Diagnostics Homework Solutions
By Matt Jones

3. First, here is a description of the prostate dataset from the cran site:

Description

The `prostate` data frame has 97 rows and 9 columns. A study on 97 men with prostate cancer who were due to receive a radical prostatectomy.

Usage

```
data(prostate)
```

Format

This data frame contains the following columns:

`lcavol` log(cancer volume)

`lweight` log(prostate weight)

`age` age

`lbph` log(benign prostatic hyperplasia amount)

`svi` seminal vesicle invasion

`lcp` log(capsular penetration)

`gleason` Gleason score

`pgg45` percentage Gleason scores 4 or 5

`lpsa` log(prostate specific antigen)

Source

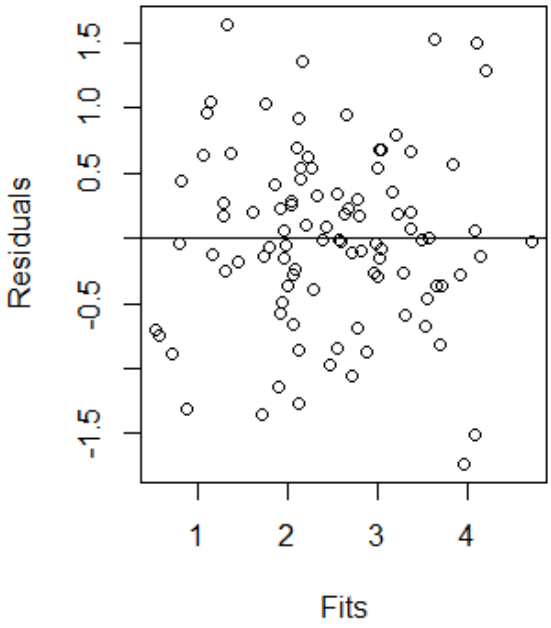
Andrews DF and Herzberg AM (1985): Data. New York: Springer-Verlag

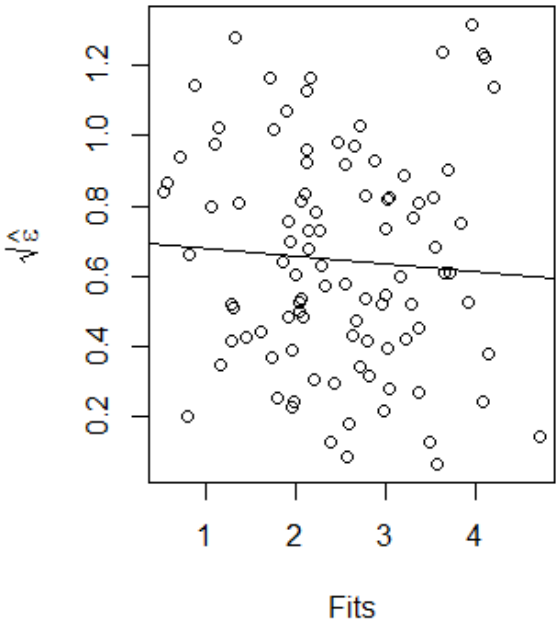
Here's a fit of the lpsa variable against the other variables:

```
> prostatelm <- lm(lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason + pgg45, data=prostate)
> summary(prostatelm)

Call:
lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
    gleason + pgg45, 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
```

So it looks like the significant variables are lcavol, lweight, svi, and perhaps lbph and/or age. The r^2 values are okay, at about 62% - 65%.

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



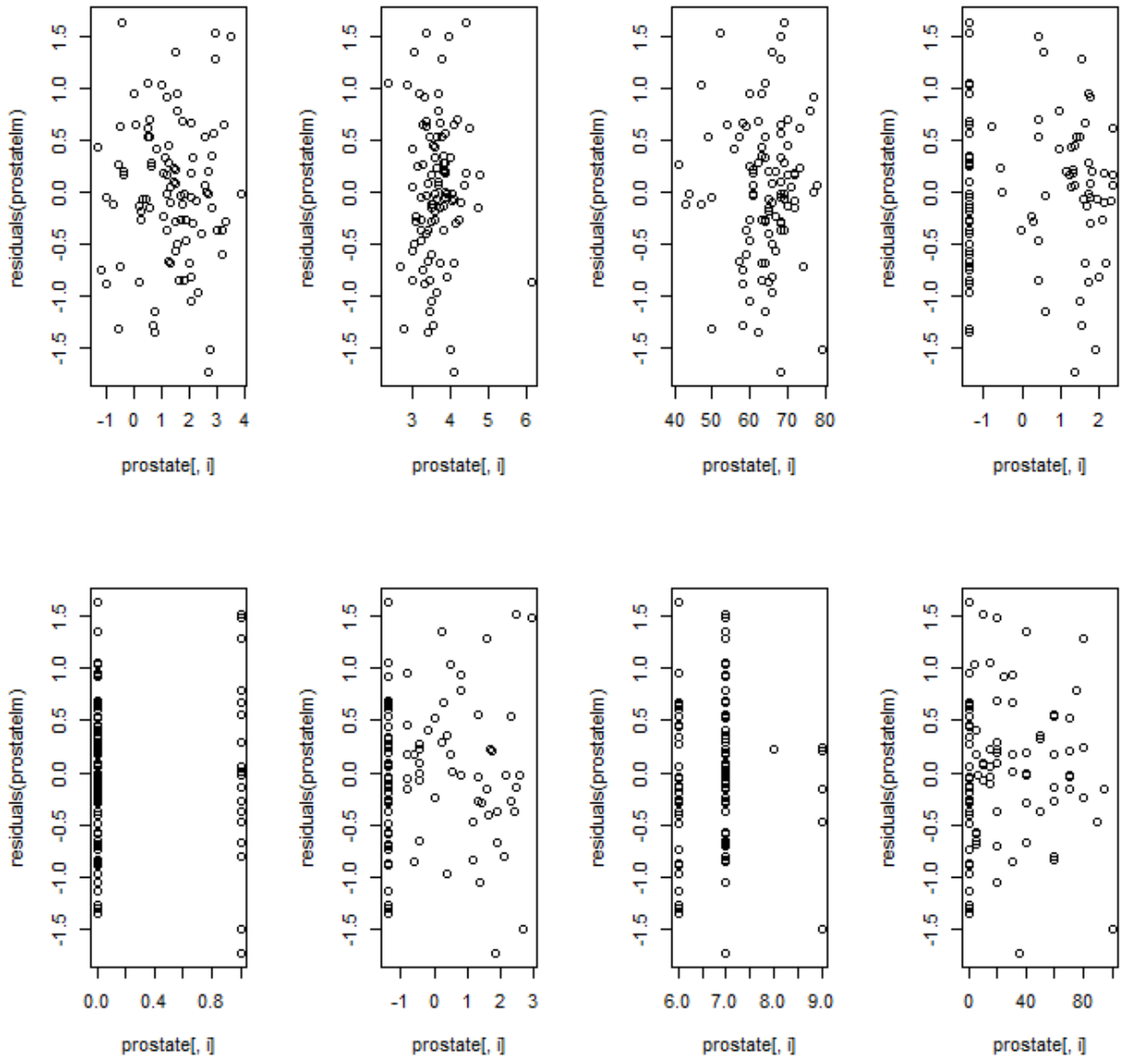



```
> par(mfrow=c(1,2))
> plot(fitted(prostatelm), residuals(prostatelm), xlab="Fits", ylab="Residuals")
> abline(h=0)
> plot(fitted(prostatelm), sqrt(abs(residuals(prostatelm))), xlab="Fits",
ylab=expression(sqrt(hat(epsilon))))
> sumary(lm(sqrt(abs(residuals(prostatelm))) ~ fitted(prostatelm)))
                Estimate Std. Error t value  Pr(>|t|)
(Intercept)       0.703381   0.090607  7.7630 9.475e-12
fitted(prostatelm) -0.021990   0.034232 -0.6424    0.5222

n = 97, p = 2, Residual SE = 0.31328, R-Squared = 0
> flipped.out <- lm(sqrt(abs(residuals(prostatelm))) ~ fitted(prostatelm))
> abline(flipped.out)
> sumary(flipped.out)
                Estimate Std. Error t value  Pr(>|t|)
(Intercept)       0.703381   0.090607  7.7630 9.475e-12
fitted(prostatelm) -0.021990   0.034232 -0.6424    0.5222

n = 97, p = 2, Residual SE = 0.31328, R-Squared = 0
```

Due to the graphs and also the p-value of .5222 for the t-test for significance of the slope of the line running through the sqrt(|residuals|) vs. fits, there does not seem to be much reason to suspect the errors have non-constant variance.

Next, I will plot the residuals versus each of the predictor variables, checking for marginal constant variance:

```
> windows()
> par(mfrow=c(2, 4))
> for(i in 1:8){plot(residuals(prostatelm) ~ prostate[,i])}
```



So the main things that catch my eye here are that the svi and gleason variables seem to be discrete. Also, the lbph variable seems to have a clustering of values at -1.386294, and there seems to be a clustering of pgg45 at 0. There might be a nonconstant variance associated with the lbph, svi, lcp, gleason, and pgg45 variables. Let's check them one at a time by running an F-test on for equality of variance, partitioning each predictor range in a "reasonable" way:

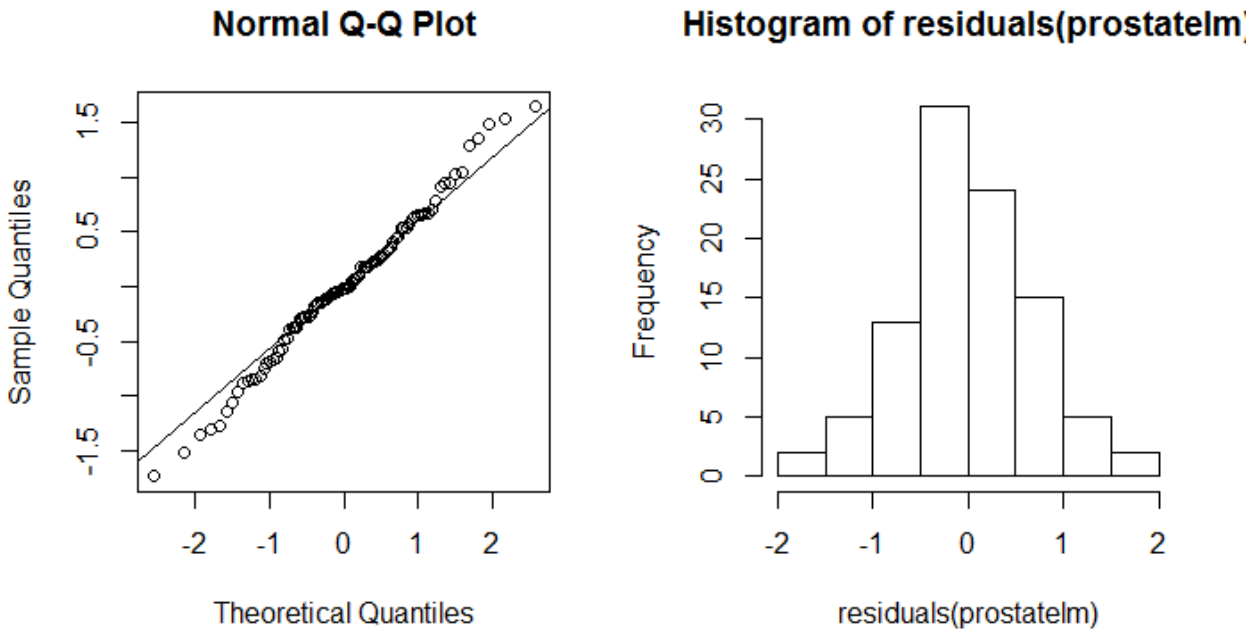
```
> var.test(residuals(prostatelm)[prostate$lbph < -1], residuals(prostatelm)[prostate$lbph > -1])

        F test to compare two variances

data:  residuals(prostatelm)[prostate$lbph < -1] and residuals(prostatelm)[prostate$lbph > -1]
F = 0.9933, num df = 42, denom df = 53, p-value = 0.9907
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.5617949 1.7910392
sample estimates:
ratio of variances
        0.9932875

> var.test(residuals(prostatelm)[prostate$svi < .4], residuals(prostatelm)[prostate$svi > .4])

        F test to compare two variances

data:  residuals(prostatelm)[prostate$svi < 0.4] and residuals(prostatelm)[prostate$svi > 0.4]
F = 0.5309, num df = 75, denom df = 20, p-value = 0.05251
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2416742 1.0067865
sample estimates:
ratio of variances
        0.5309392

> var.test(residuals(prostatelm)[prostate$lcp < -1], residuals(prostatelm)[prostate$lcp > -1])

        F test to compare two variances

data:  residuals(prostatelm)[prostate$lcp < -1] and residuals(prostatelm)[prostate$lcp > -1]
F = 0.8749, num df = 44, denom df = 51, p-value = 0.6536
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.4942496 1.5680687
sample estimates:
ratio of variances
         0.874909

> var.test(residuals(prostatelm)[prostate$gleason < 6.5], residuals(prostatelm)[prostate$gleason > 6.5])

        F test to compare two variances

data:  residuals(prostatelm)[prostate$gleason < 6.5] and residuals(prostatelm)[prostate$gleason > 6.5]
F = 1.107, num df = 34, denom df = 61, p-value = 0.7155
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.6225646 2.0732200
```

```
sample estimates:
ratio of variances
        1.107029

> var.test(residuals(prostatelm)[prostate$pgg45 < 1], residuals(prostatelm)[prostate$pgg45 > 1])

        F test to compare two variances

data:  residuals(prostatelm)[prostate$pgg45 < 1] and residuals(prostatelm)[prostate$pgg45 > 1]
F = 1.107, num df = 34, denom df = 61, p-value = 0.7155
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.6225646 2.0732200
sample estimates:
ratio of variances
        1.107029
```

The smallest p-value for these F-tests is just over 5% (for the svi variable). It could be that a data transformation would be ideal for this variable.

(b) Check the normality assumption.

```
> windows()
> par(mfrow=c(1,2))
> qqnorm(residuals(prostatelm, ylab="Residuals", main="Normal probability plot of residuals"))
> qqline(residuals(prostatelm))
> hist(residuals(prostatelm))
> shapiro.test(residuals(prostatelm))

        Shapiro-Wilk normality test

data:  residuals(prostatelm)
W = 0.9911, p-value = 0.7721
```



Given the linearity of the qq-plot and that the p-value of the Shapiro-Wilks test is not very small, I'd say the normality assumption is probably satisfied. That is, there is insufficient evidence to reject the claim that the errors are normally distributed.

(c) Check for large leverage points.

First I'll obtain the hat values, which are the diagonal elements of the hat matrix:

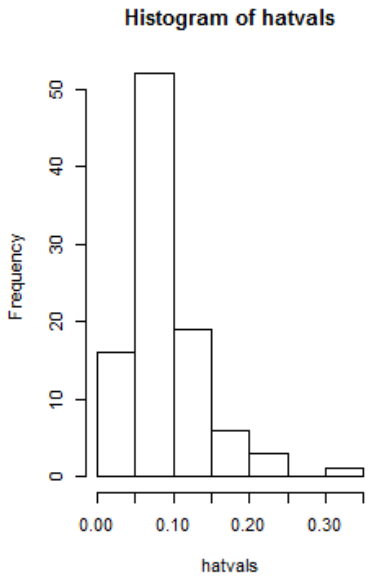
```
> hatvals <- hatvalues(prostatelm)
> sum(hatvals)
[1] 9
> head(prostate)
     lcavol lweight age      lbph svi      lcp gleason pgg45      lpsa
1 -0.5798185  2.7695  50 -1.386294   0 -1.38629       6     0 -0.43078
2 -0.9942523  3.3196  58 -1.386294   0 -1.38629       6     0 -0.16252
3 -0.5108256  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
```

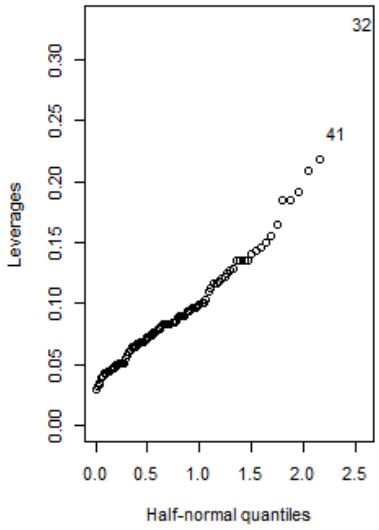
I've also checked that their sum equals p = number of model parameters = 9.

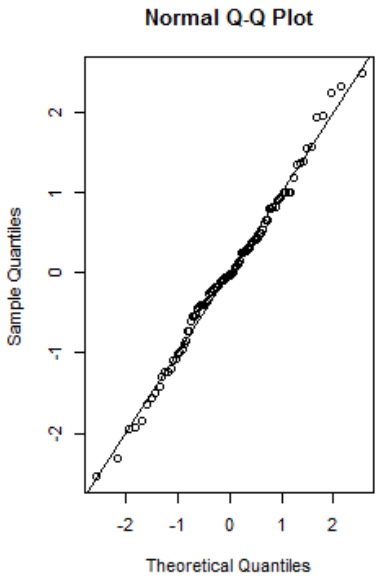
I'll make a histogram of the hat values, and also plot the leverage values (the hat values) against the half-normal quantiles, and make a qq-plot for the standardized residuals:

```
> windows()
> par(mfrow=c(1,3))
> hist(hatvals)
```

```
> row <- seq(1:nrow(prostate))
> prostate <- cbind(prostate, row)
> head(prostate)
     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
  row
1   1
2   2
3   3
4   4
5   5
6   6
> halfnorm(hatvals, labs=row, ylab="Leverages")
> qqnorm(rstandard(prostatelm))
> abline(0,1)
```







There might be a couple of leverage values in rows 41 and 32. Let us remove them and redo the regression for the full model:

```
> rm4132 <- prostate[-c(41, 32),]
> rm4132lm <- lm(lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason + pgg45, data=rm4132)
> summary(rm4132lm)

Call:
```

```
lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
    gleason + pgg45, data = rm4132)

Residuals:
     Min       1Q   Median       3Q      Max
-1.77963 -0.36315 -0.05768  0.42073  1.54611

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.291805   1.362779   0.214  0.83096
lcavol       0.566701   0.088941   6.372  8.9e-09 ***
lweight      0.620974   0.202967   3.059  0.00296 **
age         -0.020661   0.011284  -1.831  0.07058 .
lbph         0.098485   0.059172   1.664  0.09968 .
svi          0.758756   0.243759   3.113  0.00252 **
lcp         -0.095831   0.093696  -1.023  0.30927
gleason      0.027747   0.165601   0.168  0.86733
pgg45        0.004261   0.004436   0.961  0.33944
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7066 on 86 degrees of freedom
Multiple R-squared:  0.6636,    Adjusted R-squared:  0.6323
F-statistic: 21.21 on 8 and 86 DF,  p-value: < 2.2e-16
```

Comparing this with the output for the original model:

```
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
```

You want to look to see if the regression coefficients have changed much. Unfortunately, it's difficult to tell since we don't know the units on the data, and comparing magnitudes of sheer numbers is not very meaningful. The standard errors haven't changed much either, and also the R62 and R^2 adjusted values do not seem to have changed significantly. Perhaps the potential leverage points aren't having much of an influence on the regression line. Remember though, that leverage points don't have to be influential: leverage points are just ones that fall far away from the rest of the points in terms of their predictor variable coordinates.

(d) Check for outliers.

```
> stud <- rstudent(prostatelm)
> stud[which.max(abs(stud))]
    39
-2.61698
```

The studentized residual with the largest magnitude is -2.61698. Comparing that to the critical value for a two-tailed family error rate of 5%

```
> qt(.05/nrow(prostate)/2, nrow(prostate)-9)
[1] -3.605841
```

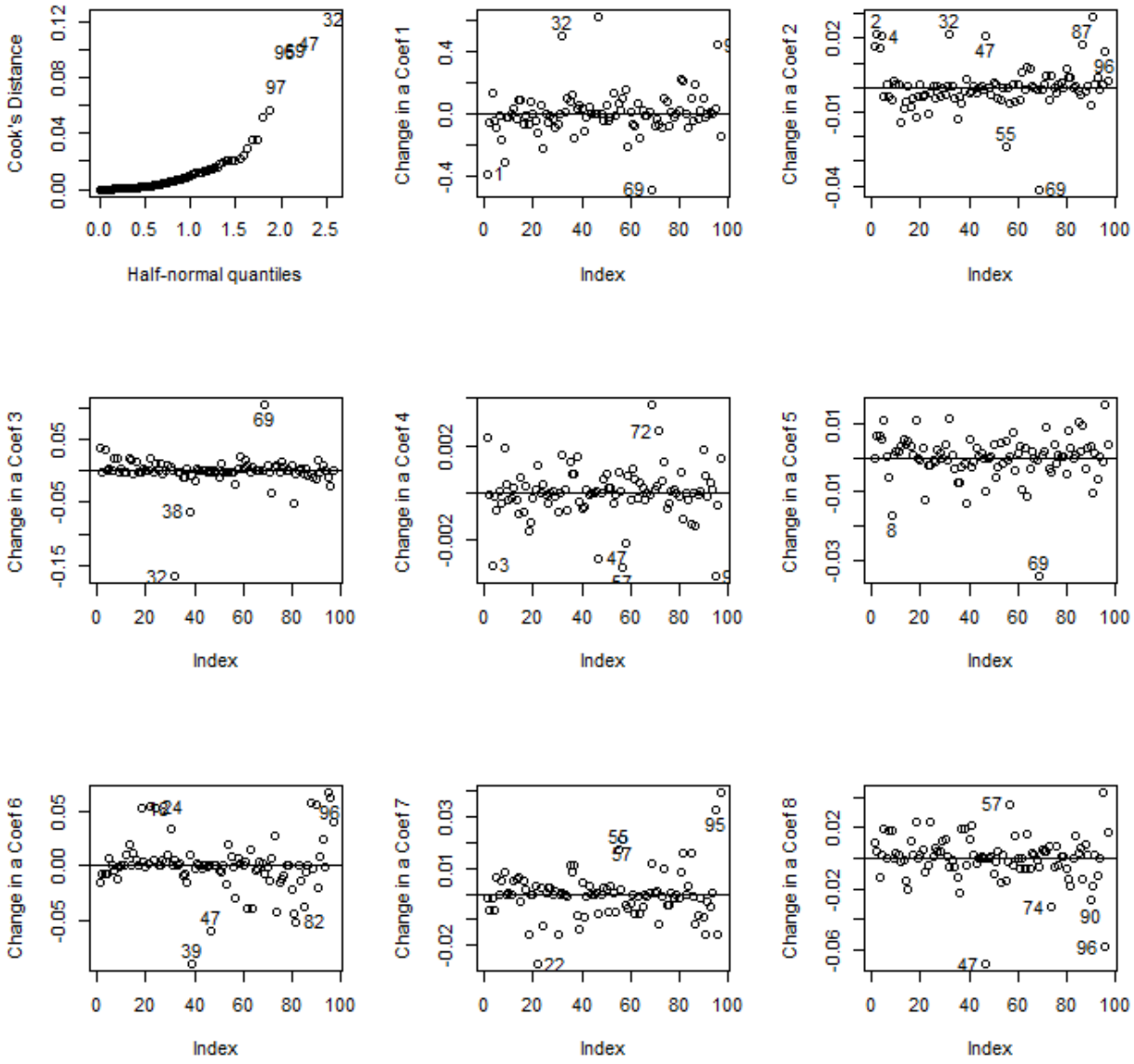
it seems the largest studentized residual is really not that large in magnitude, and so there probably aren't any outliers.

(e) Check for influential points.

See part (c) above. I addressed this in part when checking for leverage points. Now let's compute the Cook's distances:

```
> par(mfrow=c(1,2))
> halfnorm(cook, 3, labs=row, ylab="Cook's Distance")
> prostatelm.x <- lm(lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason + pgg45, prostate,
subset=(cook< max(cook)))
> sumary(prostatelm.x)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.1718627  1.3288221  0.1293 0.897391
lcavol       0.5653328  0.0884722  6.3899 7.93e-09
lweight      0.6216627  0.2020165  3.0773 0.002793
age         -0.0212715  0.0111459 -1.9085 0.059630
lbph         0.0955905  0.0585289  1.6332 0.106037
svi          0.7604232  0.2425957  3.1345 0.002347
lcp         -0.1059870  0.0903647 -1.1729 0.244045
gleason      0.0506884  0.1563842  0.3241 0.746620
pgg45        0.0044683  0.0043898  1.0179 0.311554

n = 96, p = 9, Residual SE = 0.70336, R-Squared = 0.66
> sumary(prostatelm)
```

```
             Estimate Std. Error t value  Pr(>|t|)
(Intercept)  0.6693367  1.2963875  0.5163  0.606934
lcavol       0.5870218  0.0879203  6.6767 2.111e-09
lweight      0.4544674  0.1700124  2.6731  0.008955
age         -0.0196372  0.0111727 -1.7576  0.082293
lbph         0.1070540  0.0584492  1.8316  0.070398
svi          0.7661573  0.2443091  3.1360  0.002329
lcp         -0.1054743  0.0910135 -1.1589  0.249638
gleason      0.0451416  0.1574645  0.2867  0.775033
pgg45        0.0045252  0.0044212  1.0235  0.308860

n = 97, p = 9, Residual SE = 0.70842, R-Squared = 0.65

> windows()
> par(mfrow=c(3,3))
> halfnorm(cook, 5, labs=row, ylab="Cook's Distance")
> plot(dfbeta(prostatelm)[,1],ylab="Change in a Coef 1")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,1], labels=row.names(prostate))
[1]  1 32 47 69 96
> plot(dfbeta(prostatelm)[,2],ylab="Change in a Coef 2")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,2], labels=row.names(prostate))
 [1]  2  4 32 47 55 69 87 91 96
> plot(dfbeta(prostatelm)[,3],ylab="Change in a Coef 3")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,3], labels=row.names(prostate))
[1] 32 38 69
> plot(dfbeta(prostatelm)[,4],ylab="Change in a Coef 4")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,4], labels=row.names(prostate))
[1]  3 47 57 69 72 95
> plot(dfbeta(prostatelm)[,5],ylab="Change in a Coef 5")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,5], labels=row.names(prostate))
[1]  8 69 96
> plot(dfbeta(prostatelm)[,6],ylab="Change in a Coef 6")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,6], labels=row.names(prostate))
 [1] 18 24 39 47 82 95 96
> plot(dfbeta(prostatelm)[,7],ylab="Change in a Coef 7")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,7], labels=row.names(prostate))
 [1] 22 55 57 95 97
> plot(dfbeta(prostatelm)[,8],ylab="Change in a Coef 8")
> abline(h=0)
> identify(seq(1:nrow(prostate)), dfbeta(prostatelm)[,8], labels=row.names(prostate))
[1] 47 57 74 90 95 96
```



The observations in rows 69, 47, and 32 seem to be causing noticeable changes in several of the betas when left out, and their Cook's distances are also large compared to the rest. Perhaps these points are influential. I'll try leaving them out of the model and see what happens:

```
> rm473269 <- prostate[-c(47, 32, 69),]
> rm473269lm <- lm(lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason + pgg45,
data=rm473269)
> summary(rm473269lm)

Call:
lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
   gleason + pgg45, data = rm473269)
```

```
Residuals:
   Min     1Q  Median     3Q    Max
-1.8555 -0.3062 -0.0355  0.4326  1.4898

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.144363   1.308087   0.110 0.912383
lcavol       0.588711   0.086330   6.819 1.25e-09 ***
lweight      0.479408   0.200835   2.387 0.019201 *
age         -0.021434   0.010700  -2.003 0.048334 *
lbph         0.140689   0.057561   2.444 0.016586 *
svi          0.829867   0.231338   3.587 0.000557 ***
lcp         -0.108953   0.085839  -1.269 0.207806
gleason      0.124391   0.150913   0.824 0.412099
pgg45        0.004664   0.004163   1.120 0.265755
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6668 on 85 degrees of freedom
Multiple R-squared:  0.7034,    Adjusted R-squared:  0.6755
F-statistic:  25.2 on 8 and 85 DF,  p-value: < 2.2e-16
```

So the R^2 and adjusted R^2 values have improved. Below is our full model for the full dataset:

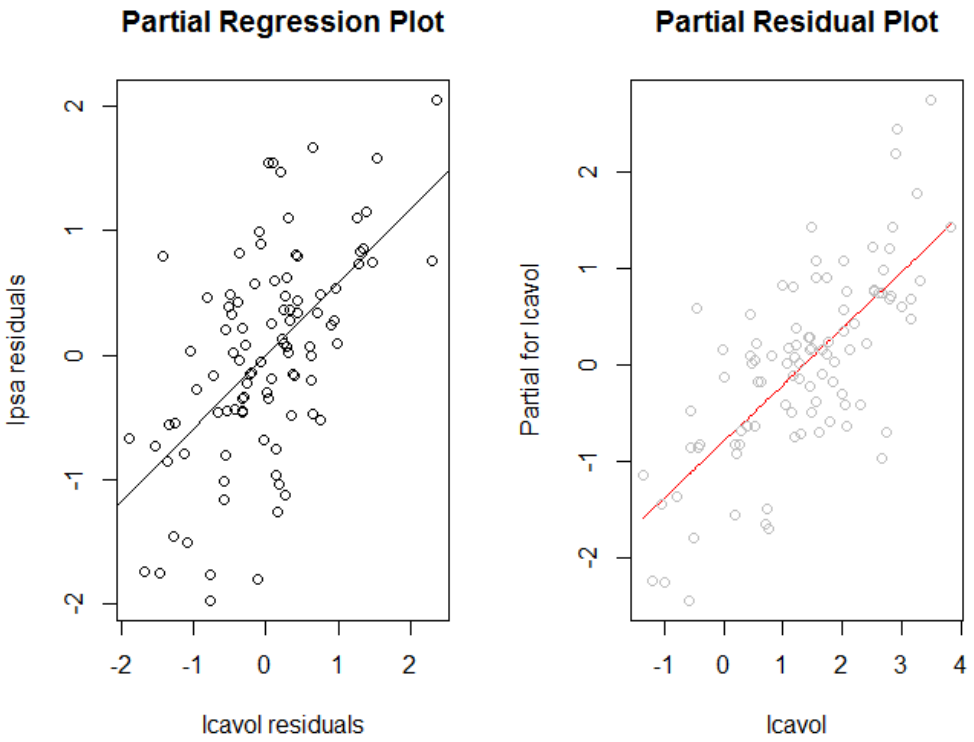
```
> sumary(prostatelm)
             Estimate Std. Error t value  Pr(>|t|)
(Intercept)  0.6693367  1.2963875  0.5163  0.606934
lcavol       0.5870218  0.0879203  6.6767 2.111e-09
lweight      0.4544674  0.1700124  2.6731  0.008955
age         -0.0196372  0.0111727 -1.7576  0.082293
lbph         0.1070540  0.0584492  1.8316  0.070398
svi          0.7661573  0.2443091  3.1360  0.002329
lcp         -0.1054743  0.0910135 -1.1589  0.249638
gleason      0.0451416  0.1574645  0.2867  0.775033
pgg45        0.0045252  0.0044212  1.0235  0.308860

n = 97, p = 9, Residual SE = 0.70842, R-Squared = 0.65
```

Besides the R^2 values improving, the age and lbph variables have become more significant. Therefore, we might want to include them in our model, and actually use the model with beta estimates calculated from the data without those three potential influential observations.

(f) Check the structure of the relationship between the predictors and the response.

Let's construct a partial regression/added variable plot for first variable, lcavol. Then we'll make a partial residual plot:

```
> windows()
> par(mfrow=c(1,2))
> d <- residuals(lm(lpsa ~ lweight + age + lbph + svi + lcp + gleason + pgg45, prostate))
> m <- residuals(lm(lcavol ~ lweight + age + lbph + svi + lcp + gleason + pgg45, prostate))
> plot(m, d, xlab="lcavol residuals", ylab="lpsa residuals", main="Partial Regression Plot")
> coef(lm(d~m))
  (Intercept)             m
-2.254521e-16  5.870218e-01
> coef(prostatelm)
 (Intercept)       lcavol     lweight         age        lbph         svi
 0.669336698  0.587021826  0.454467424 -0.019637176  0.107054031  0.766157326
         lcp      gleason        pgg45
-0.105474263  0.045141598  0.004525231
> abline(0, coef(prostatelm)['lcavol'])
> termplot(prostatelm, partial.resid=TRUE, terms=1, main="Partial Residual Plot")
```

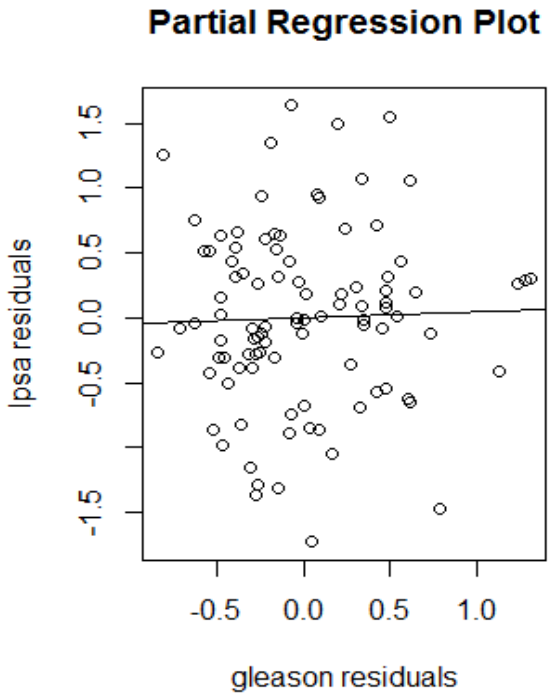


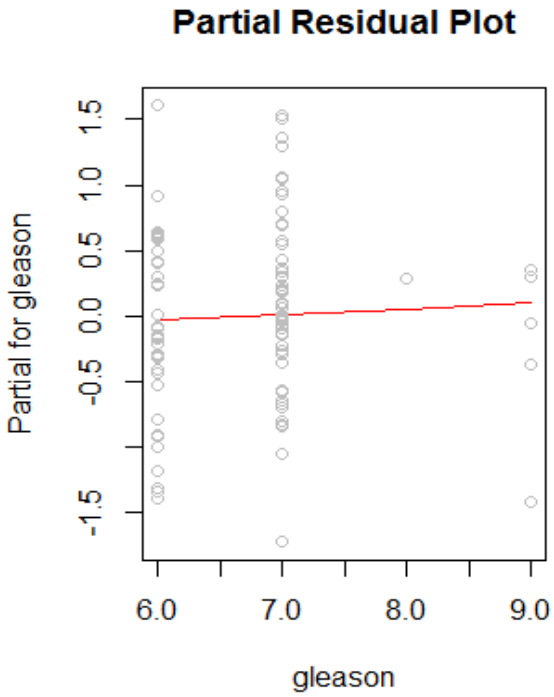
The partial regression plot above (the one of the right) shows a linear band with non-zero slope, indicating the addition of the lcavol variable might be helpful in a regression model that already contains the other variables. The partial residual plot seems to indicate that as the lcavol variable changes from

-1 to 4 that its contribution in the overall model changes. If the cluster had looked flat, it would seem that variable (lcavol) didn't matter much, and we could probably remove it from the model in favor of a possible intercept term adjustment. All of this is consistent with the fact that the p-value for the lcavol variable is extremely small when running the full model: it's 2.111 parts in a billion!

We should probably make added variable and partial residual plots for each of the eight variables, but I will just do it for one more- and choose one with a high p-value to illustrate the contrast. So lets' choose the gleason variable:

```
> windows()
> par(mfrow=c(1,2))
> d <- residuals(lm(lpsa ~ lcavol + lweight + age + lbph + svi + lcp + pgg45, prostate))
> m <- residuals(lm(gleason ~ lcavol + lweight + age + lbph + svi + lcp + pgg45, prostate))
> plot(m, d, xlab="gleason residuals", ylab="lpsa residuals", main="Partial Regression Plot")
> coef(lm(d~m))
 (Intercept)           m
1.359074e-16 4.514160e-02
> coef(prostatelm)
 (Intercept)      lcavol     lweight         age        lbph         svi
 0.669336698  0.587021826  0.454467424 -0.019637176  0.107054031  0.766157326
         lcp     gleason       pgg45
-0.105474263  0.045141598  0.004525231
> abline(0, coef(prostatelm)['gleason'])
> termplot(prostatelm, partial.resid=TRUE, terms=7, main="Partial Residual Plot")
```





As suspected, the partial regression and partial residual plots are very flat. It does not seem (at least from these plots) that gleason is an important variable for our model, and it can likely be discarded (if the other seven variables are to be included).

4. Here is a description of the swiss dataset:

Swiss Fertility and Socioeconomic Indicators (1888) Data

Description

Standardized fertility measure and socio-economic indicators for each of 47 French-speaking provinces of Switzerland at about 1888.

Usage

```
swiss
```

Format

A data frame with 47 observations on 6 variables, *each* of which is in percent, i.e., in *[0, 100]*.

[,1] Fertility	*Ig*, 'common standardized fertility measure'
[,2] Agriculture	% of males involved in agriculture as occupation
[,3] Examination	% draftees receiving highest mark on army examination
[,4] Education	% education beyond primary school for draftees.
[,5] Catholic	% 'catholic' (as opposed to 'protestant').
[,6] Infant.Mortality	live births who live less than 1 year.

All variables but 'Fertility' give proportions of the population.

Details

(paraphrasing Mosteller and Tukey):

Switzerland, in 1888, was entering a period known as the *demographic transition*; i.e., its fertility was beginning to fall from the high level typical of underdeveloped countries.

The data collected are for 47 French-speaking "provinces" at about 1888.

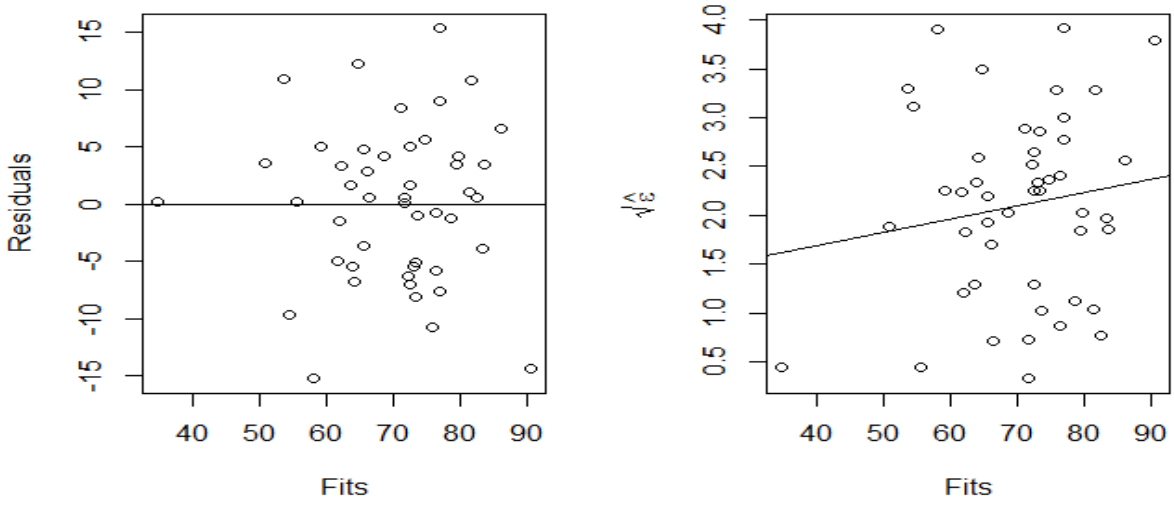
Here, all variables are scaled to *[0, 100]*, where in the original, all but `"Catholic"` were scaled to *[0, 1]*.

Here's a fit of the Fertility variable against the other variables:

```
> swisslm <- lm(Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality,
data=swiss)
> summary(swisslm)

Call:
lm(formula = Fertility ~ Agriculture + Examination + Education +
   Catholic + Infant.Mortality, data = swiss)

Residuals:
    Min      1Q  Median      3Q     Max
-15.2743  -5.2617   0.5032   4.1198  15.3213

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     66.91518   10.70604   6.250 1.91e-07 ***
Agriculture     -0.17211    0.07030  -2.448  0.01873 *
Examination     -0.25801    0.25388  -1.016  0.31546
Education       -0.87094    0.18303  -4.758 2.43e-05 ***
Catholic         0.10412    0.03526   2.953  0.00519 **
Infant.Mortality  1.07705    0.38172   2.822  0.00734 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.165 on 41 degrees of freedom
Multiple R-squared:  0.7067,    Adjusted R-squared:  0.671
F-statistic: 19.76 on 5 and 41 DF,  p-value: 5.594e-10
```

So it looks like all the variables seem significant except for "Examination". The intercept term also looks significant. The r^2 values are okay, at about 67% - 71%.

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

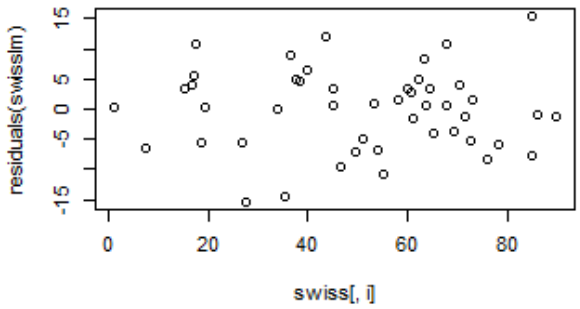


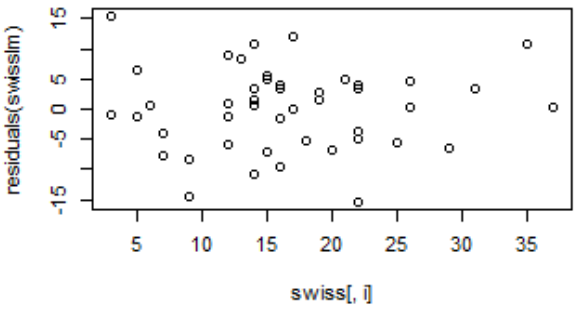

```
> par(mfrow=c(1,2))
> plot(fitted(swisslm), residuals(swisslm), xlab="Fits", ylab="Residuals")
> abline(h=0)
> plot(fitted(swisslm), sqrt(abs(residuals(swisslm))), xlab="Fits", ylab=expression(sqrt(hat(epsilon))))
> sumary(lm(sqrt(abs(residuals(swisslm))) ~ fitted(swisslm)))
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     1.150103   0.949926  1.2107   0.2323
fitted(swisslm) 0.013609   0.013397  1.0159   0.3151

n = 47, p = 2, Residual SE = 0.95417, R-Squared = 0.02
> flipped.out <- lm(sqrt(abs(residuals(swisslm))) ~ fitted(swisslm))
> abline(flipped.out)
> sumary(flipped.out)
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     1.150103   0.949926  1.2107   0.2323
fitted(swisslm) 0.013609   0.013397  1.0159   0.3151

n = 47, p = 2, Residual SE = 0.95417, R-Squared = 0.02
```

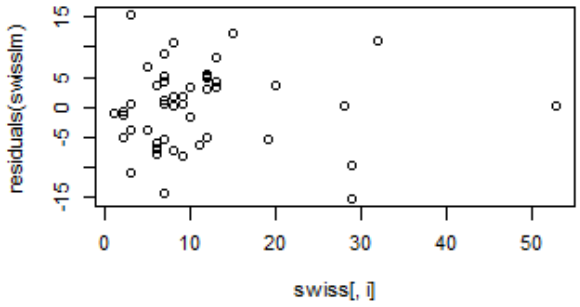
Due to the graphs I think there does not seem to be much reason to suspect the errors have non-constant variance. Note the line in the plot for sqrt|epsilon| seems to have a positive pitch. However, this is perhaps due to the scaling, and also if we removed the one point in the lower-left corner, it would likely look pretty flat.

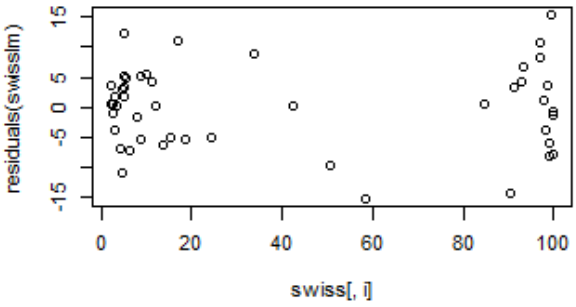
Next, I will plot the residuals versus each of the predictor variables, checking for marginal constant variance:

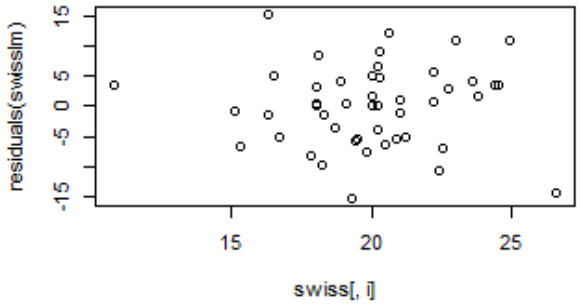
```
> windows()
> par(mfrow=c(3, 2))
> for(i in 2:6){plot(residuals(swisslm) ~ swiss[,i])}
```











So the main thing that catches my eye here is the point pattern in the residuals vs. the "Catholic" variable. I'm thinking there might be some heteroscedasticity here. I'll check further by running a an F-test on for equality of variance, partitioning the Catholic varable at <50 and > 50…:

```
> var.test(residuals(swisslm)[swiss$Catholic < 50], residuals(swisslm)[swiss$Catholic > 50])

        F test to compare two variances

data:  residuals(swisslm)[swiss$Catholic < 50] and residuals(swisslm)[swiss$Catholic > 50]
F = 0.443, num df = 28, denom df = 17, p-value = 0.05434
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.1758757 1.0154879
```

sample estimates:
ratio of variances
0.4429865

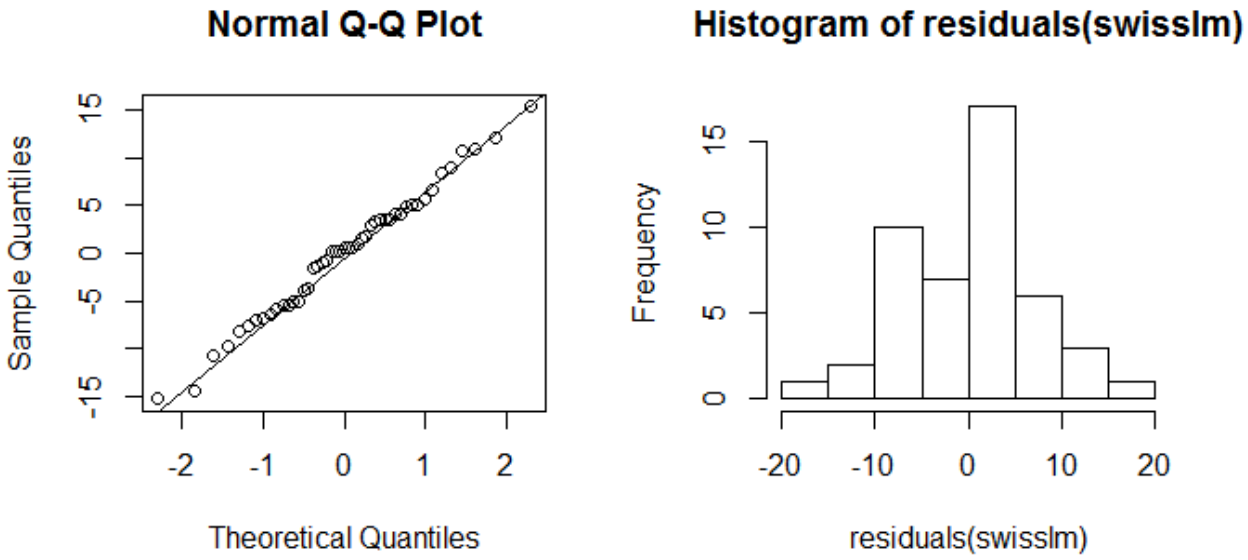
The p-value here is just over 5%, so it could be that a data transformation is in order (like Box-Cox or something).

(b) Check the normality assumption.

```
> windows()
> par(mfrow=c(1,2))
> qqnorm(residuals(swisslm, ylab="Residuals", main="Normal prob plot of resids"))
> qqline(residuals(swisslm))
> hist(residuals(swisslm))
> shapiro.test(residuals(swisslm))
```

Shapiro-Wilk normality test

data: residuals(swisslm)
W = 0.9889, p-value = 0.9318



Given the linearity of the qq-plot and that the p-value of the Shapiro-Wilks test is not small at all, I'd say the normality assumption is indeed satisfied. That is, there is insufficient evidence to reject the claim that the errors are normally distributed.

(c) Check for large leverage points.

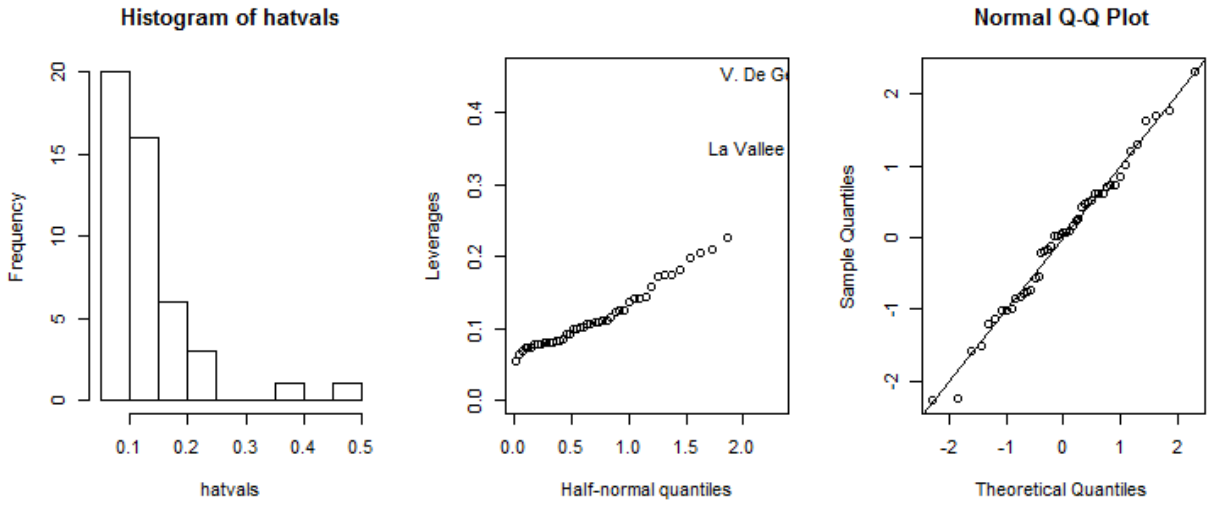
First I'll obtain the hat values, which are the diagonal elements of the hat matrix:

```
> hatvals <- hatvalues(swisslm)
> sum(hatvals)
[1] 6
> head(swiss)
             Fertility Agriculture Examination Education Catholic Infant.Mortality
Courtelary        80.2        17.0          15        12     9.96             22.2
Delemont          83.1        45.1           6         9    84.84             22.2
Franches-Mnt      92.5        39.7           5         5    93.40             20.2
Moutier           85.8        36.5          12         7    33.77             20.3
Neuveville        76.9        43.5          17        15     5.16             20.6
Porrentruy        76.1        35.3           9         7    90.57             26.6
```

I've also checked that their sum equals p = number of model parameters = 6.

I'll make a histogram of the hat values, and also plot the leverages (the hat values) against the half-normal quantiles, and make a qq-plot for the standardized residuals:

```
> windows()
> par(mfrow=c(1,3))
> hist(hatvals)
> row <- seq(1:nrow(swiss))
> swiss <- cbind(swiss, row)
> head(swiss)
             Fertility Agriculture Examination Education Catholic Infant.Mortality row
Courtelary        80.2        17.0          15        12     9.96             22.2   1
Delemont          83.1        45.1           6         9    84.84             22.2   2
Franches-Mnt      92.5        39.7           5         5    93.40             20.2   3
Moutier           85.8        36.5          12         7    33.77             20.3   4
Neuveville        76.9        43.5          17        15     5.16             20.6   5
Porrentruy        76.1        35.3           9         7    90.57             26.6   6
> halfnorm(hatvals, labs=row.names(swiss), ylab="Leverages")
> qqnorm(rstandard(swisslm))
> abline(0,1)
```



The histogram and the half-normal plots indicate the V. De Geneve (row 45) and La Vallee (row 19) observations might have some large leveraging. I'll remove them and redo the regression for the full model (with all the variables):

```
> rm4519 <- swiss[-c(45, 19),]
> rm4519lm <- lm(Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality, data=rm4519)
> summary(rm4519lm)

Call:
lm(formula = Fertility ~ Agriculture + Examination + Education +
    Catholic + Infant.Mortality, data = rm4519)

Residuals:
     Min       1Q   Median       3Q      Max
-15.0549  -5.0770   0.3589   4.8427  15.5572

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      63.83097   12.06915   5.289 5.02e-06 ***
Agriculture      -0.16094    0.07433  -2.165 0.036555 *
Examination      -0.29372    0.26555  -1.106 0.275473
Education        -0.85803    0.21980  -3.904 0.000365 ***
Catholic          0.09912    0.03690   2.686 0.010565 *
Infant.Mortality  1.22848    0.45958   2.673 0.010919 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.311 on 39 degrees of freedom
Multiple R-squared:   0.63,    Adjusted R-squared:  0.5826
```

```
F-statistic: 13.28 on 5 and 39 DF,  p-value: 1.408e-07
```

Comparing this with the output for the original model:

```
Call:
lm(formula = Fertility ~ Agriculture + Examination + Education +
    Catholic + Infant.Mortality, data = swiss)

Residuals:
    Min      1Q  Median      3Q     Max
-15.2743  -5.2617   0.5032   4.1198  15.3213

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      66.91518   10.70604   6.250 1.91e-07 ***
Agriculture      -0.17211    0.07030  -2.448  0.01873 *
Examination      -0.25801    0.25388  -1.016  0.31546
Education        -0.87094    0.18303  -4.758 2.43e-05 ***
Catholic          0.10412    0.03526   2.953  0.00519 **
Infant.Mortality  1.07705    0.38172   2.822  0.00734 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.165 on 41 degrees of freedom
Multiple R-squared:  0.7067,    Adjusted R-squared:  0.671
F-statistic: 19.76 on 5 and 41 DF,  p-value: 5.594e-10
```

It doesn’t look as if the regression coefficients have changed much, although it can be difficult to tell sometimes. In this case, all the predictor variables are percentages, so we can actually gauge this some. It doesn’t seem that they have changed substantially. The R^2 values have actually fallen, although this does not necessarily mean we should return those two points to the model. Remember leverage points don’t have to be influential: leverage points are just ones that fall far away from the rest of the points in terms of their predictor variable coordinates. So it could be that these leverage points lie pretty much in the path of the regression line, and far from the rest of the other data points, but that by removing them, the cloud of scattered points that remains seems relatively more disperse around the remaining model… Can you visualize this in a two-dimensional scario? More will be revealed...

(d) Check for outliers.

```
> stud <- rstudent(swisslm)
> stud[which.max(abs(stud))]
  Sierre
2.445227
```

The studentized residual with the largest magnitude is 2.445227. Comparing that to the critical value for a two-tailed family error rate of 5%

```
> qt(.05/nrow(swiss)/2, nrow(prostate)-6)
[1] -3.381552
```

it seems the largest studentized residual is really not that large in magnitude, and so there probably aren't any outliers.

(e) Check for influential points.

See part (c) above. I addressed this in part when checking for leverage points. Now let's compute the Cook's distances:

```
> cook <- cooks.distance(swisslm)
> par(mfrow=c(1,2))
> halfnorm(cook, 4, labs=row, ylab="Cook's Distance")
> swisslm.x <- lm(Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality,
swiss, subset=(cook< max(cook)))
> sumary(swisslm.x)
                 Estimate Std. Error t value  Pr(>|t|)
(Intercept)     65.455541  10.169984  6.4362 1.152e-07
Agriculture     -0.210343   0.068589 -3.0667  0.003871
Examination     -0.322776   0.242273 -1.3323  0.190308
Education       -0.895060   0.173843 -5.1487 7.364e-06
Catholic         0.112686   0.033626  3.3511  0.001767
Infant.Mortality  1.315665   0.375714  3.5018  0.001152

n = 46, p = 6, Residual SE = 6.79407, R-Squared = 0.74
```

Comparing with the original…

```
> sumary(swisslm)
```

```
                Estimate Std. Error t value  Pr(>|t|)
(Intercept)     66.915182  10.706038  6.2502 1.906e-07
Agriculture     -0.172114   0.070304 -2.4481  0.018727
Examination     -0.258008   0.253878 -1.0163  0.315462
Education       -0.870940   0.183029 -4.7585 2.431e-05
Catholic         0.104115   0.035258  2.9530  0.005190
Infant.Mortality  1.077048   0.381720  2.8216  0.007336

n = 47, p = 6, Residual SE = 7.16537, R-Squared = 0.71
```

It does seem the Infant Mortality rate coefficient has increased a bit (around 30%), and also the Examination coefficient has moved about 25% in magnitude. Also, R^2 has improved some. Perhaps we should set this point aside. Just for fun, I'll throw out the two points with the highest Cook's distance measurements:

```
> swisslm.xx <- lm(Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality,
swiss, subset=(cook< .14))
> sumary(swisslm.xx)
                Estimate Std. Error t value  Pr(>|t|)
(Intercept)     60.596154   9.657196  6.2747 2.152e-07
Agriculture     -0.216876   0.063993 -3.3891  0.001616
Examination     -0.251373   0.227472 -1.1051  0.275900
Education       -0.910972   0.162185 -5.6169 1.763e-06
Catholic         0.105816   0.031457  3.3639  0.001734
Infant.Mortality  1.520781   0.358730  4.2393  0.000133

n = 45, p = 6, Residual SE = 6.33410, R-Squared = 0.76
```

Comparing with the original, you can see the intercept has changed, and the Infant Mortality coefficient has increased in magnitude aby roughly 50%. Also, the new R^2 value is the highest we've seen. The Examination variable does not seem significant, and perhaps we can remove it from our model.

```
> windows()
> par(mfrow=c(3,2))
> halfnorm(cook, 3, labs=row, ylab="Cook's Distance")
> plot(dfbeta(swisslm)[,2],ylab="Change in Ag Coef")
> abline(h=0)
> identify(seq(1:nrow(swiss)), dfbeta(swisslm)[,2], labels=row.names(swiss))
[1] 1 3 4 6
> plot(dfbeta(swisslm)[,3],ylab="Change in Exam Coef")
> abline(h=0)
> identify(seq(1:nrow(swiss)), dfbeta(swisslm)[,3], labels=row.names(swiss))
[1]  5 37 40 42 46
> plot(dfbeta(swisslm)[,4],ylab="Change in Education")
> abline(h=0)
> identify(seq(1:nrow(swiss)), dfbeta(swisslm)[,4], labels=row.names(swiss))
```

```
[1]  5 40 42 46 47
> plot(dfbeta(swisslm)[,5],ylab="Change in Catholic Coef")
> abline(h=0)
> identify(seq(1:nrow(swiss)), dfbeta(swisslm)[,5], labels=row.names(swiss))
[1]  5  6  8 22 46
> plot(dfbeta(swisslm)[,6],ylab="Change in Inf. Mort. Rate")
> abline(h=0)
> identify(seq(1:nrow(swiss)), dfbeta(swisslm)[,6], labels=row.names(swiss))
[1]  6  8 19 37 42
```

The observations in rows 5, 6, 37,40, and 42 seem to be causing noticeable changes in several of the betas when left out, and their Cook's distances are also large compared to the rest. Perhaps these points are influential. I'll try leaving them out of the model and see what happens:

```
> rm5.6.37.40.42 <- swiss[-c(5, 6, 37, 40, 42),]
> rm5.6.37.40.42lm <- lm(Fertility ~ Agriculture + Examination + Education + Catholic +
Infant.Mortality, data= rm5.6.37.40.42)
> summary(rm5.6.37.40.42lm)

Call:
lm(formula = Fertility ~ Agriculture + Examination + Education +
   Catholic + Infant.Mortality, data = rm5.6.37.40.42)

Residuals:
   Min     1Q  Median     3Q    Max
-13.885  -4.847   1.316   3.905   9.640

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     65.76400    9.32560   7.052 2.81e-08 ***
Agriculture     -0.25878    0.06212  -4.166 0.000185 ***
Examination     -0.15196    0.22730  -0.669 0.508032
Education       -1.08955    0.16354  -6.662 9.14e-08 ***
Catholic         0.12793    0.03038   4.210 0.000162 ***
Infant.Mortality  1.32472    0.34121   3.882 0.000424 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.826 on 36 degrees of freedom
Multiple R-squared:  0.8133,    Adjusted R-squared:  0.7873
F-statistic: 31.36 on 5 and 36 DF,  p-value: 3.61e-12
```

So the R^2 and adjusted R^2 values have improved, and the p-values for significance of the model coefficients are all really small except the one corresponding to Examination. I believe we could throw away these potential influential points and use the model above, but without the Examination variable. In fact, I'm going to run that now:

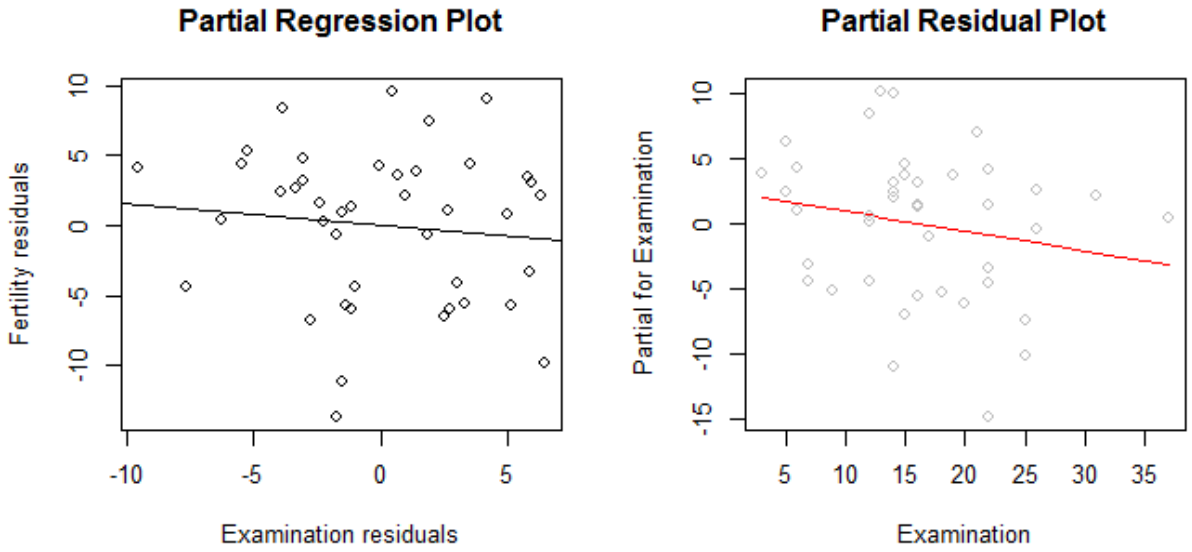
```
> a.good.model.lm <- lm(Fertility ~ Agriculture + Education + Catholic + Infant.Mortality, data=
rm5.6.37.40.42)
> summary(a.good.model.lm)

Call:
lm(formula = Fertility ~ Agriculture + Education + Catholic +
   Infant.Mortality, data = rm5.6.37.40.42)

Residuals:
   Min     1Q  Median     3Q    Max
-13.626  -4.404   1.143   3.773   9.572

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     63.00313    8.29875   7.592 4.68e-09 ***
Agriculture     -0.25256    0.06096  -4.143 0.000191 ***
Education       -1.15418    0.13092  -8.816 1.27e-10 ***
Catholic         0.13966    0.02462   5.673 1.74e-06 ***
Infant.Mortality  1.33388    0.33838   3.942 0.000346 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.782 on 37 degrees of freedom
Multiple R-squared:  0.811,     Adjusted R-squared:  0.7905
F-statistic: 39.68 on 4 and 37 DF,  p-value: 6.612e-13
```

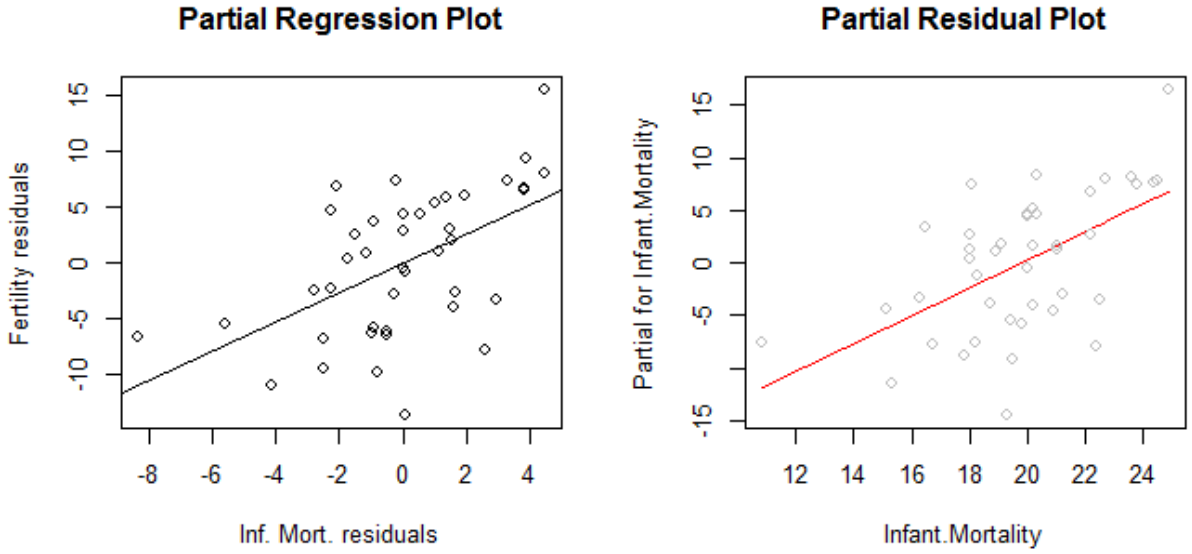
This model looks really good to me. Well done!

(f) Check the structure of the relationship between the predictors and the response.

Let's construct a partial regression/added variable plot, and also a partial residual plot for just a couple variables to illustrate… I'll do these on the model with all variables (including Examination), but without those five potential influential observations. Let's do them for the Examination variable, and then also for the Infant.Mortality variable.

```
> windows()
> par(mfrow=c(2,2))
> dEx <- residuals(lm(Fertility ~ Agriculture + Education + Catholic + Infant.Mortality, data=
rm5.6.37.40.42))
> mEx <- residuals(lm(Examination ~ Agriculture +  Education + Catholic + Infant.Mortality, data=
rm5.6.37.40.42))
> plot(mEx, dEx, xlab="Examination residuals", ylab="Fertility residuals", main="Partial Regression Plot")
> coef(lm(dEx~mEx))
  (Intercept)          mEx
 4.944528e-16 -1.519641e-01
> coef(rm5.6.37.40.42lm)
    (Intercept)     Agriculture     Examination       Education        Catholic Infant.Mortality
     65.7639979      -0.2587834      -0.1519641      -1.0895451       0.1279288       1.3247190
> abline(0, coef(rm5.6.37.40.42lm)['Examination'])
> termplot(rm5.6.37.40.42lm, partial.resid=TRUE, terms=2, main="Partial Residual Plot")
> dIM <- residuals(lm(Fertility ~ Agriculture + Examination + Education + Catholic, data= rm5.6.37.40.42))
> mIM <- residuals(lm(Infant.Mortality ~ Agriculture + Examination + Education + Catholic, data=
rm5.6.37.40.42))
> plot(mIM, dIM, xlab="Inf. Mort. residuals", ylab="Fertility residuals", main="Partial Regression Plot")
> coef(lm(dIM~mIM))
  (Intercept)          mIM
-7.332887e-16  1.324719e+00
> coef(rm5.6.37.40.42lm)
    (Intercept)     Agriculture     Examination       Education        Catholic Infant.Mortality
     65.7639979      -0.2587834      -0.1519641      -1.0895451       0.1279288       1.3247190
> abline(0, coef(rm5.6.37.40.42lm)['Infant.Mortality'])
> termplot(rm5.6.37.40.42lm, partial.resid=TRUE, terms=5, main="Partial Residual Plot")
```





These plots are consistent with what we suspected: that the examination variable is not adding anything to the model- that is, given the other variables in the model, adding it does not seem to help explain the variation in the response any better, and so if the other predictor variables are kept in the model, it can likely be discarded. On the other hand, the infant mortality rate variable seems to be significant.