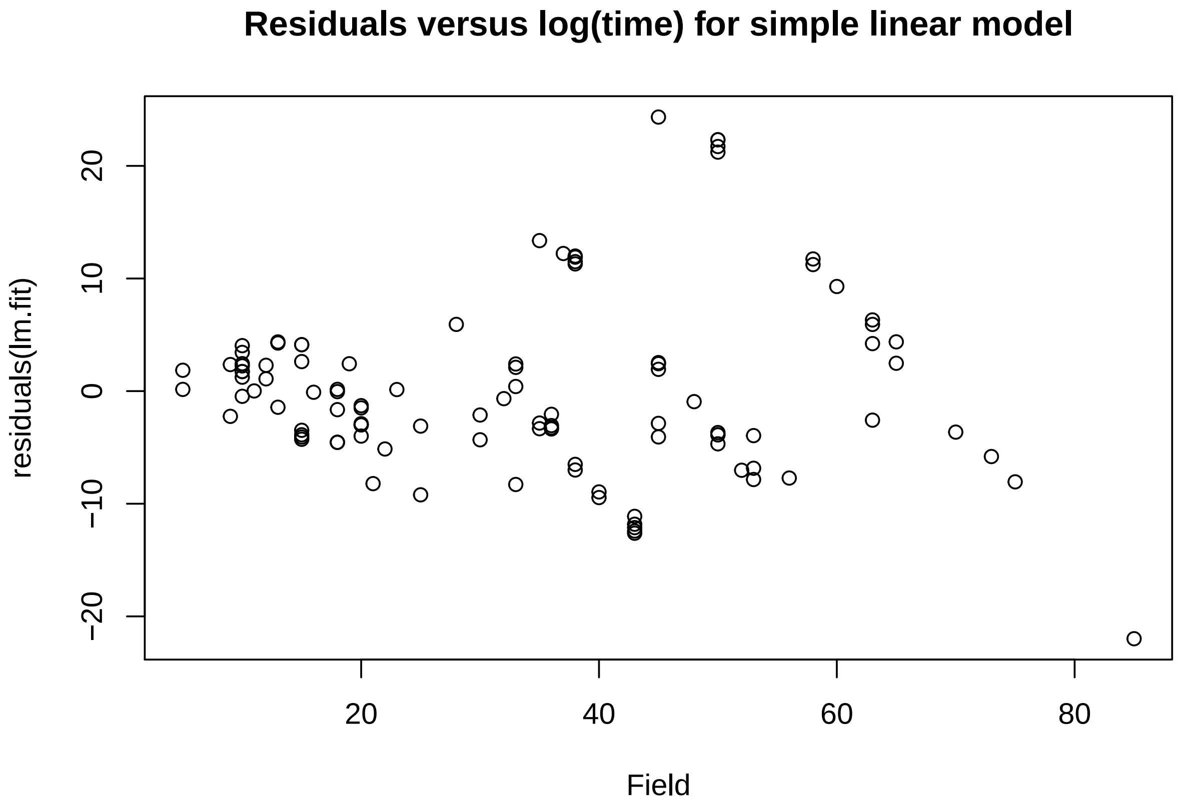
8.1 NIST pipeline Data

Researchers at National Institutes of Standards and Technology (NIST) collected pipeline data on ultrasonic measurements of the depth of defects in the Alaska pipeline in the field. The depth of the defects were then remeasured in the laboratory. These measurements were performed in six different batches. It turns out that this batch effect is not significant and so can be ignored in the analysis that follows. The laboratory measurements are more accurate than the in-field measurements, but more time consuming and expensive. We want to develop a regression equation for correcting the in-field measurements.

(a) Fit a regression model $Lab \sim Field$. Check for non-constant variance.

```
##
## Call:
## lm(formula = Lab ~ Field, data = pipeline)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.985  -4.072  -1.431   2.504  24.334
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.96750    1.57479  -1.249    0.214
## Field        1.22297    0.04107  29.778   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.865 on 105 degrees of freedom
## Multiple R-squared:  0.8941, Adjusted R-squared:  0.8931
## F-statistic: 886.7 on 1 and 105 DF,  p-value: < 2.2e-16
```


Lab ~ Field
pipeline$Lab
pipeline$Field
80
60
40
20
20
40
60
80
Batch
1
2
3
4
5
6

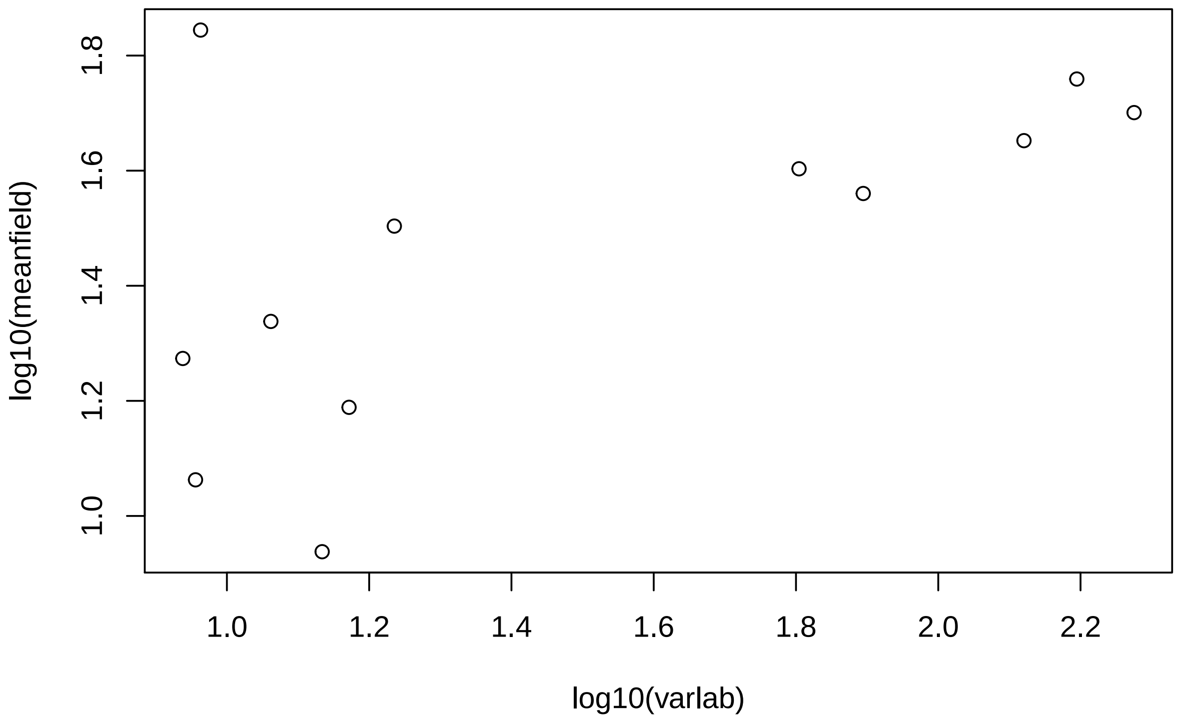


Based on the residual plot we see that we have evidence of non-constant variance. Variance increases with increasing predictor value.

(b) We wish to use weights to account for the non-constant variance.

Here we split the range of Field into 12 groups of size nine (except for the last group which has only eight values). Within each group, we compute the variance of Lab as varlab and the mean of Field as meanfield. Supposing pipeline is the name of your data frame, the following R code will make the needed computations:

Suppose we guess that the the variance in the response is linked to the predictor in the following way: $var(Lab) = a0\ Field^{a1}$ Regress $log(varlab)$ on $log(meanfield)$ to estimate $a0$ and $a1$. (You might choose to remove the last point.) Use this to determine appropriate weights in a WLS fit of Lab on Field. Show the regression summary.

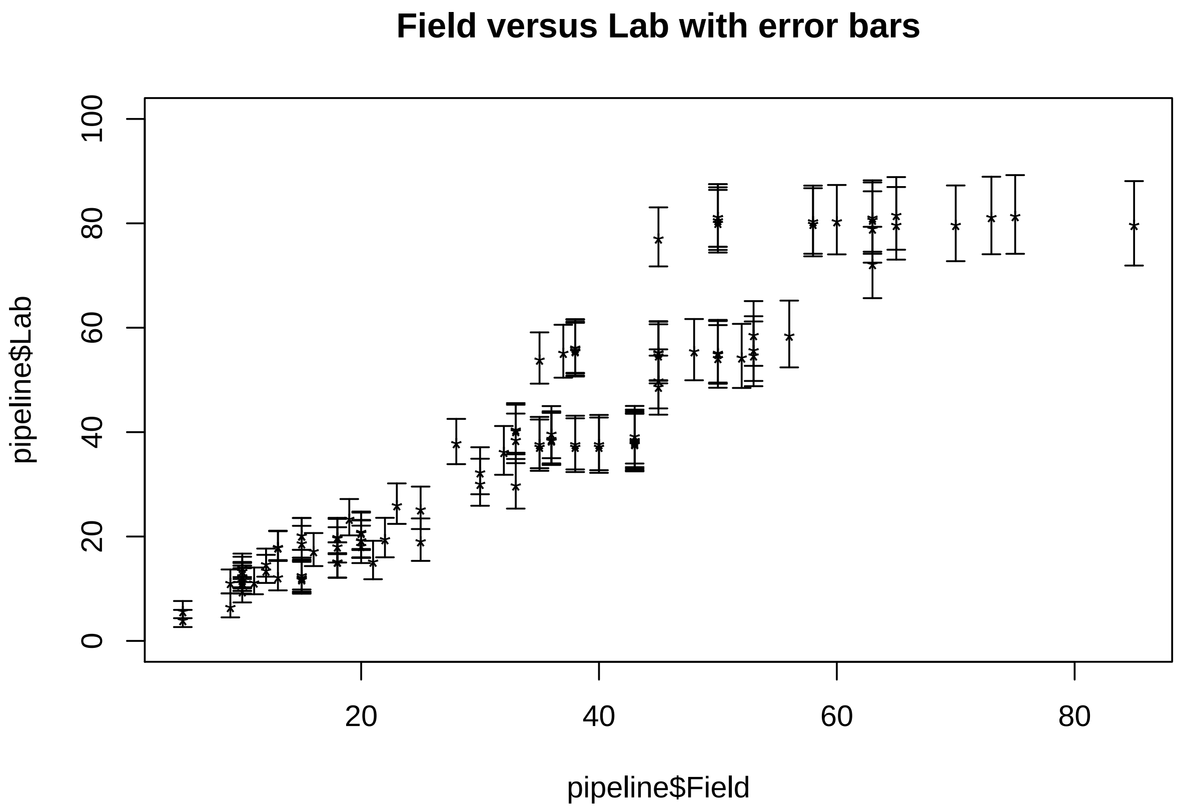


```
##
## Call:
## lm(formula = log(varlab) ~ log(meanfield), data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2038 -0.6729  0.1656  0.7205  1.1891
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.3538     1.5715  -0.225   0.8264
## log(meanfield)   1.1244     0.4617   2.435   0.0351 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.018 on 10 degrees of freedom
## Multiple R-squared:  0.3723, Adjusted R-squared:  0.3095
## F-statistic: 5.931 on 1 and 10 DF,  p-value: 0.03513
```

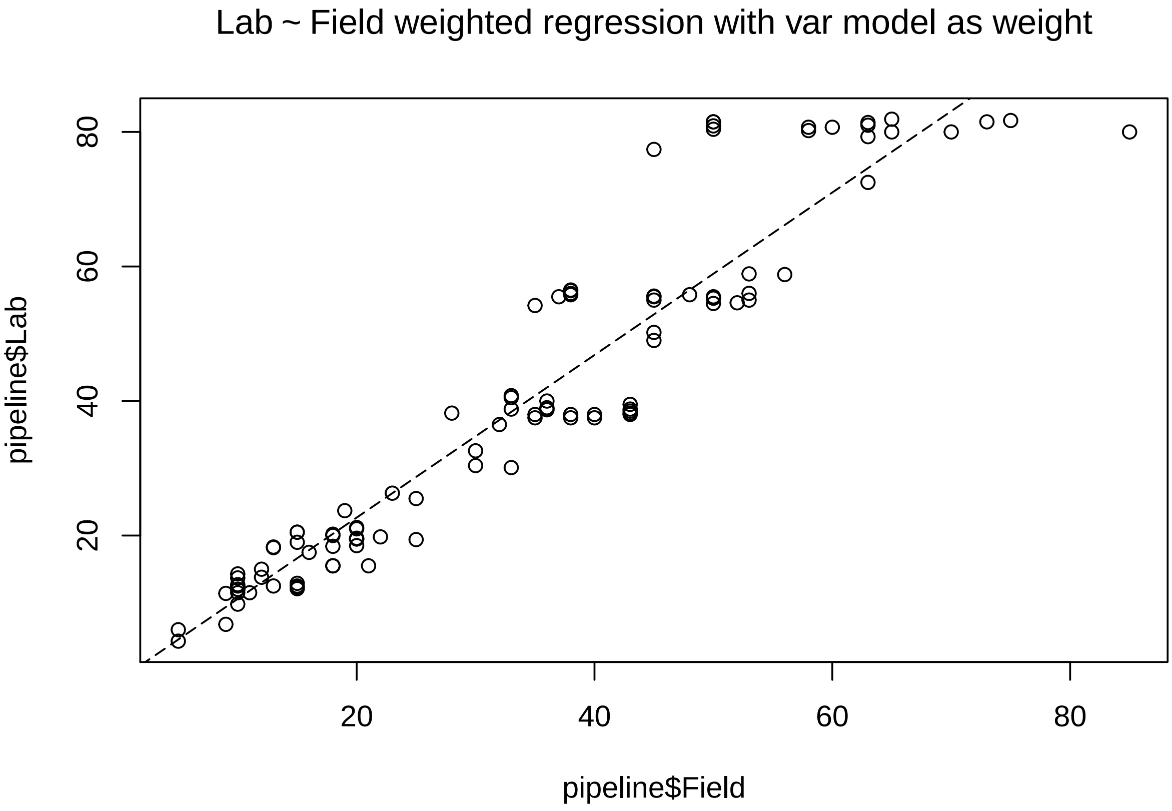
Since $var(Lab) = a0\ Field^{a1}$ we have that $log(var(Lab)) = log(a0) + a1\ log(Field)$ and from the model we fit our estimates of $a0$ and $a1$ are $10^{-0.3538} = 0.4427922$ and 1.1244.

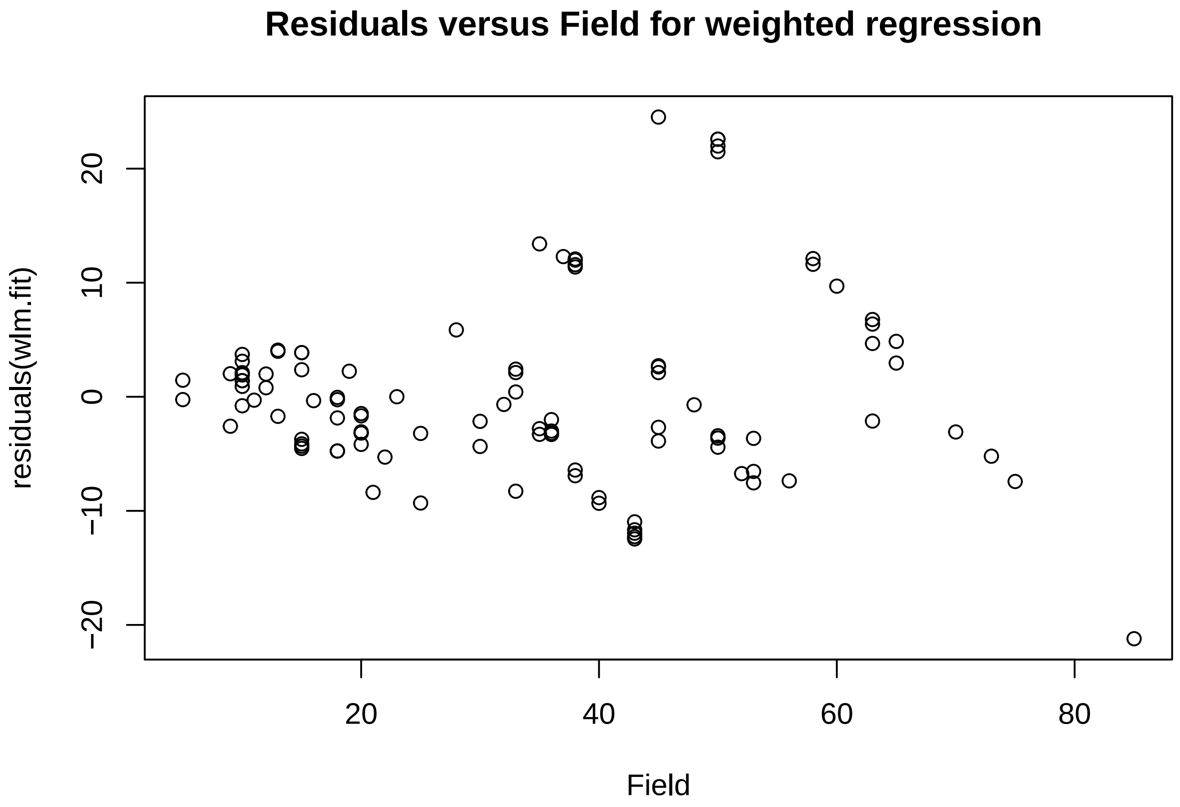
Now we calculate our weight vector with the variances obtained from our model.

Let plot the data with some error bars from the estimated variances just to make sure everything looks reasonable.



```
## Generalized least squares fit by REML
##   Model: Lab ~ Field
##   Data: pipeline
##        AIC      BIC    logLik
##   708.1764 716.1383 -351.0882
##
## Variance function:
##  Structure: fixed weights
##  Formula: ~var.lab
##
## Coefficients:
##                 Value Std.Error  t-value p-value
## (Intercept) -1.494365 0.9070661 -1.64747  0.1025
## Field        1.208276 0.0348839 34.63704  0.0000
##
##  Correlation:
##       (Intr)
## Field -0.809
##
## Standardized residuals:
##        Min         Q1        Med         Q3        Max
## -1.7814680 -0.6930709 -0.2727923  0.5313567  2.9450736
##
## Residual standard error: 1.47198
## Degrees of freedom: 107 total; 105 residual
```

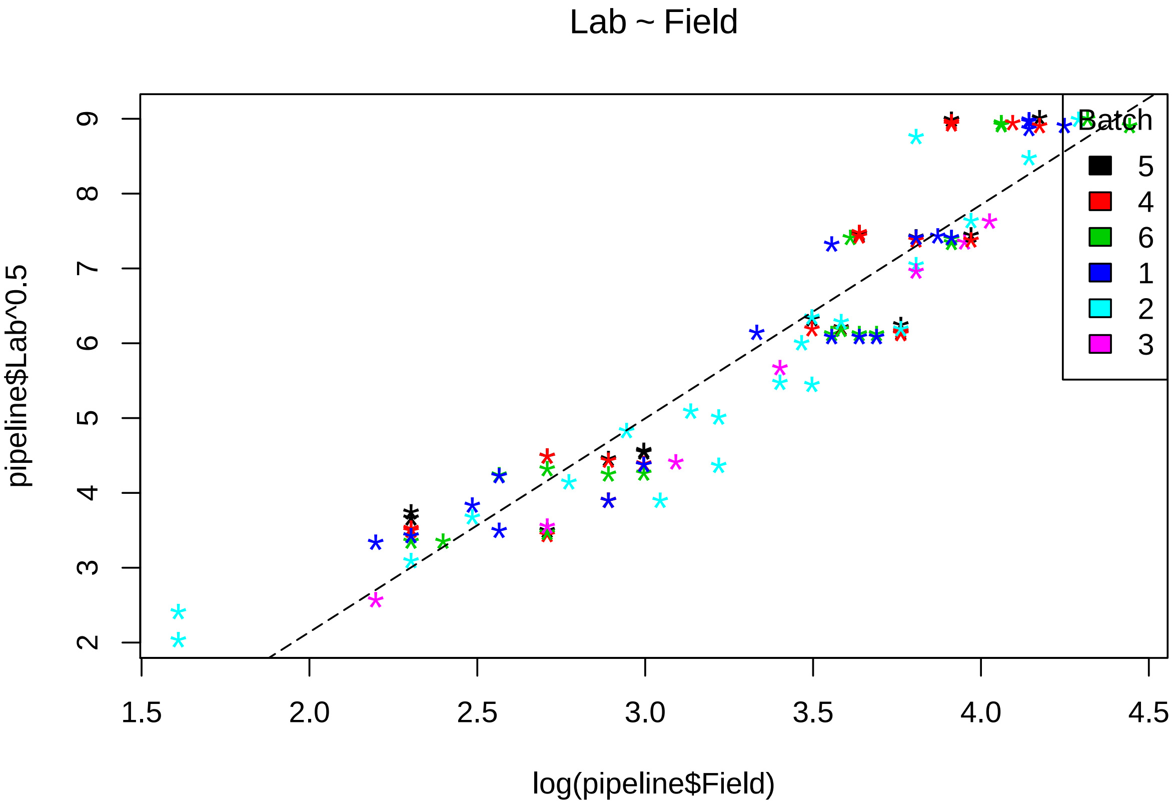


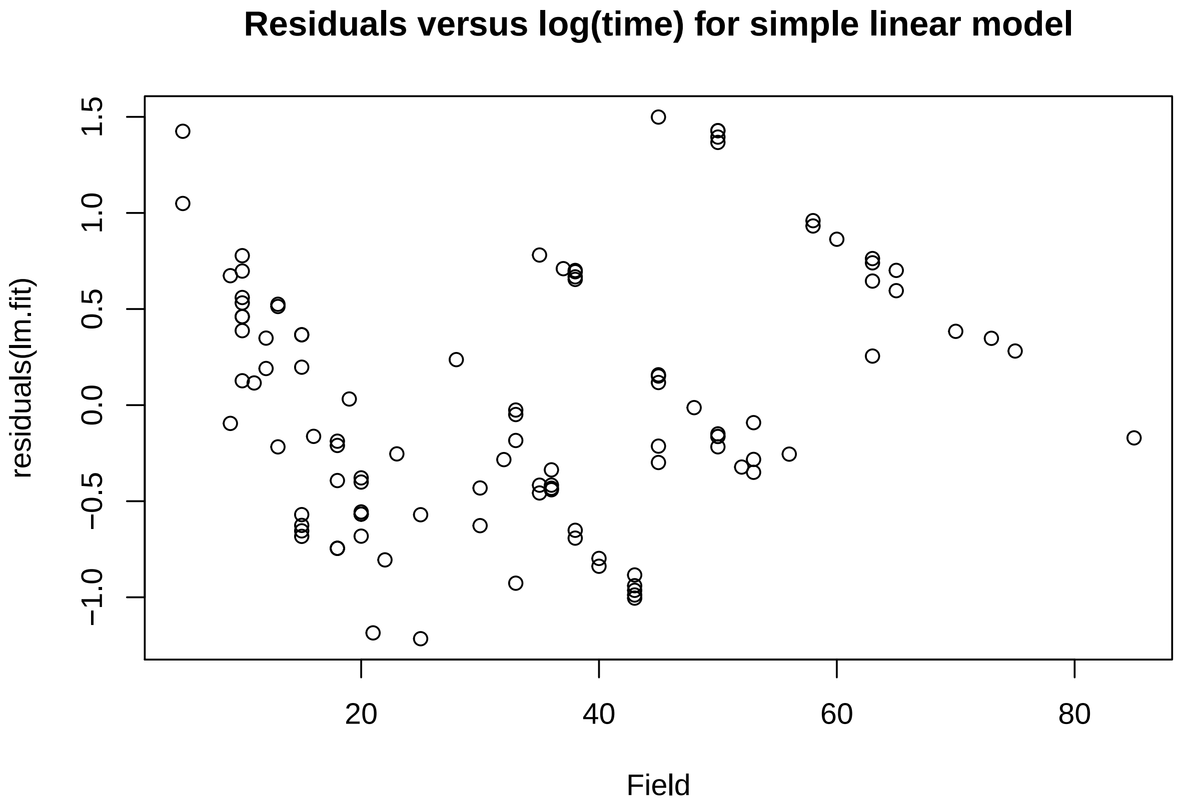


(c) An alternative to weighting is transformation. Find transformations on Lab and/or Field so that in the transformed scale the relationship is approximately linear with constant variance. You may restrict your choice of transformation to square root, log and inverse.

```
##
## Call:
## lm(formula = (Lab)^0.5 ~ log(Field), data = pipeline)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2157 -0.5066 -0.1625  0.5191  1.4991
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.57141    0.33200  -10.76   <2e-16 ***
## log(Field)   2.85554    0.09786   29.18   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.6553 on 105 degrees of freedom
## Multiple R-squared:  0.8902, Adjusted R-squared:  0.8892
## F-statistic: 851.4 on 1 and 105 DF,  p-value: < 2.2e-16
```





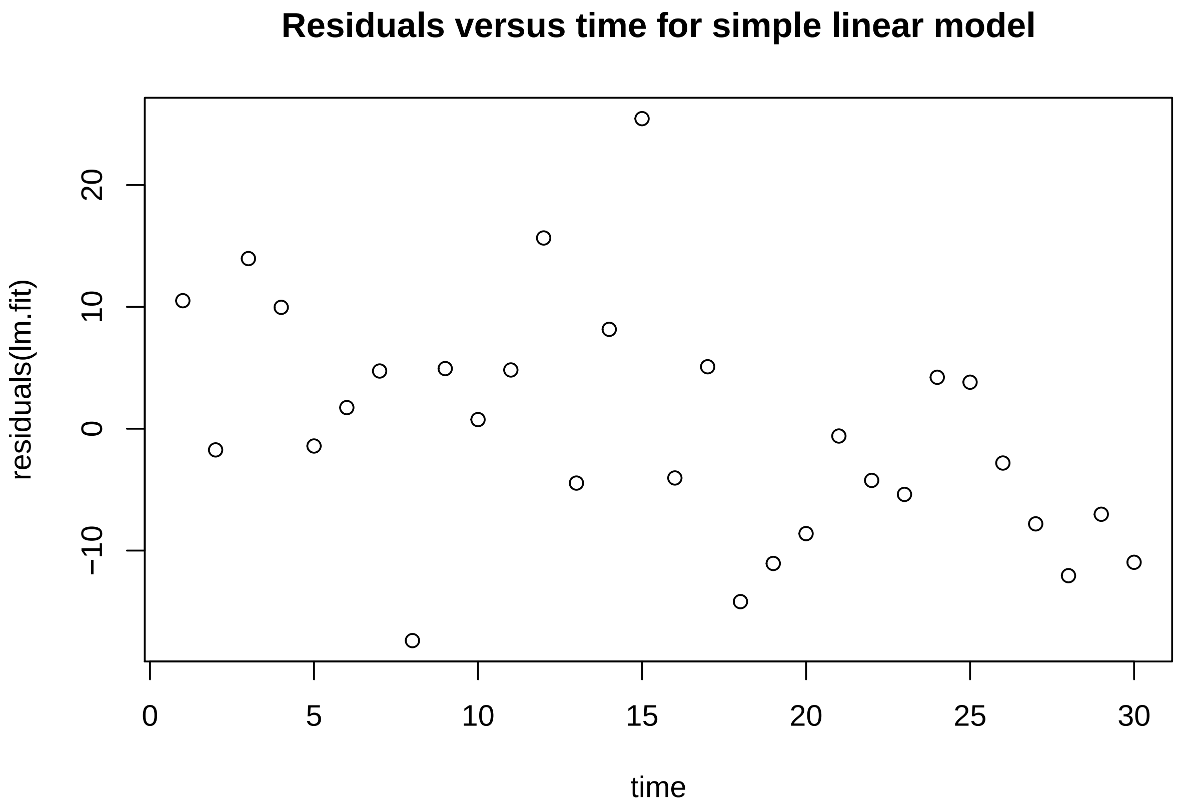
The RSE of this model is lower than the weighted model. I'm not sure I'd use that alone as a criteria for selecting a model. If we had a physical reason for the variance model we used - then we might opt to stick with the weighted regression.

8.6 Analysis of cheddar data

Using the cheddar data, fit a linear model with taste as the response and the other three variables as predictors.

```
##
## Call:
## lm(formula = taste ~ ., data = cheddar)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.390  -6.612  -1.009   4.908  25.449
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -28.8768    19.7354  -1.463  0.15540
## Acetic        0.3277     4.4598   0.073  0.94198
## H2S           3.9118     1.2484   3.133  0.00425 **
## Lactic       19.6705     8.6291   2.280  0.03108 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.13 on 26 degrees of freedom
## Multiple R-squared:  0.6518, Adjusted R-squared:  0.6116
## F-statistic: 16.22 on 3 and 26 DF,  p-value: 3.81e-06
```

(a) Suppose that the observations were taken in time order. Create a time variable. Plot the residuals of the model against time and comment on what can be seen.



Fitting a linear model we can get an estimate of the correlation.

```
##
## Call:
## lm(formula = X1 ~ X2, data = df)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -21.0152  -5.2405  -0.7975   4.1784  25.2072
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.4910     3.2764   2.286   0.0300 *
## X2           -0.4833     0.1846  -2.619   0.0141 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.749 on 28 degrees of freedom
## Multiple R-squared:  0.1967, Adjusted R-squared:  0.168
## F-statistic: 6.857 on 1 and 28 DF,  p-value: 0.01409
```

Here we calculate the lag(1) correlation on the residuals.

Table 1: lag(1) correlation on residuals

cor.residuals
0.1787

(b) Fit a GLS model with same form as above but now allow for an AR(1) correlation among the errors. Is there evidence of such a correlation?

```
## Generalized least squares fit by REML
##   Model: taste ~ Acetic + H2S + Lactic
##   Data: na.omit(cheddar)
##      AIC      BIC  logLik
##   214.94 222.4886 -101.47
##
## Correlation Structure: AR(1)
##  Formula: ~time
##  Parameter estimate(s):
##       Phi
## 0.2641944
##
## Coefficients:
##                  Value Std.Error   t-value p-value
## (Intercept) -30.332472 20.273077 -1.496195  0.1466
## Acetic        1.436411  4.876581  0.294553  0.7707
## H2S           4.058880  1.314283  3.088284  0.0047
## Lactic       15.826468  9.235404  1.713674  0.0985
##
##  Correlation:
```

```
##        (Intr) Acetic H2S
## Acetic -0.899
## H2S     0.424 -0.395
## Lactic  0.063 -0.416 -0.435
##
## Standardized residuals:
##         Min          Q1         Med          Q3         Max
## -1.64546468 -0.63861716 -0.06641714  0.52255676  2.41323021
##
## Residual standard error: 10.33276
## Degrees of freedom: 30 total; 26 residual
```

(c) Fit a LS model but with time now as an additional predictor. Investigate the significance of time in the model.

```
##
## Call:
## lm(formula = taste ~ ., data = cheddar)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.3523  -4.9735  -0.5089   4.8531  23.1311
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -36.6127    17.9845  -2.036  0.05250 .
## Acetic        4.1275     4.2556   0.970  0.34139
## H2S           3.5387     1.1315   3.127  0.00444 **
## Lactic       17.9527     7.7875   2.305  0.02973 *
## time         -0.5459     0.2043  -2.672  0.01306 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.112 on 25 degrees of freedom
## Multiple R-squared:  0.7291, Adjusted R-squared:  0.6858
## F-statistic: 16.83 on 4 and 25 DF,  p-value: 8.205e-07
```

Time is significant in this model at a level of $\alpha = 0.05$. The coefficient tells us that when all other variables are held constant an increasing time value results in a decreasing value of taste. We see the value of the coefficient is close to the model we fit of residuals of the original model versus time ~ -0.5.

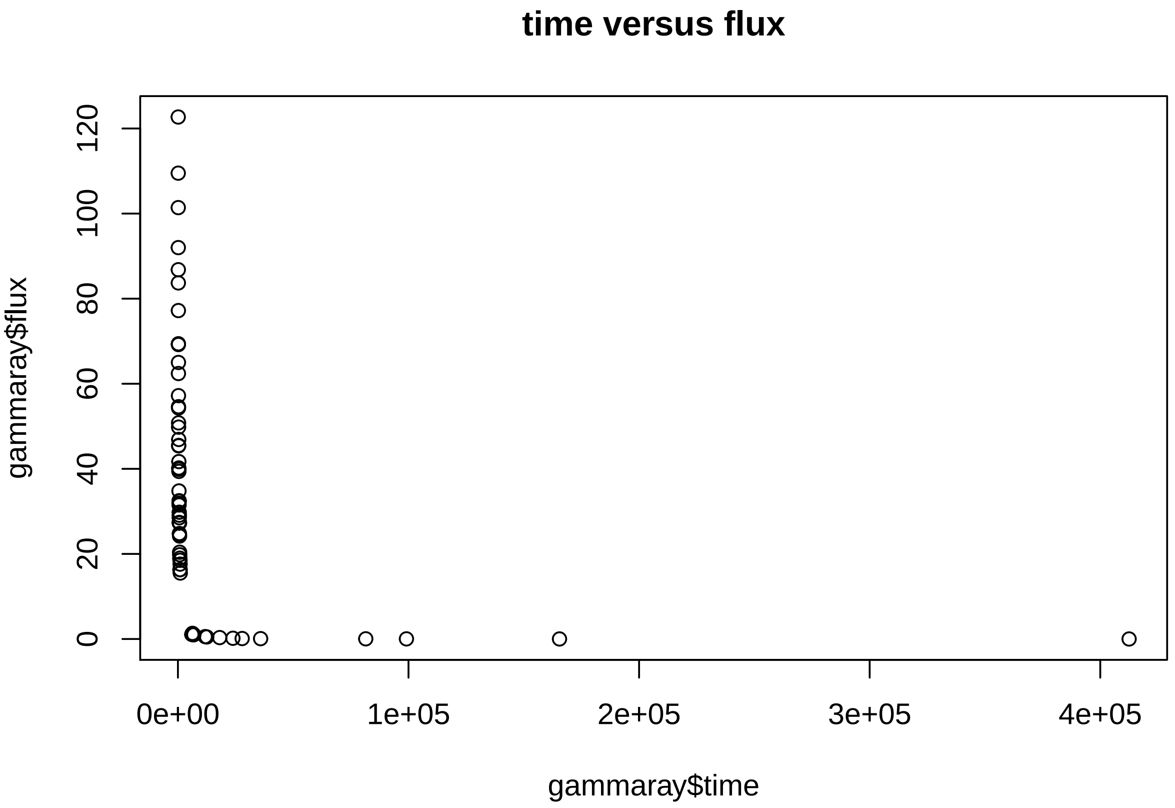
(d) The last two models have both allowed for an effect of time. Explain how they do this differently.

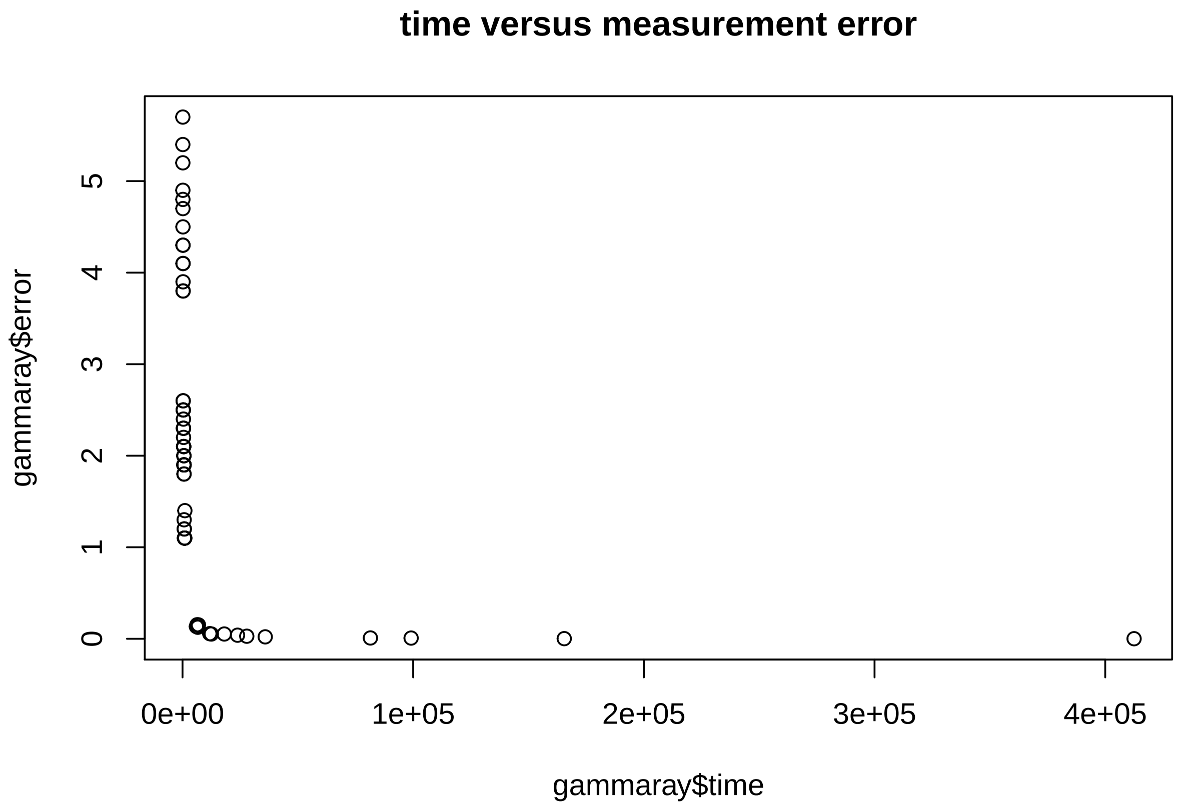
Obviously the LS model accounts for time by explicitly including it as a predictor. The GLS model we account for time through the error structure. We construct an estimate of the variance covariance matrix of the regression equation $\Sigma = S^t S$ from an AR(1) model of the residuals.

8.8 Gammaray analysis

The gammaray dataset shows the x-ray decay light curve of a gamma ray burst. Build a model to predict the flux as a function time that uses appropriate weights.

First we plot the data

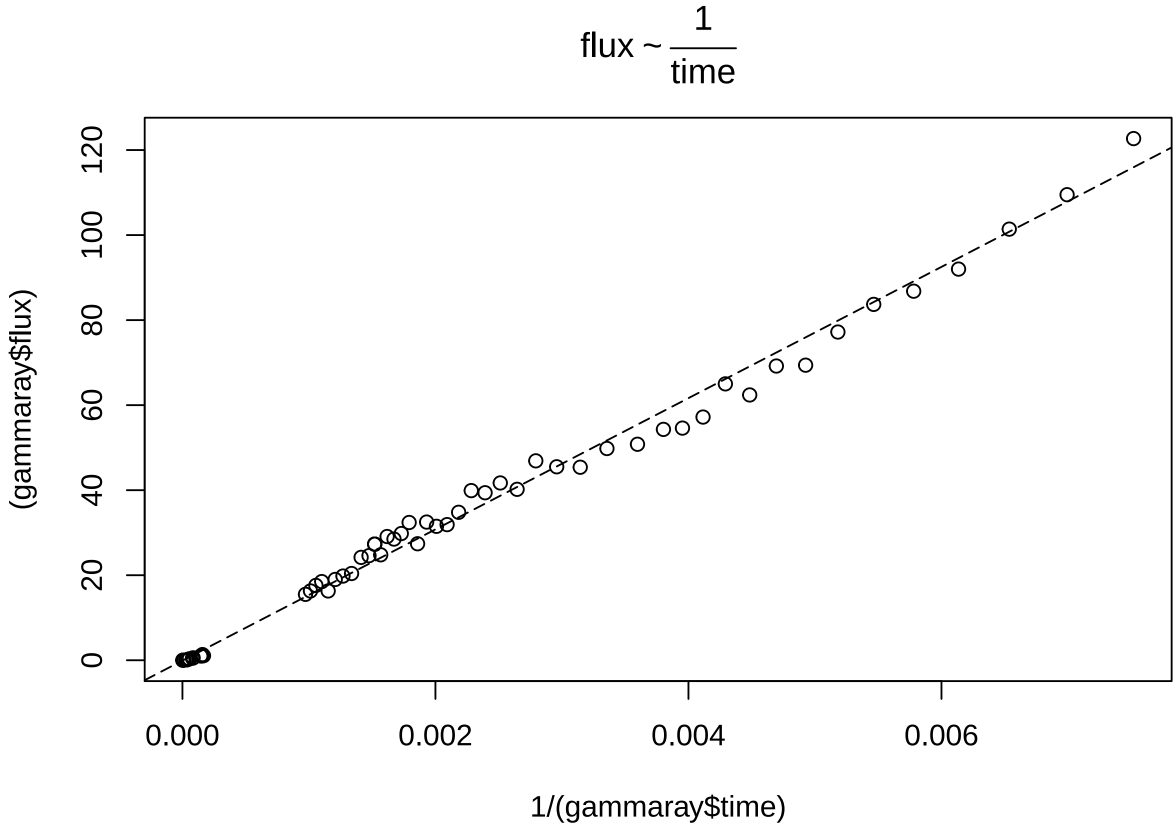


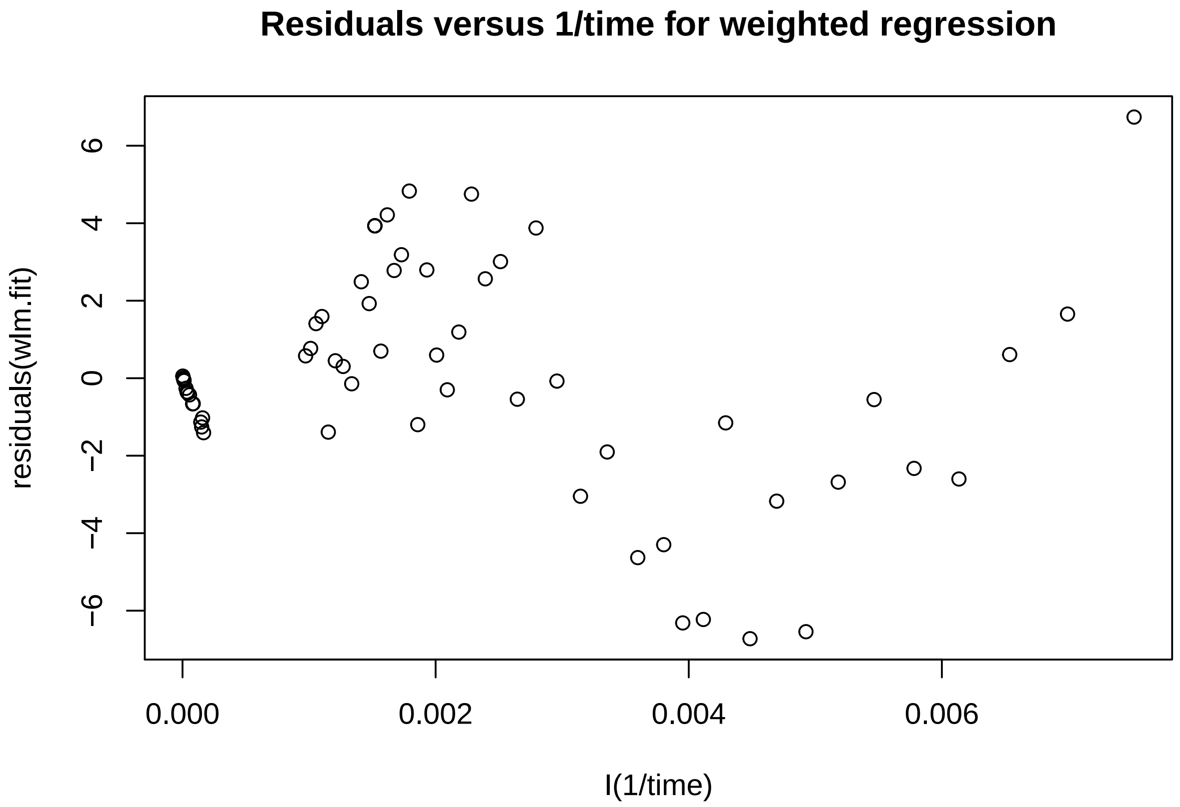


We see a hyperbolic relationship btween the predictor and the response so We fit a weighted regression $flux \sim \frac{1}{time}$ with weights equal to the error

```
## Generalized least squares fit by REML
##   Model: flux ~ I(1/time)
##   Data: gammaray
##        AIC     BIC    logLik
##   261.1843 267.517 -127.5922
##
## Variance function:
##  Structure: fixed weights
##  Formula: ~error
##
## Coefficients:
##                Value Std.Error  t-value p-value
## (Intercept)    -0.09   0.04068 -2.21963  0.0302
## I(1/time)   15434.73 171.88649 89.79603  0.0000
##
##  Correlation:
##           (Intr)
## I(1/time) -0.112
```

```
##
## Standardized residuals:
##         Min          Q1         Med          Q3         Max
## -1.81209736 -0.87077410 -0.02224618  0.91280791  1.57197856
##
## Residual standard error: 2.120048
## Degrees of freedom: 63 total; 61 residual
```

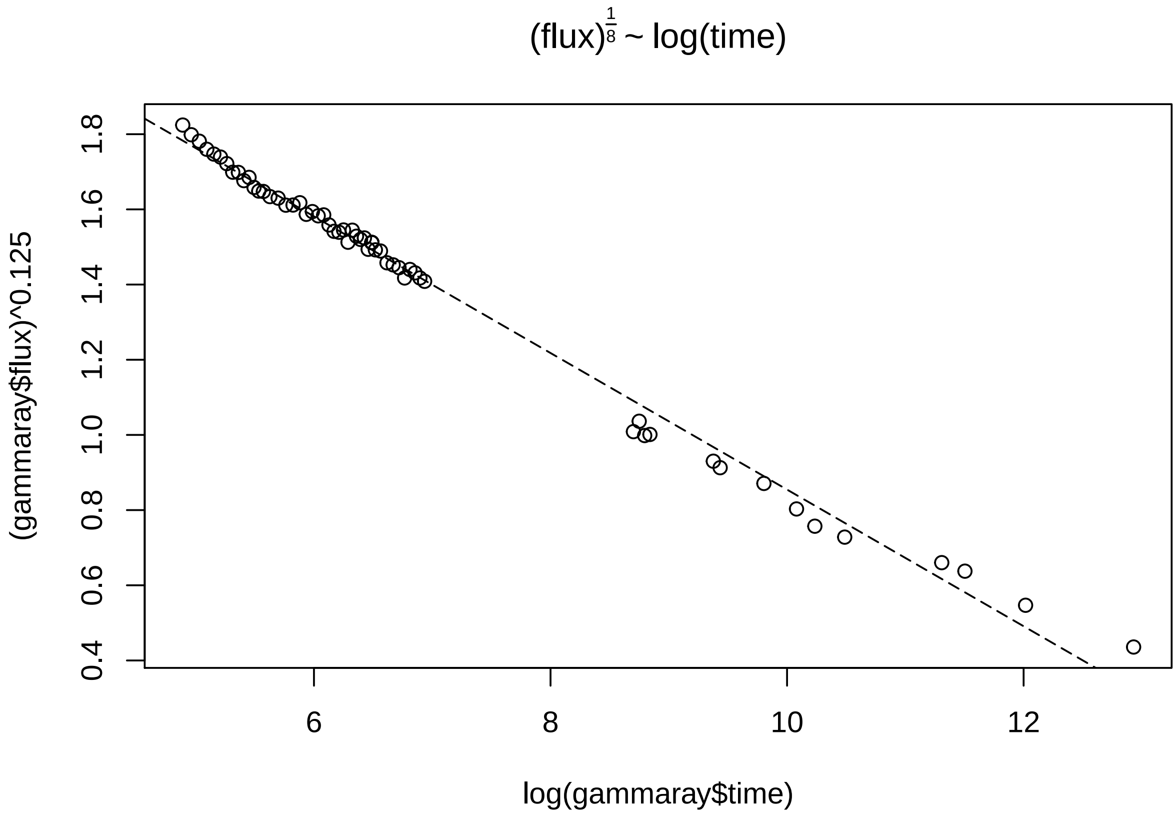


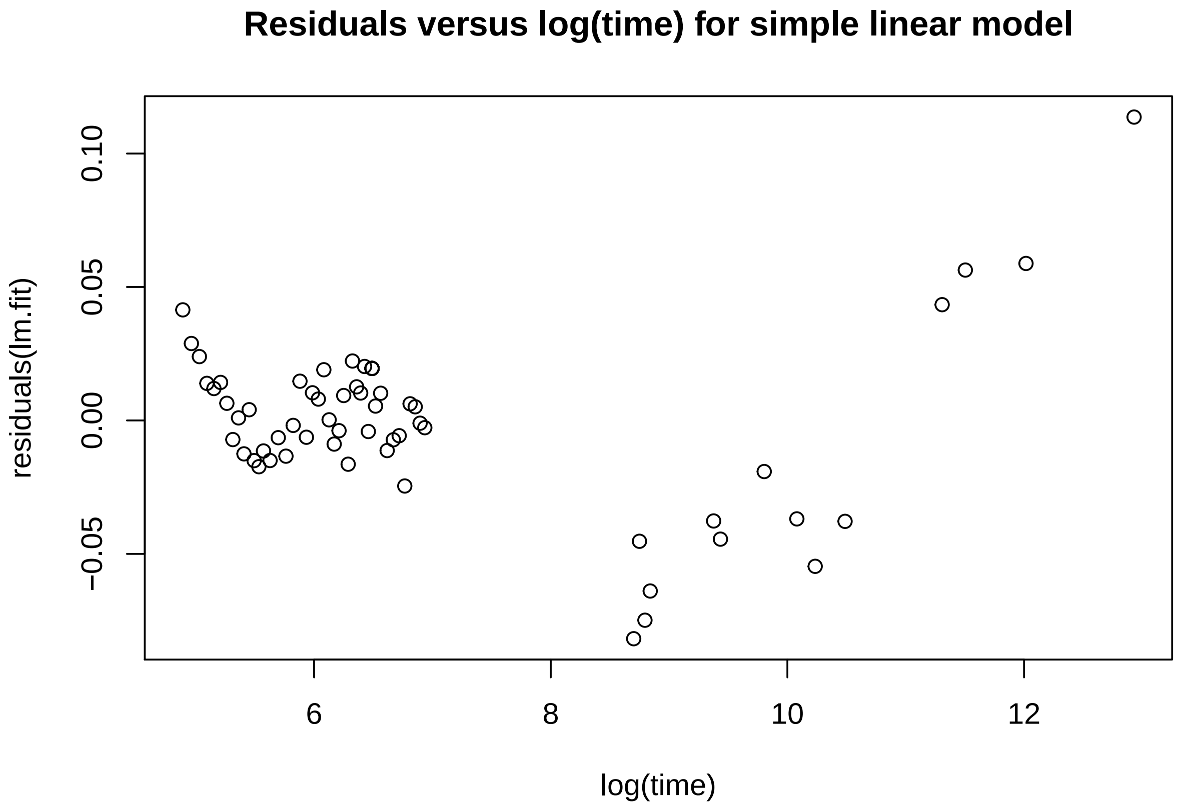


We try a trasformation - for reference, and for fun. We Chose this model after some experimenting

$$(flux)^{\frac{1}{8}} \sim log(time)$$

```
##
## Call:
## lm(formula = (flux)^0.125 ~ log(time), data = gammaray)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.081779 -0.012954  0.000958  0.016848  0.113664
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.671490   0.015088  177.06   <2e-16 ***
## log(time)   -0.181701   0.002094  -86.79   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0314 on 61 degrees of freedom
## Multiple R-squared:  0.992,  Adjusted R-squared:  0.9918
## F-statistic:  7532 on 1 and 61 DF,  p-value: < 2.2e-16
```




Residuals versus log(time) for simple linear model
residuals(lm.fit)
0.10
0.05
0.00
−0.05
6
8
10
12
log(time)

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

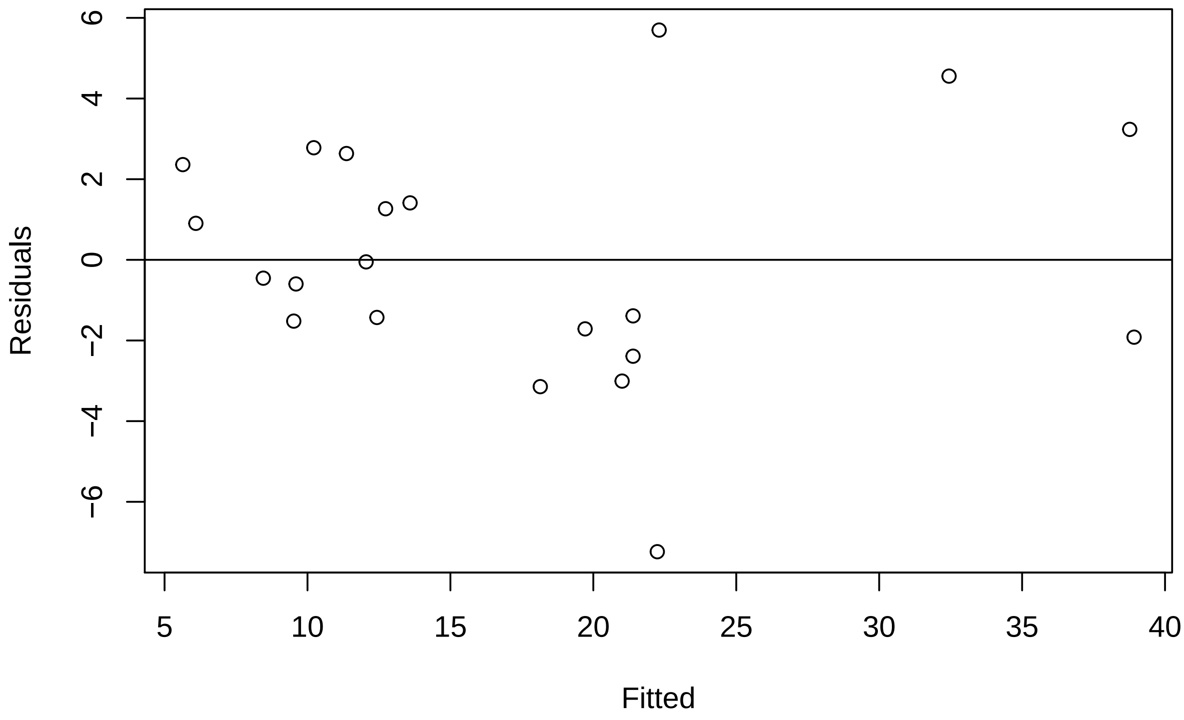
23 October, 2017

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 `quantreg::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

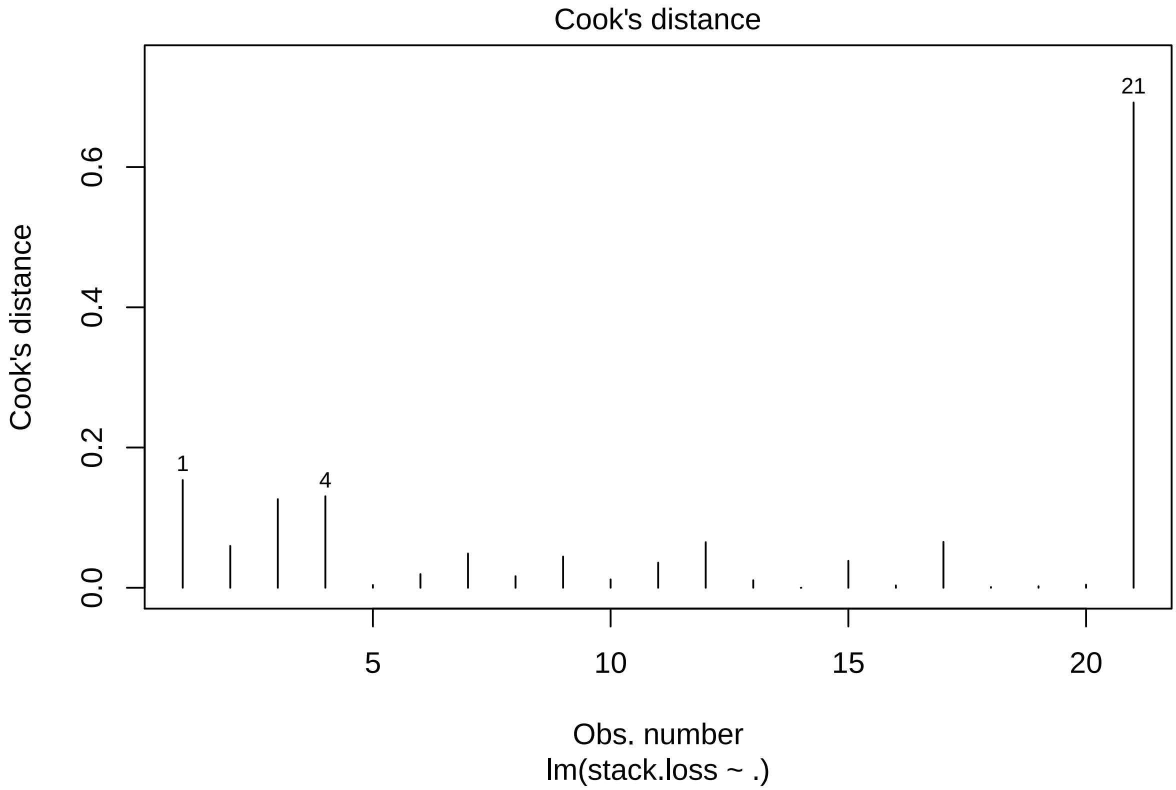
Table 3: Bonferroni corrected t-value

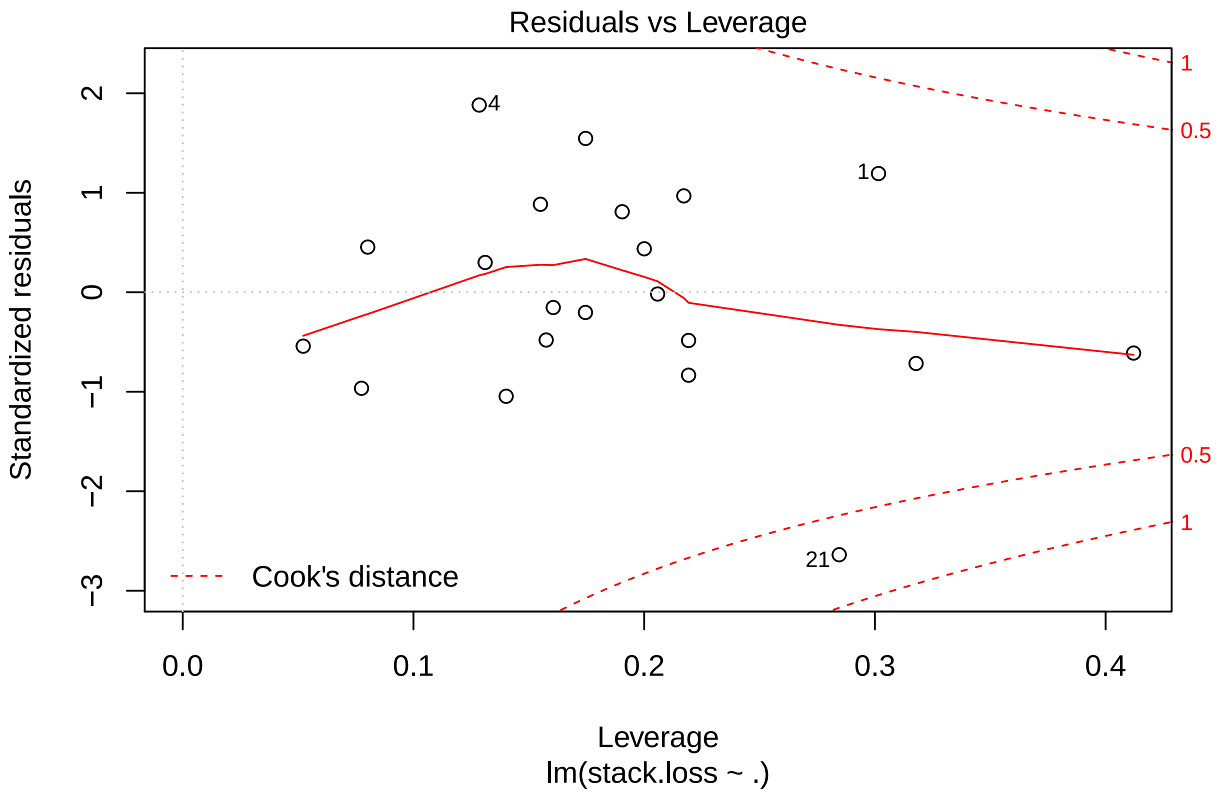
t.val.alpha
-3.604

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

Check for influential points.

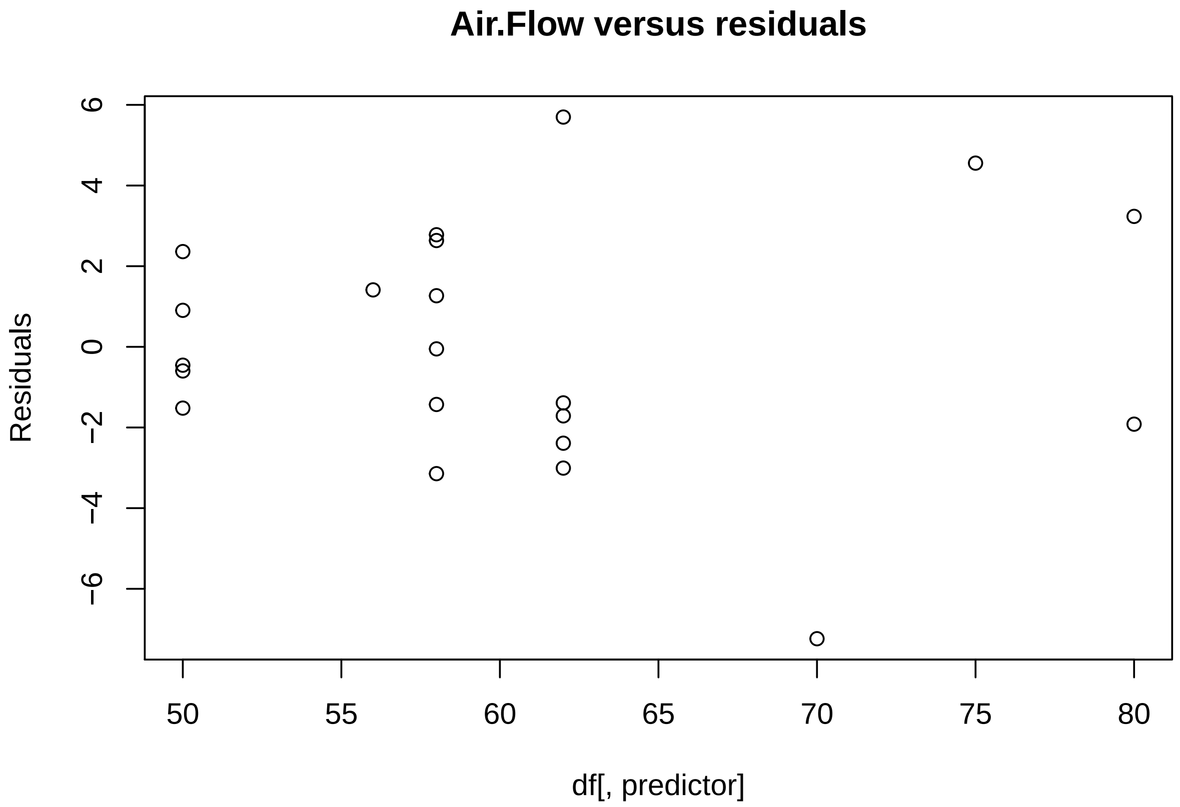
We plot the Cook's distances and the residual-leverage plot with level set contours of the Cook distance.

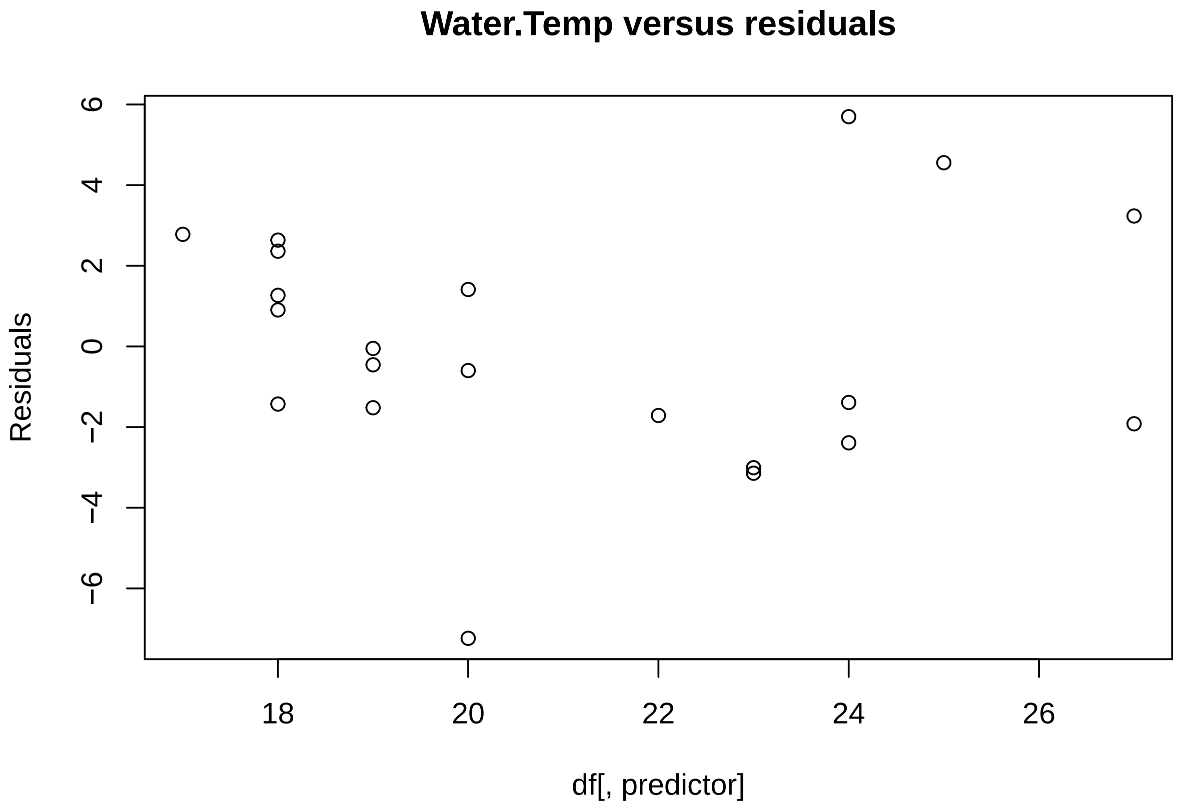


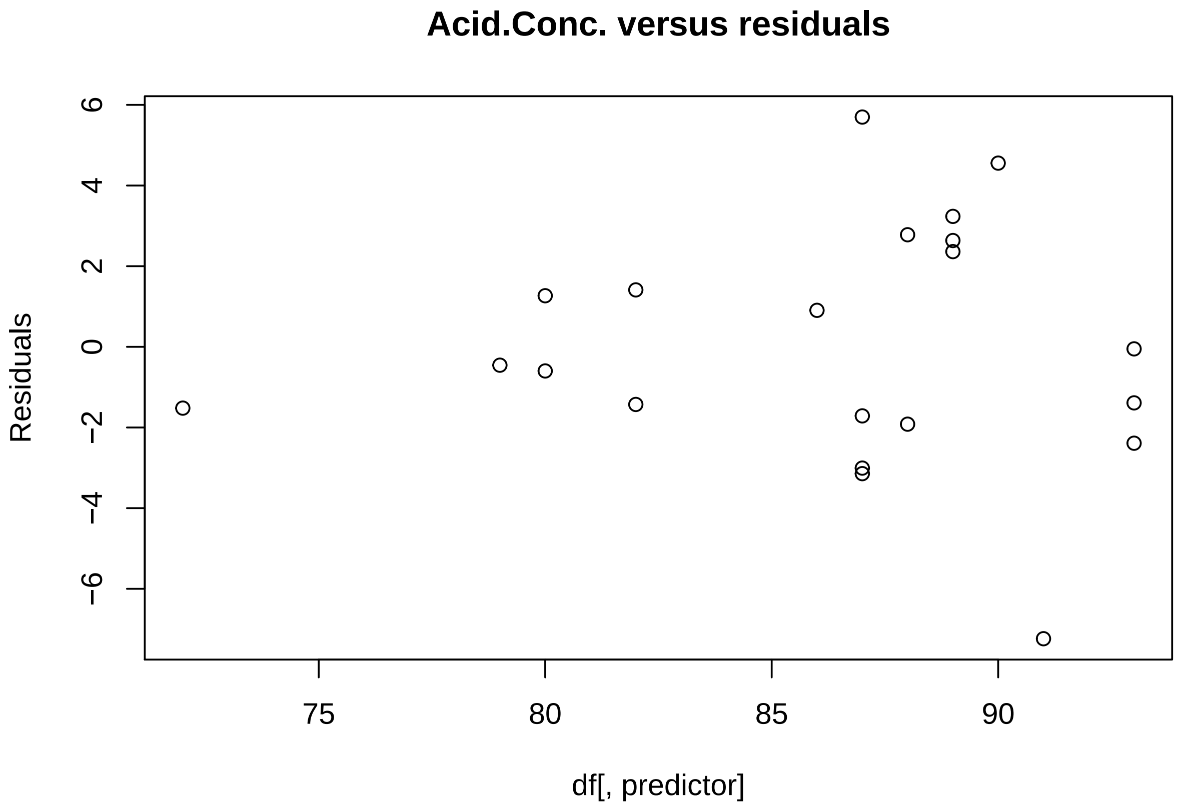


Check for structure in the model.

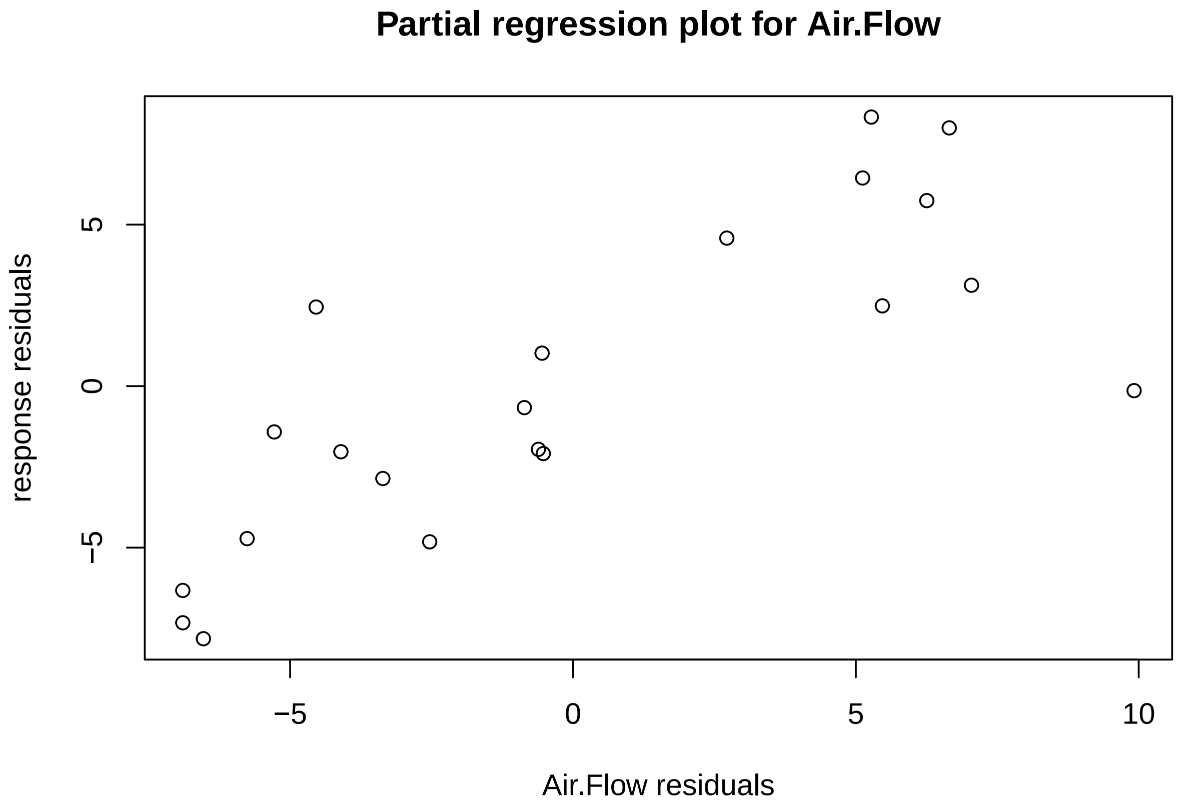
Plot residuals versus predictors

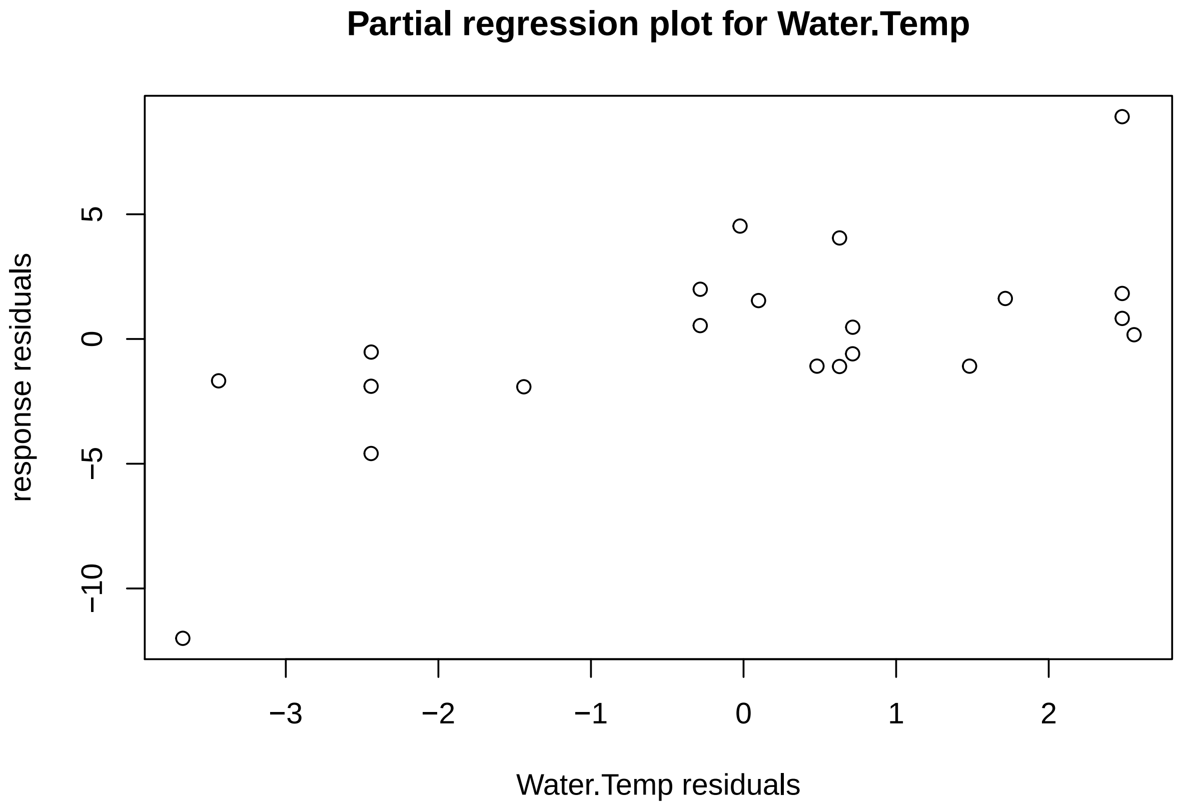

Air.Flow versus residuals
Residuals
6
4
2
0
−2
−4
−6
50
55
60
65
70
75
80
df[, predictor]

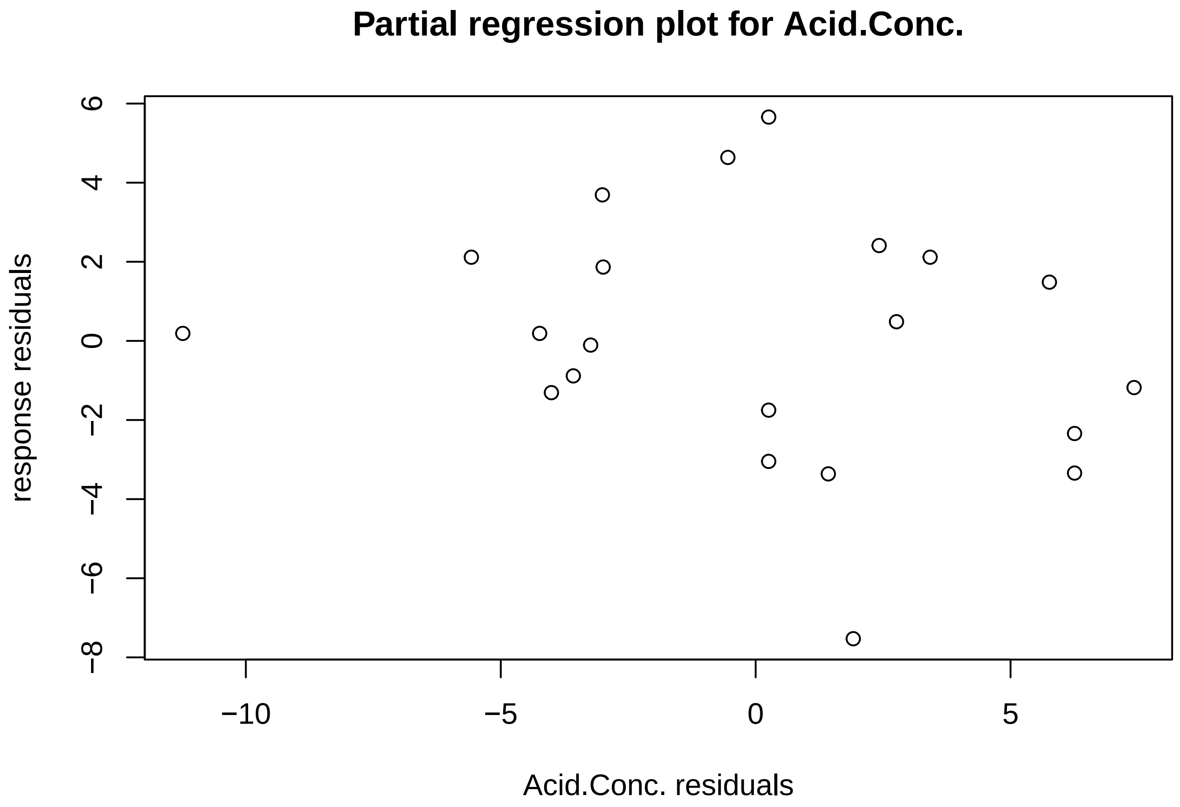

Water.Temp versus residuals
Residuals
6
4
2
0
−2
−4
−6
18
20
22
24
26
df[, predictor]



Perform partial regression


Partial regression plot for Air.Flow
response residuals
−5
0
5
−5
0
5
10
Air.Flow residuals


Partial regression plot for Water.Temp
response residuals
−10
−5
0
5
−3
−2
−1
0
1
2
Water.Temp residuals

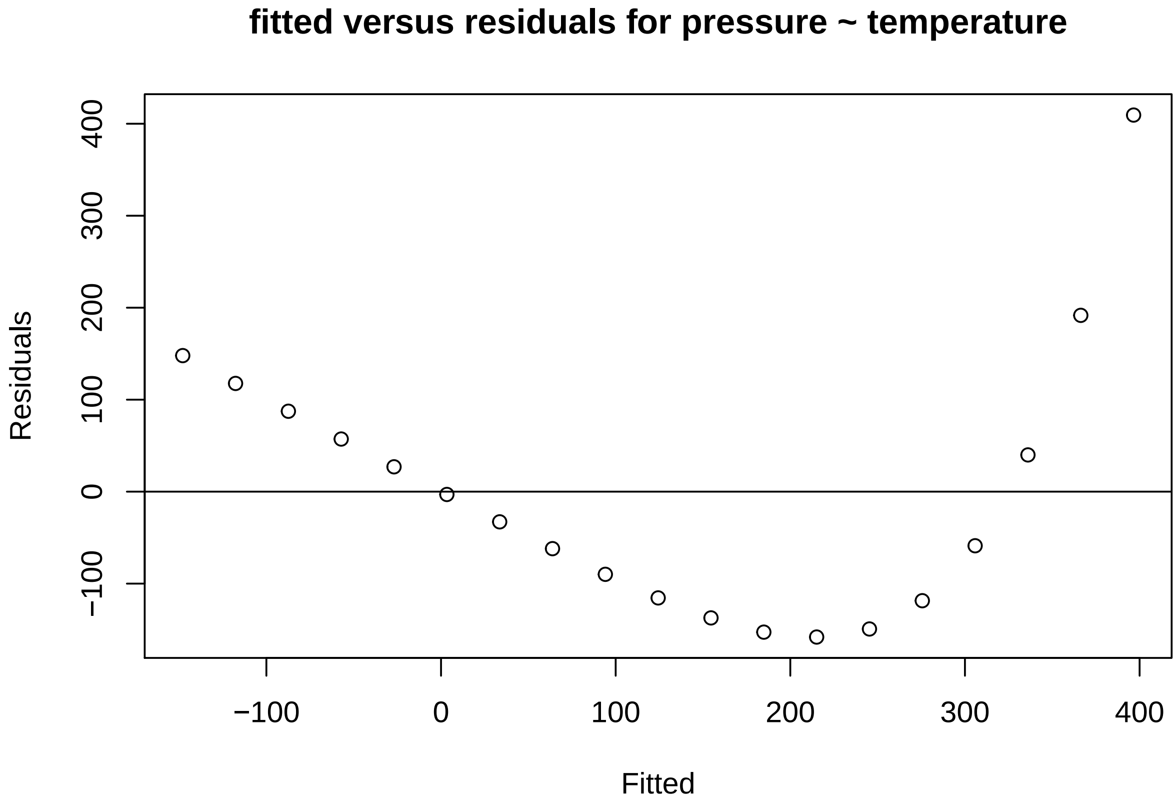


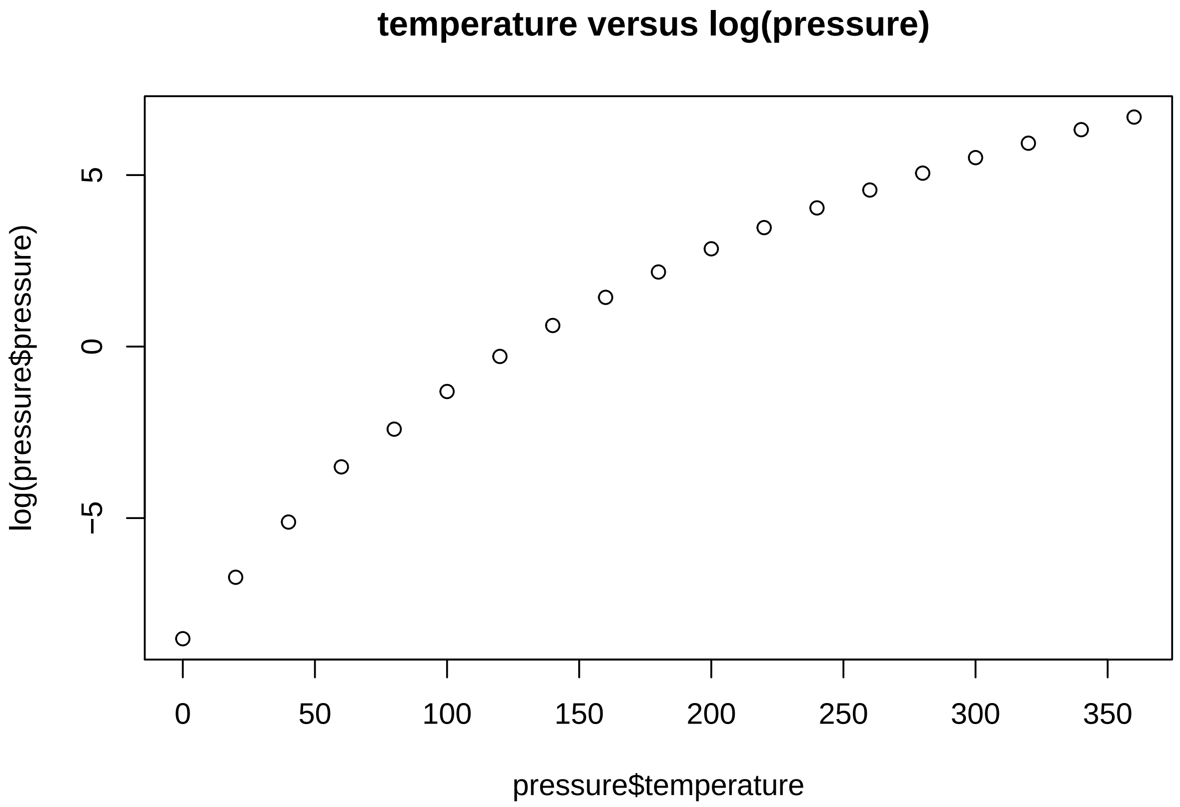
9.4 Using transformations in model of pressure data

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
##
```

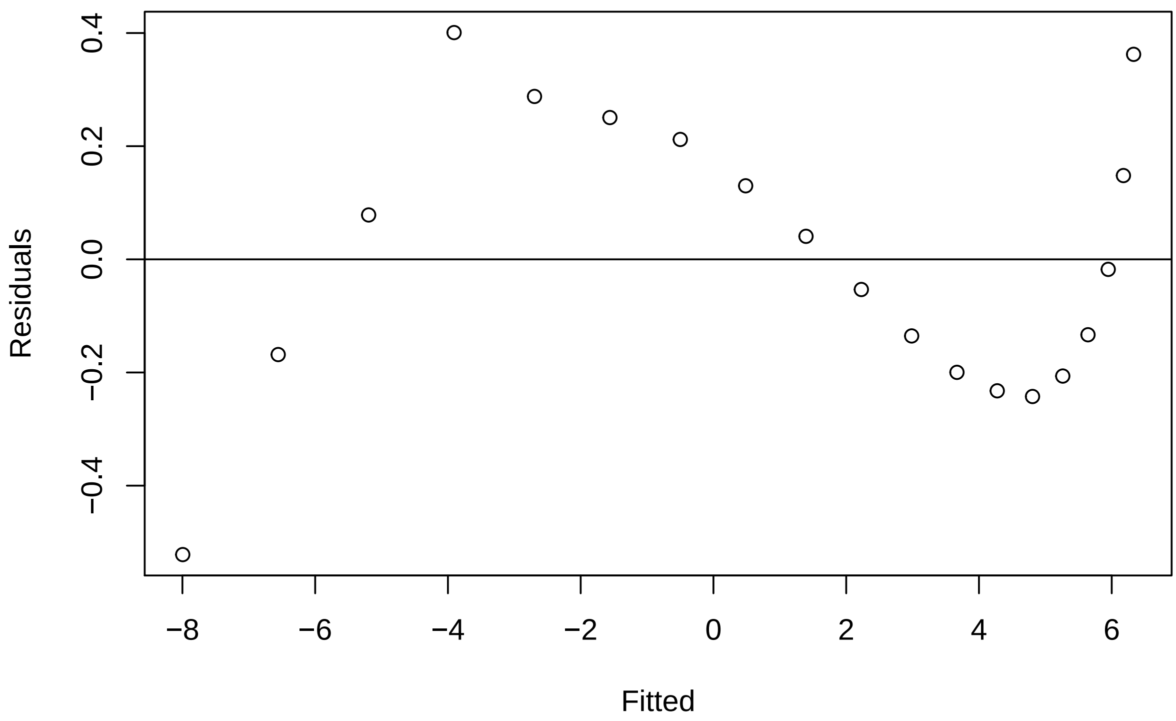
```
## Residual standard error: 150.8 on 17 degrees of freedom
## Multiple R-squared:  0.5742, Adjusted R-squared:  0.5492
## F-statistic: 22.93 on 1 and 17 DF,  p-value: 0.000171
```



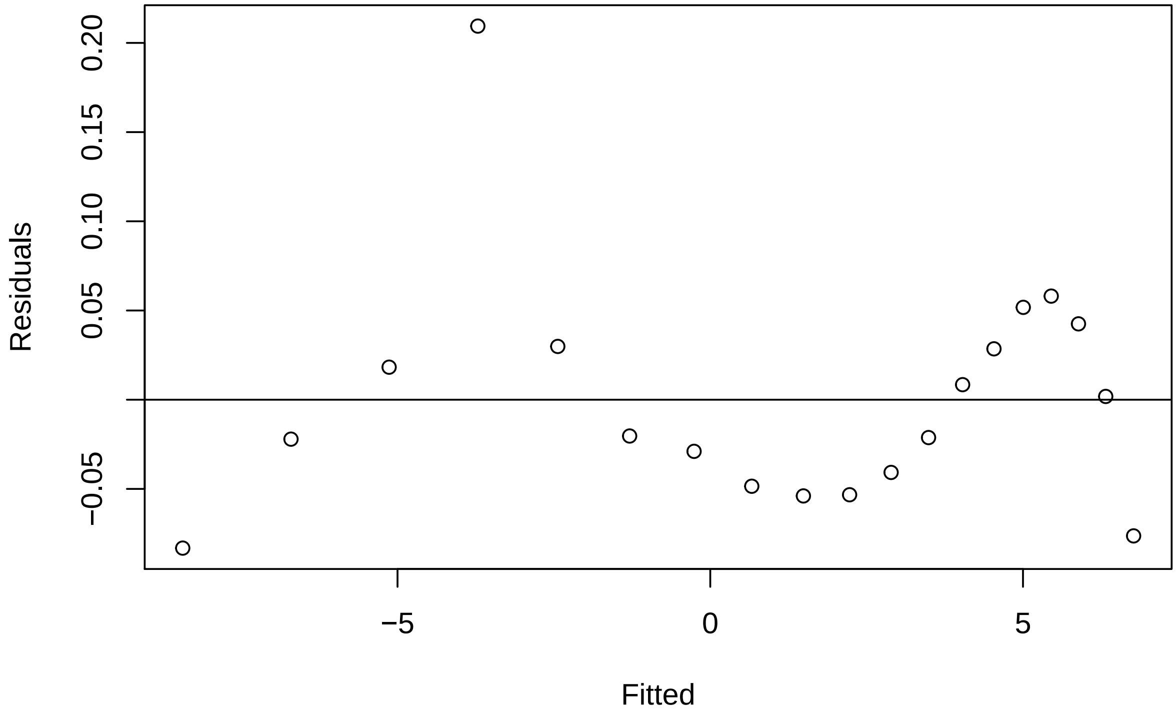


Based on the plots above we look into fitting a series of models of the form $log(pressure) \sim \sum b_i temperature^i$ We note this data looks highly regular, and appears to originate from a physical process. There's obviously some functional relationship between these variables. Knowing this may help 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
```



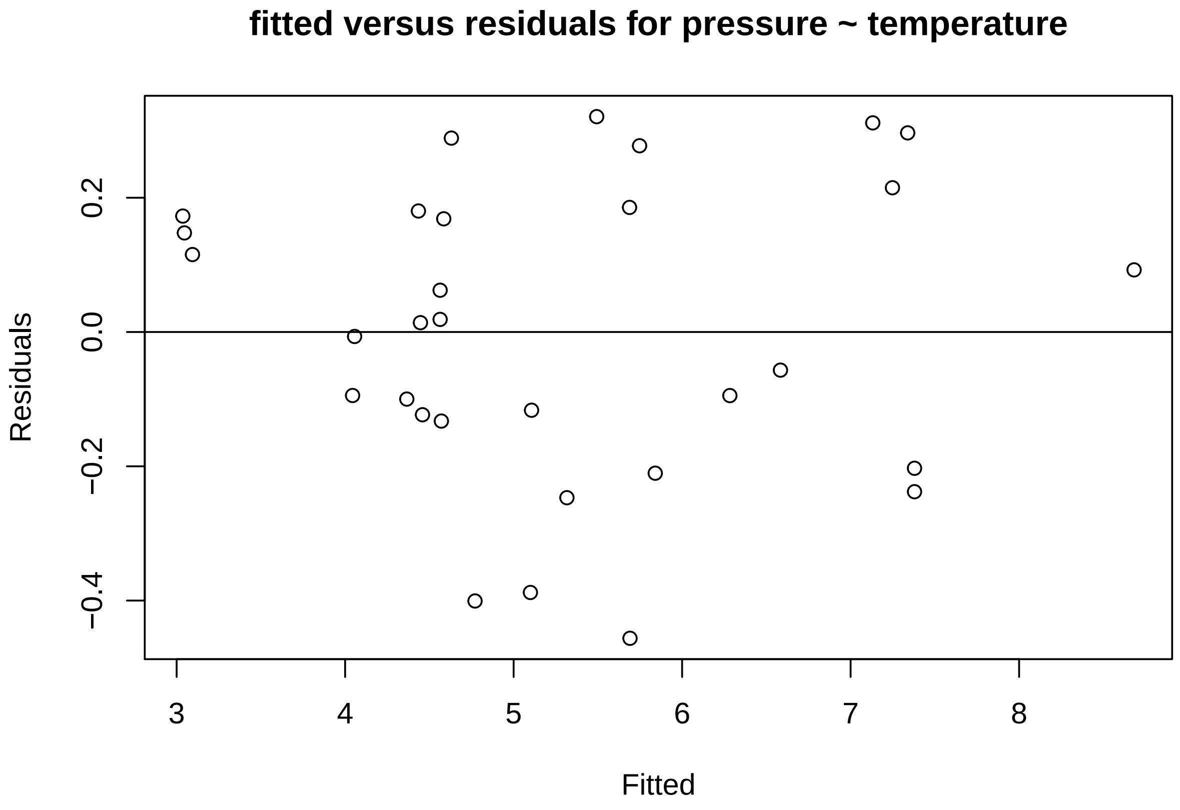
```
##
## Call:
## lm(formula = log(pressure) ~ temperature + I(temperature^2) +
##     I(temperature^3), data = pressure)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.08319 -0.04463 -0.02035  0.02919  0.20942
##
## Coefficients:
##                    Estimate Std. Error t value 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
```

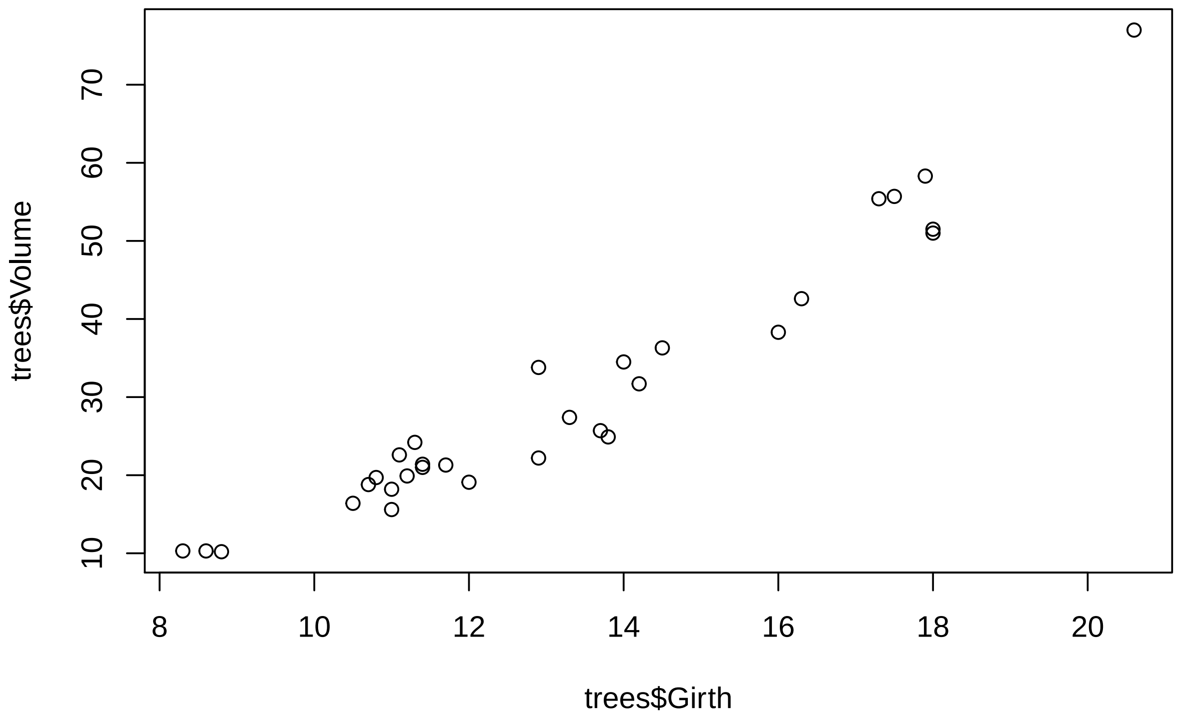


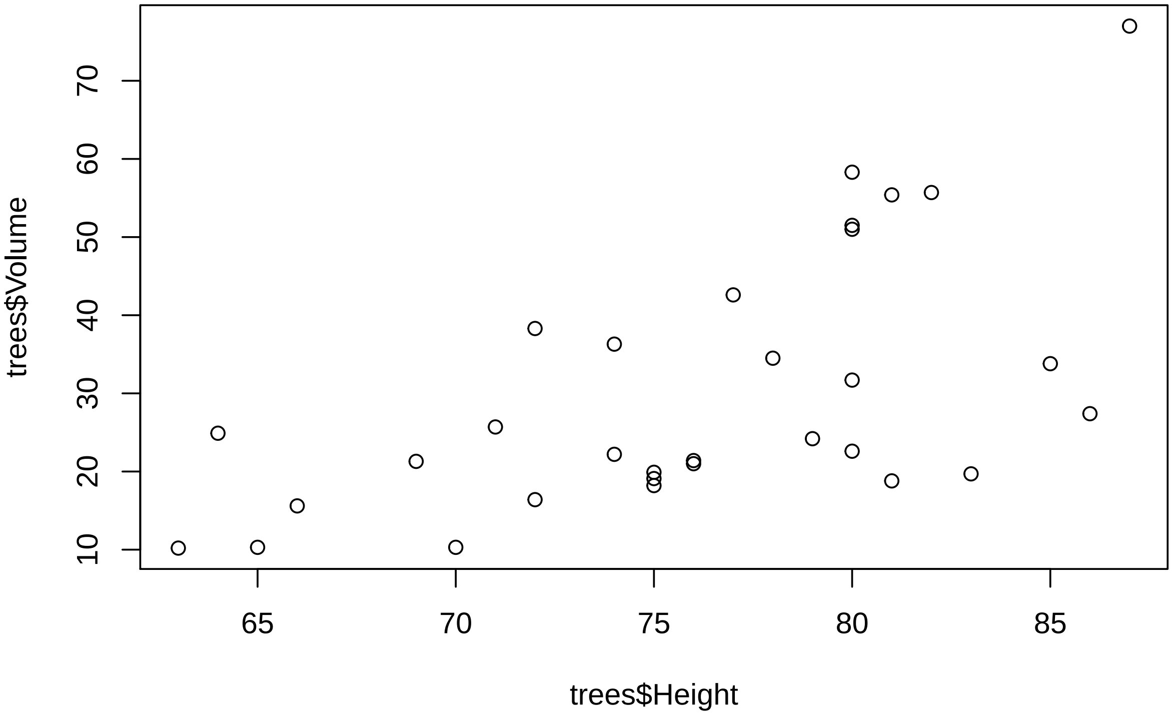
9.5 Use transformations to find a good model for volume in terms of girth and height using the trees data.

```
##
## 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:
```

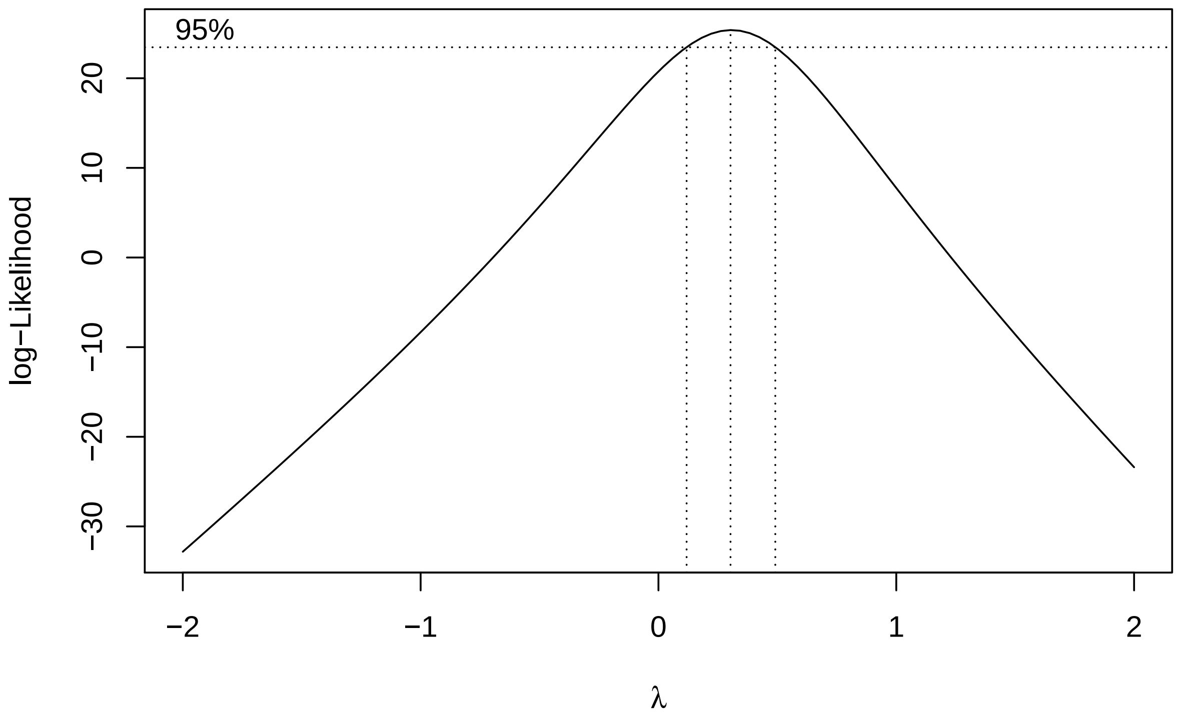
```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.769955   0.509414  -5.438 8.40e-06 ***
## Girth        0.404922   0.015584  25.983  < 2e-16 ***
## Height       0.035758   0.007675   4.659 7.05e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## 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
```

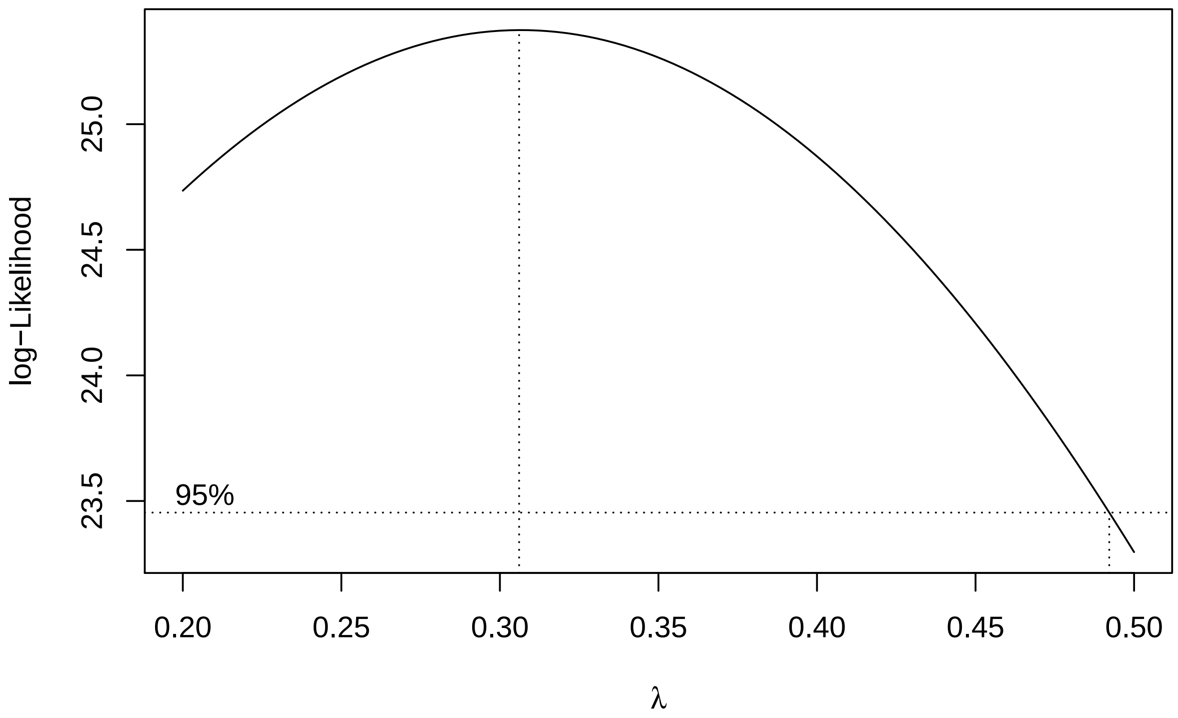



trees$Volume
10
20
30
40
50
60
70
8
10
12
14
16
18
20
trees$Girth



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
−30
−20
−10
0
10
20
−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

NCSU ST 503 HW 9

Probems 10.1 (a - c), 10.4, and 10.5 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

27 October, 2017

10.1 1 (a - c) Subset Selection with prostate data

For 10.1 (a): Please use Backward Elimination in 3 ways: (i) a 0.05 p-value criterion as the stopping rule, (ii) using AIC as the stopping rule, and (iii) using BIC as the stopping rule.

For 10.1 (b-c): You should be comparing all possible subsets.

Use the prostate data with lpsa as the response and the other variables as predictors. Implement the following variable selection methods to determine the "best" model:

(a) Backward elimination

It was not clear to be that it is possible to use regsubsets with the backward method to perform Backward Elimination based on p-value.

```
rm(list = ls())
data(prostate, package="faraway");
df <- prostate
n <-nrow(df)

lm.fit <- lm(lpsa ~ ., data=prostate)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = lpsa ~ ., data = prostate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7331 -0.3713 -0.0170  0.4141  1.6381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.669337   1.296387   0.516  0.60693    
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 ** 
## age         -0.019637   0.011173  -1.758  0.08229 .  
## lbph         0.107054   0.058449   1.832  0.07040 .  
## svi          0.766157   0.244309   3.136  0.00233 ** 
## lcp         -0.105474   0.091013  -1.159  0.24964    
## gleason      0.045142   0.157465   0.287  0.77503    
## pgg45        0.004525   0.004421   1.024  0.30886    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234 
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.fit,. ~ . - gleason)
summary(lm.subset1)
```

```
## 
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp + 
##     pgg45, data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.73117 -0.38137 -0.01728  0.43364  1.63513 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.953926   0.829439   1.150  0.25319    
## lcavol       0.591615   0.086001   6.879 8.07e-10 ***
## lweight      0.448292   0.167771   2.672  0.00897 ** 
## age         -0.019336   0.011066  -1.747  0.08402 .  
## lbph         0.107671   0.058108   1.853  0.06720 .  
## svi          0.757734   0.241282   3.140  0.00229 ** 
## lcp         -0.104482   0.090478  -1.155  0.25127    
## pgg45        0.005318   0.003433   1.549  0.12488    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7048 on 89 degrees of freedom
## Multiple R-squared:  0.6544, Adjusted R-squared:  0.6273 
## F-statistic: 24.08 on 7 and 89 DF,  p-value: < 2.2e-16
```

```r
lm.subset1 <- update(lm.subset1,. ~ . - lcp)
summary(lm.subset1)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + pgg45,
##     data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.77711 -0.41708  0.00002  0.40676  1.59681
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.980085   0.830665   1.180  0.24116
## lcavol       0.545770   0.076431   7.141 2.31e-10 ***
## lweight      0.449450   0.168078   2.674  0.00890 **
## age         -0.017470   0.010967  -1.593  0.11469
## lbph         0.105755   0.058191   1.817  0.07249 .
## svi          0.641666   0.219757   2.920  0.00442 **
## pgg45        0.003528   0.003068   1.150  0.25331
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7061 on 90 degrees of freedom
## Multiple R-squared:  0.6493, Adjusted R-squared:  0.6259
## F-statistic: 27.77 on 6 and 90 DF,  p-value: < 2.2e-16
```

```r
lm.subset1 <- update(lm.subset1,. ~ . - pgg45)
summary(lm.subset1)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83505 -0.39396  0.00414  0.46336  1.57888
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.95100    0.83175   1.143 0.255882
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *
## age         -0.01489    0.01075  -1.385 0.169528
## lbph         0.11184    0.05805   1.927 0.057160 .
## svi          0.72095    0.20902   3.449 0.000854 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.subset1,. ~ . - age)
summary(lm.subset1)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + lbph + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.82653 -0.42270  0.04362  0.47041  1.48530
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.14554    0.59747   0.244  0.80809
## lcavol       0.54960    0.07406   7.422 5.64e-11 ***
## lweight      0.39088    0.16600   2.355  0.02067 *
## lbph         0.09009    0.05617   1.604  0.11213
## svi          0.71174    0.20996   3.390  0.00103 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7108 on 92 degrees of freedom
## Multiple R-squared:  0.6366, Adjusted R-squared:  0.6208
## F-statistic: 40.29 on 4 and 92 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.subset1,. ~ . - lbph)
summary(lm.subset1)
```

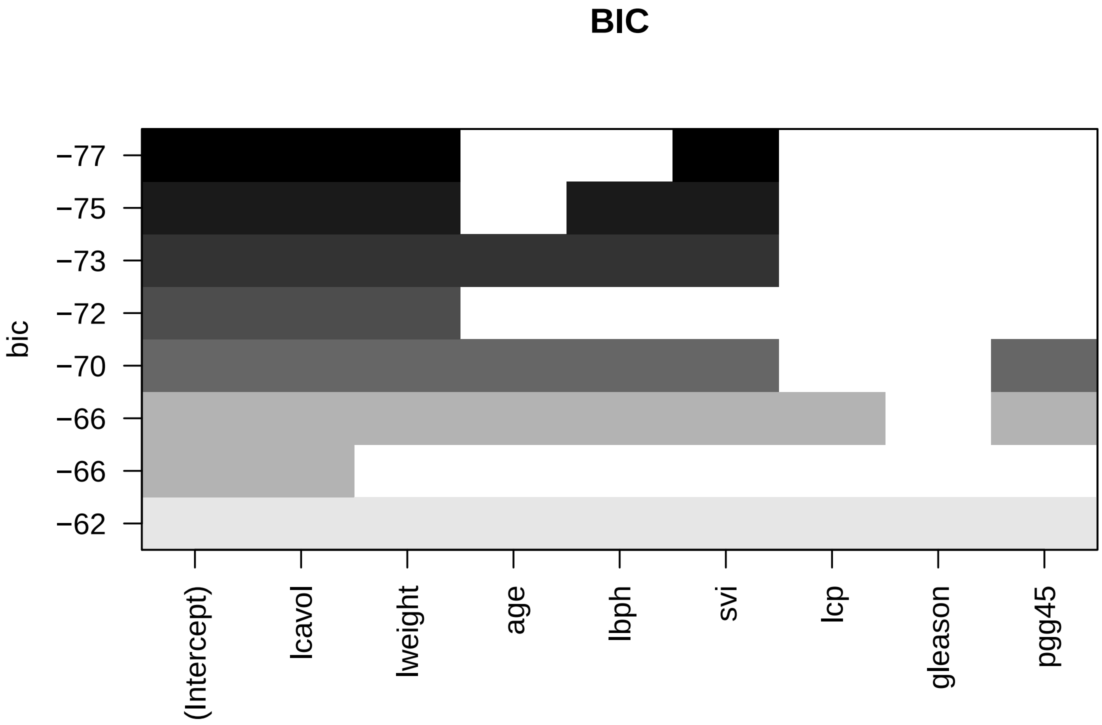
```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + svi, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.72964 -0.45764  0.02812  0.46403  1.57013
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.26809    0.54350  -0.493  0.62298
## lcavol       0.55164    0.07467   7.388  6.3e-11 ***
## lweight      0.50854    0.15017   3.386  0.00104 **
## svi          0.66616    0.20978   3.176  0.00203 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7168 on 93 degrees of freedom
## Multiple R-squared:  0.6264, Adjusted R-squared:  0.6144
## F-statistic: 51.99 on 3 and 93 DF,  p-value: < 2.2e-16
```

Backward BIC

We can use regsubsets with the backwards method to find the best model by the BIC criteria. The plot method will show us the top models. Interestingly there does not appear a way to use the plot with the AIC.

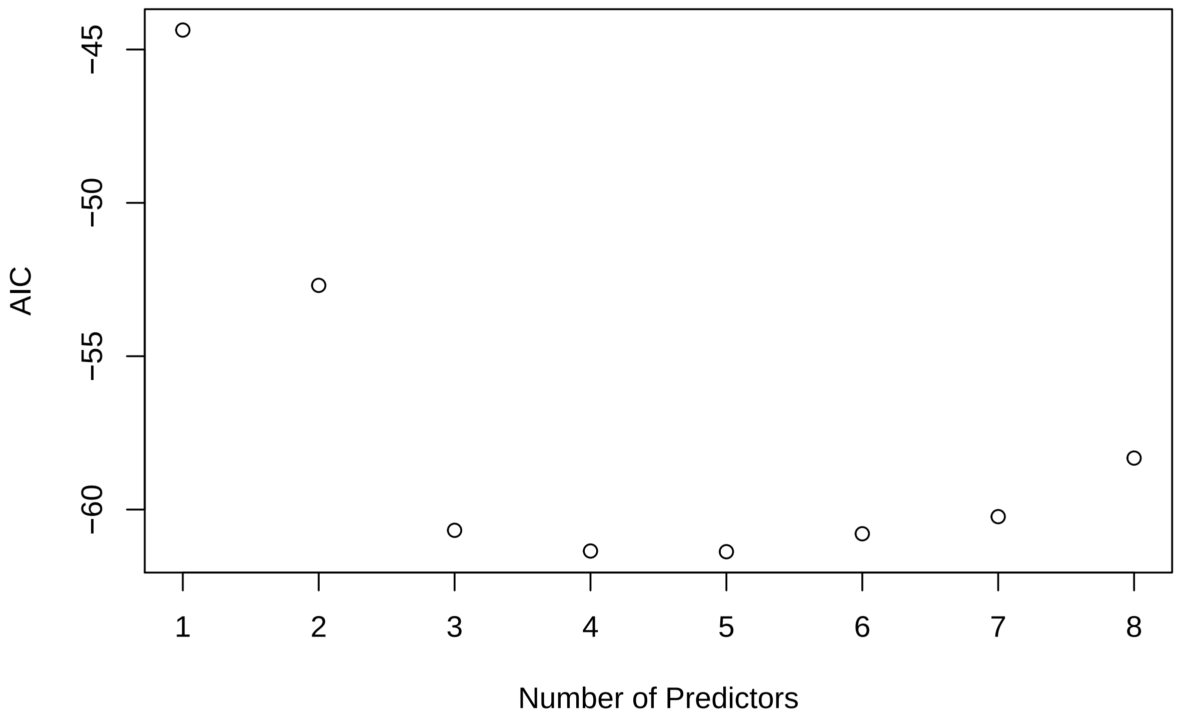
```
##   (Intercept) lcavol lweight   age  lbph   svi   lcp gleason pgg45
## 1        TRUE   TRUE   FALSE FALSE FALSE FALSE FALSE   FALSE FALSE
## 2        TRUE   TRUE    TRUE FALSE FALSE FALSE FALSE   FALSE FALSE
## 3        TRUE   TRUE    TRUE FALSE FALSE  TRUE FALSE   FALSE FALSE
## 4        TRUE   TRUE    TRUE FALSE  TRUE  TRUE FALSE   FALSE FALSE
## 5        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE FALSE
## 6        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE  TRUE
## 7        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE   FALSE  TRUE
## 8        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE    TRUE  TRUE
```



There does not appear to be a scale="aci" option for the regsubsets plot. This is interesting to note.
We plot the AIC for the models here and compare to the exhaustive search for reference.

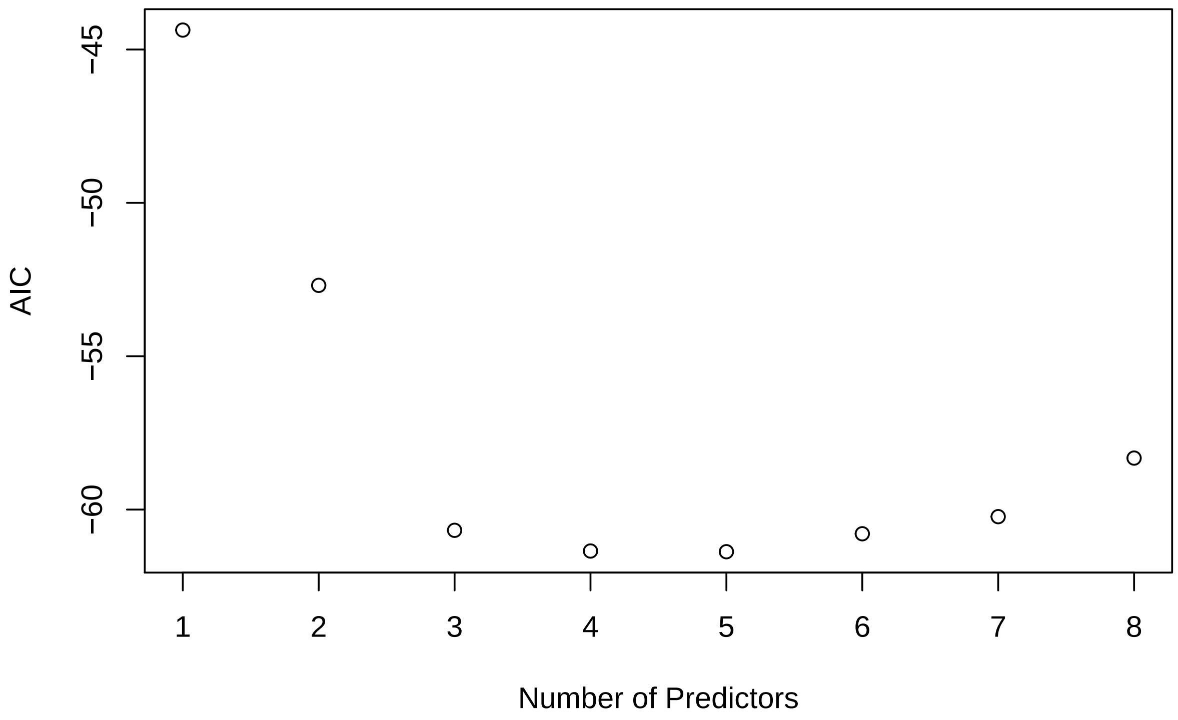
Backward AIC

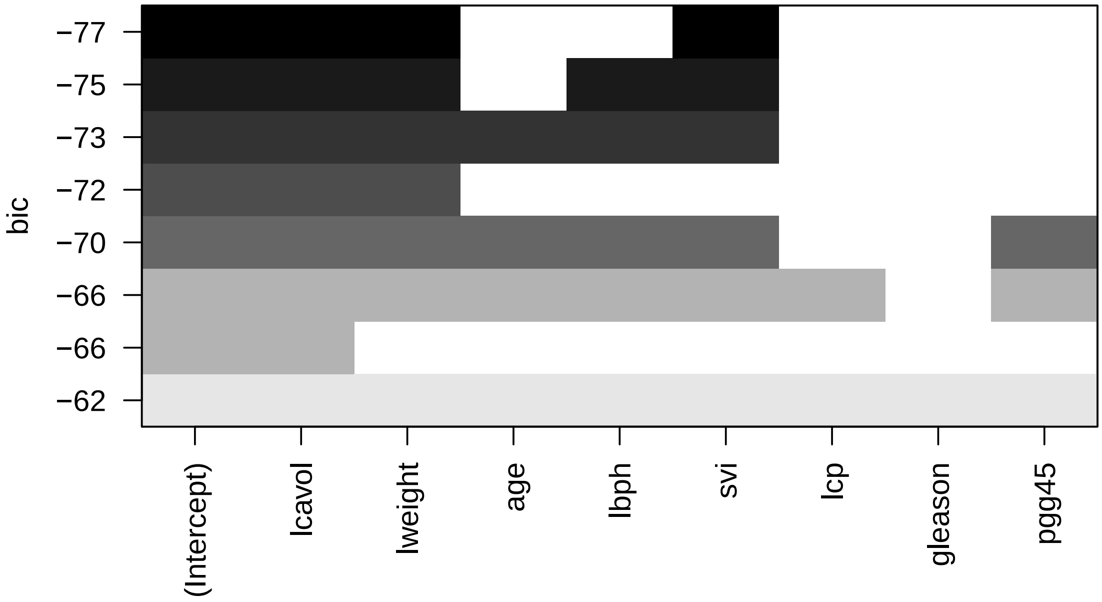
```
##   (Intercept) lcavol lweight   age  lbph   svi   lcp gleason pgg45
## 1        TRUE   TRUE   FALSE FALSE FALSE FALSE FALSE   FALSE FALSE
## 2        TRUE   TRUE    TRUE FALSE FALSE FALSE FALSE   FALSE FALSE
## 3        TRUE   TRUE    TRUE FALSE FALSE  TRUE FALSE   FALSE FALSE
## 4        TRUE   TRUE    TRUE FALSE  TRUE  TRUE FALSE   FALSE FALSE
## 5        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE FALSE
## 6        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE  TRUE
## 7        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE   FALSE  TRUE
## 8        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE    TRUE  TRUE
```



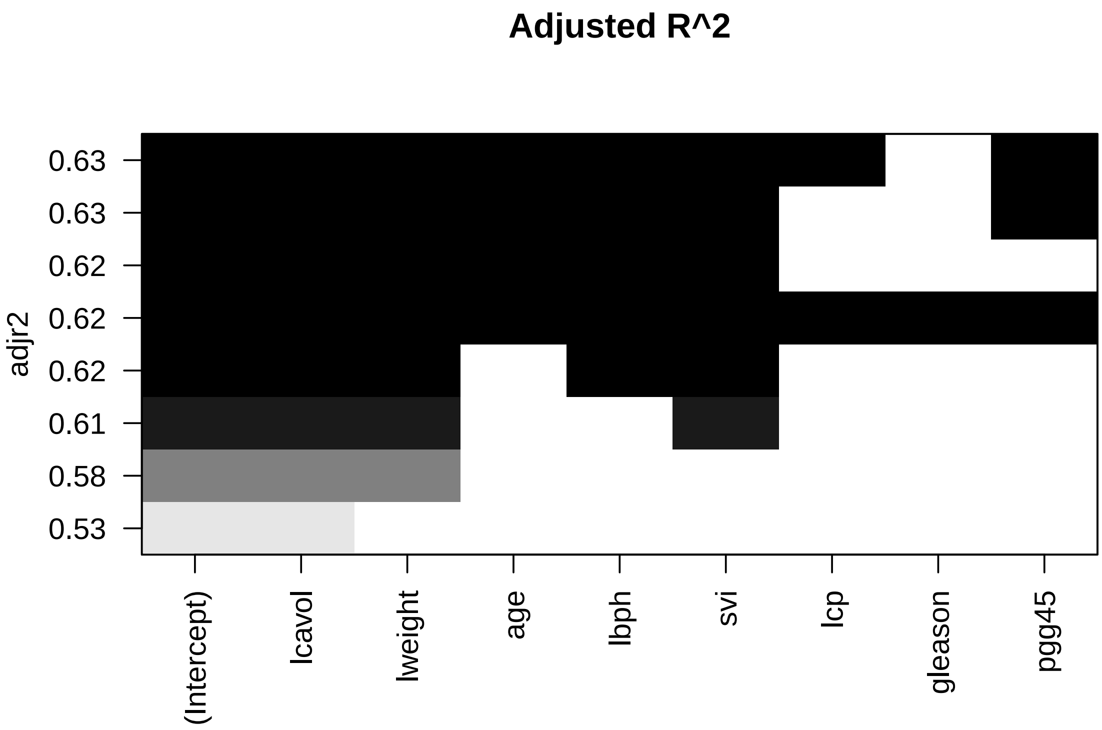
(b) exhaustive AIC

```
##   (Intercept) lcavol lweight   age  lbph   svi   lcp gleason pggg45
## 1        TRUE   TRUE   FALSE FALSE FALSE FALSE FALSE   FALSE FALSE
## 2        TRUE   TRUE    TRUE FALSE FALSE FALSE FALSE   FALSE FALSE
## 3        TRUE   TRUE    TRUE FALSE FALSE  TRUE FALSE   FALSE FALSE
## 4        TRUE   TRUE    TRUE FALSE  TRUE  TRUE FALSE   FALSE FALSE
## 5        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE FALSE
## 6        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE  TRUE
## 7        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE   FALSE  TRUE
## 8        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE    TRUE  TRUE
```


−45
−50
−55
−60
AIC
1
2
3
4
5
6
7
8
Number of Predictors


bic
−77
−75
−73
−72
−70
−66
−66
−62
(Intercept)
lcavol
lweight
age
lbph
svi
lcp
gleason
pgg45

(c) exhaustive Adjusted R^2



10.4 Simplifying trees model

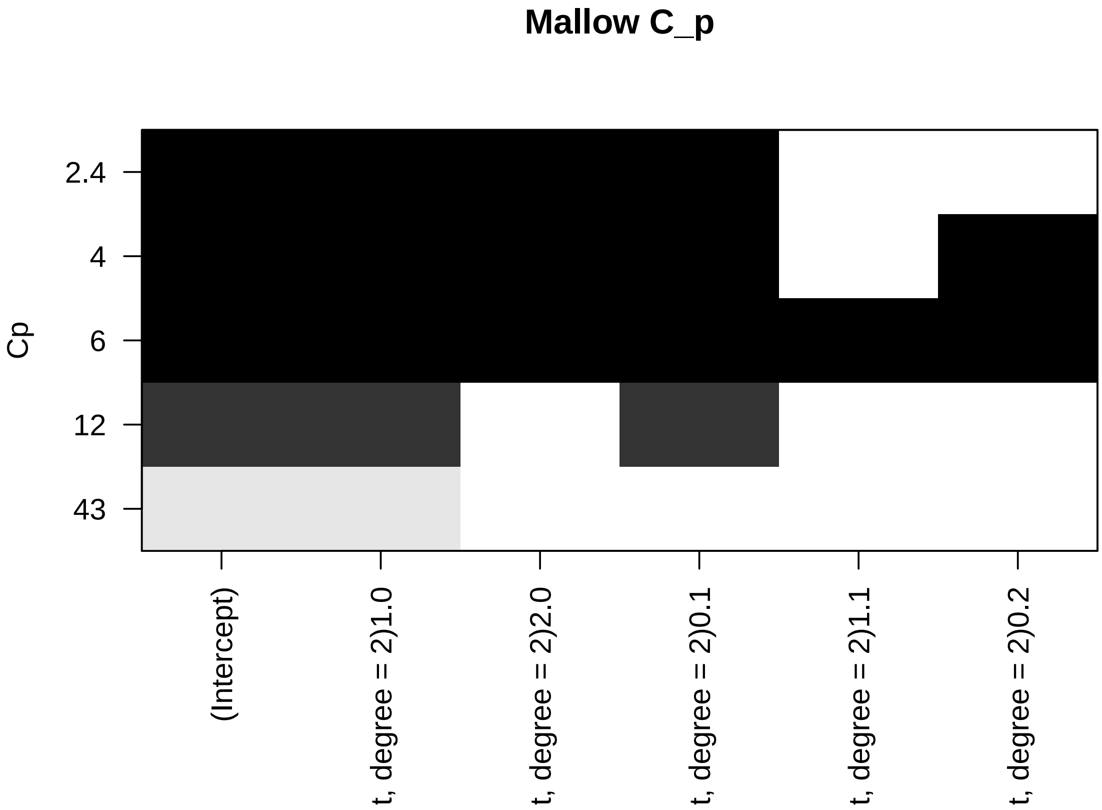
Using the trees data, fit a model with log(Volume) as the response and a second-order polynomial (including the interaction term) in Girth and Height. Determine whether the model may be reasonably simplified.

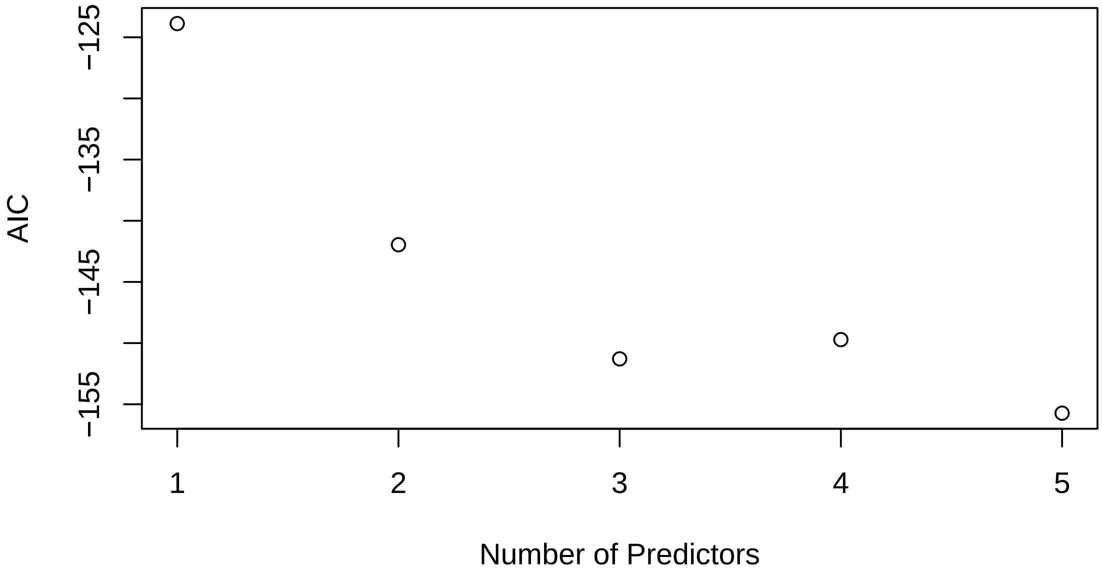
```
##
## Call:
## lm(formula = log(Volume) ~ polym(Girth, Height, degree = 2),
##     data = trees)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.159718 -0.041905 -0.003371  0.055167  0.133780
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         3.27472    0.02370 138.163  < 2e-16
## polym(Girth, Height, degree = 2)1.0  2.51882    0.11972  21.039  < 2e-16
## polym(Girth, Height, degree = 2)2.0 -0.24312    0.18449  -1.318    0.200
## polym(Girth, Height, degree = 2)0.1  0.54249    0.11339   4.784 6.52e-05
## polym(Girth, Height, degree = 2)1.1 -0.11845    1.08511  -0.109    0.914
## polym(Girth, Height, degree = 2)0.2 -0.05025    0.10402  -0.483    0.633
##
## (Intercept)                         ***
## polym(Girth, Height, degree = 2)1.0 ***
## polym(Girth, Height, degree = 2)2.0
## polym(Girth, Height, degree = 2)0.1 ***
## polym(Girth, Height, degree = 2)1.1
## polym(Girth, Height, degree = 2)0.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08469 on 25 degrees of freedom
## Multiple R-squared:  0.9784, Adjusted R-squared:  0.9741
## F-statistic: 226.7 on 5 and 25 DF,  p-value: < 2.2e-16
```

Now we run subset selection. We'll use an exhaustive method since there are not too many predictors. And we'll use the Mallow C_p as our criteria.

```
##                         (Intercept) polym(Girth, Height, degree = 2)1.0
##                          3.27471581                          2.51881720
## polym(Girth, Height, degree = 2)2.0 polym(Girth, Height, degree = 2)0.1
##                         -0.24312237                          0.54248964
## polym(Girth, Height, degree = 2)1.1 polym(Girth, Height, degree = 2)0.2
##                         -0.11844598                         -0.05024754
```

```
##   (Intercept) polym(Girth, Height, degree = 2)1.0
## 1        TRUE                                TRUE
## 2        TRUE                                TRUE
## 3        TRUE                                TRUE
## 4        TRUE                                TRUE
## 5        TRUE                                TRUE
##   polym(Girth, Height, degree = 2)2.0 polym(Girth, Height, degree = 2)0.1
## 1                               FALSE                               FALSE
## 2                               FALSE                                TRUE
## 3                                TRUE                                TRUE
## 4                                TRUE                                TRUE
## 5                                TRUE                                TRUE
##   polym(Girth, Height, degree = 2)1.1 polym(Girth, Height, degree = 2)0.2
## 1                               FALSE                               FALSE
## 2                               FALSE                               FALSE
## 3                               FALSE                               FALSE
## 4                               FALSE                                TRUE
## 5                                  TRUE                                  TRUE
```


Mallow C_p
Cp
2.4
4
6
12
43
(Intercept)
t, degree = 2)1.0
t, degree = 2)2.0
t, degree = 2)0.1
t, degree = 2)1.1
t, degree = 2)0.2



The AIC criterion indicates the best model is the full model with all the polynomial terms, while the Mallow Cp indicates the best model is a reduced one with the first three terms of the polynomial expansion :

$log(Volume) \sim polym(Girth, Height, degree = 2)1.0 + polym(Girth, Height, degree = 2)2.0 + polym(Girth, Height, degree = 2)0.1$

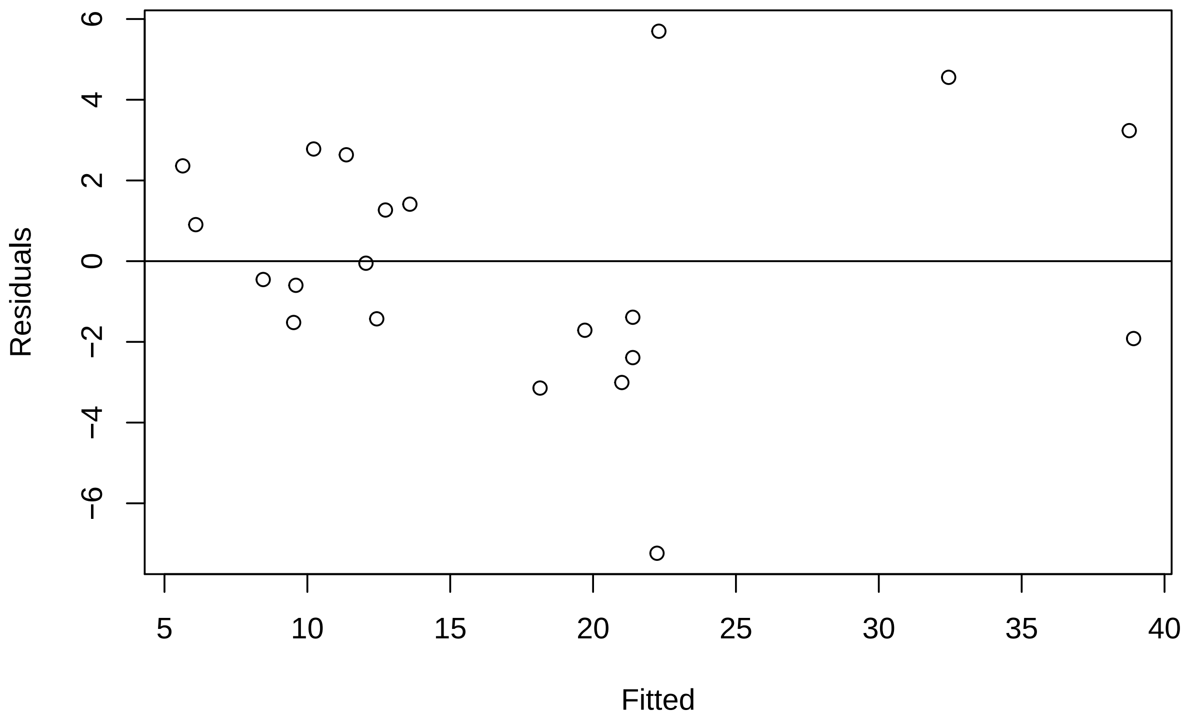
$log(Volume) \sim Girth + Girth^2, +Height$

10.5 Model reduction in stackloss data

Fit a linear model to the stackloss data with stack.loss as the predictor and the other variables as predictors.

```
##
## Call:
## lm(formula = stack.loss ~ ., data = stackloss)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2377 -1.7117 -0.4551  2.3614  5.6978
```

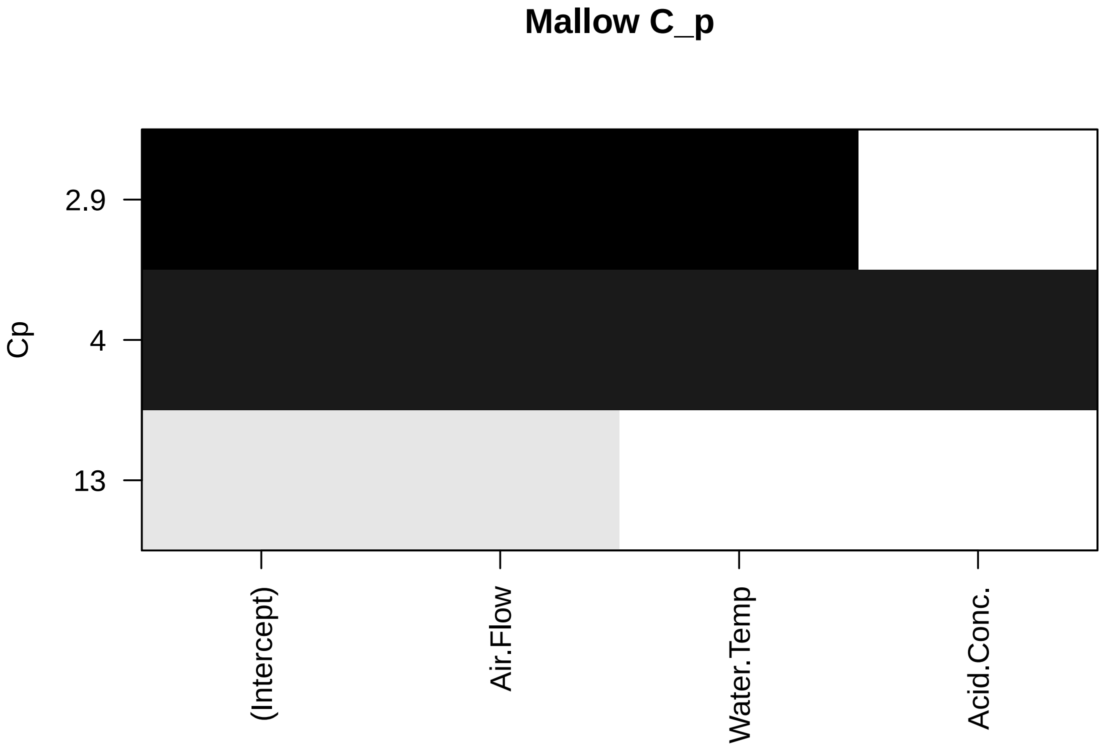
```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -39.9197    11.8960  -3.356  0.00375 **
## Air.Flow      0.7156     0.1349   5.307  5.8e-05 ***
## Water.Temp    1.2953     0.3680   3.520  0.00263 **
## Acid.Conc.   -0.1521     0.1563  -0.973  0.34405
## ---
## 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
```

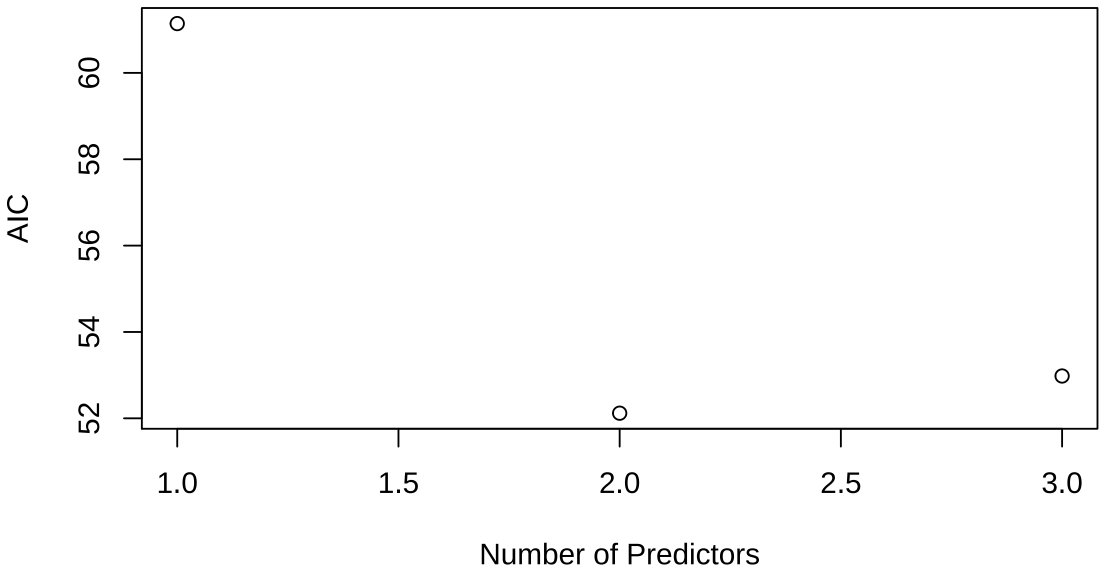


Simplify the model if possible.

Now we run subset selection. We'll use an exhaustive method since there are not too many predictors. And we'll use the Mallow C_p as our criteria.

```
##   (Intercept) Air.Flow Water.Temp Acid.Conc.
## 1        TRUE     TRUE      FALSE      FALSE
## 2          TRUE     TRUE        TRUE       FALSE
## 3          TRUE     TRUE        TRUE        TRUE
```





The reduced model with the lowest AIC has 2 variables and is $stack.loss \sim Air.Flow + Water.Temp$. We note this is the same model indicated by the Mallow Cp criterion.

Check the model for outliers and influential points.

Check for outliers.

Table 1: Range of Studentized residuals

range.residuals.left	range.residuals.right
-3.471	2.027

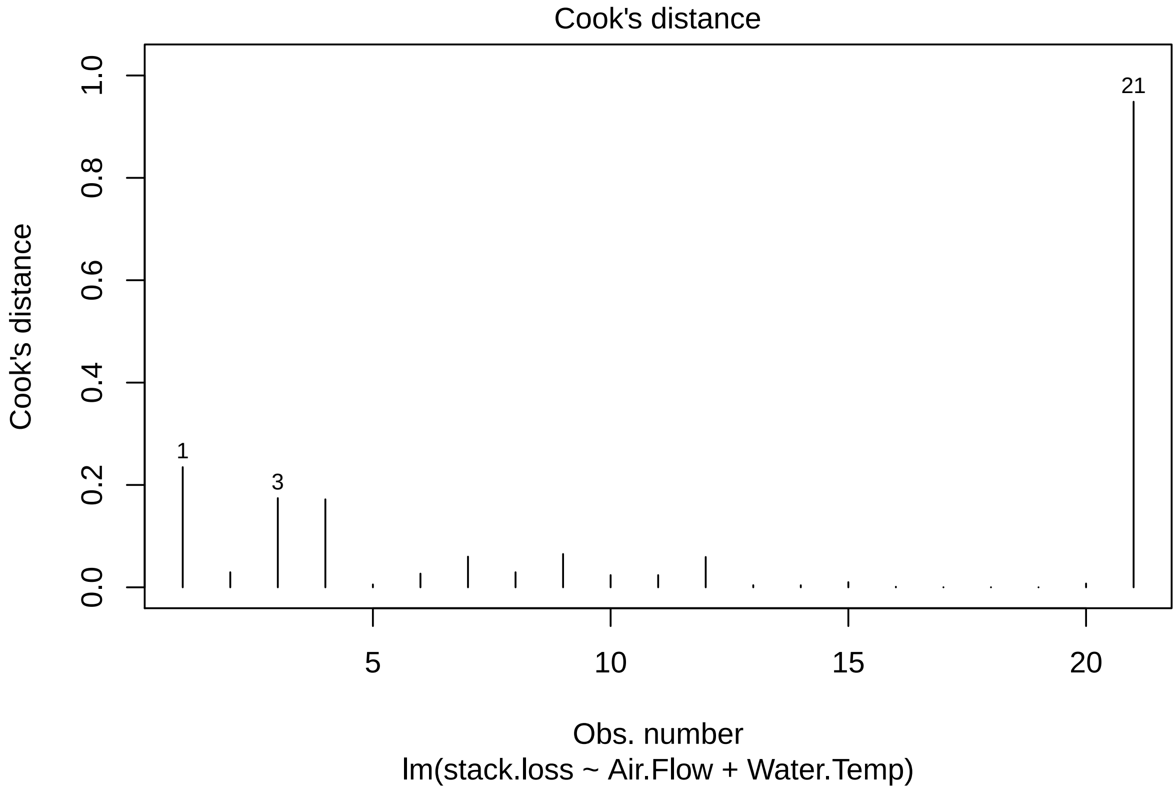
Table 2: Bonferroni corrected t-value

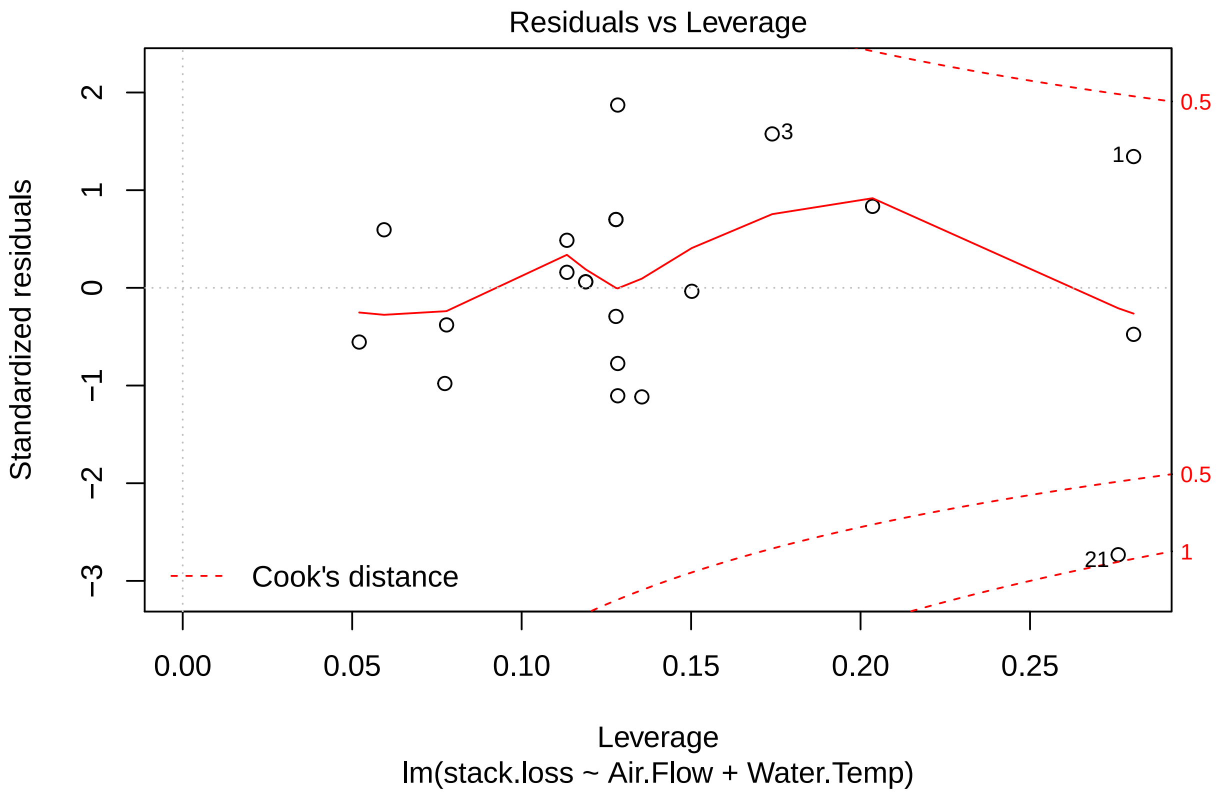
t.val.alpha
-3.565

Here we look for studentized residuals that fall outside the interval given by the Bonferroni corrected t-values. In the case of the reduced model we do not see any outliers.

Check for influential points.

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





We see that data element 21 is an influential point for the reduced model under the criteria $D_i > \frac{1}{2}$. Elements 1 and 3 are also influential under the criteria $D_i > \frac{4}{n}$

Now return to the full model, determine whether there are any outliers or influential points

Check for outliers.

Table 3: Range of Studentized residuals

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

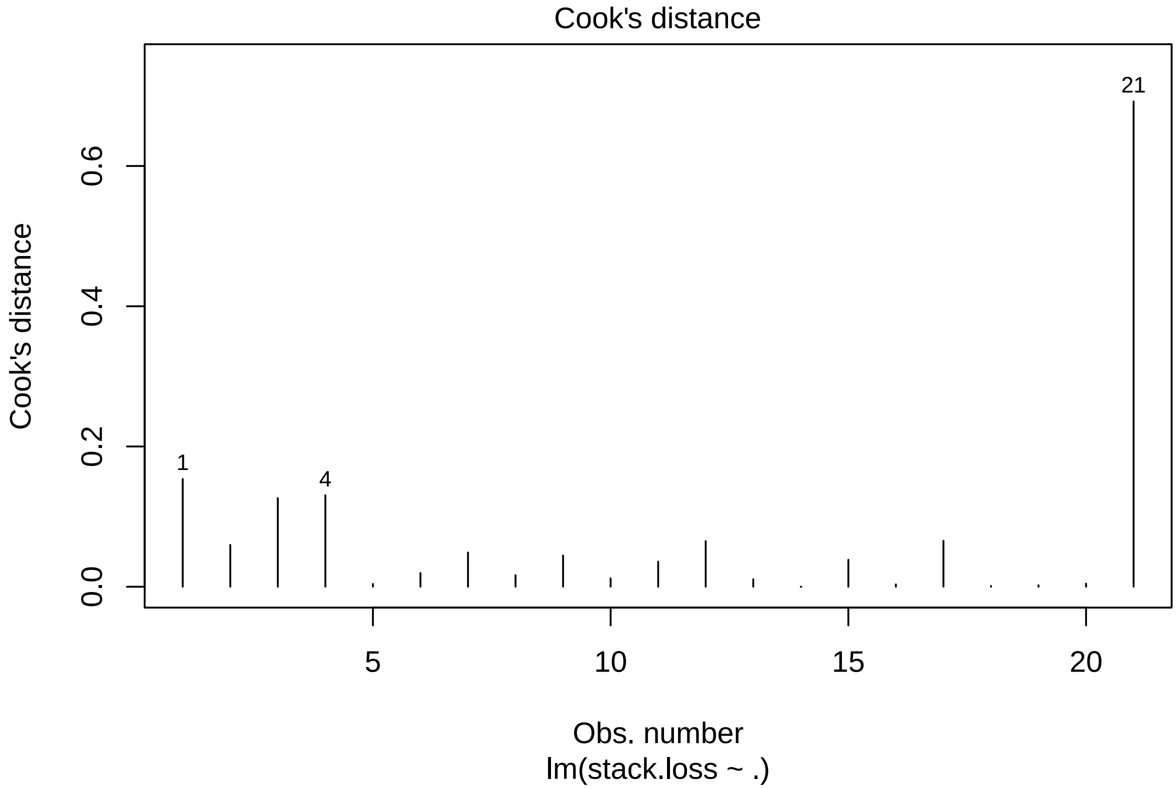
Table 4: Bonferroni corrected t-value

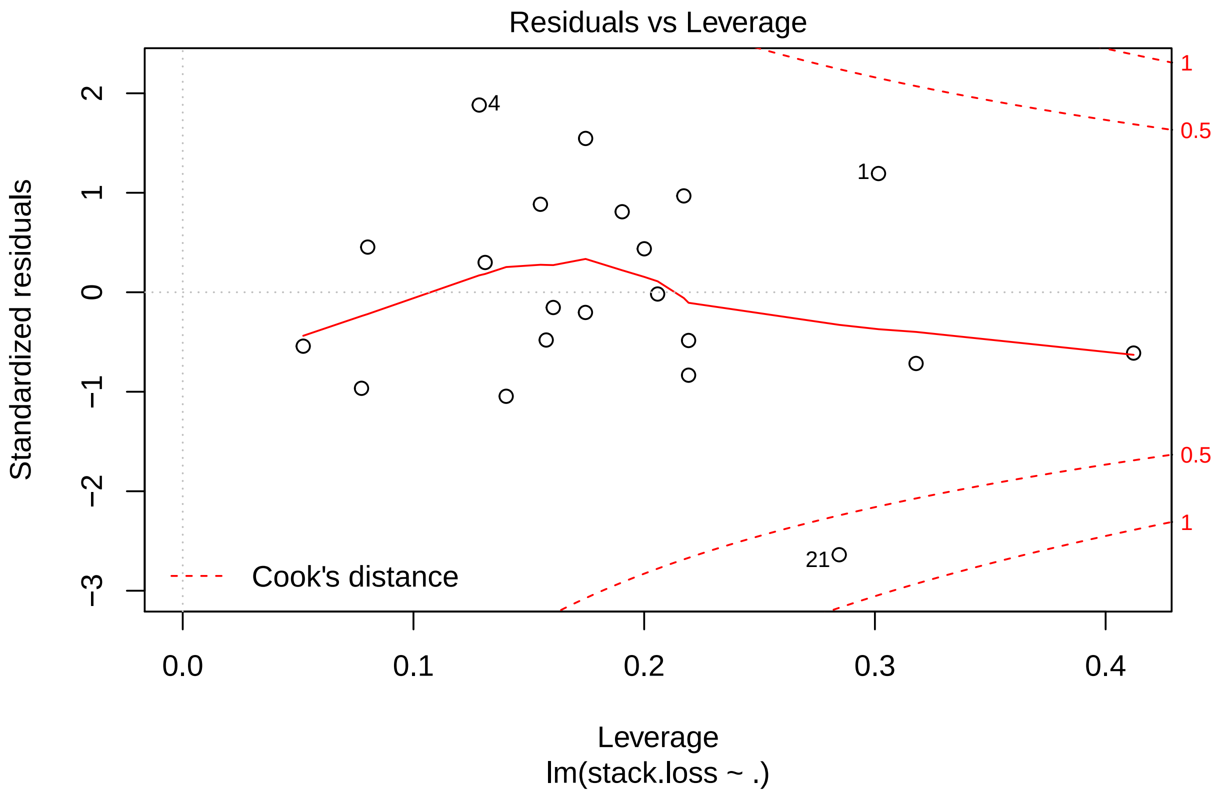
t.val.alpha
-3.604

Here we look for studentized residuals that fall outside the interval given by the Bonferroni corrected t-values. we see there are no outliers for the full model

Check for influential points.

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

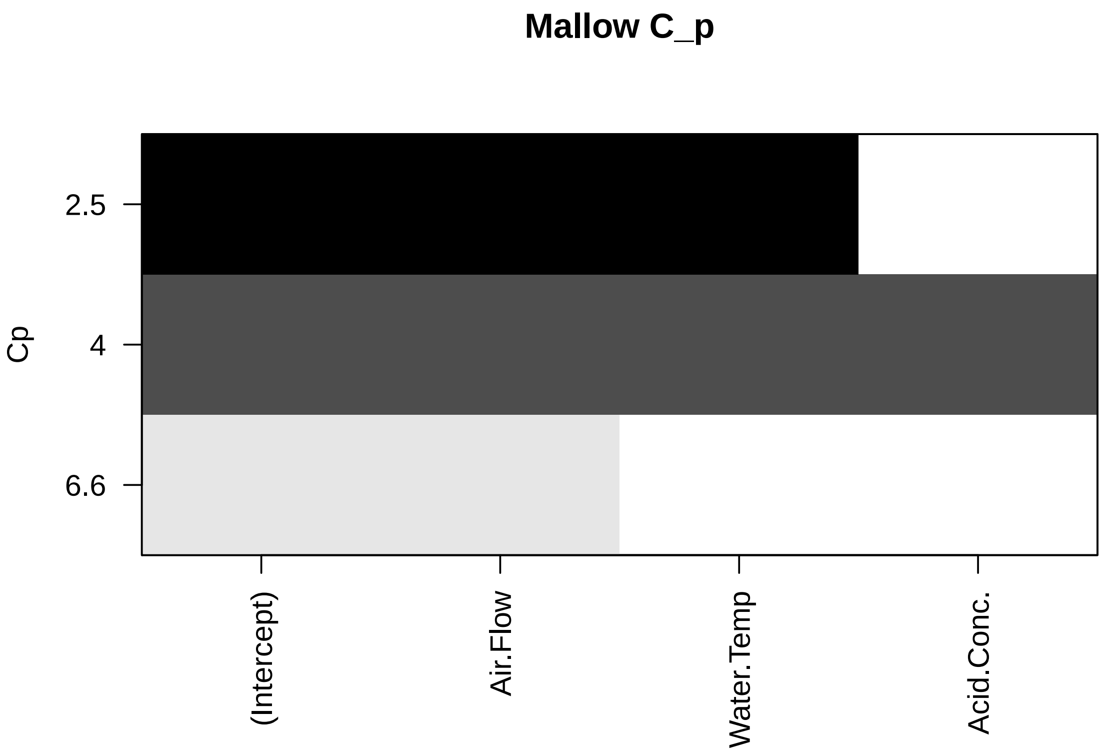


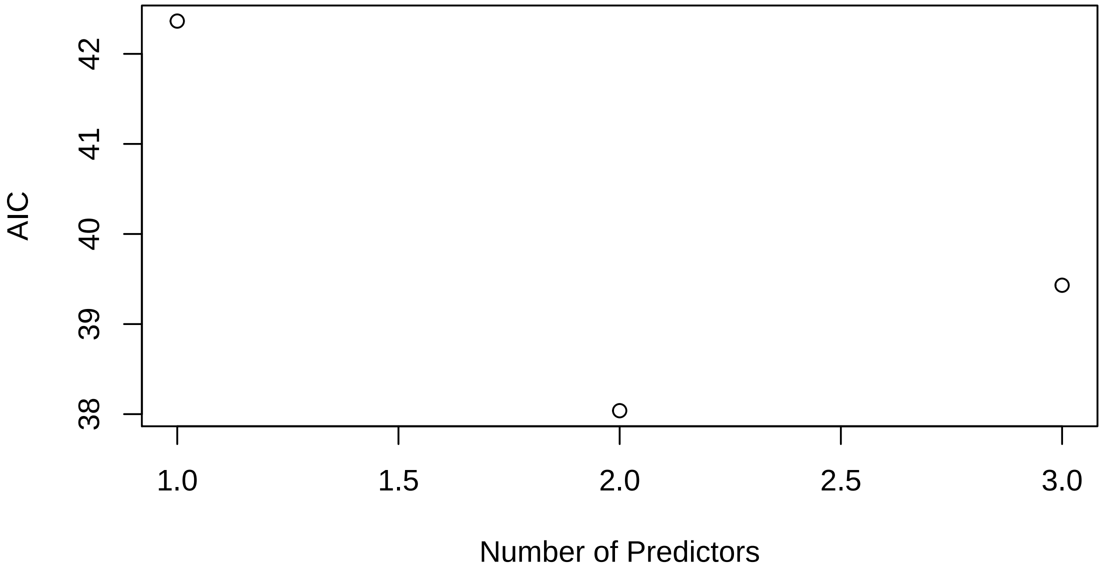


We see element 21 is an influential point, and that 1 and 4 are also influential under the criteria $D_i > \frac{4}{n}$. Since element 1 is an influential in both the full and reduced model we remove that along with element 21.

Eliminate the outliers and influential points for the full model and then repeat the variable selection procedures.

```
##   (Intercept) Air.Flow Water.Temp Acid.Conc.
## 1        TRUE     TRUE      FALSE      FALSE
## 2        TRUE     TRUE       TRUE      FALSE
## 3        TRUE     TRUE       TRUE       TRUE
```


Mallow C_p
Cp
2.5
4
6.6
(Intercept)
Air.Flow
Water.Temp
Acid.Conc.



We see that the subset selection routine has chosen the same model $stack.loss \sim Air.Flow + Water.Temp$.

NCSU ST 503 HW 9

Probems 10.1 (a - c), 10.4, and 10.5 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

27 October, 2017

10.1 1 (a - c) Subset Selection with prostate data

For 10.1 (a): Please use Backward Elimination in 3 ways: (i) a 0.05 p-value criterion as the stopping rule, (ii) using AIC as the stopping rule, and (iii) using BIC as the stopping rule.

For 10.1 (b-c): You should be comparing all possible subsets.

Use the prostate data with lpsa as the response and the other variables as predictors. Implement the following variable selection methods to determine the "best" model:

(a) Backward elimination

It was not clear to be that it is possible to use regsubsets with the backward method to perform Backward Elimination based on p-value.

```
rm(list = ls())
data(prostate, package="faraway");
df <- prostate
n <-nrow(df)

lm.fit <- lm(lpsa ~ ., data=prostate)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = lpsa ~ ., data = prostate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7331 -0.3713 -0.0170  0.4141  1.6381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.669337   1.296387   0.516  0.60693    
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 ** 
## age         -0.019637   0.011173  -1.758  0.08229 .  
## lbph         0.107054   0.058449   1.832  0.07040 .  
## svi          0.766157   0.244309   3.136  0.00233 ** 
## lcp         -0.105474   0.091013  -1.159  0.24964    
## gleason      0.045142   0.157465   0.287  0.77503    
## pgg45        0.004525   0.004421   1.024  0.30886    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234 
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.fit,. ~ . - gleason)
summary(lm.subset1)
```

```
## 
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp + 
##     pgg45, data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.73117 -0.38137 -0.01728  0.43364  1.63513 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.953926   0.829439   1.150  0.25319    
## lcavol       0.591615   0.086001   6.879 8.07e-10 ***
## lweight      0.448292   0.167771   2.672  0.00897 ** 
## age         -0.019336   0.011066  -1.747  0.08402 .  
## lbph         0.107671   0.058108   1.853  0.06720 .  
## svi          0.757734   0.241282   3.140  0.00229 ** 
## lcp         -0.104482   0.090478  -1.155  0.25127    
## pgg45        0.005318   0.003433   1.549  0.12488    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7048 on 89 degrees of freedom
## Multiple R-squared:  0.6544, Adjusted R-squared:  0.6273 
## F-statistic: 24.08 on 7 and 89 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.subset1,. ~ . - lcp)
summary(lm.subset1)
```

```
## 
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + pgg45, 
##     data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.77711 -0.41708  0.00002  0.40676  1.59681 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.980085   0.830665   1.180  0.24116    
## lcavol       0.545770   0.076431   7.141 2.31e-10 ***
## lweight      0.449450   0.168078   2.674  0.00890 ** 
## age         -0.017470   0.010967  -1.593  0.11469    
## lbph         0.105755   0.058191   1.817  0.07249 .  
## svi          0.641666   0.219757   2.920  0.00442 ** 
## pgg45        0.003528   0.003068   1.150  0.25331    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7061 on 90 degrees of freedom
## Multiple R-squared:  0.6493, Adjusted R-squared:  0.6259 
## F-statistic: 27.77 on 6 and 90 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.subset1,. ~ . - pgg45)
summary(lm.subset1)
```

```
## 
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.83505 -0.39396  0.00414  0.46336  1.57888 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.95100    0.83175   1.143 0.255882    
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *  
## age         -0.01489    0.01075  -1.385 0.169528    
## lbph         0.11184    0.05805   1.927 0.057160 .  
## svi          0.72095    0.20902   3.449 0.000854 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245 
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.subset1,. ~ . - age)
summary(lm.subset1)
```

```
## 
## Call:
## lm(formula = lpsa ~ lcavol + lweight + lbph + svi, data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.82653 -0.42270  0.04362  0.47041  1.48530 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.14554    0.59747   0.244  0.80809    
## lcavol       0.54960    0.07406   7.422 5.64e-11 ***
## lweight      0.39088    0.16600   2.355  0.02067 *  
## lbph         0.09009    0.05617   1.604  0.11213    
## svi          0.71174    0.20996   3.390  0.00103 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7108 on 92 degrees of freedom
## Multiple R-squared:  0.6366, Adjusted R-squared:  0.6208 
## F-statistic: 40.29 on 4 and 92 DF,  p-value: < 2.2e-16
```

```
lm.subset1 <- update(lm.subset1,. ~ . - lbph)
summary(lm.subset1)
```

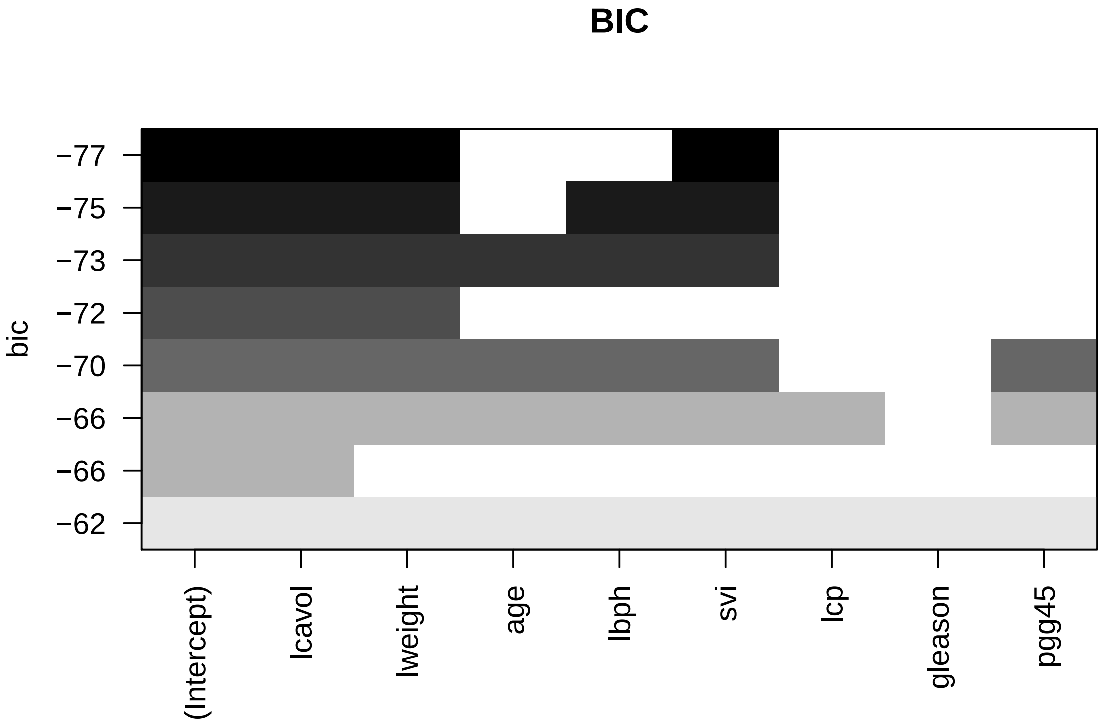
```
## 
## Call:
## lm(formula = lpsa ~ lcavol + lweight + svi, data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.72964 -0.45764  0.02812  0.46403  1.57013 
## 
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.26809    0.54350  -0.493  0.62298
## lcavol       0.55164    0.07467   7.388  6.3e-11 ***
## lweight      0.50854    0.15017   3.386  0.00104 **
## svi          0.66616    0.20978   3.176  0.00203 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7168 on 93 degrees of freedom
## Multiple R-squared:  0.6264, Adjusted R-squared:  0.6144
## F-statistic: 51.99 on 3 and 93 DF,  p-value: < 2.2e-16
```

Backward BIC

We can use regsubsets with the backwards method to find the best model by the BIC criteria. The plot method will show us the top models. Interestingly there does not appear a way to use the plot with the AIC.

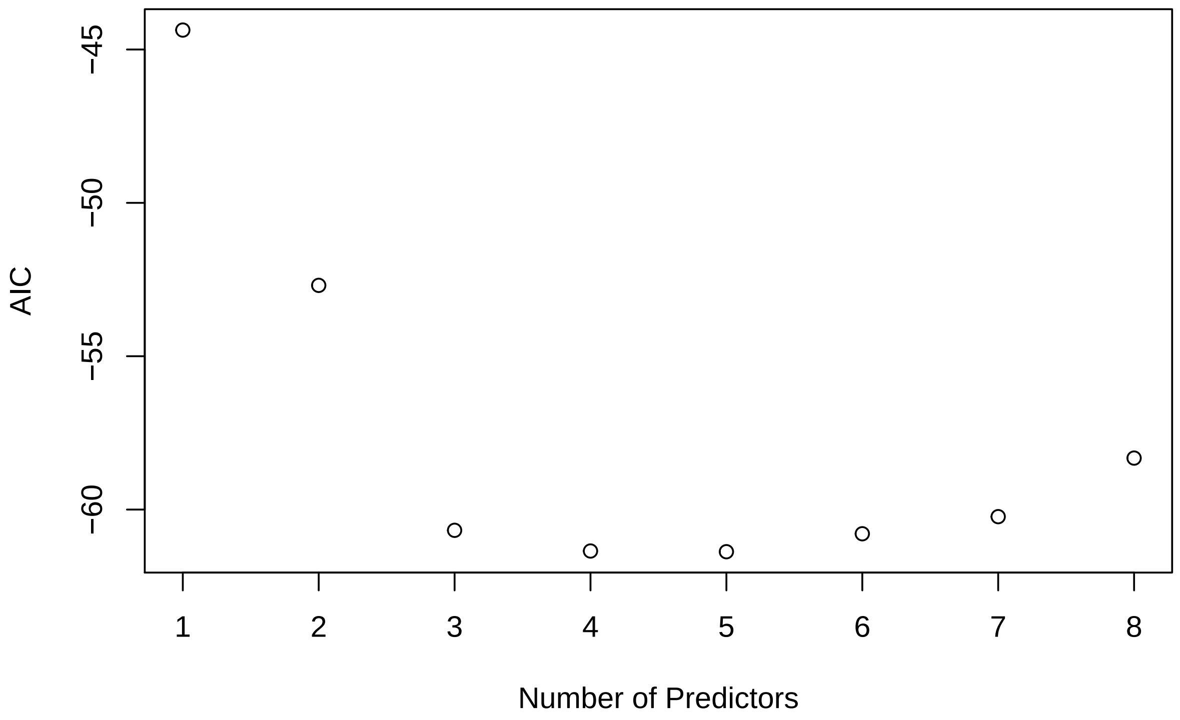
```
##   (Intercept) lcavol lweight   age  lbph   svi   lcp gleason pgg45
## 1        TRUE   TRUE   FALSE FALSE FALSE FALSE FALSE   FALSE FALSE
## 2        TRUE   TRUE    TRUE FALSE FALSE FALSE FALSE   FALSE FALSE
## 3        TRUE   TRUE    TRUE FALSE FALSE  TRUE FALSE   FALSE FALSE
## 4        TRUE   TRUE    TRUE FALSE  TRUE  TRUE FALSE   FALSE FALSE
## 5        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE FALSE
## 6        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE  TRUE
## 7        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE   FALSE  TRUE
## 8        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE    TRUE  TRUE
```



There does not appear to be a scale="aci" option for the regsubsets plot. This is interesting to note.
We plot the AIC for the models here and compare to the exhaustive search for reference.

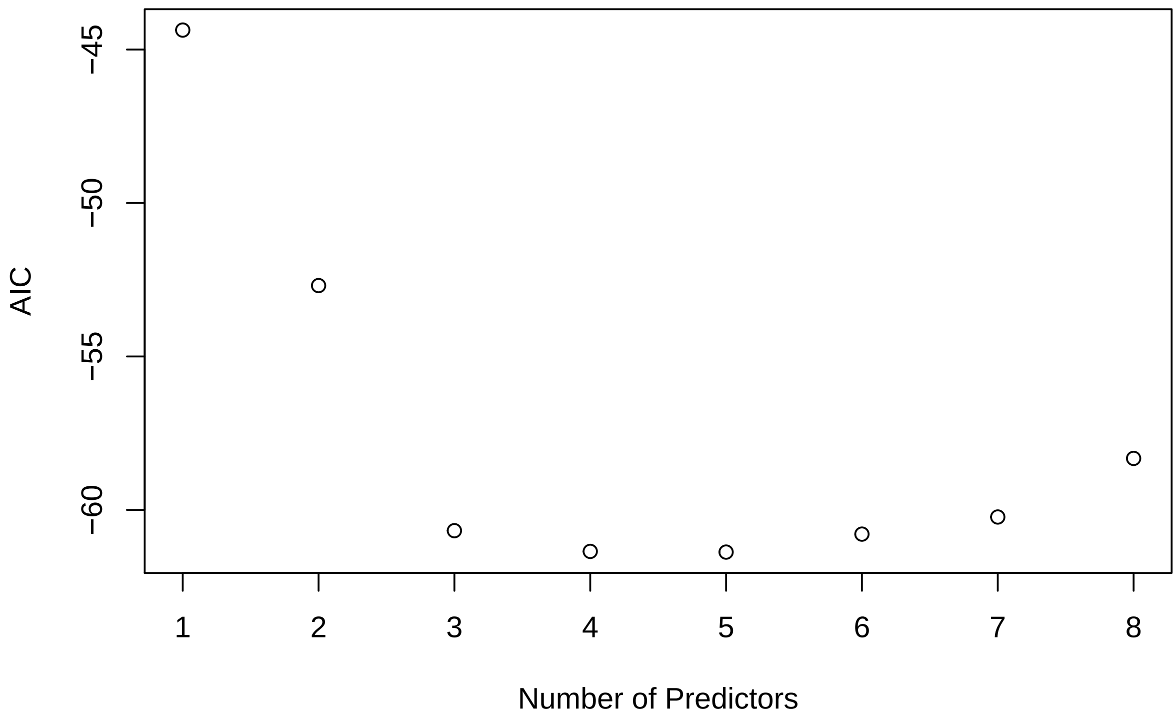
Backward AIC

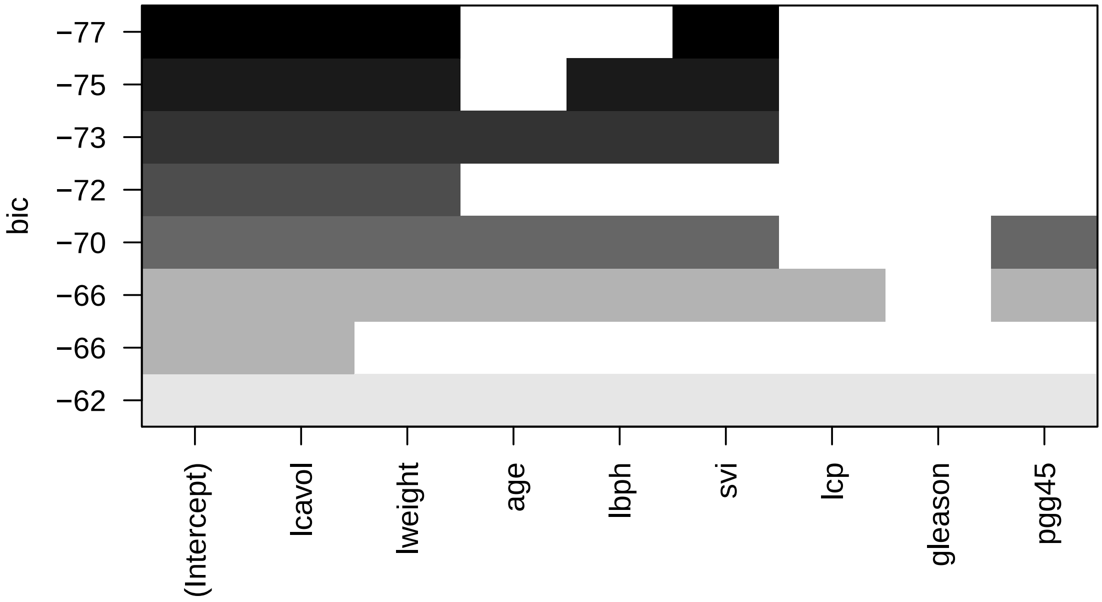
```
##   (Intercept) lcavol lweight   age  lbph   svi   lcp gleason pgg45
## 1        TRUE   TRUE   FALSE FALSE FALSE FALSE FALSE   FALSE FALSE
## 2        TRUE   TRUE    TRUE FALSE FALSE FALSE FALSE   FALSE FALSE
## 3        TRUE   TRUE    TRUE FALSE FALSE  TRUE FALSE   FALSE FALSE
## 4        TRUE   TRUE    TRUE FALSE  TRUE  TRUE FALSE   FALSE FALSE
## 5        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE FALSE
## 6        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE  TRUE
## 7        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE   FALSE  TRUE
## 8        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE    TRUE  TRUE
```



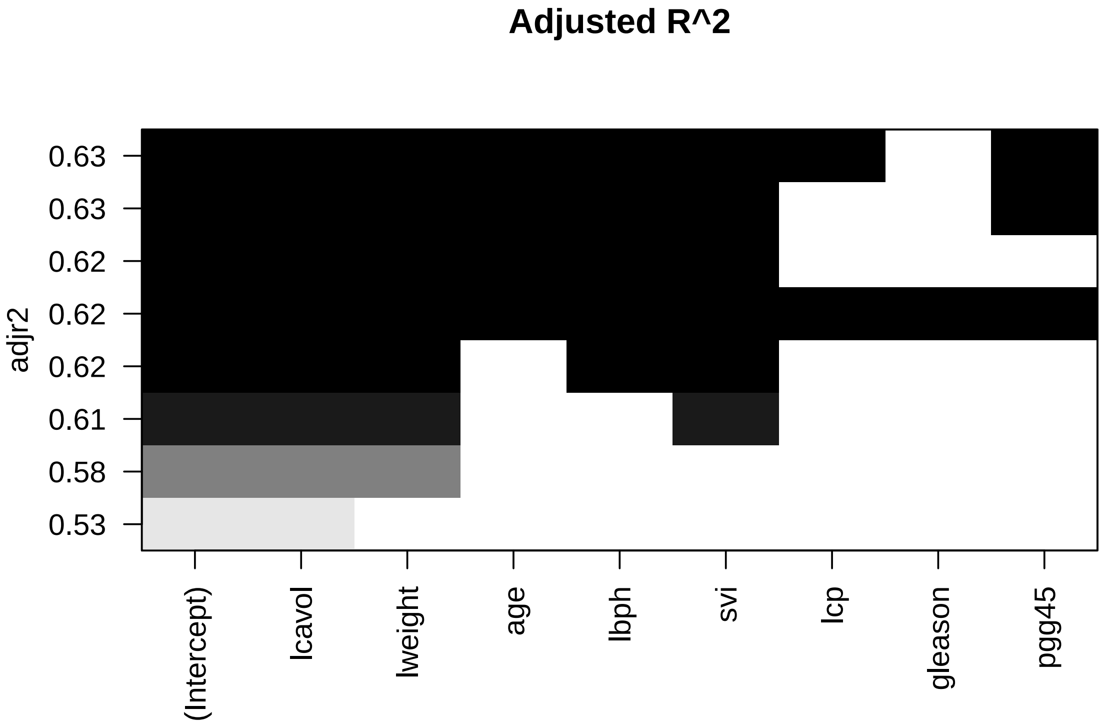
(b) exhaustive AIC

```
##   (Intercept) lcavol lweight   age  lbph   svi   lcp gleason pgg45
## 1        TRUE   TRUE   FALSE FALSE FALSE FALSE FALSE   FALSE FALSE
## 2        TRUE   TRUE    TRUE FALSE FALSE FALSE FALSE   FALSE FALSE
## 3        TRUE   TRUE    TRUE FALSE FALSE  TRUE FALSE   FALSE FALSE
## 4        TRUE   TRUE    TRUE FALSE  TRUE  TRUE FALSE   FALSE FALSE
## 5        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE FALSE
## 6        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE FALSE   FALSE  TRUE
## 7        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE   FALSE  TRUE
## 8        TRUE   TRUE    TRUE  TRUE  TRUE  TRUE  TRUE    TRUE  TRUE
```


−45
−50
−55
−60
AIC
1
2
3
4
5
6
7
8
Number of Predictors


bic
−77
−75
−73
−72
−70
−66
−66
−62
(Intercept)
lcavol
lweight
age
lbph
svi
lcp
gleason
pgg45

(c) exhaustive Adjusted R^2



10.4 Simplifying trees model

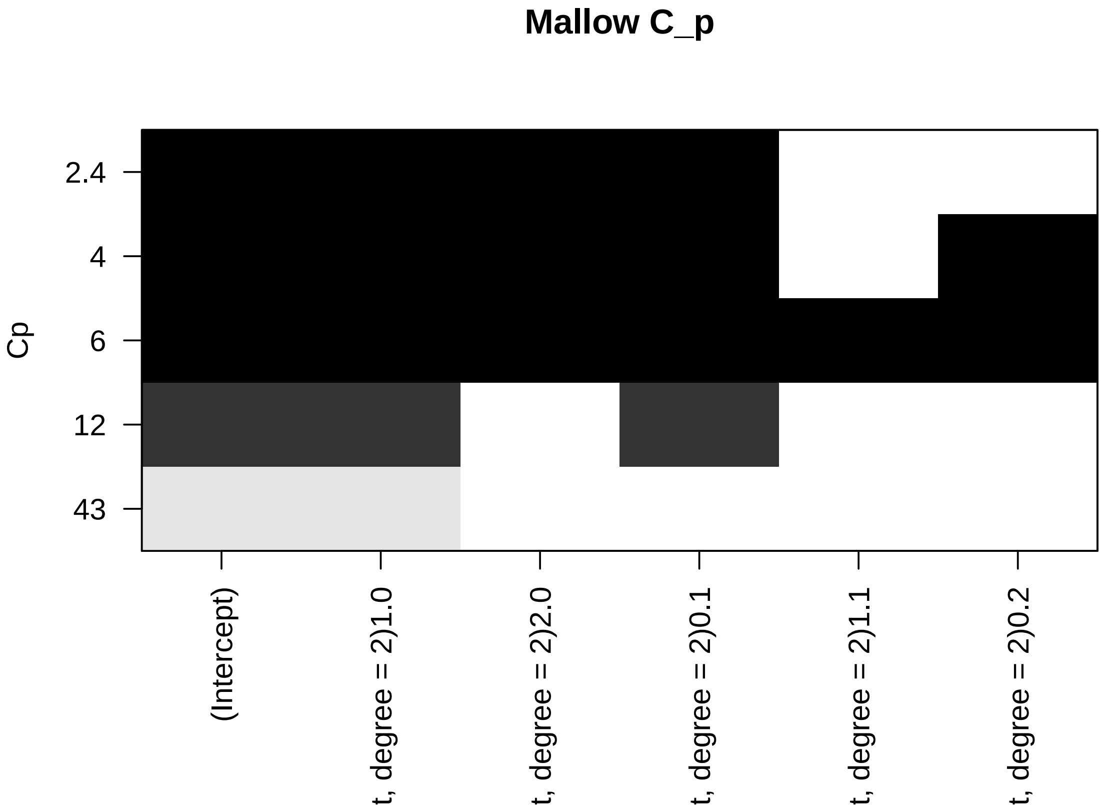
Using the trees data, fit a model with log(Volume) as the response and a second-order polynomial (including the interaction term) in Girth and Height. Determine whether the model may be reasonably simplified.

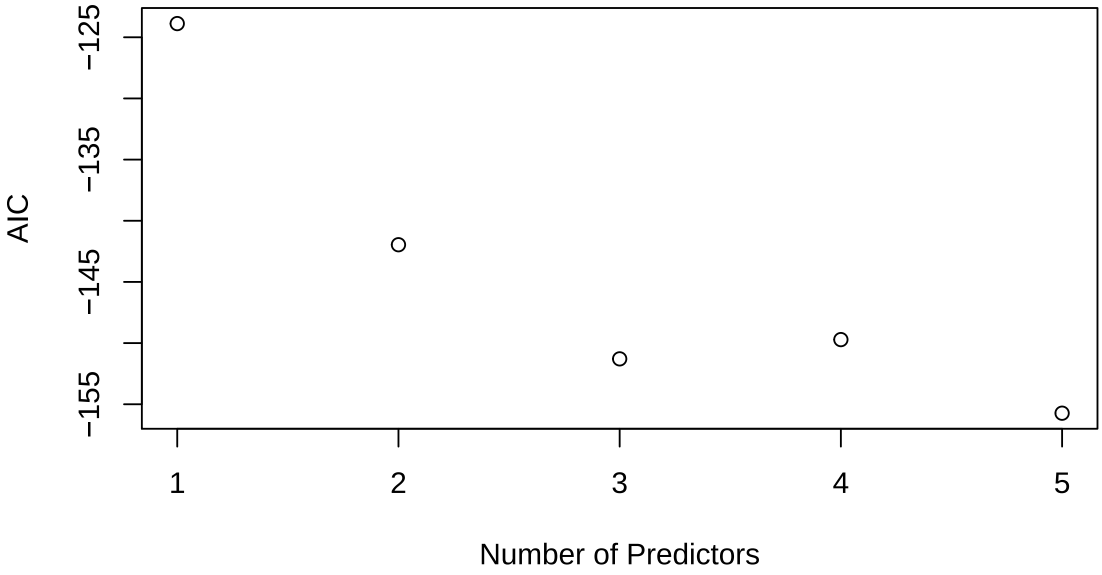
```
##
## Call:
## lm(formula = log(Volume) ~ polym(Girth, Height, degree = 2),
##     data = trees)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.159718 -0.041905 -0.003371  0.055167  0.133780
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         3.27472    0.02370 138.163  < 2e-16
## polym(Girth, Height, degree = 2)1.0  2.51882    0.11972  21.039  < 2e-16
## polym(Girth, Height, degree = 2)2.0 -0.24312    0.18449  -1.318    0.200
## polym(Girth, Height, degree = 2)0.1  0.54249    0.11339   4.784 6.52e-05
## polym(Girth, Height, degree = 2)1.1 -0.11845    1.08511  -0.109    0.914
## polym(Girth, Height, degree = 2)0.2 -0.05025    0.10402  -0.483    0.633
##
## (Intercept)                         ***
## polym(Girth, Height, degree = 2)1.0 ***
## polym(Girth, Height, degree = 2)2.0
## polym(Girth, Height, degree = 2)0.1 ***
## polym(Girth, Height, degree = 2)1.1
## polym(Girth, Height, degree = 2)0.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08469 on 25 degrees of freedom
## Multiple R-squared:  0.9784, Adjusted R-squared:  0.9741
## F-statistic: 226.7 on 5 and 25 DF,  p-value: < 2.2e-16
```

Now we run subset selection. We'll use an exhaustive method since there are not too many predictors. And we'll use the Mallow C_p as our criteria.

```
##                         (Intercept) polym(Girth, Height, degree = 2)1.0
##                          3.27471581                          2.51881720
## polym(Girth, Height, degree = 2)2.0 polym(Girth, Height, degree = 2)0.1
##                         -0.24312237                          0.54248964
## polym(Girth, Height, degree = 2)1.1 polym(Girth, Height, degree = 2)0.2
##                         -0.11844598                         -0.05024754
```

```
##   (Intercept) polym(Girth, Height, degree = 2)1.0
## 1        TRUE                                TRUE
## 2        TRUE                                TRUE
## 3        TRUE                                TRUE
## 4        TRUE                                TRUE
## 5        TRUE                                TRUE
##   polym(Girth, Height, degree = 2)2.0 polym(Girth, Height, degree = 2)0.1
## 1                               FALSE                               FALSE
## 2                               FALSE                                TRUE
## 3                                TRUE                                TRUE
## 4                                TRUE                                TRUE
## 5                                TRUE                                TRUE
##   polym(Girth, Height, degree = 2)1.1 polym(Girth, Height, degree = 2)0.2
## 1                               FALSE                               FALSE
## 2                               FALSE                               FALSE
## 3                               FALSE                               FALSE
## 4                               FALSE                                TRUE
## 5                          TRUE                            TRUE
```


Mallow C_p
Cp
2.4
4
6
12
43
(Intercept)
t, degree = 2)1.0
t, degree = 2)2.0
t, degree = 2)0.1
t, degree = 2)1.1
t, degree = 2)0.2



The AIC criterion indicates the best model is the full model with all the polynomial terms, while the Mallow Cp indicates the best model is a reduced one with the first three terms of the polynomial expansion :

$log(Volume) \sim polym(Girth, Height, degree = 2)1.0 + polym(Girth, Height, degree = 2)2.0 + polym(Girth, Height, degree = 2)0.1$

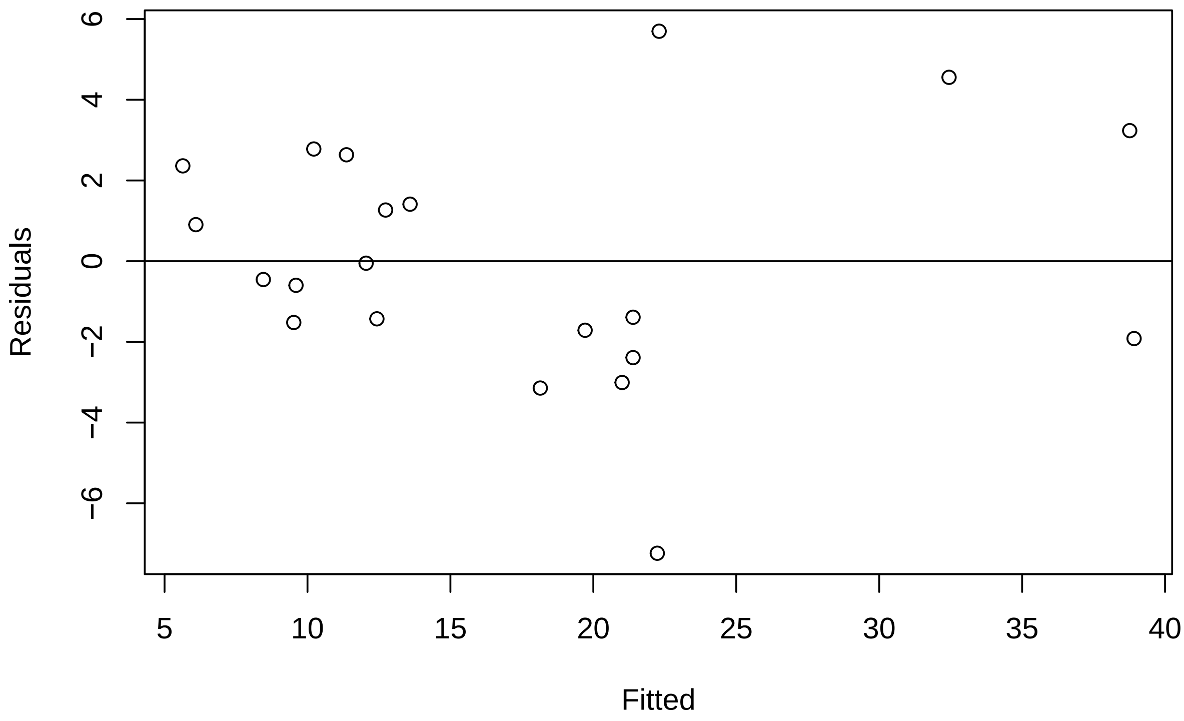
$log(Volume) \sim Girth + Girth^2, +Height$

10.5 Model reduction in stackloss data

Fit a linear model to the stackloss data with stack.loss as the predictor and the other variables as predictors.

```
##
## Call:
## lm(formula = stack.loss ~ ., data = stackloss)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2377 -1.7117 -0.4551  2.3614  5.6978
```

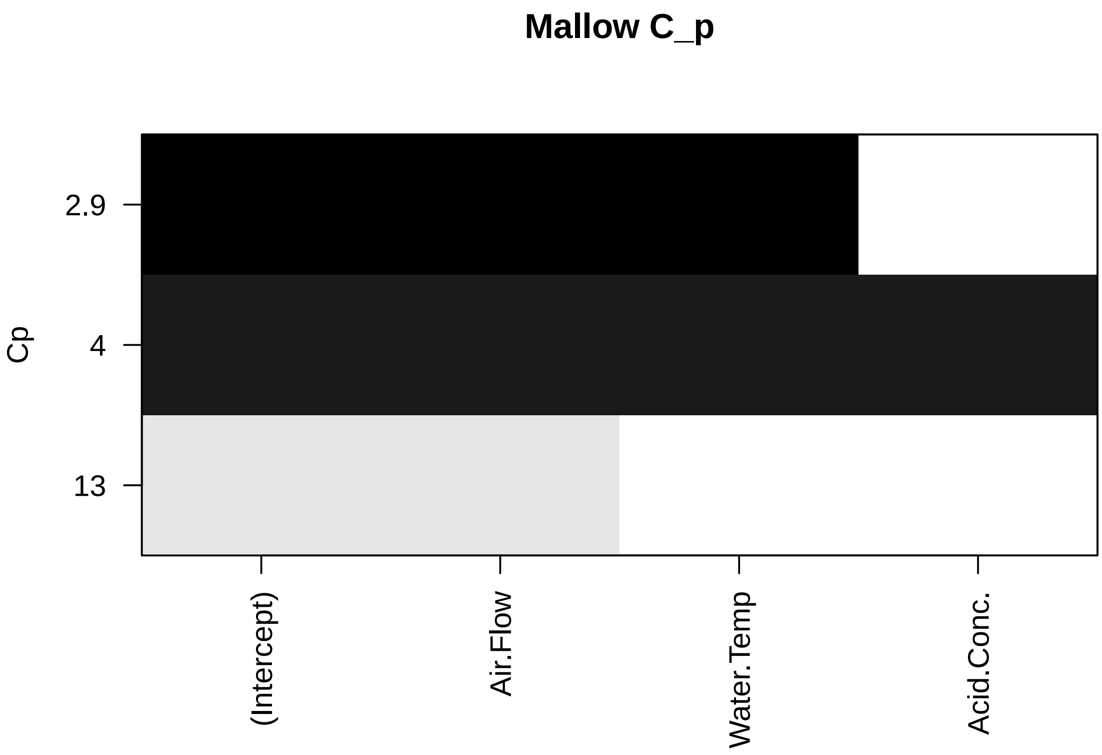
```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -39.9197    11.8960  -3.356  0.00375 **
## Air.Flow      0.7156     0.1349   5.307  5.8e-05 ***
## Water.Temp    1.2953     0.3680   3.520  0.00263 **
## Acid.Conc.   -0.1521     0.1563  -0.973  0.34405
## ---
## 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
```

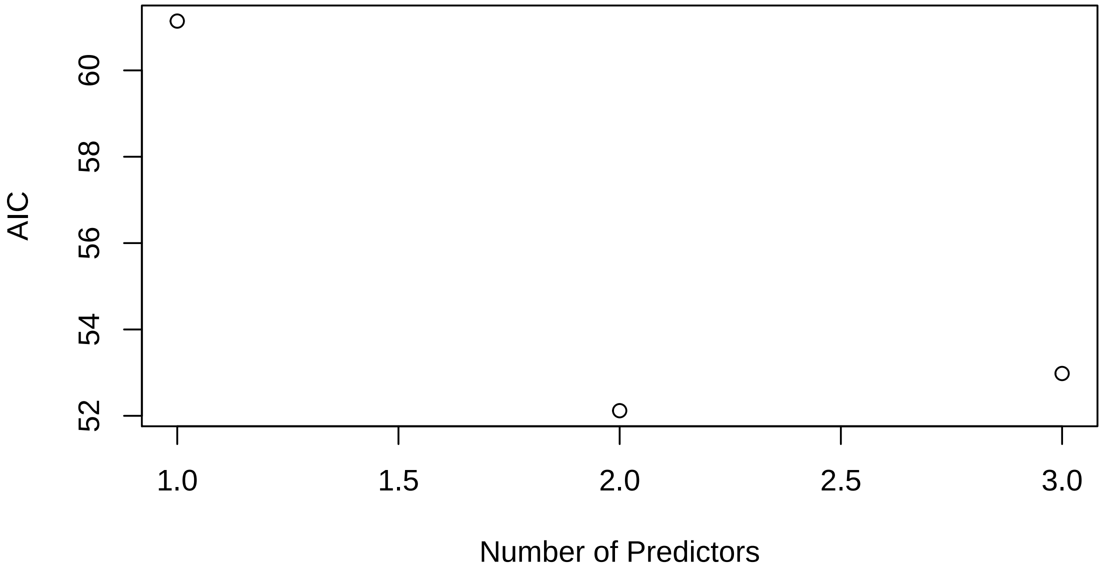


Simplify the model if possible.

Now we run subset selection. We'll use an exhaustive method since there are not too many predictors. And we'll use the Mallow C_p as our criteria.

```
##   (Intercept) Air.Flow Water.Temp Acid.Conc.
## 1        TRUE     TRUE      FALSE      FALSE
## 2          TRUE     TRUE        TRUE       FALSE
## 3          TRUE     TRUE        TRUE        TRUE
```





The reduced model with the lowest AIC has 2 variables and is $stack.loss \sim Air.Flow + Water.Temp$. We note this is the same model indicated by the Mallow Cp criterion.

Check the model for outliers and influential points.

Check for outliers.

Table 1: Range of Studentized residuals

range.residuals.left	range.residuals.right
-3.471	2.027

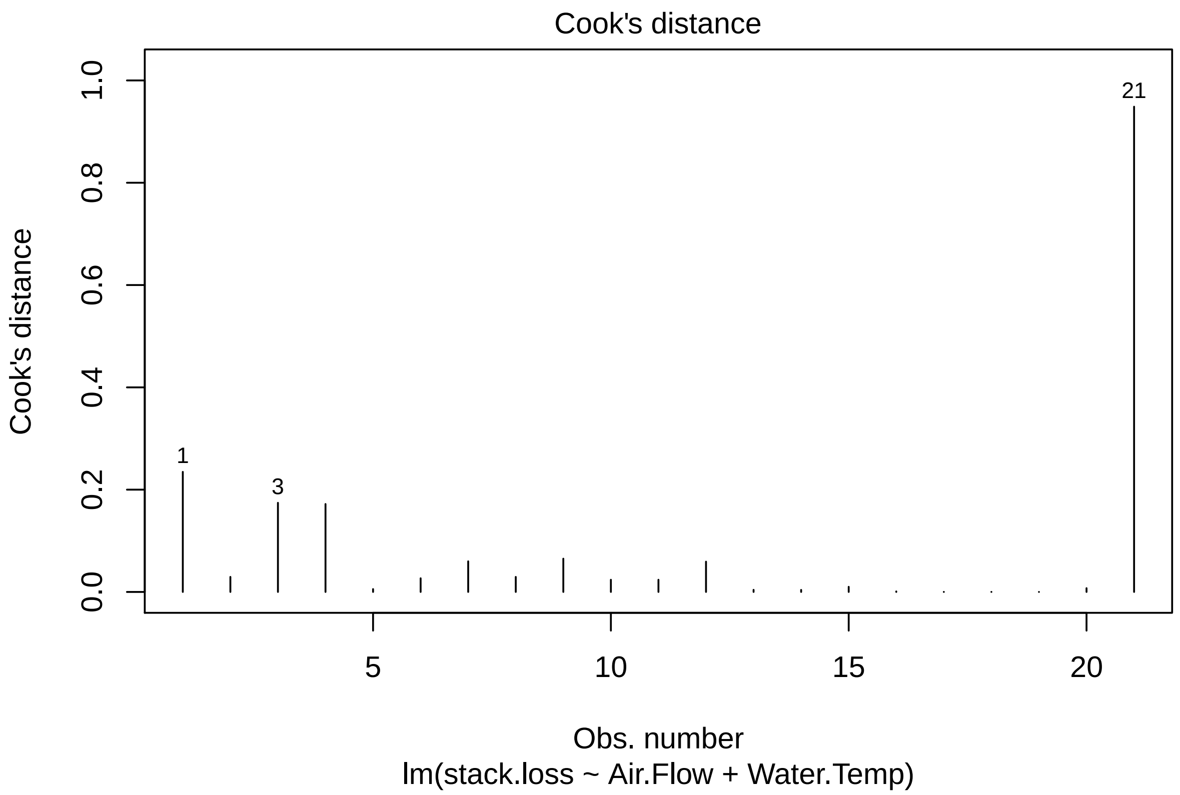
Table 2: Bonferroni corrected t-value

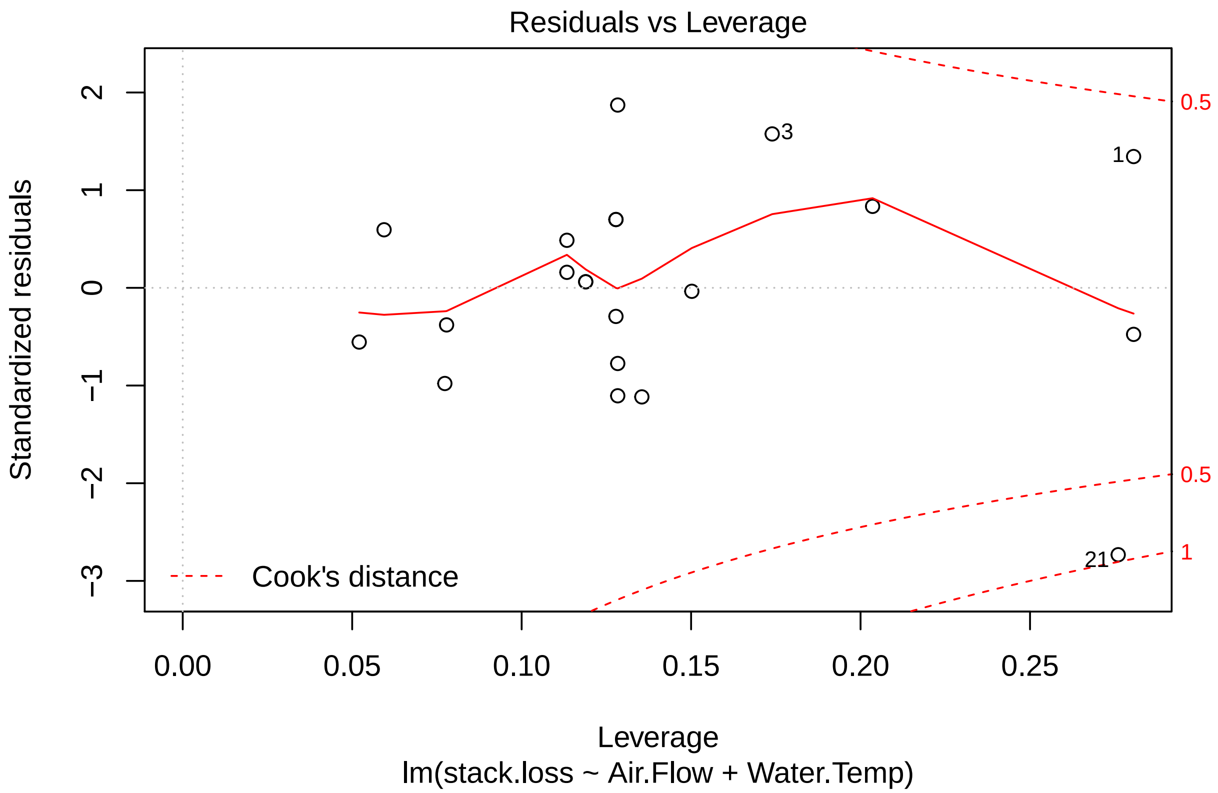
t.val.alpha
-3.565

Here we look for studentized residuals that fall outside the interval given by the Bonferroni corrected t-values. In the case of the reduced model we do not see any outliers.

Check for influential points.

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





We see that data element 21 is an influential point for the reduced model under the criteria $D_i > \frac{1}{2}$. Elements 1 and 3 are also influential under the criteria $D_i > \frac{4}{n}$

Now return to the full model, determine whether there are any outliers or influential points

Check for outliers.

Table 3: Range of Studentized residuals

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

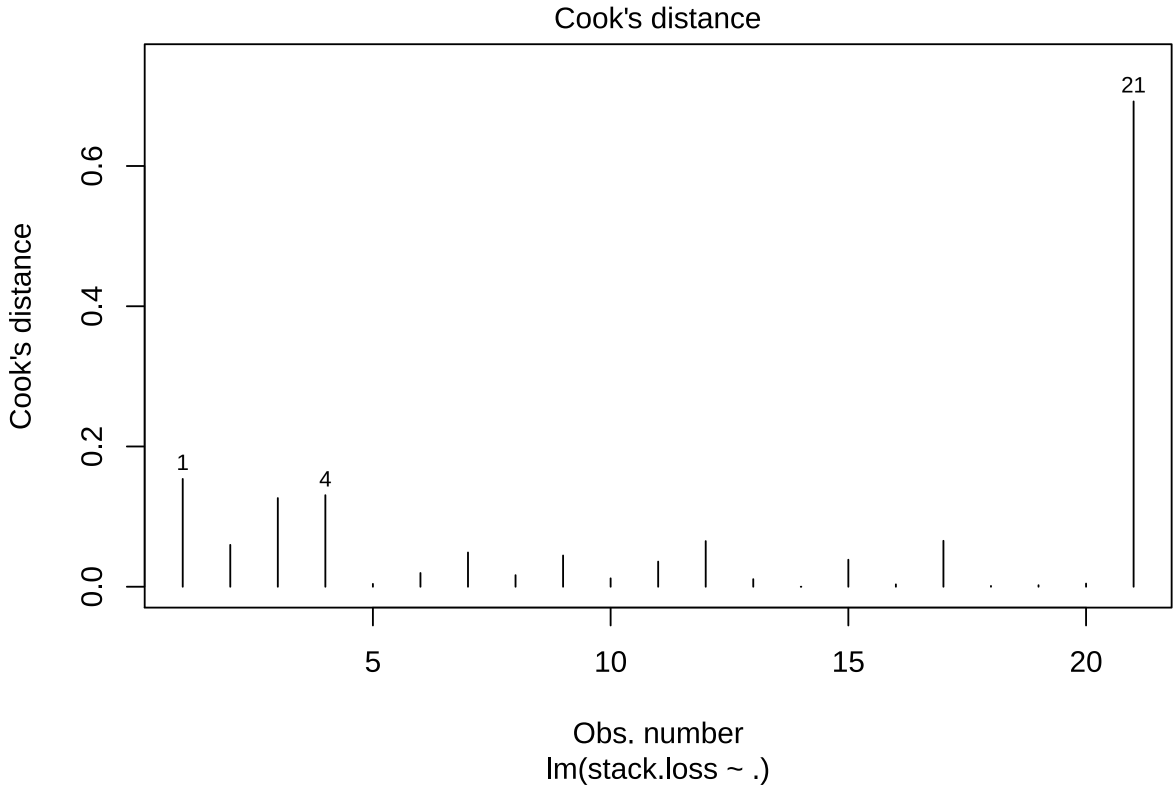
Table 4: Bonferroni corrected t-value

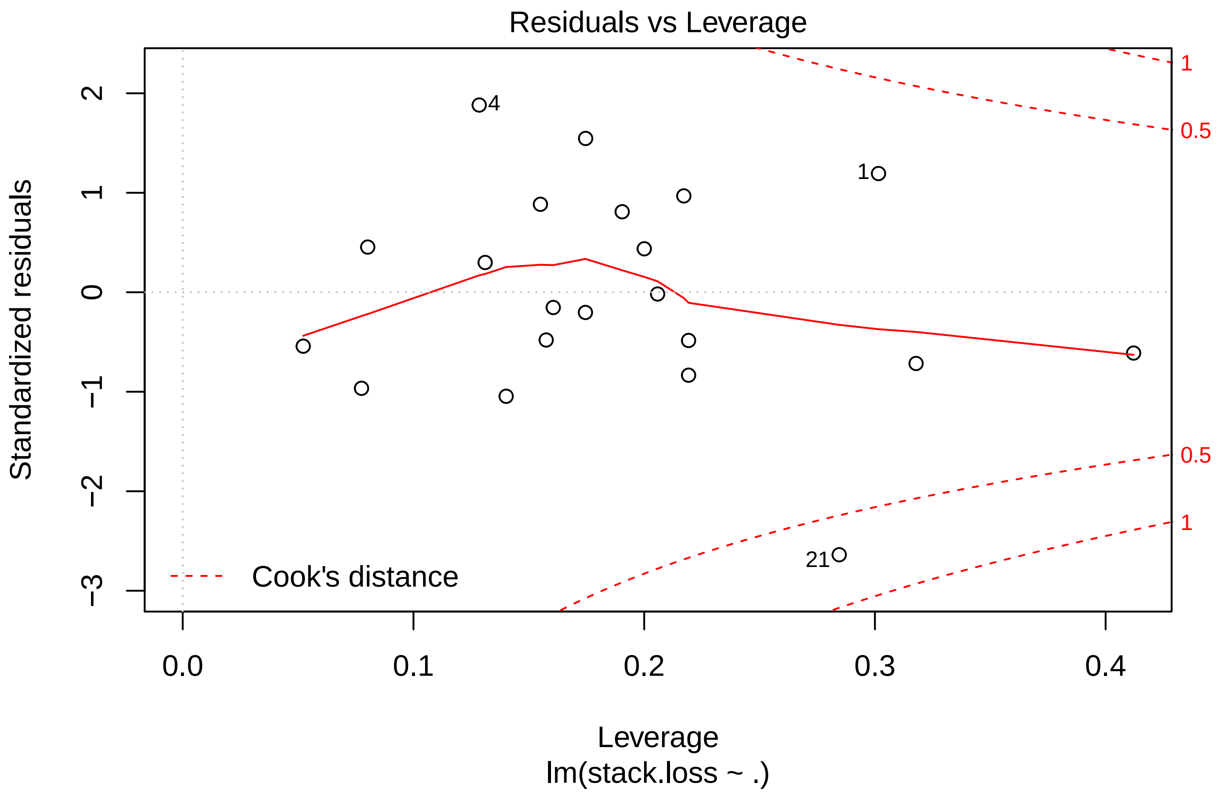
t.val.alpha
-3.604

Here we look for studentized residuals that fall outside the interval given by the Bonferroni corrected t-values. we see there are no outliers for the full model

Check for influential points.

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

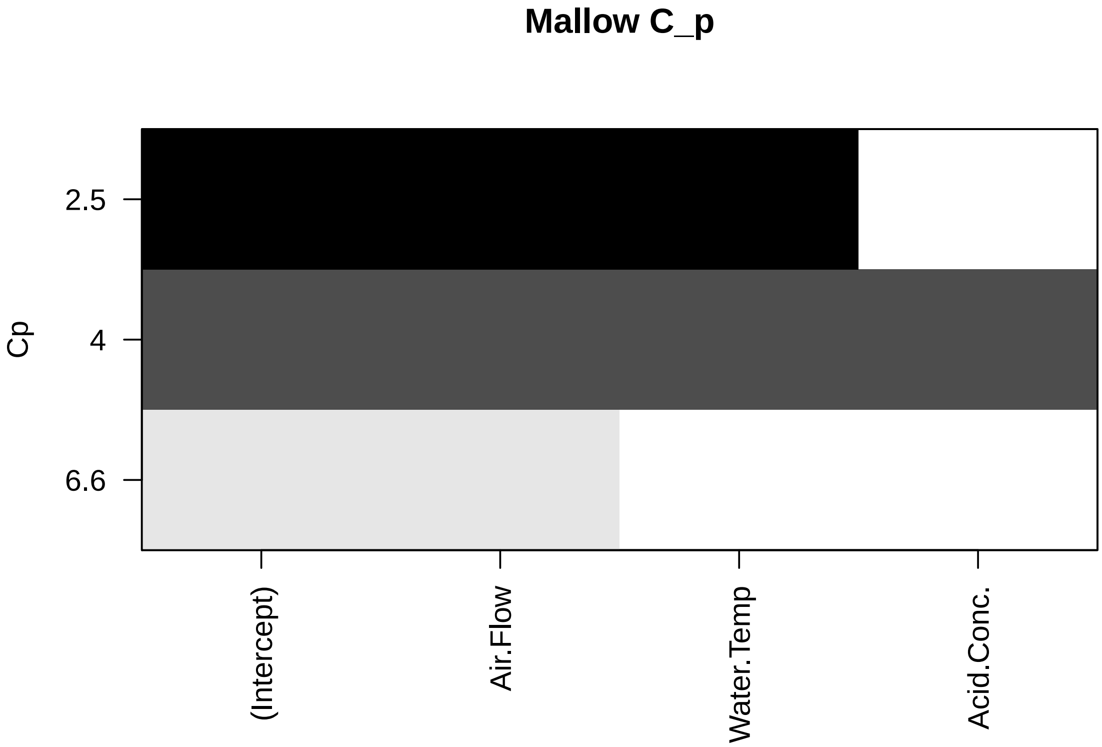


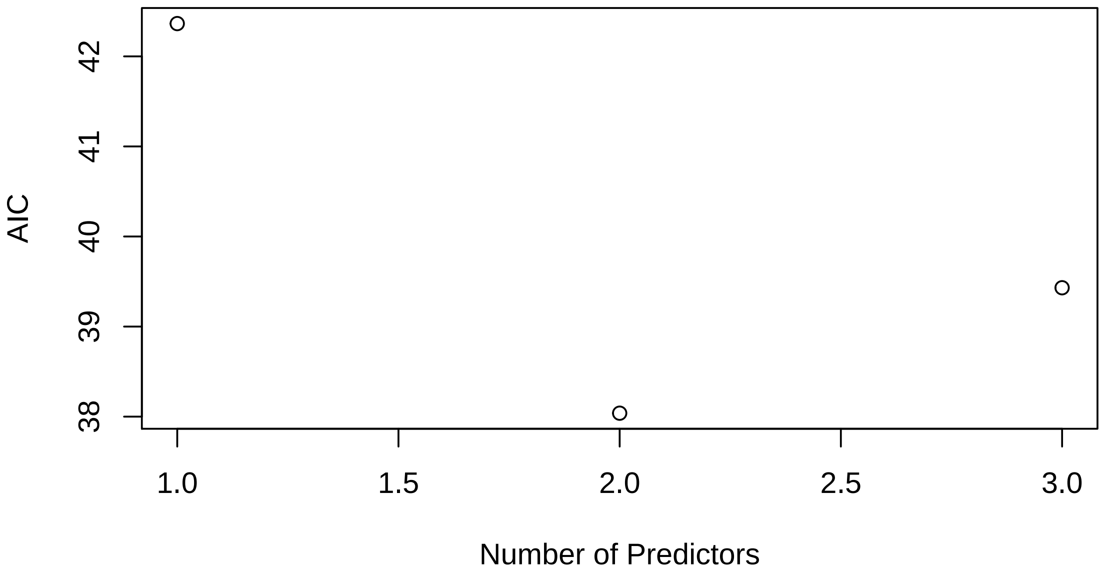


We see element 21 is an influential point, and that 1 and 4 are also influential under the criteria $D_i > \frac{4}{n}$. Since element 1 is an influential in both the full and reduced model we remove that along with element 21.

Eliminate the outliers and influential points for the full model and then repeat the variable selection procedures.

```
##   (Intercept) Air.Flow Water.Temp Acid.Conc.
## 1        TRUE     TRUE      FALSE      FALSE
## 2        TRUE     TRUE       TRUE      FALSE
## 3        TRUE     TRUE       TRUE       TRUE
```


Mallow C_p
Cp
2.5
4
6.6
(Intercept)
Air.Flow
Water.Temp
Acid.Conc.



We see that the subset selection routine has chosen the same model $stack.loss \sim Air.Flow + Water.Temp$.

NCSU ST 503 HW 11

Probem 2.1,2.2,4.1 Faraway, Julian J. Extending the Linear Model with R CRC Press.

Bruce Campbell

14 November, 2017

Complete exercises # 1 (d, e, f, and h), and # 2 (a - c, h) from Chapter 2. Complete exercise # 1 (a - f) from Chapter 4.

For 2.1, you have already done parts (e, f, and h) using the full 9 variable model in the group discussion. Repeat this time for the model that is chosen in part (d). Compare your results using this model to the results from the full model.

For 4.1, the question does not match the data completely.

(1) The problem says to ignore the variable “volact”. Apparently, it was already ignored well enough since it does not appear in the dataset, so that is fine.

(2) For part (b) in 4.1, it says to fit the model using the other “5 variables” as predictors. There are 6 predictors to use, 5 numeric and one binary class variable, not 5, so use the 6.

2.1 wbca analysis

The dataset wbca comes from a study of breast cancer in Wisconsin. There are 681 cases of potentially cancerous tumors of which 238 are actually malignant. Determining whether a tumor is really malignant is traditionally determined by an invasive surgical procedure. The purpose of this study was to determine whether a new procedure called fine needle aspiration, which draws only a small sample of tissue, could be effective in determining tumor status.

We split the data into a training and test set for model evaluation. One third of the data is reserved for the test set.

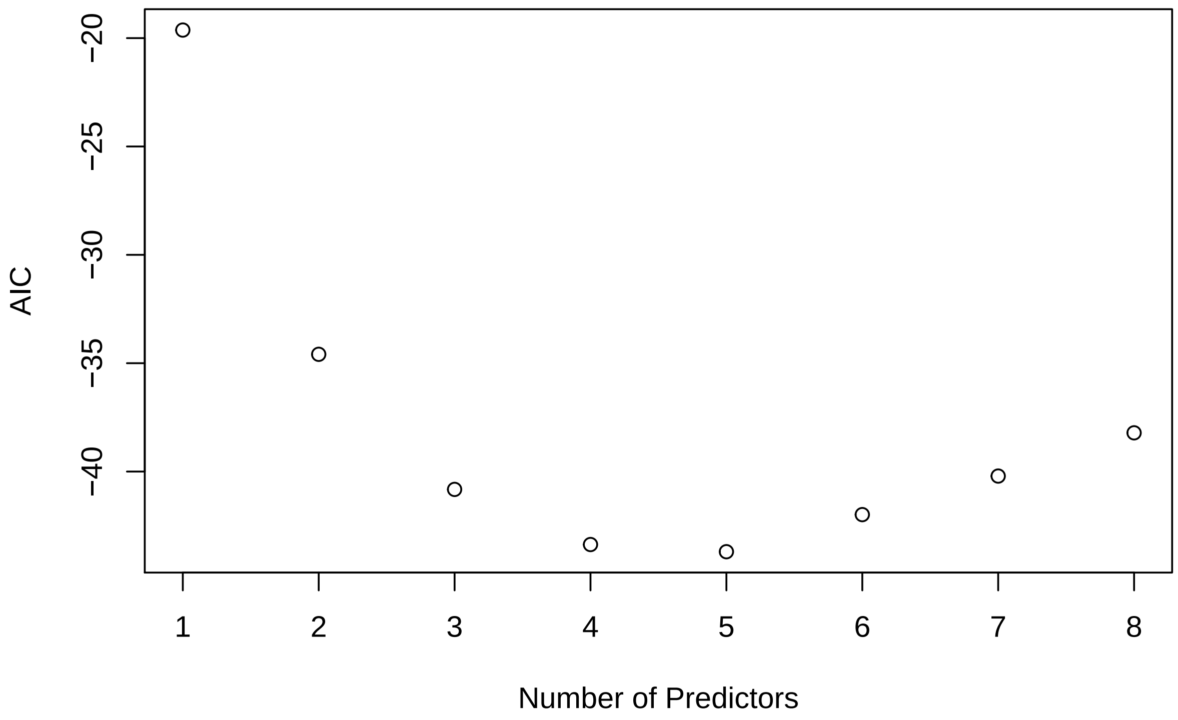
Fit a binary regression with Class as the response and the other nine variables as predictors.

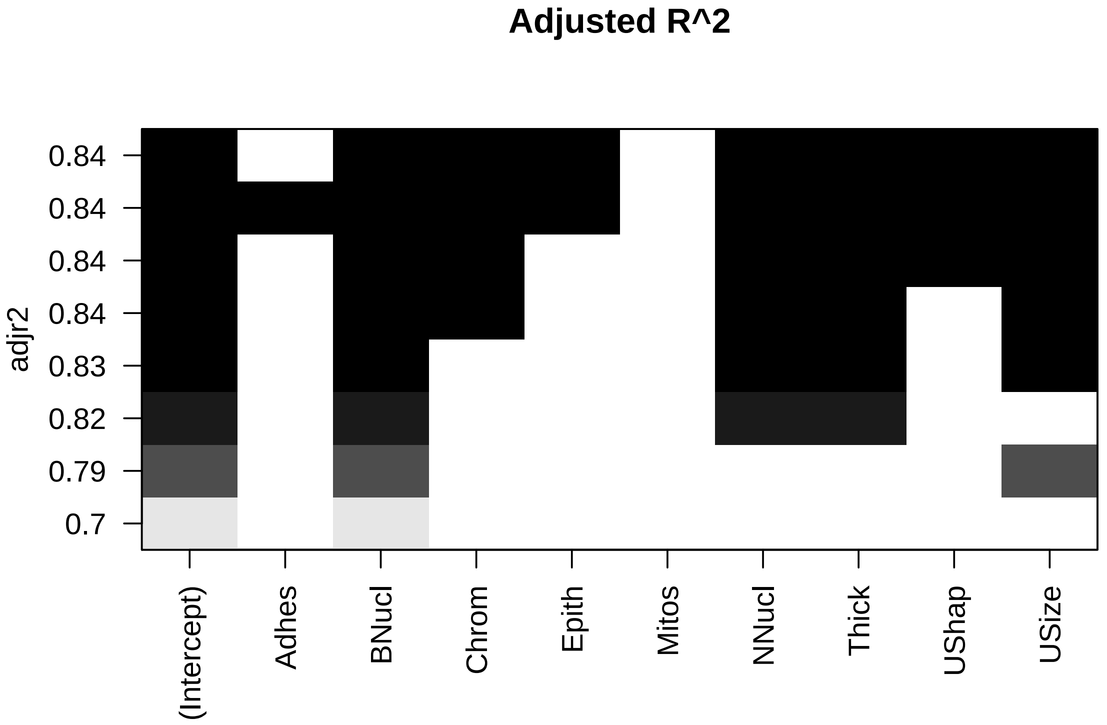
```
##
```

```
## Call:
## glm(formula = Class ~ ., family = binomial, data = DFTrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3520  -0.0123   0.0406   0.0969   3.1771
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 11.65211    1.84168   6.327  2.5e-10 ***
## Adhes       -0.39057    0.19726  -1.980 0.047708 *
## BNucl       -0.46793    0.13566  -3.449 0.000562 ***
## Chrom       -0.65578    0.23207  -2.826 0.004716 **
## Epith       -0.02961    0.24675  -0.120 0.904469
## Mitos       -0.57934    0.51120  -1.133 0.257094
## NNucl       -0.25630    0.15143  -1.693 0.090543 .
## Thick       -0.80067    0.21174  -3.781 0.000156 ***
## UShap       -0.23802    0.26885  -0.885 0.375988
## USize        0.17773    0.24495   0.726 0.468109
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 593.945  on 453  degrees of freedom
## Residual deviance:  62.044  on 444  degrees of freedom
## AIC: 82.044
##
## Number of Fisher Scoring iterations: 9
```

(d) Use AIC as the criterion to determine the best subset of variables. (Use the step function.)

```
##   (Intercept) Adhes BNucl Chrom Epith Mitos NNucl Thick UShap USize
## 1        TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 2        TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
## 3        TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE
## 4        TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE
## 5        TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE
## 6        TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
## 7        TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
## 8        TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
```


AIC
−20
−25
−30
−35
−40
1
2
3
4
5
6
7
8
Number of Predictors



We see that the AIC is minimized at 4 predictors. The best 4 predictor model is $Class \sim BNucl + NNucl + Thick + USize$ We now fit that reduced model on the training data.

```
##
## Call:
## glm(formula = Class ~ BNucl + NNucl + Thick + USize, family = binomial,
##     data = DFTrain)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.12132  -0.04102   0.05276   0.15489   3.01438
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   8.6336     1.1110   7.771 7.80e-15 ***
## BNucl        -0.5696     0.1164  -4.892 9.97e-07 ***
## NNucl        -0.3741     0.1213  -3.084  0.00204 **
## Thick        -0.7198     0.1638  -4.394 1.11e-05 ***
## USize        -0.3936     0.1620  -2.429  0.01512 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 593.945  on 453  degrees of freedom
## Residual deviance:  82.118  on 449  degrees of freedom
## AIC: 92.118
##
## Number of Fisher Scoring iterations: 8
```

(e) Suppose that a cancer is classified as benign if p > 0.5 and malignant if p < 0.5. Compute the number of errors of both types that will be made if this method is applied to the current data with the reduced model.

Full model confusion matrix on training data.

```
##             class.predicted
## factor.class FALSE TRUE
##            0   157    7
##            1     7  283
```

$p = 0.5$ Full model error = `0.030837`

Reduced model confusion matrix on training data.

```
##             class.predicted
## factor.class FALSE TRUE
##            0   155    9
##            1     6  284
```

$p = 0.5$ Reduced model error = `0.0330396`

(f) Suppose we change the cutoff to 0.9 so that p < 0.9 is classified as malignant and p > 0.9 as benign. Compute the number of errors in this case.

Full model confusion matrix on training data.

```
##             class.predicted
## factor.class FALSE TRUE
##            0   163    1
##            1    12  278
```

$p = 0.9$ Full model error = `0.0286344`

Reduced model confusion matrix on training data.

```
##             class.predicted
## factor.class FALSE TRUE
##            0   164    0
##                 1      14   276
```

$p = 0.9$ Reduced model error = `0.030837`

(h) It is usually misleading to use the same data to fit a model and test its predictive ability. To investigate this, split the data into two parts - assign every third observation to a test set and the remaining two thirds of the data to a training set. Use the training set to determine the model and the test set to assess its predictive performance. Compare the outcome to the previously obtained results.

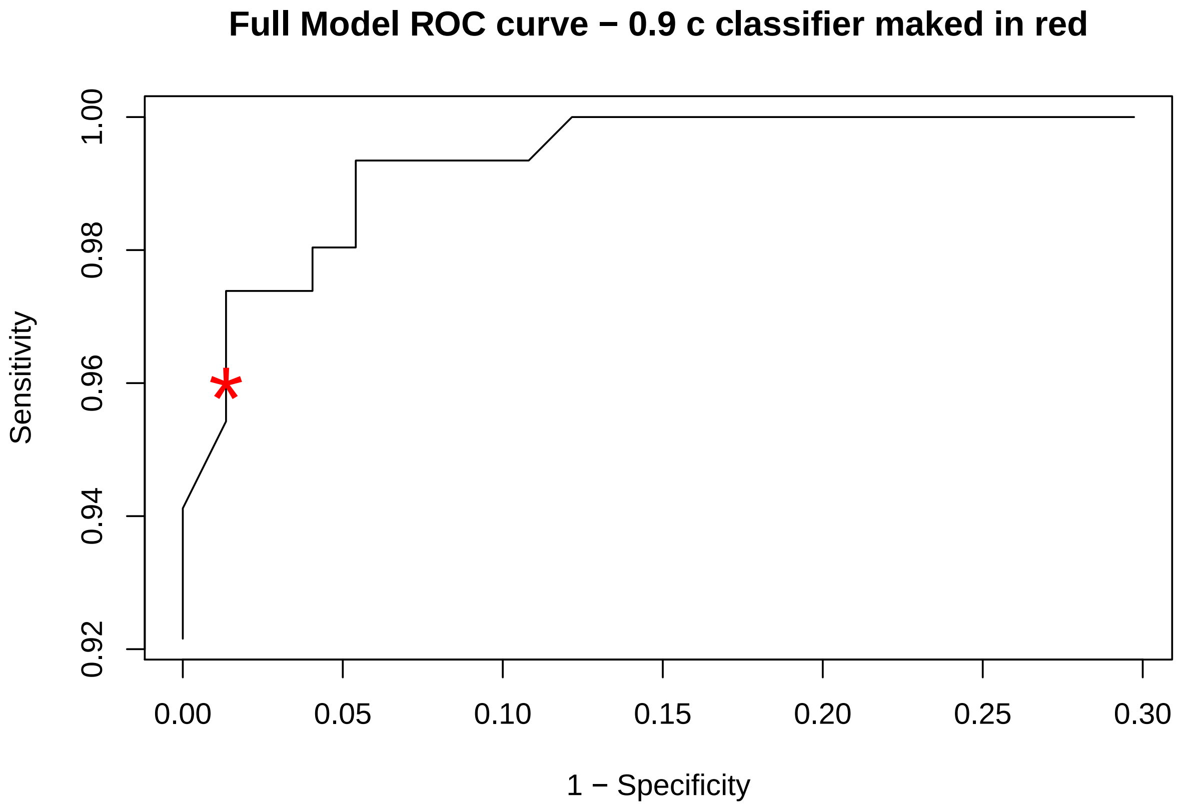
Full model test set evaluation

Table 1: Confusion matrix p=0.9

	FALSE	TRUE
0	73	1
1	6	147

Table 2: Confusion matrix p=0.5

	FALSE	TRUE
0	70	4
1	3	150



For the full model we have an accuracy of `0.030837` for the $p = 0.9$ cutoff and ‘`0.030837` for the $p = 0.5$ cutoff. Note that even though the accuracy is the same rate for both of these, the sensitivity and specificity will be different and there may be test implementation considerations that determine which one is preferable.

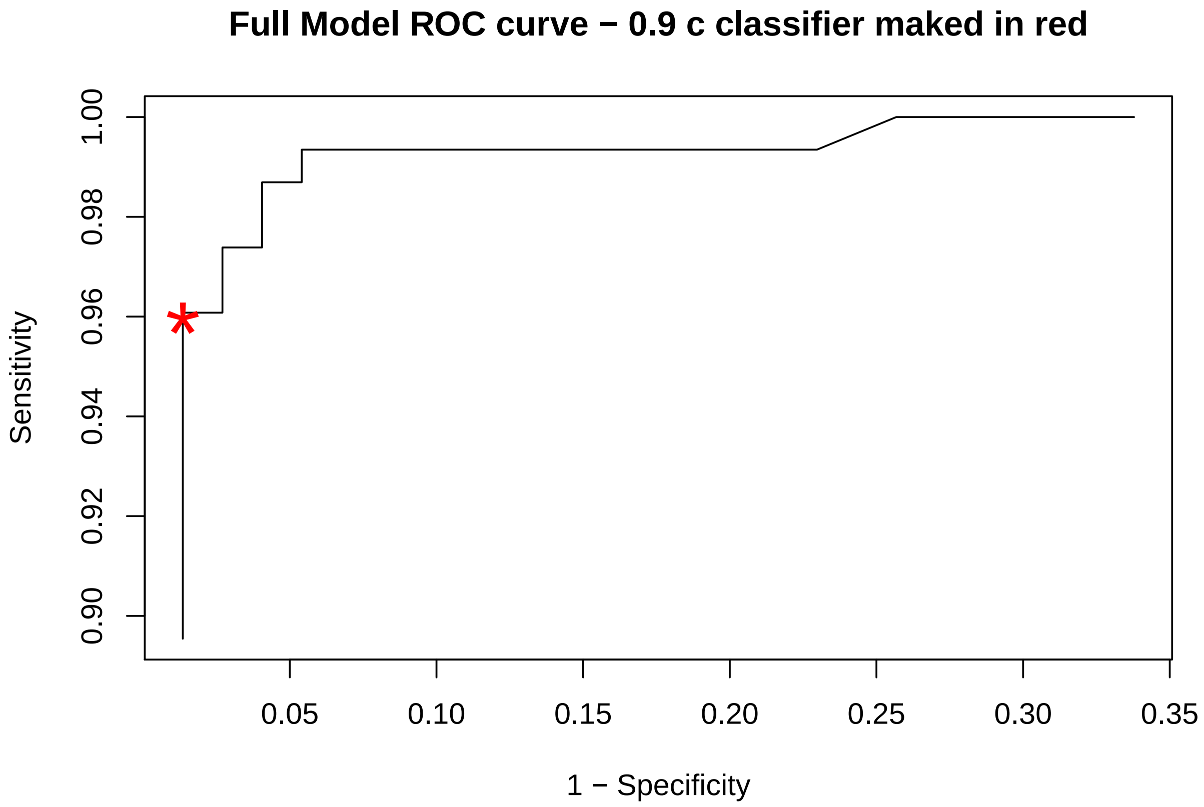
Reduced model test set evaluation

Table 3: Confusion matrix p=0.9

	FALSE	TRUE
0	73	1
1	6	147

Table 4: Confusion matrix p=0.5

	FALSE	TRUE
0	70	4
1	2	151



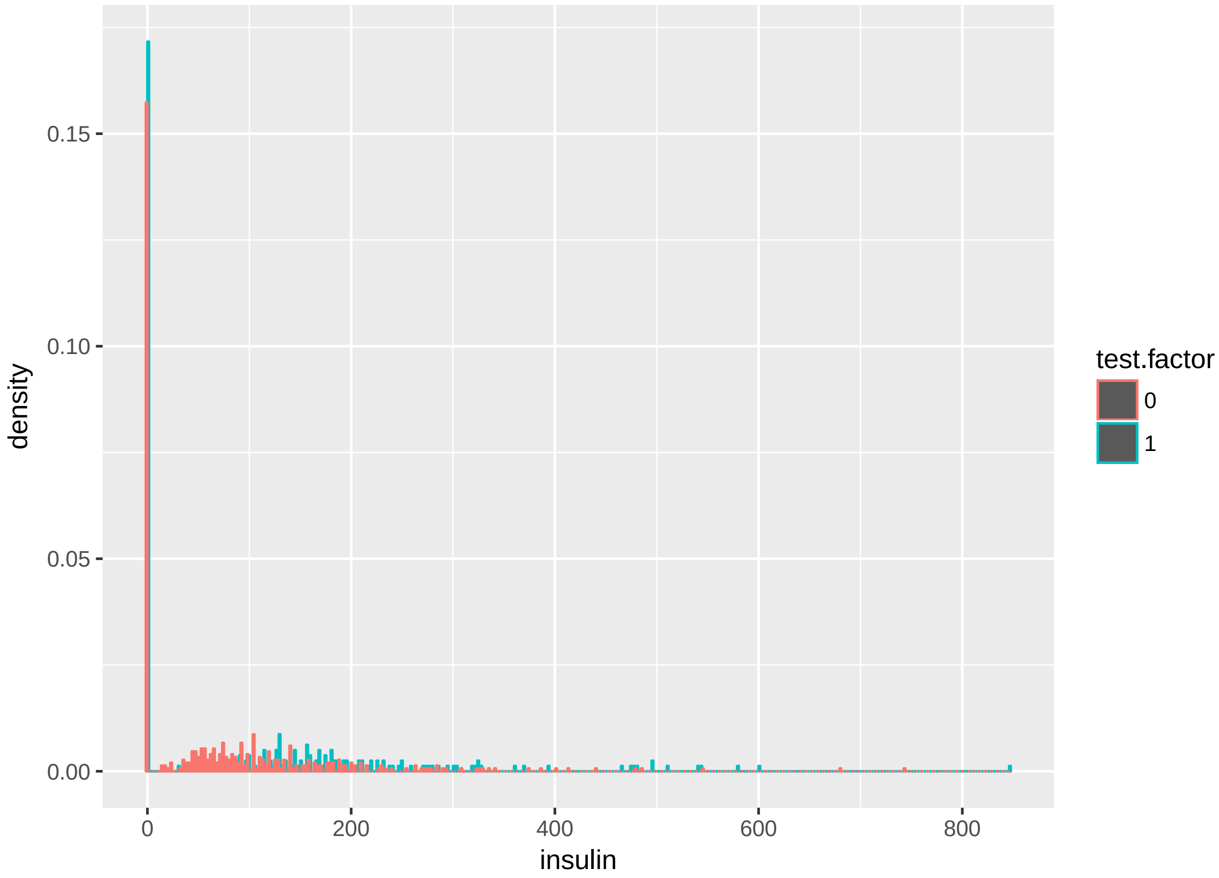
For the reduced model we have an accuracy of `0.030837` for the $p = 0.9$ cutoff and '`0.0220264` for the $p = 0.5$ cutoff.

The reduced model performs slightly better at the $p = 0.5$ cutoff. We would choose this model - all other things being equal- due to it's parsimony in the number of predictors used.

2.2 pima data analysis

The National Institute of Diabetes and Digestive and Kidney Diseases conducted a study on 768 adult female Pima Indians living near Phoenix. The purpose of the study was to investigate factors related to diabetes. The data may be found in the the dataset pima.

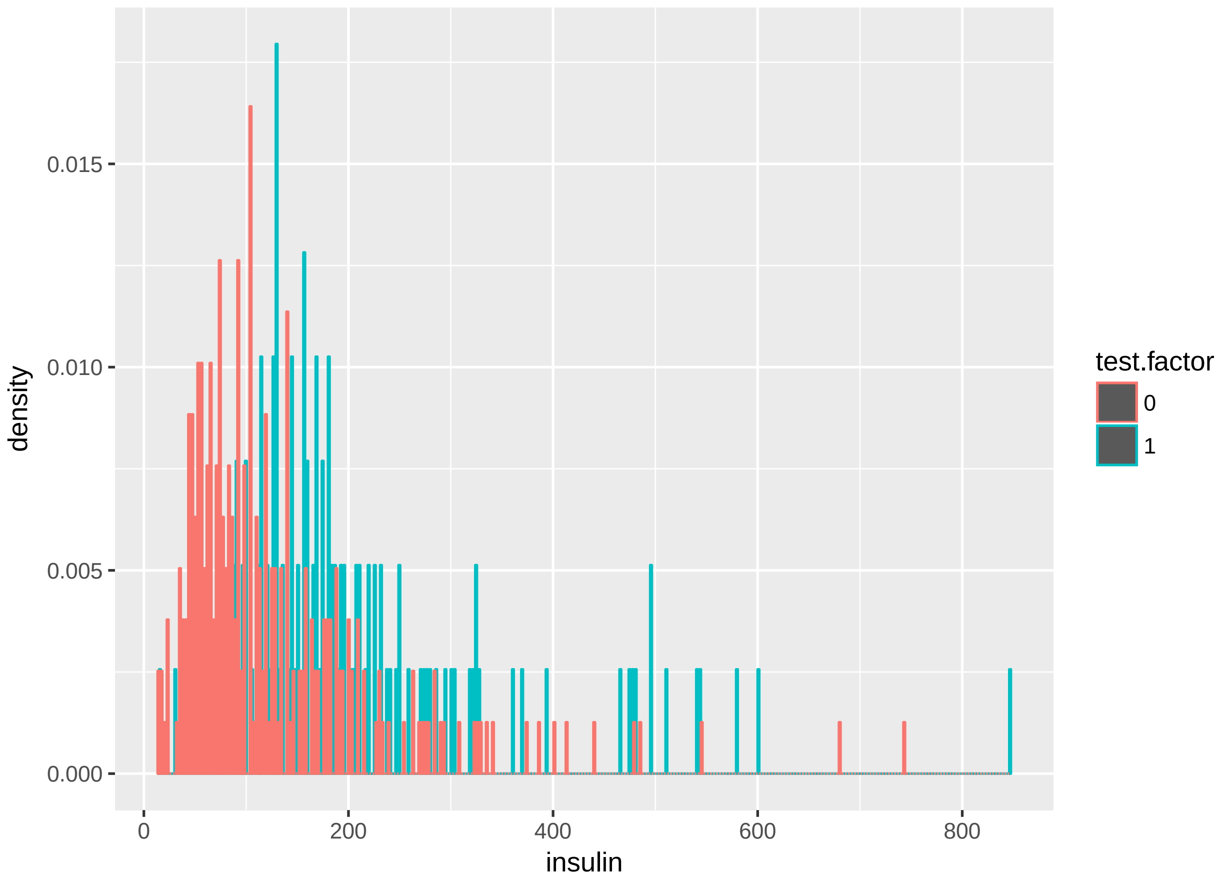
(a) Create a factor version of the test results and use this to produce an interleaved histogram to show how the distribution of insulin differs between those testing positive and negative. Do you notice anything unbelievable about the plot?



We note a number of measurements of 0 insulin. This is likely a placeholder for when no measurement was available.

(b) Replace the zero values of insulin with the missing value code NA. Recreate the interleaved histogram plot and comment on the distribution.

```
df[(df$insulin==0),]$insulin =NA
ggplot(df, aes(x=insulin, color=test.factor)) + geom_histogram(position="dodge", binwidt
```



(c) Replace the incredible zeroes in other variables with the missing value code. Fit a model with the result of the diabetes test as the response and all the other variables as predictors. How many observations were used in the model fitting? Why is this less than the number of observations in the data frame.

```
##
## Call:
## glm(formula = test ~ glucose + diastolic + triceps + insulin +
##     bmi + diabetes + age, family = binomial, data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7814  -0.6675  -0.3699   0.6474   2.5697
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.016e+01  1.209e+00  -8.409  < 2e-16 ***
## glucose      3.819e-02  5.783e-03   6.605 3.97e-11 ***
## diastolic   -1.085e-03  1.174e-02  -0.092 0.926379
## triceps      1.169e-02  1.715e-02   0.681 0.495593
## insulin     -9.424e-04  1.327e-03  -0.710 0.477683
## bmi          6.660e-02  2.712e-02   2.456 0.014046 *
## diabetes     1.079e+00  4.228e-01   2.551 0.010729 *
## age          5.203e-02  1.425e-02   3.652 0.000261 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 346.24  on 384  degrees of freedom
##   (376 observations deleted due to missingness)
## AIC: 362.24
##
## Number of Fisher Scoring iterations: 5
```

392 data elements were used to fit the model.

(d) Refit the model but now without the insulin and triceps predictors. How many observations were used in fitting this model? Devise a test to compare this model with that in the previous question.

```
##
## Call:
## glm(formula = test ~ glucose + diastolic + bmi + diabetes + age,
##     family = binomial, data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7561  -0.7236  -0.4105   0.7246   2.3652
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.872764   0.743686 -11.931  < 2e-16 ***
## glucose      0.035831   0.003551  10.092  < 2e-16 ***
## diastolic   -0.012574   0.005252  -2.394  0.01667 *
## bmi          0.093449   0.014911   6.267 3.68e-10 ***
## diabetes     0.864359   0.300074   2.880  0.00397 **
## age          0.033123   0.008124   4.077 4.55e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 974.75  on 751  degrees of freedom
## Residual deviance: 710.46  on 746  degrees of freedom
##   (16 observations deleted due to missingness)
## AIC: 722.46
##
## Number of Fisher Scoring iterations: 5
```

Only 16 observations were removed due to missing data.

We will use $D_S - D_L \sim \chi^2_{df(L)-df(S)}$

Our test statistic is `-364` with $df(L) = 384$ $df(s) = 751$

and the p-value `0.534429` is not significant so insulin and triceps are not significant in models that already have the predictors used in the smaller model.

(e) Use AIC to select a model. You will need to take account of the missing values. Which predictors are selected? How many cases are used in your selected model?

```
##                Estimate  Std. Error z value  Pr(>|z|)
## (Intercept) -10.0920180   1.0802511 -9.3423 < 2.2e-16
## glucose       0.0361890   0.0049819  7.2640 3.757e-13
## bmi           0.0744485   0.0202667  3.6734 0.0002393
## diabetes      1.0871286   0.4194084  2.5921 0.0095405
## age           0.0530121   0.0134395  3.9445 7.997e-05
##
## n = 392 p = 5
## Deviance = 347.23499 Null Deviance = 498.09781 (Difference = 150.86281)
```

The model with the minimum AIC has four predictors and the selection method found that $test \sim glucose + bmi + diabetes + age$ is the best three predictor model.

(f) Create a variable that indicates whether the case contains a missing value. Use this variable as a predictor of the test result. Is missingness associated with the test result? Refit the selected model, but now using as much of the data as reasonable. Explain why it is appropriate to do this.

```
##
## Call:
## glm(formula = which.na ~ test, family = binomial, data = df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.203  -1.137  -1.137   1.218   1.218
##
```

```
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.09607    0.08955  -1.073    0.283
## test         0.15579    0.15152   1.028    0.304
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1064.3  on 767  degrees of freedom
## Residual deviance: 1063.3  on 766  degrees of freedom
## AIC: 1067.3
##
## Number of Fisher Scoring iterations: 3
```

We see that test is significant at a level of $\alpha = 0.3$ if this were closer to 0.2 we'd begin to wonder if there was something to investigate. This test is reasonable to execute because there may be latent variables related to the test outcome that are represented in the missing value distribution. One could imagine that insulin is hard to measure in some cases that may be related to the disease status. Likewise with the other predictors.

(g) Using the last fitted model of the previous question, what is the difference in the odds of testing positive for diabetes for a woman with a BMI at the first quartile compared with a woman at the third quartile, assuming that all other factors are held constant? Give a confidence interval for this difference.

```
##
## Call:
## glm(formula = test ~ glucose + bmi + diabetes + age, family = binomial,
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8228  -0.6617  -0.3759   0.6702   2.5881
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.092018   1.080251  -9.342  < 2e-16 ***
## glucose       0.036189   0.004982   7.264 3.76e-13 ***
## bmi           0.074449   0.020267   3.673 0.000239 ***
## diabetes      1.087129   0.419408   2.592 0.009541 **
## age           0.053012   0.013439   3.945 8.00e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 347.23  on 387  degrees of freedom
## AIC: 357.23
##
## Number of Fisher Scoring iterations: 5
```

We know that a unit change in *bmi* yields a change in the odds ration of $exp(\beta_{bmi})$ - if all other predictors are held constant. Now we calculate the quartiles.

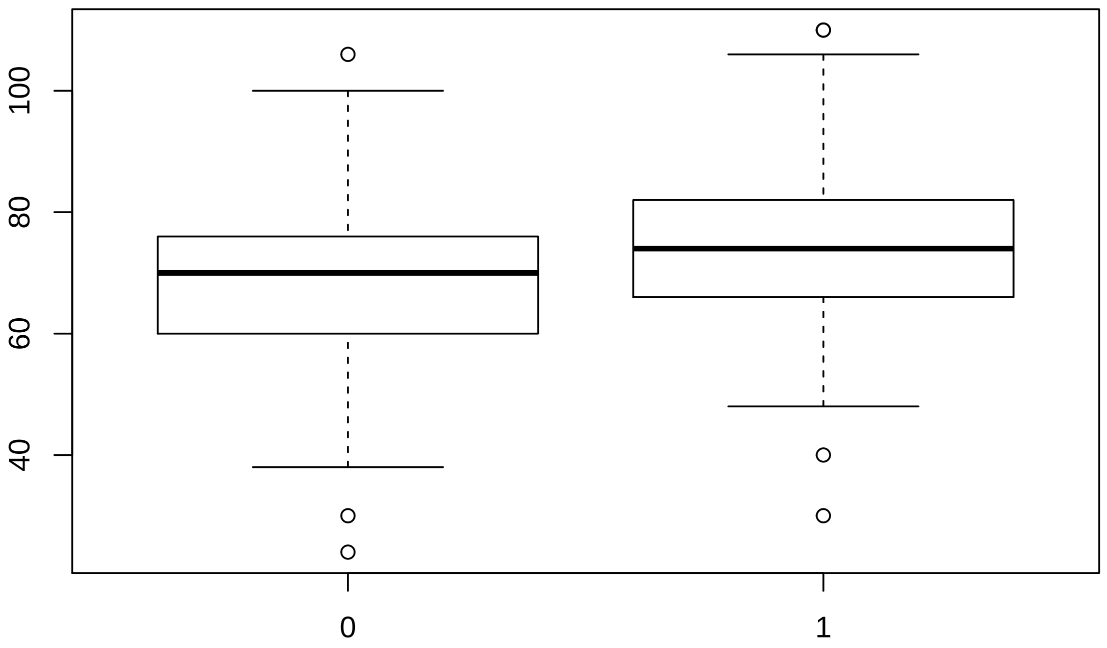
```r
q1 <- quantile(df$bmi,.25)

q3 <- quantile(df$bmi,.75)

dx <- q3-q1

odds.ration.change <- exp(lm.logistic.bss$coefficients["bmi"])*dx

pander(data.frame(odds.ration.change))
```

	odds.ration.change
bmi	9.372

We see that if all other predictors are held constant - and the BMI changes from the first quratile to the third quartile, the odds ratio changes by a factor of 9.372. Making the probability of a positive test 9.372 times more likely.

(h) Do women who test positive have higher diastolic blood pressures? Is the diastolic blood pressure significant in the regression model? Explain the distinction between the two questions and discuss why the answers are only apparently contradictory.

In the full model diastolic is not significant. This may be due to colliniearity. Let's plot the feature first.



We see evidence in the boxplot that elevated diastolic is associated with a positive test result. Let's create a univariate model to see.

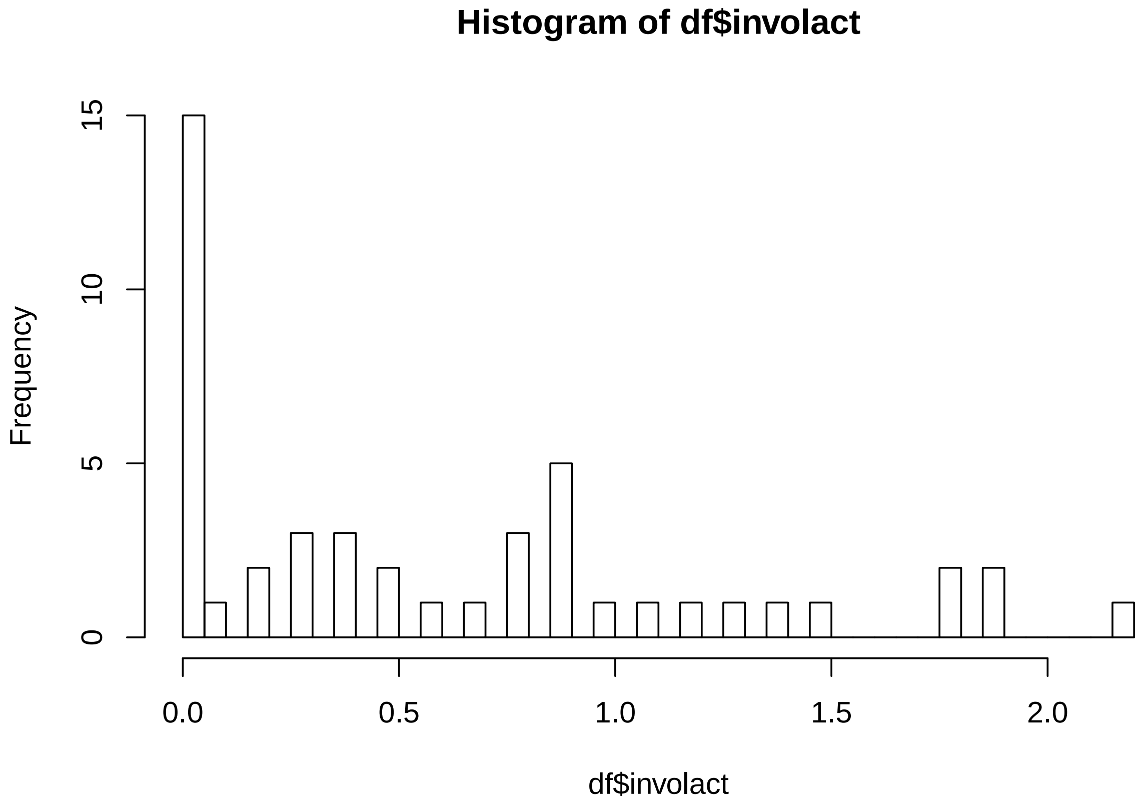
```
##
## Call:
## glm(formula = test ~ diastolic, family = binomial, data = df.na.removed)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3905  -0.9295  -0.7586   1.3246   2.1191
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.168012   0.676646  -4.682 2.84e-06 ***
## diastolic    0.034492   0.009233   3.736 0.000187 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 483.16  on 390  degrees of freedom
## AIC: 487.16
##
## Number of Fisher Scoring iterations: 4
```

Indeed we have confirmation of a relationship between the test and diastolic.

4.1 chredlin data snalysis

The Chicago insurance dataset found in chredlin concerns the problem of redlining in insurance. Read the help page for background. Use involact as the response and ignore volact.

(a) Plot a histogram of the distribution of involact taking care to choose the bin width to illustrate the issue with zero values. What fraction of the responses is zero?



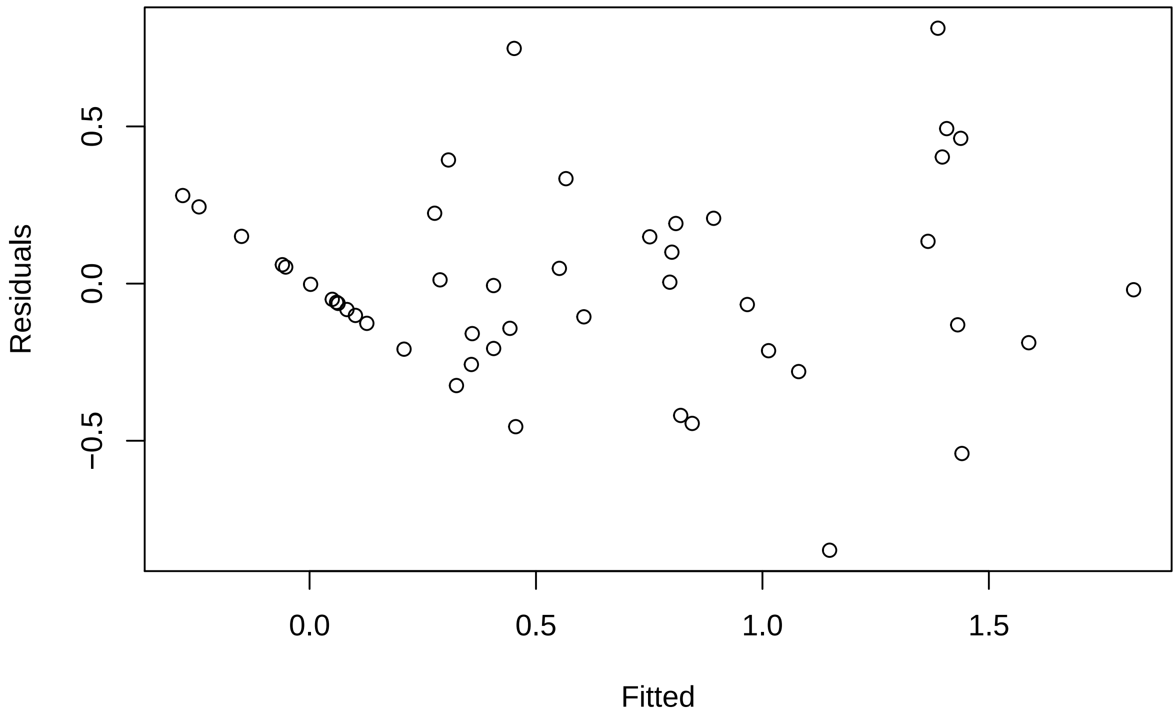
We see the proportion of zeroes is `0.3191489` Interestingly, setting freq=FALSE and using breaks did not work as expected with the `hist` function. We'll revisit this if there is time.

(b) Fit a Gaussian linear model with involact as the response with the other five variables as predictors. Use a log transformation for income. Describe the relationship between these predictors and the response.

```
##
## Call:
## lm(formula = involact ~ race + fire + theft + age + log(income) +
##     side, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.84832 -0.17365 -0.01962  0.17067  0.81249
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.190806   1.114419  -1.069  0.29168
## race         0.009330   0.002847   3.277  0.00217 **
## fire         0.039994   0.008936   4.475 6.19e-05 ***
## theft       -0.010227   0.002899  -3.528  0.00107 **
## age          0.008423   0.002857   2.948  0.00532 **
## log(income)  0.343419   0.405406   0.847  0.40198
## sides        0.016287   0.124922   0.130  0.89692
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3386 on 40 degrees of freedom
## Multiple R-squared:  0.7518, Adjusted R-squared:  0.7146
## F-statistic:  20.2 on 6 and 40 DF,  p-value: 1.059e-10
```

All but two of the predictors are significant in this model.

(c) Plot the residuals against the fitted values. How are the zero response values manifested on the plot? What impact do these cases have on the interpretation of the plot?

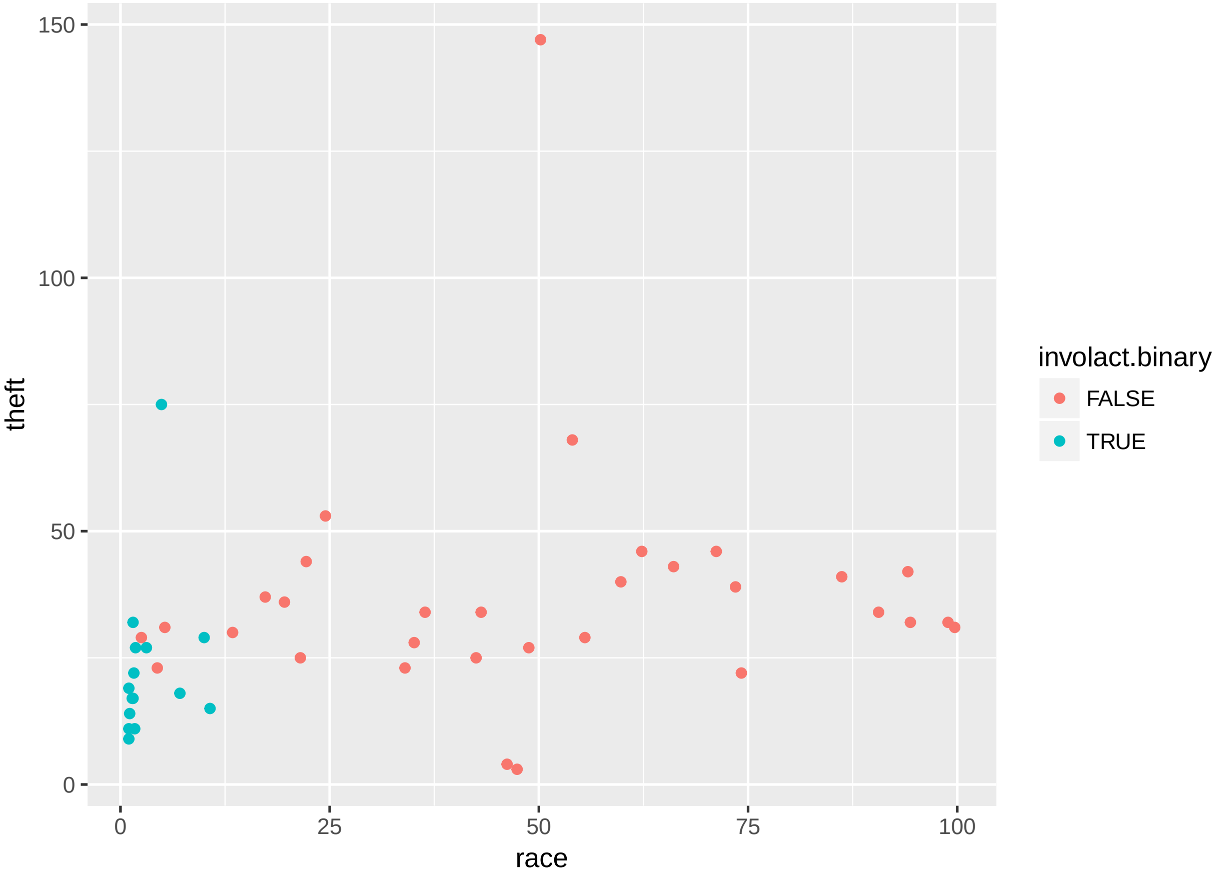


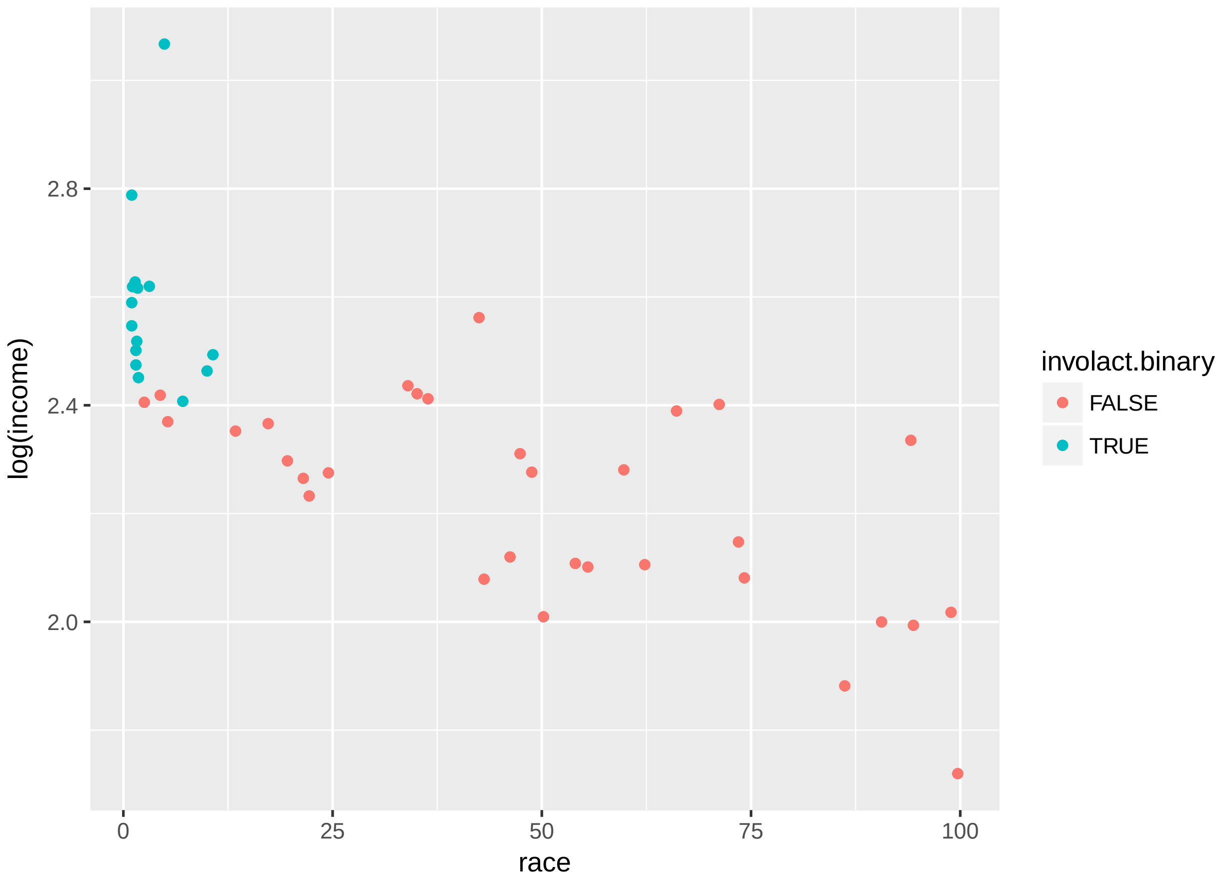
The residuals for the zero response have values have a linear association.

(d) Create a binary response variable which distinguishes zero values of involact. Fit a logistic regression model with this response but with the same five predictors. What problem occurred during this fit? Explain why this happened.

```
##
## Call:
## glm(formula = involact.binary ~ race + fire + theft + age + log(income) +
##     side, family = binomial, data = df)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -8.398e-05  -2.100e-08  -2.100e-08   2.100e-08   7.852e-05
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1092.733 482732.211  -0.002    0.998
## race            -3.739   4174.573  -0.001    0.999
## fire            -3.861  21554.214   0.000    1.000
## theft           -3.546   2084.079  -0.002    0.999
## age             -2.635   2138.273  -0.001    0.999
## log(income)    596.601 214833.228   0.003    0.998
## sides          -62.371  28593.855  -0.002    0.998
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5.8865e+01  on 46  degrees of freedom
## Residual deviance: 2.2855e-08  on 40  degrees of freedom
## AIC: 14
##
## Number of Fisher Scoring iterations: 25
```

We suspect we have perfect class separation which causes instability in the model fitting procedure. We plotted a whole bunch of 2d predictor combinations looking for this but did not encounter it. Some are below. We did not try all combinations, and some came very close. We know that if even one point crosses the linear separating hyperplane then the fitting algorithm should converge.





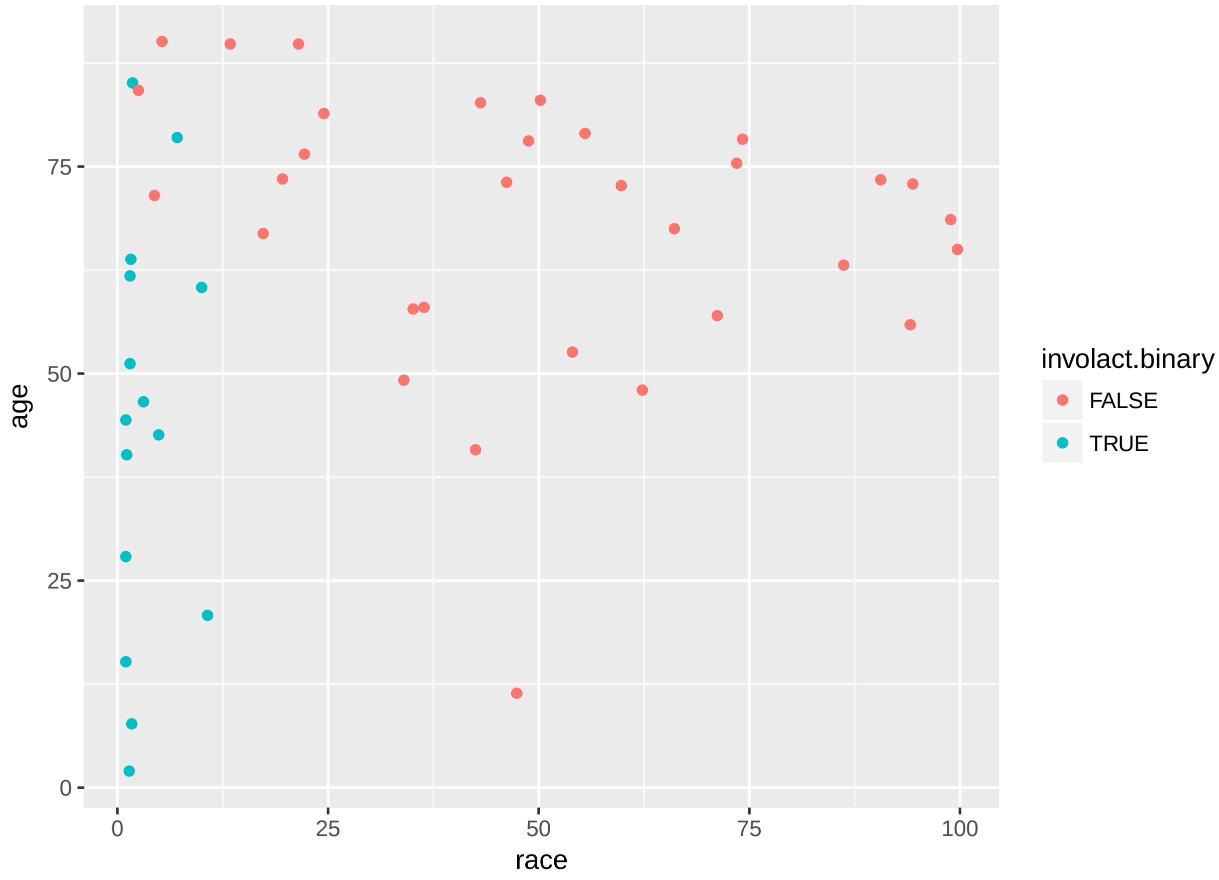
(e) Fit a smaller model using only race and age. Interpret the z-statistics. Test for the signficance of the two predictors using the difference-in-deviances test. Which test for the significance of the predictors should be preferred?

```
##
## Call:
## glm(formula = involact.binary ~ race + age, family = binomial,
##     data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.69864  -0.04390  -0.00014   0.01286   1.50010
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 13.09746    7.44557   1.759   0.0786 .
## race        -0.32539    0.16602  -1.960   0.0500 *
## age         -0.14675    0.08794  -1.669   0.0952 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 58.8653  on 46  degrees of freedom
## Residual deviance:  9.2286  on 44  degrees of freedom
## AIC: 15.229
##
## Number of Fisher Scoring iterations: 10
```

The p-values indicate that all predictors are significant at a level of $\alpha = 0.1$ or below.

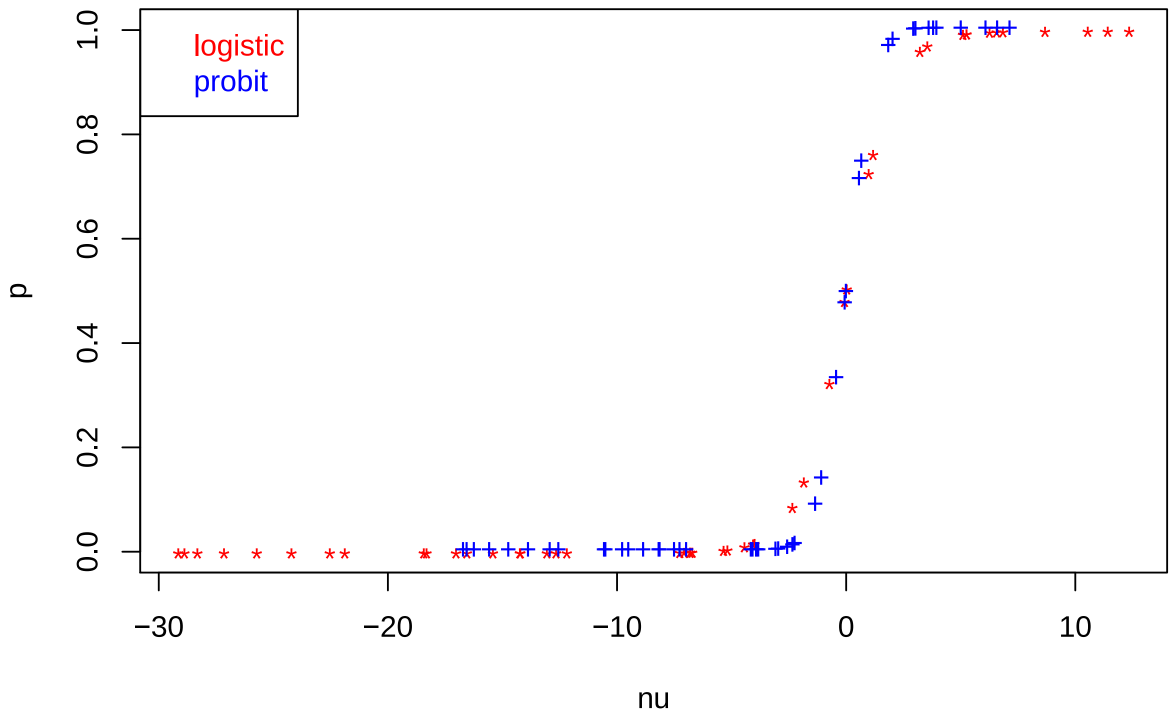
(f) Make plot of race against age which also distinguishes the two levels of the response variable. Interpret the plot and connect it to the previous model output.



(g) Refit the logit model but use a probit link. Compare the model output between the logit and probit models. Which parts are similar and which parts differ substantively? Plot the predicted values on the probability scale against each other and comment on what you see.

```
##
## Call:
## glm(formula = involact.binary ~ race + age, family = binomial(link = probit),
##     data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.65340  -0.00789   0.00000   0.00040   1.48896
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.55984    4.16874   1.813   0.0698 .
## race        -0.18655    0.08913  -2.093   0.0364 *
## age         -0.08503    0.04939  -1.722   0.0851 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 58.8653  on 46  degrees of freedom
## Residual deviance:  8.9786  on 44  degrees of freedom
## AIC: 14.979
##
## Number of Fisher Scoring iterations: 11
```

We fit $involact.binary \sim race + age$ with binomial family and probit link. We will now show the relationship between the logistic and probit models


logistic
probit
p
1.0
0.8
0.6
0.4
0.2
0.0
−30
−20
−10
0
10
nu

NCSU ST 503 HW 12

Probem 5.1,8.5,8.6 Faraway, Julian J. Extending the Linear Model with R CRC Press.

Bruce Campbell

21 November, 2017

# 1 from Chapter 5.

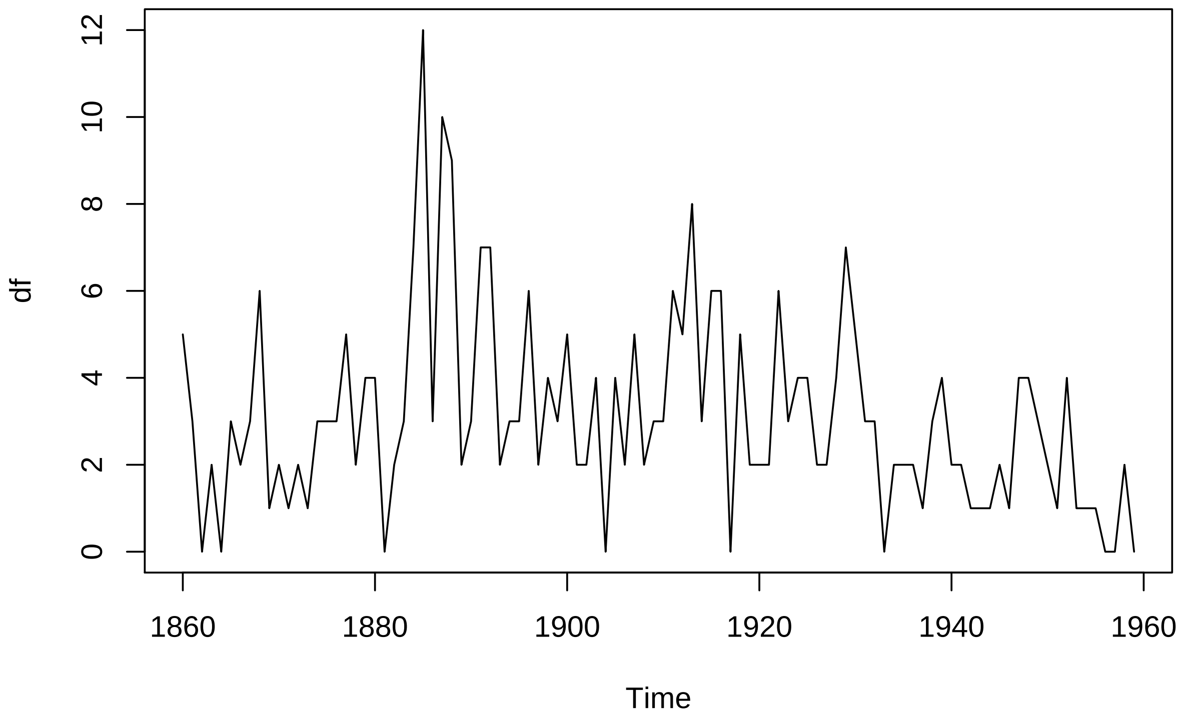
Complete exercises # 5 (a - d) and # 6 (a, b, d - f)) from Chapter 8

5.1 discoveries analysis

The dataset discoveries lists the numbers of "great" inventions and scientific discoveries in each year from 1860 to 1959.

(a) Plot the discoveries over time and comment on the trend, if any.

```
rm(list = ls())
library(faraway)
data("discoveries", package="faraway")
df <- discoveries
plot(df)
```



(b) Fit a Poisson response model with a constant term. Now compute the mean number of discoveries per year. What is the relationship between this mean and the coefficient seen in the model?

```
ddf <- data.frame(df)
model.pois <- glm(df ~ 1, family=poisson, ddf)
sumary(model.pois)
```

```
##             Estimate Std. Error z value  Pr(>|z|)
## (Intercept) 1.131402   0.056796   19.92 < 2.2e-16
##
## n = 100 p = 1
## Deviance = 164.68460 Null Deviance = 164.68460 (Difference = 0.00000)
```

```
pander(data.frame(lambda.hat =mean(df)))
```

lambda.hat
3.1

Our model is $Y_i \sim Pois(\mu_i)$, $log(\mu_i) = x^\intercal \beta$ and we have that $e^{\beta_0} \sim \hat{\lambda}$

(c) Use the deviance from the model to check whether the model fits the data. What does this say about whether the rate of discoveries is constant over time?

The deviance $D \sim \chi^2_{n-1}$ is significant and we conclude the null model is not a good fit for the data. We can conclude - and see in the plot - where the rate changes over time.

(d) Make a table of how many years had zero, one, two, three, etc. discoveries. Collapse eight or more into a single category. Under an appropriate Poisson distribution, calculate the expected number of years with each number of discoveries. Plot the observed against the expected using a different plotting character to denote the number of discoveries. How well do they agree?

```r
tbl <- table(df)
tt <- tbl[1:9]
sumover8 <- sum(tbl[9:length(tbl)])
tt[9] <- sumover8

pander(tt, caption = "freqss")
```

Table 2: freqss

0	1	2	3	4	5	6	7	8
9	12	26	20	12	7	6	4	4

```r
propo <- tt /sum(tt)

pander(propo, caption = "proportion")
```

Table 3: proportion

0	1	2	3	4	5	6	7	8
0.09	0.12	0.26	0.2	0.12	0.07	0.06	0.04	0.04

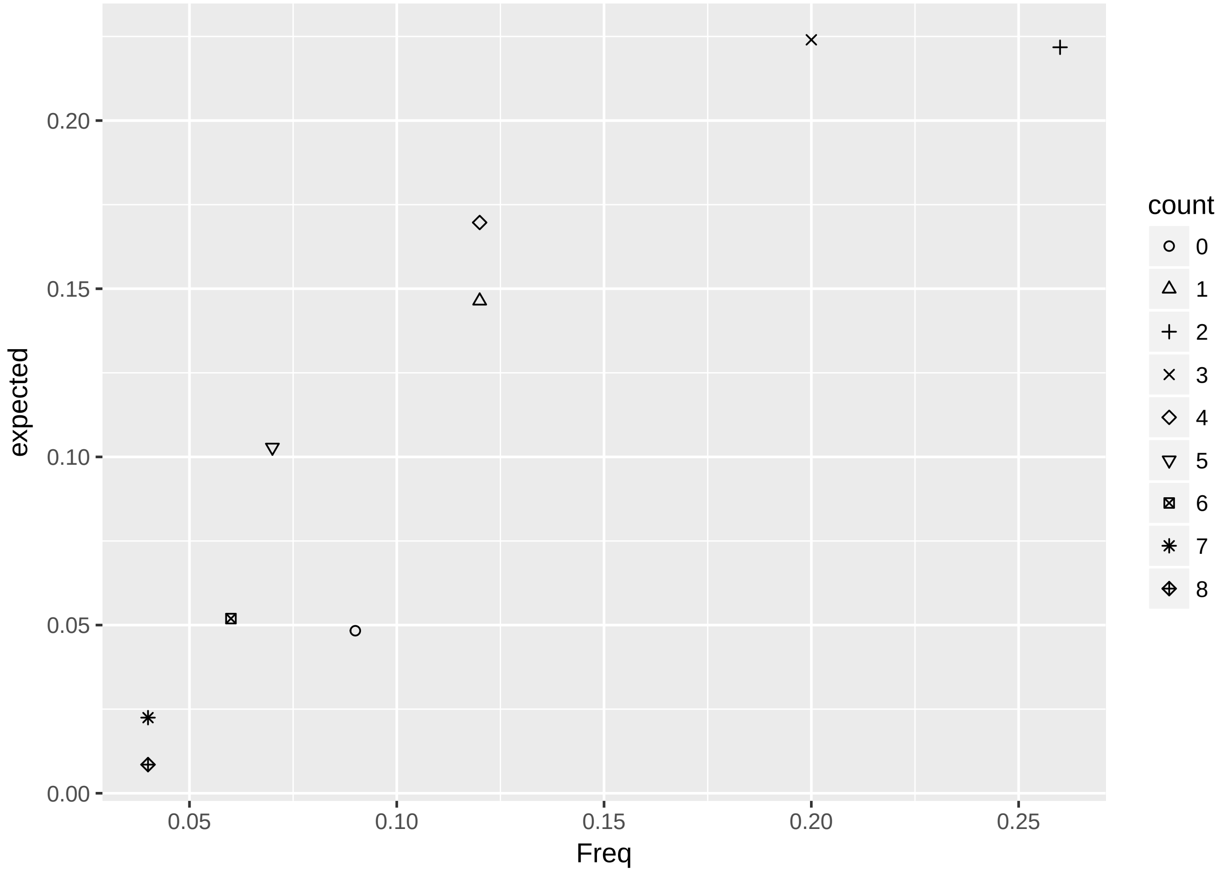
```r
lambda <- sum(0:8 * propo)

expected <- dpois(0:8,lambda = lambda)

#n=1000000
```

```r
#sum(0:n * dpois(0:n,lambda = lambda))

pander(data.frame(t(expected)), caption = "expected")
```

Table 4: expected

X1	X2	X3	X4	X5	X6	X7	X8	X9
0.04832	0.1464	0.2218	0.224	0.1697	0.1028	0.05193	0.02248	0.008514

```r
dfp <-data.frame(count=as.factor(0:8),propo,expected)

(p <- ggplot(dfp, aes(x=Freq, y=expected, shape=count)) +geom_point()+ scale_shape_manu
```



(e) Use the Pearson's Chi-squared test to check whether the observed numbers are consistent with the expected numbers. Interpret the result.

We have to deal with the fact that we binned the counts $9 : \infty$

```r
observed <- tt

elast <- 1- sum(dpois(0:8,lambda = lambda))

ee <- expected
ee[8]<- elast
expected.counts <- ee*sum(tt)

ctbl <-data.frame(observed,expected.counts)
ctbl$df <- NULL
chisq.test(ctbl)
```

```
##
##  Pearson's Chi-squared test
##
## data:  ctbl
## X-squared = 8.4433, df = 8, p-value = 0.3914
```

We have evidence that the observed numbers are cinsistent with the expected numbers

(f) Fit a Poisson response model that is quadratic in the year. Test for the significance of the quadratic term. What does this say about the presence of a trend in discovery?

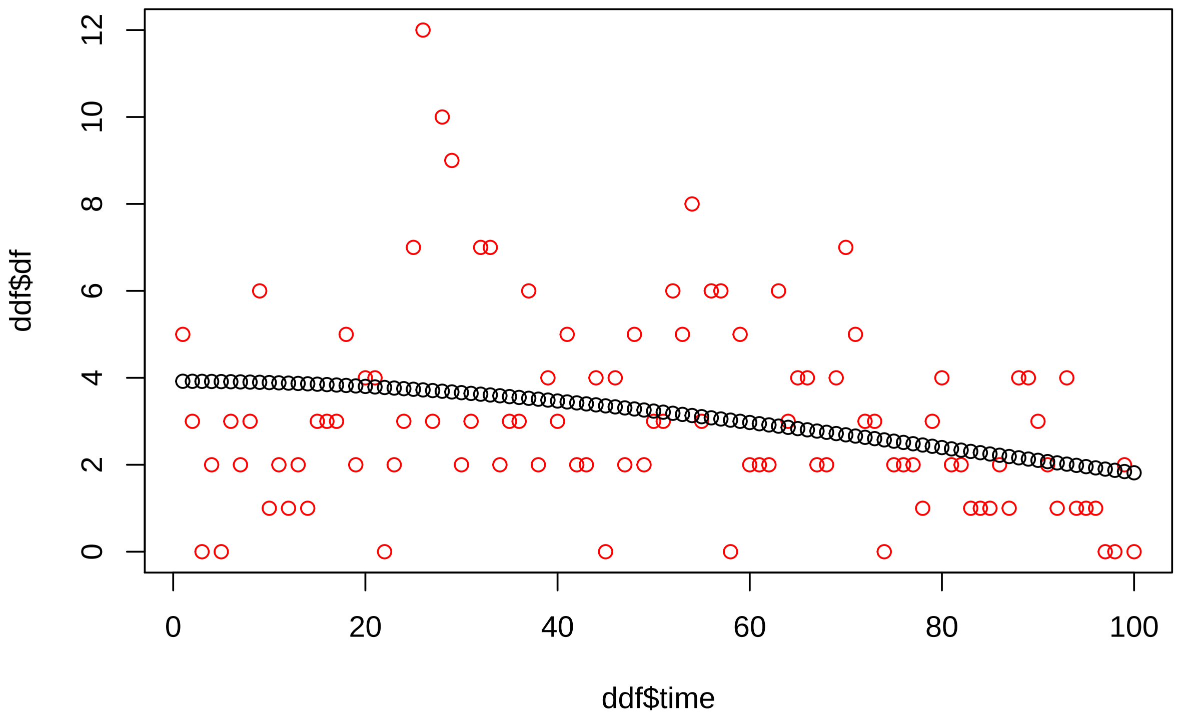
```r
ddf <- data.frame(time=1:length(df),df)
model.pois <- glm(df ~ I(time^2), family=poisson, ddf)
sumary(model.pois)
```

```
##                Estimate  Std. Error z value  Pr(>|z|)
## (Intercept)  1.3664e+00  8.0153e-02 17.0476 < 2.2e-16
## I(time^2)   -7.6975e-05  2.0628e-05 -3.7315 0.0001903
##
## n = 100 p = 2
## Deviance = 149.82413 Null Deviance = 164.68460 (Difference = 14.86047)
```

This model confirms our observation that the rate changes

(g) Compute the predicted number of discoveries each year and show show these predictions as a line drawn over the data. Comment on what you see.

```r
plot(ddf$time,ddf$df,col=2)
points(ddf$time,fitted(model.pois),col=1)
```



We note the rate does change over time in concordance with the data except for a small period of exceptional discovery early in the series.

8.5 Galapagos

Again using the Galápagos data, fit a Poisson model to the species response with the five geographic variables as predictors. Do not use the endemics variable. The purpose of this question is to compare six different ways of testing the significance of the elevation predictor, i.e., $H0 : \beta_{Elev} = 0$. In each case, report the p-value.

(a) Use the z-statistic from the model summary.

```
rm(list = ls())
library(faraway)
data("gala", package="faraway")
df <- gala
model.pois <- glm( Species~ Area+Elevation+Nearest+Scruz+Adjacent, family=poisson, df)
sumary(model.pois)
```

```
##                Estimate  Std. Error  z value  Pr(>|z|)
## (Intercept)  3.1548e+00  5.1750e-02  60.9630 < 2.2e-16
## Area        -5.7994e-04  2.6273e-05 -22.0737 < 2.2e-16
## Elevation    3.5406e-03  8.7407e-05  40.5070 < 2.2e-16
## Nearest      8.8256e-03  1.8213e-03   4.8459 1.261e-06
## Scruz       -5.7094e-03  6.2562e-04  -9.1260 < 2.2e-16
## Adjacent    -6.6303e-04  2.9328e-05 -22.6078 < 2.2e-16
##
## n = 30 p = 6
## Deviance = 716.84577 Null Deviance = 3510.72862 (Difference = 2793.88284)
```

The p-value is `< 2.2e-16`

(b) Fit a model without elevation and use the difference in deviances to make the test.

```
model.pois.reduced <- glm( Species~ Area+Nearest+Scruz+Adjacent, family=poisson, df)
sumary(model.pois.reduced)
```

```
##                Estimate  Std. Error  z value Pr(>|z|)
## (Intercept)  4.3447e+00  3.1137e-02 139.5352   <2e-16
## Area         4.1901e-04  1.2043e-05  34.7930   <2e-16
## Nearest      2.0805e-02  1.6999e-03  12.2390   <2e-16
## Scruz       -6.7806e-03  5.4561e-04 -12.4275   <2e-16
## Adjacent    -2.5642e-06  2.9390e-05  -0.0872   0.9305
##
## n = 30 p = 5
## Deviance = 2389.56888 Null Deviance = 3510.72862 (Difference = 1121.15974)
```

Our test statistic is $\frac{(2389.56888-716.84577)}{\hat{\phi}}$

We need to estimate ϕ

```
(dp <- sum(residuals(model.pois,type="pearson")^2)/model.pois$df.res)
```

```
## [1] 31.74914
```

(c) Use the Pearson Chi-squared statistic in place of the deviance in the previous test.

```
anova(model.pois,test="Chi")
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
```

```
##
## Response: Species
##
## Terms added sequentially (first to last)
##
##
##           Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                        29     3510.7
## Area       1   895.14        28     2615.6 < 2.2e-16 ***
## Elevation  1   802.05        27     1813.5 < 2.2e-16 ***
## Nearest    1    15.71        26     1797.8 7.378e-05 ***
## Scruz      1   456.37        25     1341.5 < 2.2e-16 ***
## Adjacent   1   624.61        24      716.8 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Again the p-value for *Elevation* is `< 2.2e-16`

(d) Fit the Poisson model with a free dispersion parameter as described in Section 5.2. Make the test using the model summary.

```
sumary(model.pois,dispersion=dp)
```

```
##                Estimate  Std. Error z value  Pr(>|z|)
## (Intercept)  3.15480788  0.29158975 10.8193 < 2.2e-16
## Area        -0.00057994  0.00014804 -3.9175 8.947e-05
## Elevation    0.00354059  0.00049251  7.1889 6.530e-13
## Nearest      0.00882557  0.01026214  0.8600    0.3898
## Scruz       -0.00570942  0.00352514 -1.6196    0.1053
## Adjacent    -0.00066303  0.00016525 -4.0123 6.013e-05
##
## Dispersion parameter = 31.74914
## n = 30 p = 6
## Deviance = 716.84577 Null Deviance = 3510.72862 (Difference = 2793.88284)
```

The p-value for *Elevation* is `6.530e-13` so the pvalue is 0.001511532

(e) Use the sandwich estimation method for the standard errors in the original model. Use these to compute z-statistics.

```
library(sandwich)

(sebeta <- sqrt(diag(vcovHC(model.pois))))
```

```
##  (Intercept)         Area    Elevation      Nearest        Scruz
## 0.3185550946 0.0012176343 0.0011939774 0.0228021845 0.0065382200
##     Adjacent
## 0.0006212657
```

our z-value is `3.5406e-03/0.0011939774` which yields a pvalue of

(f) Use the robust GLM estimation method and report the test result from the summary.

```
library(robust)
rmodpla <- glmRob(Species~ log(Area)+log(Elevation)+Nearest+Scruz+Adjacent, family=poiss
summary(rmodpla)
```

```
##
## Call: glmRob(formula = Species ~ log(Area) + log(Elevation) + Nearest +
##     Scruz + Adjacent, family = poisson, data = gala)
## Deviance Residuals:
##        Min         1Q     Median         3Q        Max
## -5.420e+31  1.914e+01  4.374e+01  9.263e+01  1.911e+02
##
## Coefficients:
##                Estimate Std. Error    z value Pr(>|z|)
## (Intercept)    -7647.58     0.3972 -19254.477   0.0000
## log(Area)       -781.53     0.1334  -5856.749   0.0000
## log(Elevation)   878.42     0.8083   1086.707   0.0000
## Nearest         -371.08     5.3795    -68.980   0.0000
## Scruz             50.85    45.7806      1.111   0.2666
## Adjacent         598.49     0.4496   1331.141   0.0000
##
## (Dispersion Parameter for poisson family taken to be 1 )
##
##     Null Deviance: 21190 on 29 degrees of freedom
##
## Residual Deviance: NaN on 24 degrees of freedom
##
## Number of Iterations: 50
##
## Correlation of Coefficients:
##                (Intercept) log(Area) log(Elevation) Nearest Scruz
## log(Area)      2.5177
## log(Elevation) 0.3972      1.0000
## Nearest        0.3972      1.0000    1.0000
## Scruz          0.3972      1.0000    1.0000         1.0000
## Adjacent        0.3972      1.0000    1.0000          1.0000  1.0000
```

No clue why this would not converge unless I log transformed elevation and area!

(g) Compare all six results. Pick the best one and justify your choice.

unfinished :(

Chapter 14 Problem 6 Rice

Rice, John A. Mathematical Statistics and Data Analysis, Cengage

Bruce Campbell

08 September, 2017

Two objects of unknown weights w1 and w2 are weighed on an error-prone pan balance in the following way:

- (1) object 1 is weighed by itself, and the measurement is 3 g;
- (2) object 2 is weighed by itself, and the result is 3 g;
- (3) the difference of the weights (the weight of object 1 minus the weight of object 2) is measured by placing the objects in different pans, and the result is 1 g;
- (4) the sum of the weights is measured as 7g.

The problem is to estimate the true weights of the objects from these measurements.

- a. Set up a linear model, $Y = X\beta + \epsilon$.
- b. Find the least squares estimates of $w1$ and $w2$.
- c. Find the estimate of σ^2
- d. Find the estimated standard errors of the least squares estimates of part (b).
- e. Estimate w1-W2 and its standard error.
- f. Test the null hypothesis $H_0 : w1 = w2$

For notational convenience we denote $w_1 = \beta_1$ and $w_2 = \beta_2$

Also, we do not include an intercept in this model

Our model is

$$\mathbf{Y} = \mathbf{X}\,\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

The response \$ **Y**\$ is our measurements from the scale, the two column model matrix **X** describes the configuration of weights on the scale, and $\boldsymbol{\epsilon}$ is the error of the scale. For now we assume the errors are iid mean zero random variables, in physical systems we may want to make the additional assumption that $\epsilon_i \sim N(0, \sigma) \;\; \forall \; i$

We seek $\hat{\beta}$ such that

$$\mathbf{Y} = \mathbf{X}\,\hat{\boldsymbol{\beta}}$$

where

$$Y = \begin{pmatrix} 3 \\ 3 \\ 1 \\ 7 \end{pmatrix}$$

and

$$X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & -1 \\ 1 & 1 \end{pmatrix}$$

Since

$$\mathbf{X} \in \mathbb{R}^{4x2}$$

we know that

$$\mathbf{X}^\intercal \mathbf{X} \in \mathbb{R}^{2x2}$$

and we have hope that this can be easily calculated by hand.

$$\mathbf{X}^\intercal \mathbf{X} == \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}$$

which gives

$$(\mathbf{X}^\intercal \mathbf{X})^{-1} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$$

Now

$$(\mathbf{X}^\intercal \mathbf{Y}) = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 3 \\ 1 \\ 7 \end{pmatrix} = \begin{pmatrix} 11 \\ 9 \end{pmatrix}$$

Finally we have that

$$(\mathbf{X}^\intercal \mathbf{X})^{-1}\ \mathbf{X}^\intercal \mathbf{Y} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix} \begin{pmatrix} 11 \\ 9 \end{pmatrix} = \begin{pmatrix} \frac{11}{3} \\ \frac{9}{3} \end{pmatrix} = \hat{\boldsymbol{\beta}}$$

Calculate s^2

$$\hat{\boldsymbol{\epsilon}} = \mathbf{Y} - \mathbf{X}\ \hat{\boldsymbol{\beta}} == \begin{pmatrix} \frac{-2}{3} \\ 0 \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix}$$

And

$$\frac{\hat{\boldsymbol{\epsilon}}^\intercal \hat{\boldsymbol{\epsilon}}}{n-p} = s^2$$

Where $n = 4$ and $p = 2$ in this case. Note if we had used an intercept we would have $p = 3$. Putting the values in we get that

$$s^2 = \frac{1}{3}$$

Calculate $se(\beta_i)$

Let

$$\mathbf{C} = (\mathbf{X}^\intercal \mathbf{X})^{-1}$$

Then

$$se(\beta_i) = s\sqrt{C_{i\,i}}$$

and putting our values in from above we have that

$$se(\beta_1) = se(\beta_2) = \frac{1}{\sqrt{3}}\frac{1}{\sqrt{3}} = \frac{1}{3}$$

###Estimate $\beta_1 - \beta 2$ and find the it's standard error

If $\mathbf{x}_0 \in \mathbb{R}^p$ is a vector of predictor variables then the prediction Y_0 is given by $\mu_0 = \mathbf{x}_0^\intercal \hat{\boldsymbol{\beta}}$. We saw that the covariance matrix of $\hat{\boldsymbol{\beta}}$ is

$$\Sigma_{\hat{\boldsymbol{\beta}}\,\hat{\boldsymbol{\beta}}} = \sigma^2\,(\mathbf{X}^\intercal \mathbf{X})^{-1}$$

under the assumption that the errors are iid with constant variance. We can use this with the theorem about linear transforms of random vectors to get that

$$Var(\mu_0) = \mathbf{x}_0^\intercal\,\Sigma_{\hat{\boldsymbol{\beta}}\,\hat{\boldsymbol{\beta}}}\,\mathbf{x}_0$$

Now we're looking to estimate $\beta_1 - \beta_2$ so our predictor vectors is going to be

$$x_0 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Putting all our values from above in to the expression for μ_0 and $Var(\mu_0)$ we get that

$$\mu_0 = \begin{pmatrix} 1 & -1 \end{pmatrix} \begin{pmatrix} \frac{11}{3} \\ 3 \end{pmatrix} = \frac{2}{3}$$

$$se(\widehat{\beta_1 - \beta_2}) = \sqrt{Var(\mu_0)} = \sqrt{\begin{pmatrix} 1 & -1 \end{pmatrix} \frac{1}{3} \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix}} = \frac{\sqrt{2}}{3}$$

Note that we've use the estimate s^2 for σ^2

Test $H_0 : \beta_1 = \beta_2$

If we adopt the additional assumption that the errors are normally distributed, we will have that the $(1-\alpha)100\%$ CI for $\hat{\mu_0}$ is

$$\hat{\mu_0} \pm t_{n-p}(\frac{\alpha}{2})s_{\hat{\mu_0}}$$

```r
mu0 <- 2/3
se_mu0 <- sqrt(2)/3
n = 4
p = 2
t_alpha <- qt(0.95, n - p)
leftCI <- mu0 - t_alpha * se_mu0
rightCI <- mu0 + t_alpha * se_mu0
pander(data.frame(left = leftCI, right = rightCI), caption = "95% CI")
```

Table 1: 95% CI

left	right
-0.7098	2.043

Since our null hypotheses H_0 is in the CI we do not have enough evidence to reject it.

Checking ealier calculations in R.

We calculated the solution by hand (attached below) and then checked portions of it in R.

```r
data <- data.frame(X1 = c(1, 0, 1, 1), X2 = c(0, 1, -1, 1), Y = c(3, 3, 1, 7))


lm.fit.nointercept <- lm(Y ~ X1 + X2 - 1, data = data)

summary(lm.fit.nointercept)
```

```
## 
## Call:
## lm(formula = Y ~ X1 + X2 - 1, data = data)
## 
## Residuals:
##          1          2          3          4 
## -6.667e-01  4.996e-16  3.333e-01  3.333e-01 
## 
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)   
## X1   3.6667     0.3333      11  0.00816 **
## X2   3.0000     0.3333       9  0.01212 * 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.5774 on 2 degrees of freedom
## Multiple R-squared:  0.9902, Adjusted R-squared:  0.9804 
## F-statistic:   101 on 2 and 2 DF,  p-value: 0.009804
```

This agrees - up to numerical precision - with the exact calculations we did by hand for β, $se(\beta_1)$, $se(\beta_2)$ and s^2