STAT 210
Applied Statistics and Data Analysis
Multiple Linear Regression 4
Diagnostics

Joaquin Ortega

Fall 2020

Diagnostics

Diagnostics

While fitting a regression model using the principle of least squares requires no distributional assumptions, using the model for inferential purposes does depend on specific assumptions.

If the model assumes $\varepsilon \sim N(0, \sigma^2 \mathbf{I})$, that is, the errors in the model are assumed to be independent and to follow a normal distribution with a mean of zero and a constant variance, then we speak of a normal error model.

Regression diagnostics play a critical role verifying these assumptions and are also used to learn about unusual observations.

The diagnostics will often dictate changes in the model selected initially. These changes emphasize the fact that model building is an iterative process.

Diagnostics

We make the assumption of normality on an unobservable quantity ϵ. However, the residuals $\hat{\epsilon}_i$ can be computed and analyzed.

While the residuals do not have the same properties as the errors, the differences between residuals and errors are slight, and examining the residuals is a reasonable approach to checking assumptions about the model's errors.

The errors are assumed to follow a normal distribution. Simple techniques such as a quantile plot of the residuals can be used to study the residuals' distribution. However, care needs to be exercised when interpreting such graphs since plots of data from a normal distribution when the sample size is small will not always look normal.

Furthermore, as we have seen, the residuals do not have a constant variance.

Unusual Observations

Identifying Unusual Observations

Quite often, in regression models, certain observations do not seem to fit the overall pattern of the data.

These cases may have a large residual and have the potential to alter the fitted regression model radically.

An observation may be an outlier with respect to its y values, its x values, or both, yet not all outlying observations will have an important impact on the fitted regression model.

One of the ways used to measure outlying y values is to evaluate standardized residuals. This is done because residuals may have substantially different variances. What we obtain are the standardized residuals we described before.

Identifying Unusual Observations

A way to make the residuals more effective in detecting outlying observations is to use *deleted residuals*.

We fit a regression model excluding the ith case and denote by $\hat{y}_{i(i)}$ the predicted value we obtain corresponding to x_i. The deleted residual, $\hat{\epsilon}_{(i)}$, is then defined as

$$\hat{\epsilon}_{(i)} = y_i - \hat{y}_{i(i)}.$$

An algebraic equivalent expression for $\hat{\epsilon}_{(i)}$ exists that does not require the computation of $\hat{y}_{i(i)}$ for each omitted case. Specifically, it can be shown that

$$\hat{\epsilon}_{(i)} = y_i - \hat{y}_{i(i)} = \frac{\hat{\epsilon}_i}{1 - h_{ii}}.$$

Identifying Unusual Observations

The estimated variance is

$$Var(\hat{\epsilon}_{(i)}) = \frac{\hat{\sigma}^2_{(i)}}{1 - h_{ii}} = \frac{MSE_{(i)}}{1 - h_{ii}}.$$

The studentized deleted residual or **externally studentized residual** is defined as

$$r_i^* = \frac{\hat{\epsilon}_{(i)}}{\sqrt{Var(\hat{\epsilon}_{(i)})}} = \frac{\hat{\epsilon}_i}{\sqrt{MSE_{(i)}/(1 - h_{ii})}}.$$

Again, there is an algebraic equivalent that avoids doing n regressions. The algebraic equivalent definition of r_i^* is

$$r_i^* = r_i \Big(\frac{n - p - 1}{n - p - r_i^2} \Big)^{1/2} \sim t_{n-p-1}. \tag{1}$$

Bonferroni Correction

When the model is correct, each studentized deleted residual follows a t-distribution with $n - p - 1$ degrees of freedom. Even though it is very likely only a few 'large' r_i^*s will be of interest, by identifying them as large, all cases have implicitly been tested.

Assume we have m hypotheses $H_1, \ldots, H_m$ with corresponding p-values $p_1, \ldots, p_m$.

The familywise error rate (FWER) is the probability of rejecting at least one true hypothesis, i.e., making an error of Type I.

If all tests are at level α' then

$$FWER = P\Big(\bigcup_{i=1}^{m}\{p_i \leq \alpha'\}\Big|H_0\Big) \leq \sum_{i=1}^{m} P(p_i \leq \alpha'|H_0) = m\alpha'$$
$$> \alpha' \text{ if } m > 1$$

To have $FWER = \alpha$ select $\alpha' = \alpha/m$.

Identifying Unusual Observations

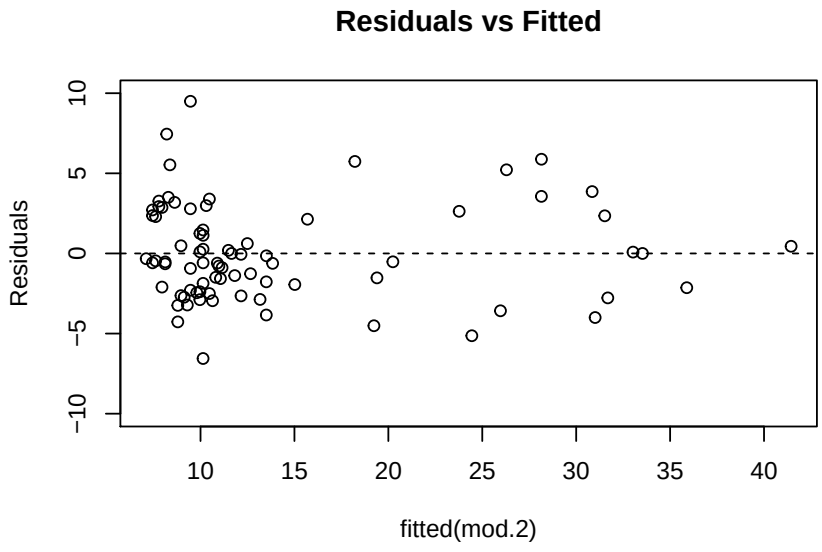
A Bonferroni approach is often used to control the overall significance level, where r_i^* values are declared significant if their absolute value exceeds $t_{1-\alpha/2n;n-p-1}$.

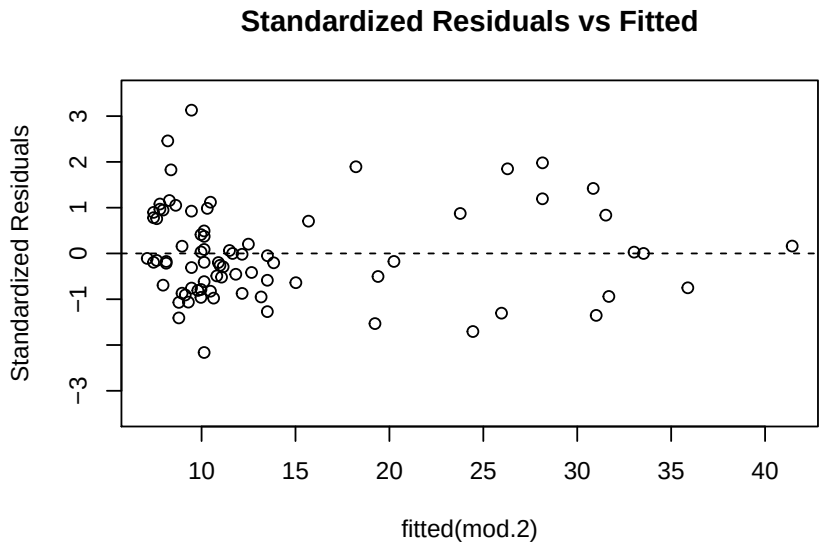
However, this approach does tend to be conservative, especially for large n.

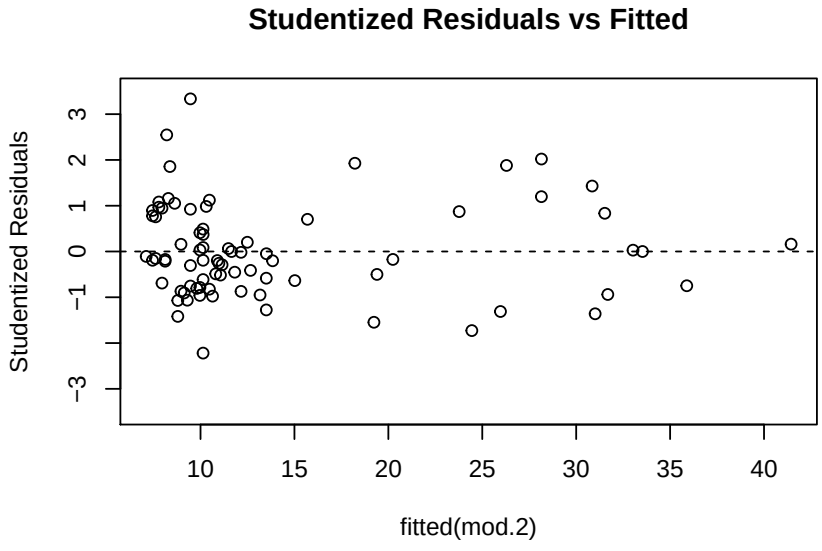
The `R` function `rstudent()` computes the studentized deleted residuals according to (1). The function `studres()` in the `MASS` package also computes these residuals.

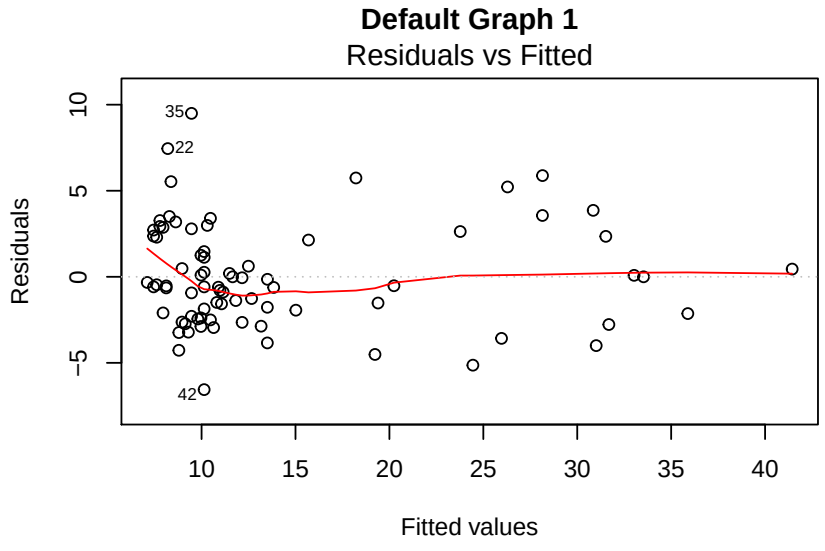
Let us use these functions to obtain and plot the different versions of residuals that we have considered for the model `HWFAT` $\sim$ `ABS + TRICEPS` using the `HSwrestler` data

Identifying Unusual Observations









Identifying Unusual Observations

```
sort(abs(resid(mod.2)))[76:78] # Extract three largest values
```

```
##       42       22       35
## 6.555825 7.449100 9.495697
```

```
sort(abs(stdres(mod.2)))[76:78] # Extract three largest values
```

```
##       42       22       35
## 2.163508 2.458597 3.129513
```

```
sort(abs(studres(mod.2)))[76:78] # Extract three largest values
```

```
##       42       22       35
## 2.219409 2.546944 3.333868
```

```
qt(1-.2/(2*78),78-3-1) # Critical value
```

```
## [1] 3.121816
```

```
detach(HSwrestler)
```

Cases 42, 22, and 35 have the largest absolute values of their plain residuals, standardized residuals, and studentized residuals. Case 35 could be considered an outlier using a significance level of $\alpha = 0.20$ since the critical value is 3.121816.

High Leverage Observations

While residuals were used to identify outlying y values, the hat matrix provides an analog for the x values.

The diagonal entry h_{ii} of the hat matrix $\mathbf{H}$ provides a measure of the distance of the ith case from the centroid of the x observations.

That is, h_{ii} can be used to assess whether an observation is outlying from the other x's.

The limits on h_{ii} are $1/n \leq h_{ii} \leq 1/c$, where c is the number of rows of X that have the same values as the ith row.

Note that the upper limit is never greater than 1.

High Leverage Observations

In general, a leverage value, h_{ii}, is considered large if it is more than twice as large as the mean leverage value $(2p/n)$.

Observations with large h_{ii} are called **high leverage points**, and each case should be investigated to see if the point estimates in the model under consideration change when the ith case is included versus excluded from the analysis.

It is important to note that not all points with high leverage will radically alter the estimation of parameters in the model. When the estimated parameters are substantially different with and without the ith case, the ith case is said to be **influential**.

Not all high leverage observations are influential.

Which cases are influential (if any) may change when the model is changed.

Tools

Cook's Distance

Cook's distance evaluates the influence of the ith case on all of the n fitted values.

It is a combined measure of the standardized residual (r_i) and the leverage value (h_{ii}) that produces a number used to assess the impact of removing the ith observation on all the regression coefficients (β).

Cook's D_i is defined as

$$D_i = \frac{1}{p\hat{\sigma}^2}\sum_{i=1}^{n}(\hat{y}_{i(i)} - \hat{y}_i)^2 = \frac{(\hat{\mathbf{y}}_{(i)} - \hat{\mathbf{y}})'(\hat{\mathbf{y}}_{(i)} - \hat{\mathbf{y}})}{p\hat{\sigma}^2}$$
$$= \frac{(\hat{\beta}_{(i)} - \hat{\beta})'(\mathbf{X}'\mathbf{X})(\hat{\beta}_{(i)} - \hat{\beta})}{p\hat{\sigma}^2}$$

or equivalently,

$$D_i = \frac{\hat{\epsilon}_i}{p\hat{\sigma}^2}\frac{h_{ii}}{(1-h_{ii})^2} = \frac{r_i^2}{p}\frac{h_{ii}}{1-h_{ii}}.$$

Cook's Distance

D_i values are generally flagged for further scrutiny when they exceed $F_{0.5;p,n-p}$; however, the exact distribution of D_i is unknown, and the use of $F_{0.5;p,n-p}$ is only a suggestion.

Often, a simple graph of the D_is will indicate values that require further scrutiny.

In `R`, `cooks.distance()` will compute the D_is. The package `car` also has the function `cookd()`.

DFFITS

A measure related to D_i is DFFITS, an abbreviation for 'difference in fits.' DFFITS is a standardized measure of the amount by which the predicted value $\hat{y}_i$ changes when the ith case is deleted from the data. The definition of DFFITS is

$$DFFITS_i = \frac{\hat{y}_i - \hat{y}_{(i)}}{\hat{\sigma}_{(i)}\sqrt{h_{ii}}}$$

while a computationally equivalent definition of DFFITS is

$$DFFITS_i = r_i^* \left(\frac{h_{ii}}{1 - h_{ii}}\right)^{1/2}$$

where r_i^* is the studentized deleted residual.

DFFITS values whose absolute value exceeds $2(p/n)^{1/2}$ generally require further scrutiny. To compute DFFITS with `R`, use `dffits(linearmodel)`.

DFBETAS

A standardized measure of the amount by which the kth regression coefficient changes when the ith observation is omitted from the data set is DFBETAS.

The DFBETAS measure is defined as

$$DFBETAS_{k(i)} = \frac{\hat{\beta}_k - \hat{\beta}_{k(i)}}{\sqrt{\hat{\sigma}^2_{(i)} v_{k+1,k+1}}}$$

where $v_{k+1,k+1}$ is the $(k+1)$th diagonal entry $(k = 0, 1, \ldots, p-1)$ of $\mathbf{V} = (\mathbf{X}'\mathbf{X})^{-1}$.

A case is considered to have a large DFBETAS value if its absolute value exceeds $2/\sqrt{n}$. To compute DFBETAS with `R`, use `dfbetas(linearmodel)`.

Example

Example

The data frame `Kinder` contains the height in inches and weight in pounds of 20 children from a kindergarten class.

We use all 20 observations to construct a regression model. The results are stored in the object `mod` by regressing height on weight.

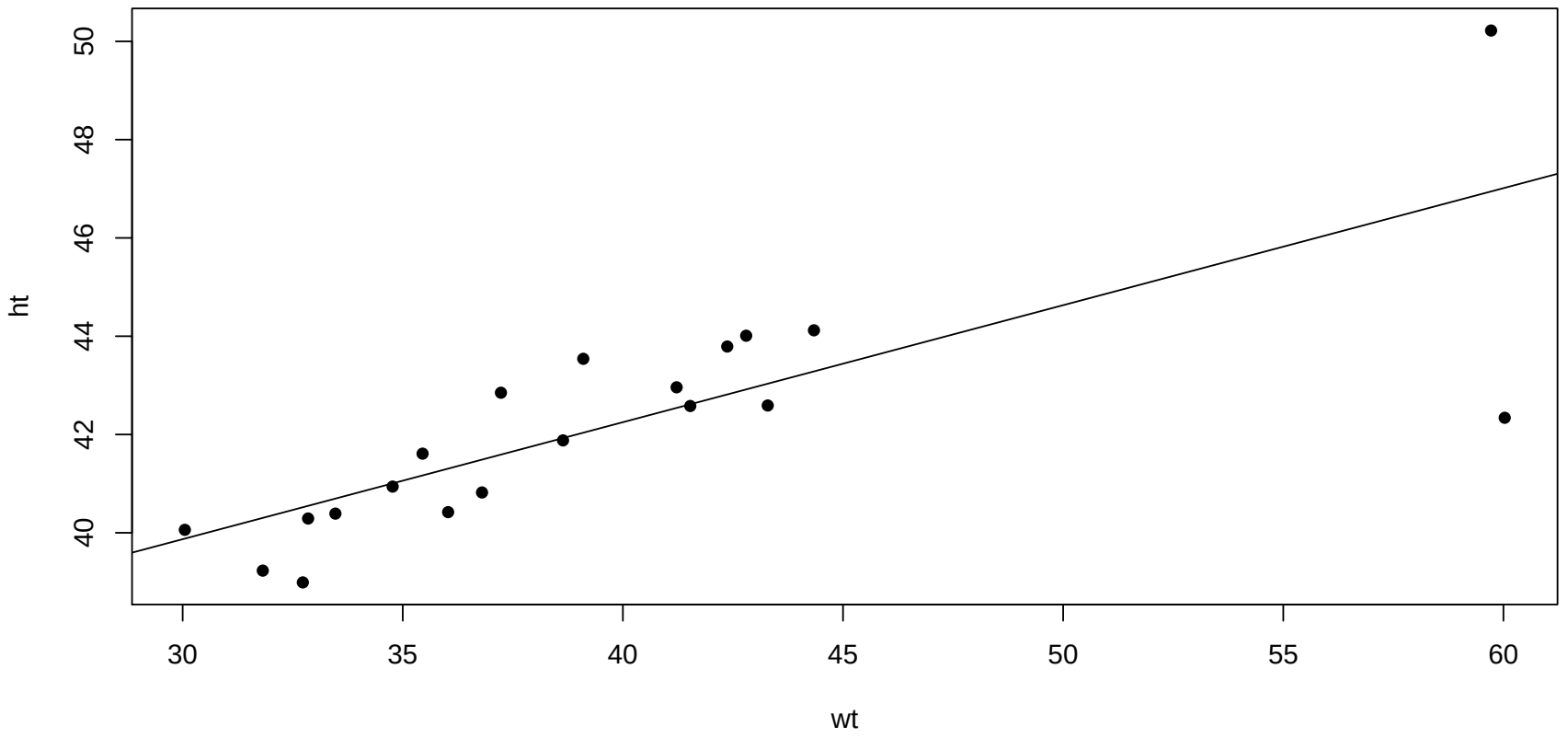
```
mod <- lm(ht ~ wt, data = Kinder)
```

We start by plotting the data

```
library(PASWR); attach(Kinder)
```

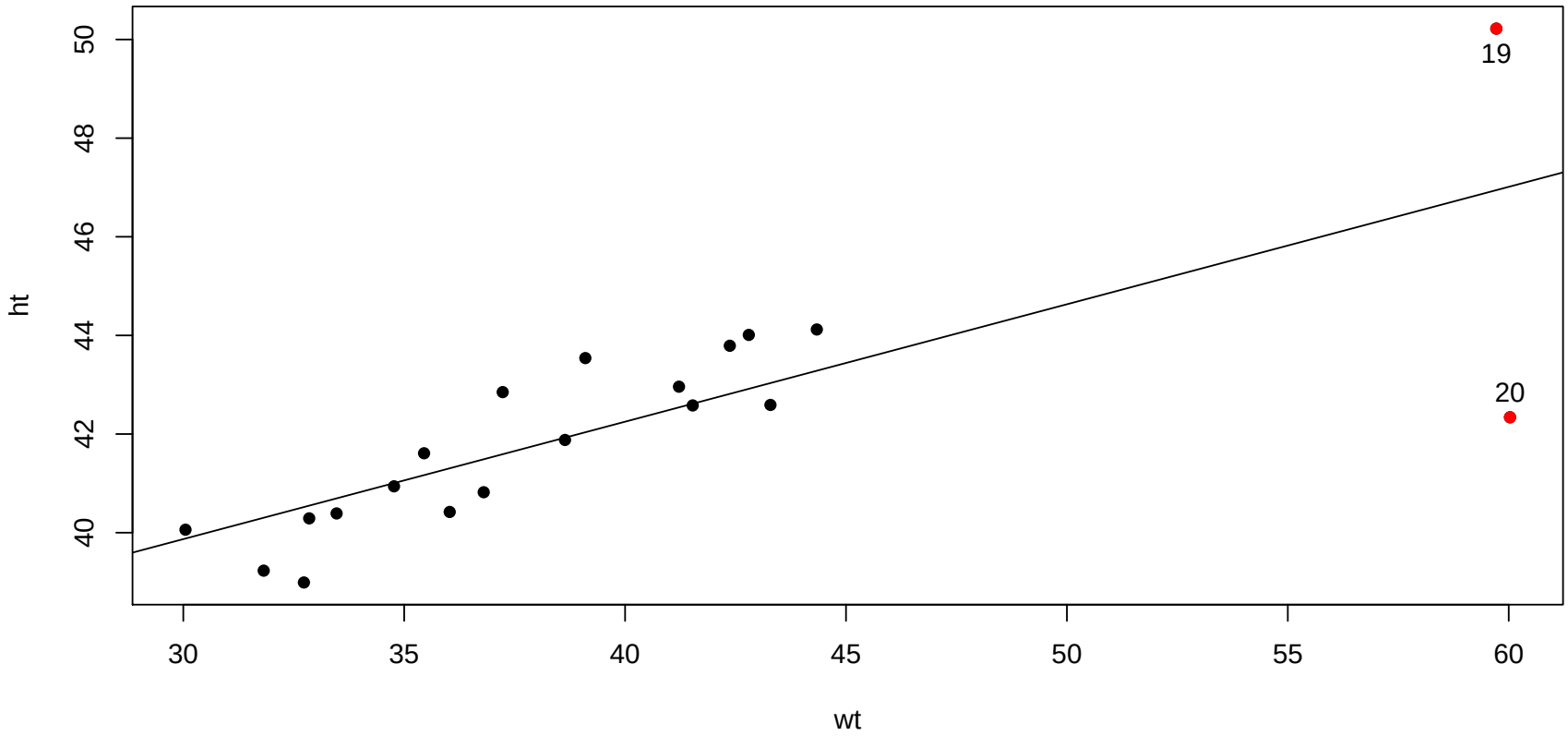
Example

```
plot(wt, ht, pch=16); abline(mod)
```



Example

```
plot(wt, ht, pch=16); points(wt[c(19,20)], ht[c(19,20)],
                             pch=16, col='red')
```



Example

A linear relation seems reasonable from the graphs, but two points require further scrutiny. We look at the hat matrix and the leverage values:

```
mod <- lm(ht~wt)
hii <- lm.influence(mod)$hat; round(hii,3)
```

```
##     1     2     3     4     5     6     7     8     9    10
## 0.067 0.090 0.126 0.082 0.052 0.070 0.053 0.065 0.060 0.061
##    11    12    13    14    15    16    17    18    19    20
## 0.058 0.050 0.055 0.056 0.051 0.057 0.101 0.088 0.375 0.385
```

Example

There are two large values, which correspond to points 19 and 20.

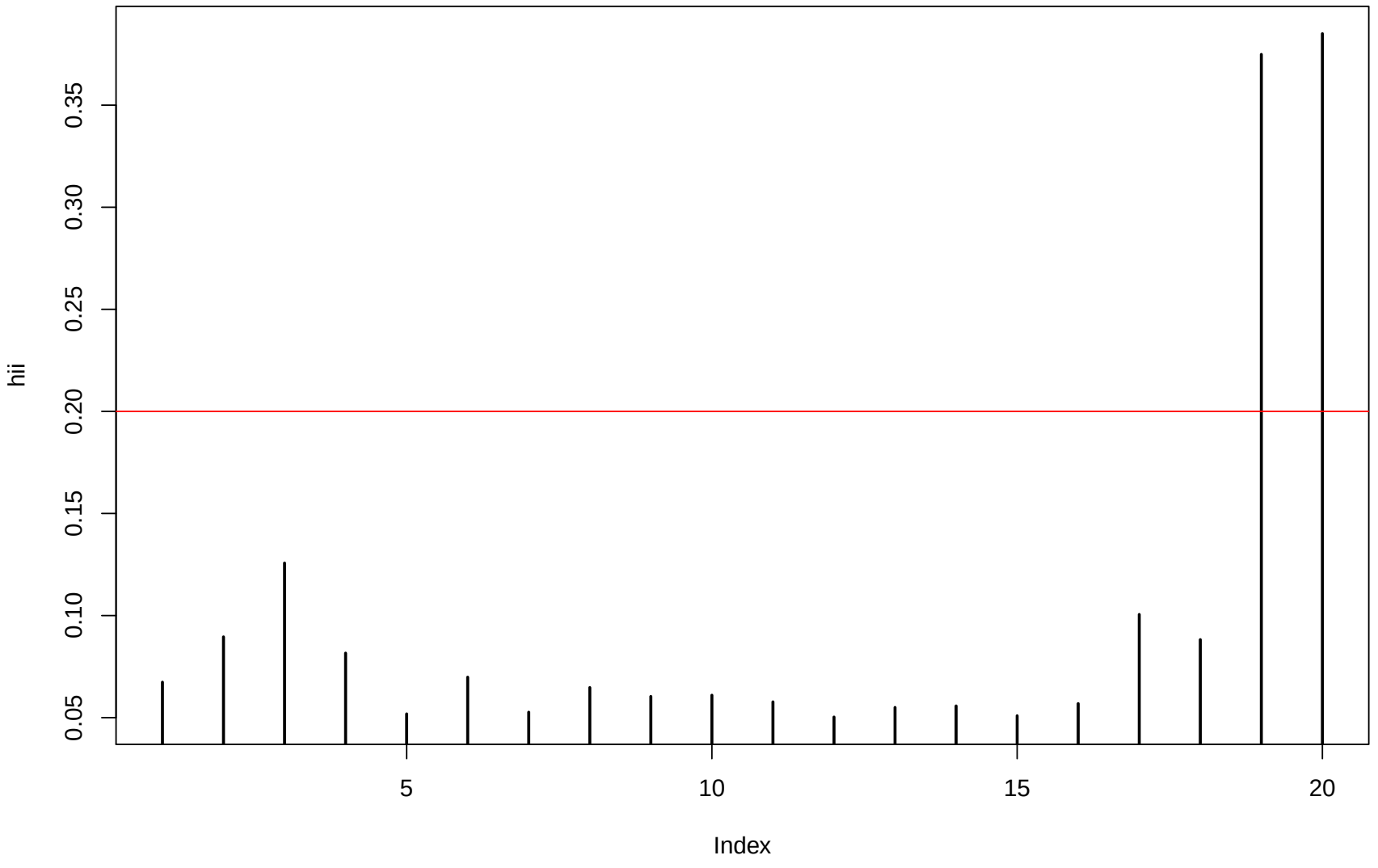
Child 19, although taller and heavier than the other children, seems to follow the linear trend of increased height with increased weight.

Child 20 appears to be right around the 50% percentile in height but has the largest weight (overweight child).

The following command produces a graph of these values and a horizontal line at $2p/n = 1/5$

Example

```
plot(hii, type = 'h', lwd=2)
abline(h=1/5, col='red')
```



Example

Observations 19 and 20 are removed from consideration, and `ht` is regressed on `wt` with the results stored in `modk`.

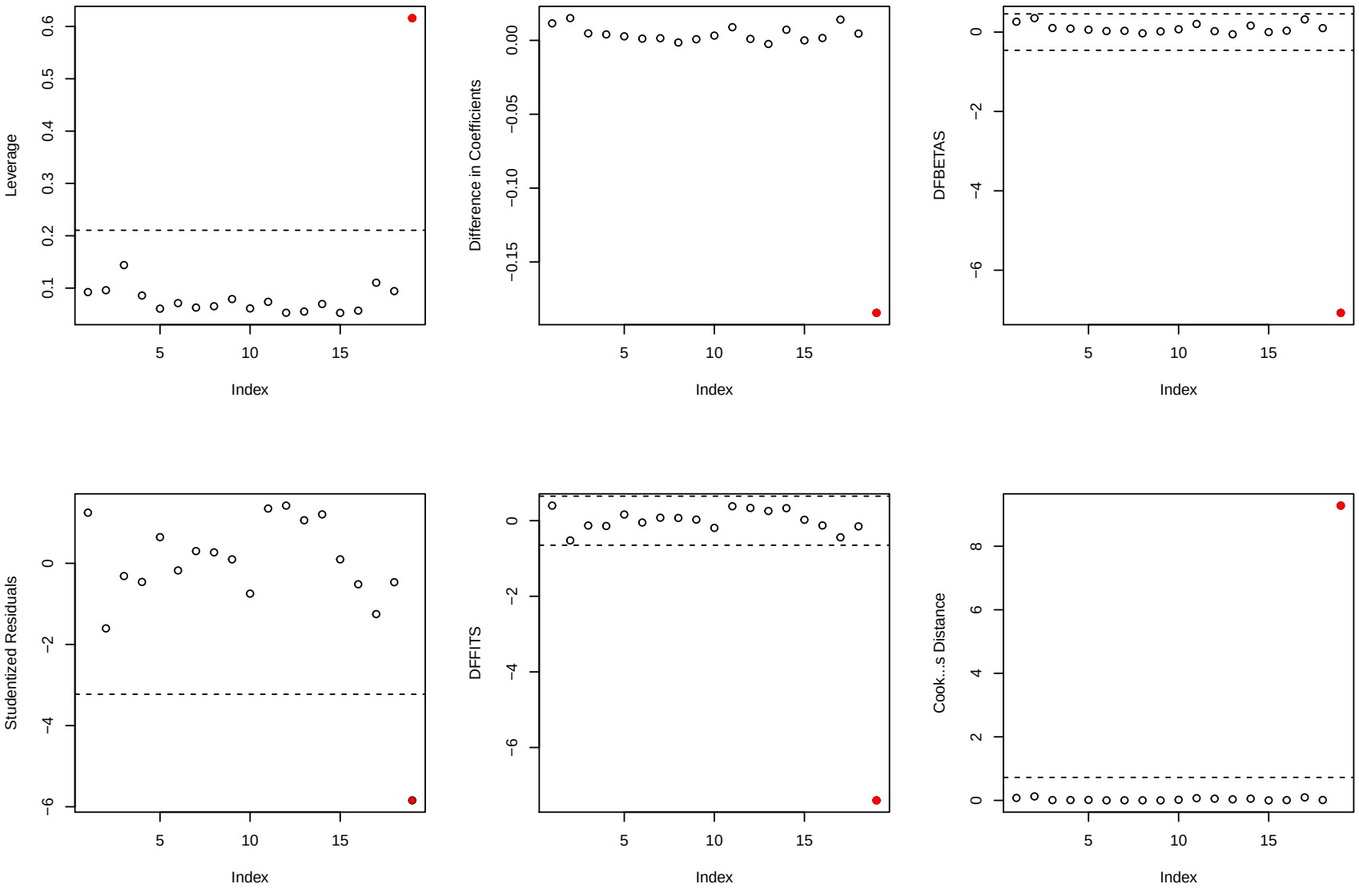
```
modk <- lm(ht[-c(19,20)]~wt[-c(19,20)])
```

We now consider a model without observation 19 and store the result in `modk19`. We want to explore if, under this new model, data point 20 is an influential observation. For this, we will consider Cook's D_i, DFFITS, and DFBETAS.

The 19th observation now (previously 20) corresponds to the 'overweight' child. From the diagnostics below, the overweight child is flagged in each graph for further scrutiny.

Note that `lm.influence(linear model)$coefficients` returns $\hat{\beta}_{k(i)} - \hat{\beta}_k$ in `R`. The `overweight` child is an observation with high leverage that is also influential.

Example

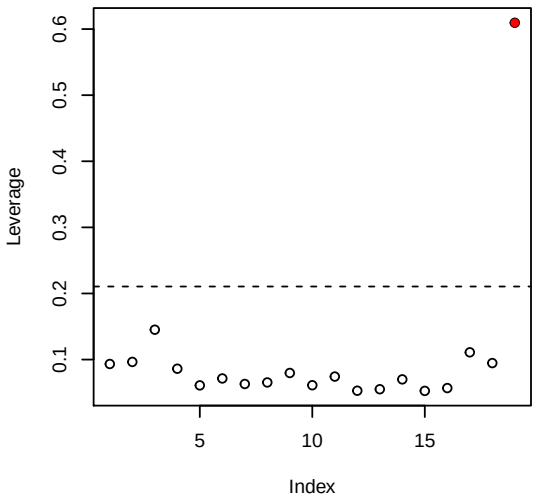

Leverage
0.1
0.2
0.3
0.4
0.5
0.6
5
10
15
Index
Difference in Coefficients
−0.15
−0.10
−0.05
0.00
5
10
15
Index
DFBETAS
−6
−4
−2
0
5
10
15
Index
Studentized Residuals
−6
−4
−2
0
5
10
15
Index
DFFITS
−6
−4
−2
0
5
10
15
Index
Cook...s Distance
0
2
4
6
8
5
10
15
Index

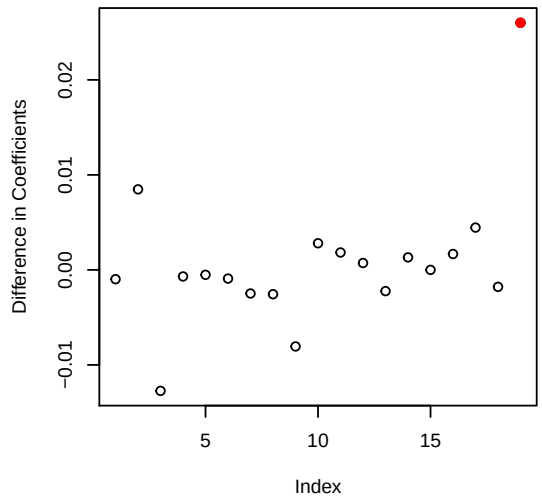
Example

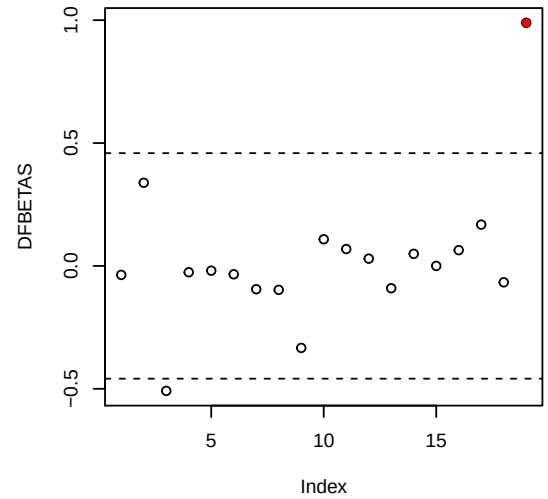
```
library(MASS) # Need for function studres()
library(car) # Need for function cookd
library(data.table)
modk19 <- lm(ht[-19]~wt[-19]); n <- 19; p <- 2
par(mfrow=c(2,3))
hiik19 <- lm.influence(modk19)$hat # extracting hii values
plot(hiik19, ylab="Leverage"); cv <- 2*p/n
points(19,hiik19[19],pch=19, col = 'red')
abline(h=cv, lty=2)
plot(lm.influence(modk19)$coefficients[,2],
     ylab="Difference in Coefficients")
points(19,lm.influence(modk19)$coefficients[19,2],pch=19, col = 'red')
plot(dfbetas(modk19)[,2], ylab="DFBETAS")
points(19,dfbetas(modk19)[19,2],pch=19, col = 'red')
cv <- 2/sqrt(n) # Critical value for DFBETAS
abline(h=c(-cv, cv), lty=2)
plot(studres(modk19), ylab="Studentized Residuals")
points(19,studres(modk19)[19], pch=16, col = 'red')
cv <- qt(1-.10/(2*n), n-p-1) # Critical value
abline(h=c(-cv, cv), lty=2)
DFFITS <- studres(modk19)*(hiik19/(1-hiik19))^.5 #See *
plot(DFFITS, ylab="DFFITS")
points(19,DFFITS[19],pch=19, col = 'red')
cv <- 2*sqrt(p/n) # Critical value for DFITS
abline(h=c(-cv, cv), lty=2)
cd <- cooks.distance(modk19) # Cook's distance
plot(cd, ylab="Cook's Distance")
points(19,cd[19],pch=19, col = 'red')
CF <- qf(.50, p, n-p) # Critical value for Cook's Distance
abline(h=CF, lty=2)
par(mfrow=c(1,1))
```

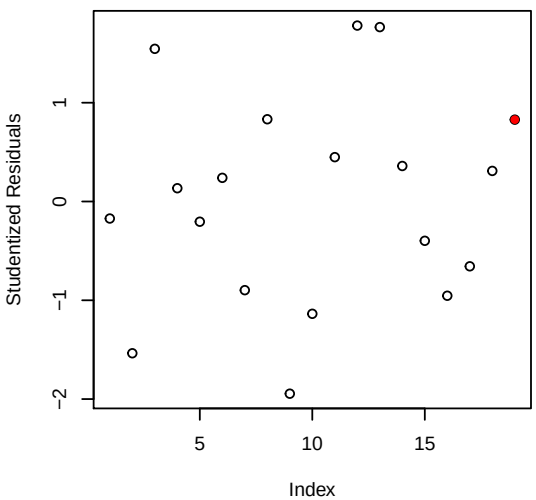
Example

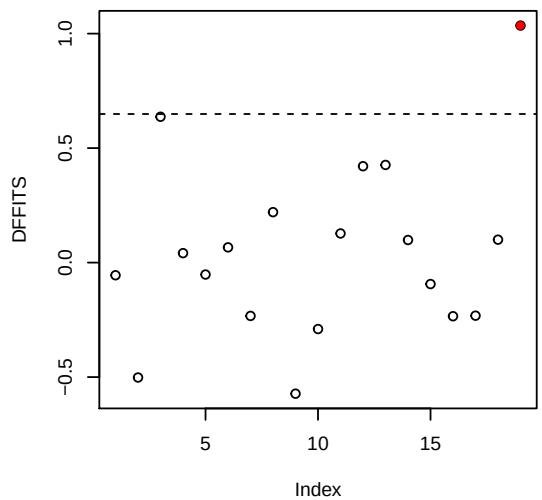
Next, we consider a model with all data points except observation 20 and do similar calculations and graphs as before.

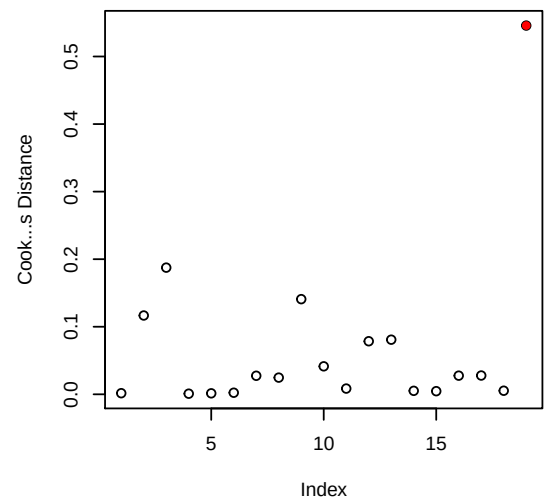










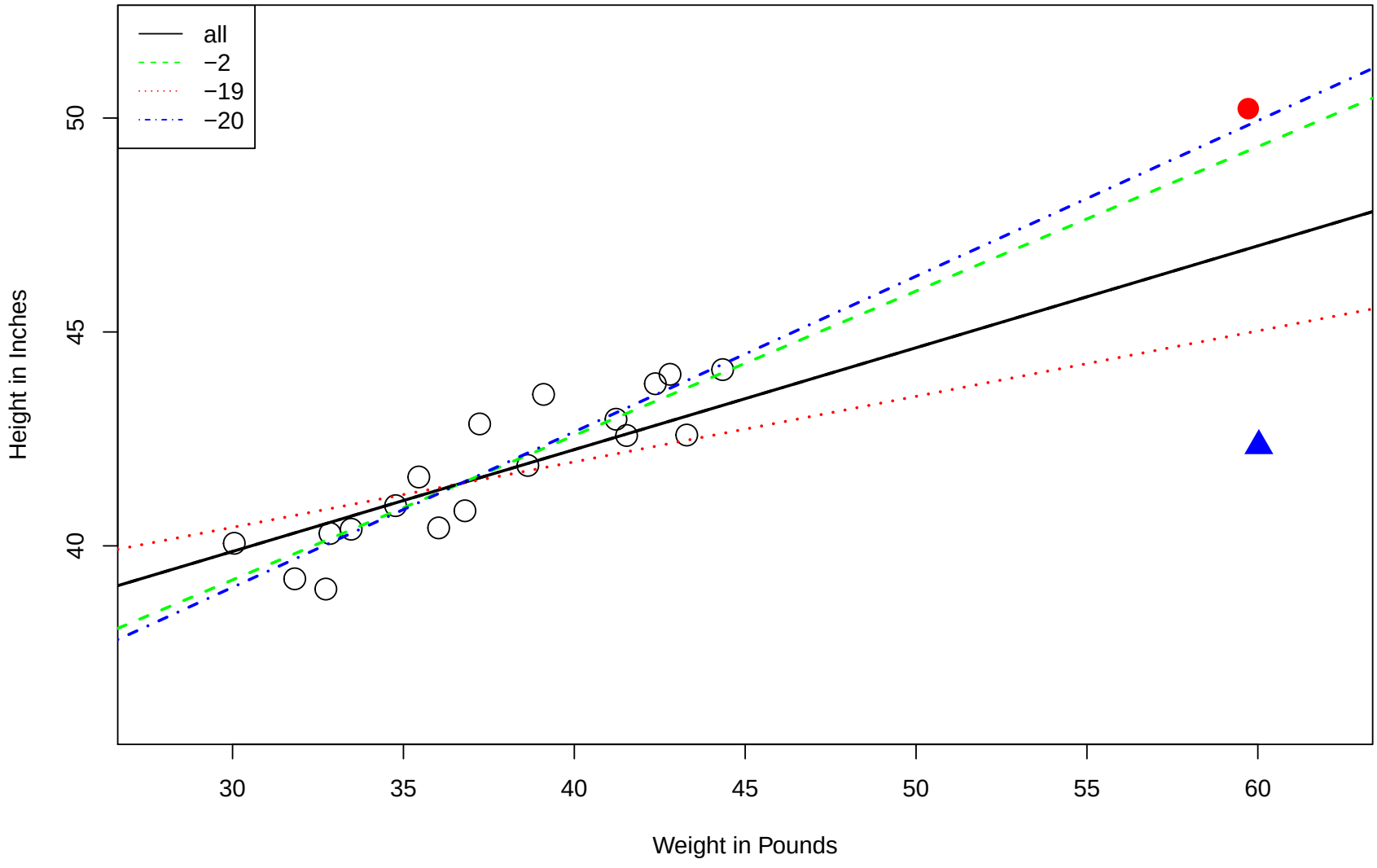


Example

```
modk20 <- lm(ht[-20]~wt[-20])
par(mfrow=c(2,3))
hiik20 <- lm.influence(modk20)$hat
plot(hiik20, ylab="Leverage")
points(19,hiik20[19],pch=16, col = 'red')
cv <- 2*p/n
abline(h=cv, lty=2)
plot(lm.influence(modk20)$coefficients[,2],
       ylab="Difference in Coefficients")
points(19,lm.influence(modk20)$coefficients[19,2],pch=19, col = 'red')
plot(dfbetas(modk20)[,2], ylab="DFBETAS")
points(19,dfbetas(modk20)[19,2],pch=16, col ='red')
cv <- 2/sqrt(n) # Critical value for DFBETAS
abline(h=c(-cv, cv), lty=2)
plot(studres(modk20), ylab="Studentized Residuals")
points(19,studres(modk20)[19], pch = 16, col = 'red')
cv <- qt(1-.10/(2*n), n-p-1) # Critical value
abline(h=c(-cv, cv), lty=2)
DFFITS <- studres(modk20)*(hiik20/(1-hiik20))^.5
plot(DFFITS, ylab="DFFITS")
points(19,DFFITS[19], pch = 16, col = 'red')
cv <- 2*sqrt(p/n) # Critical value for DFITS
abline(h=c(-cv, cv), lty=2)
cd <- cooks.distance(modk20) # Cook's distance
plot(cd, ylab="Cook's Distance")
points(19,cd[19], pch = 16, col = 'red')
CF <- qf(.50, p, n-p) # Critical value for Cook's Distance
abline(h=CF, lty=2)
par(mfrow=c(1,1))
```

Example

Finally, we plot the three regressions model we have fitted and compare the results.



Example

When all 20 cases are included in the regression, cases 19 (solid circle) and 20 (solid triangle) have large leverage values; however, if case 20 is omitted, case 19 still has a large leverage value, yet it is not very influential.

Consider the differences between the lines `modk20` (dot-dash, blue, case 20 omitted) and `modk` (dash, green, where cases 19 and 20 are omitted). There is very little difference between them.

On the other hand, if case 19 (solid circle) is omitted, the resulting regression `modk19` (dotted, red) is substantially different from modk. In other words, case 20 has high leverage and is influential when case 19 is omitted.

Other Diagnostic Tools

Other Diagnostic Tools

We now introduce other graphical diagnostic tools based on the `car` package Companion to Applied Regression) by Fox, Weisberg, and Price.

We will consider the data set `rat` from the `alr4` package. This data comes from 'an experiment in which rats were injected with a dose of a drug approximately proportional to body weight. At the end of the experiment, the animal's liver was weighed, and the fraction of the drug recovered in the liver was recorded. **The experimenter expected the response to be independent of the predictors.**'

Other Diagnostic Tools

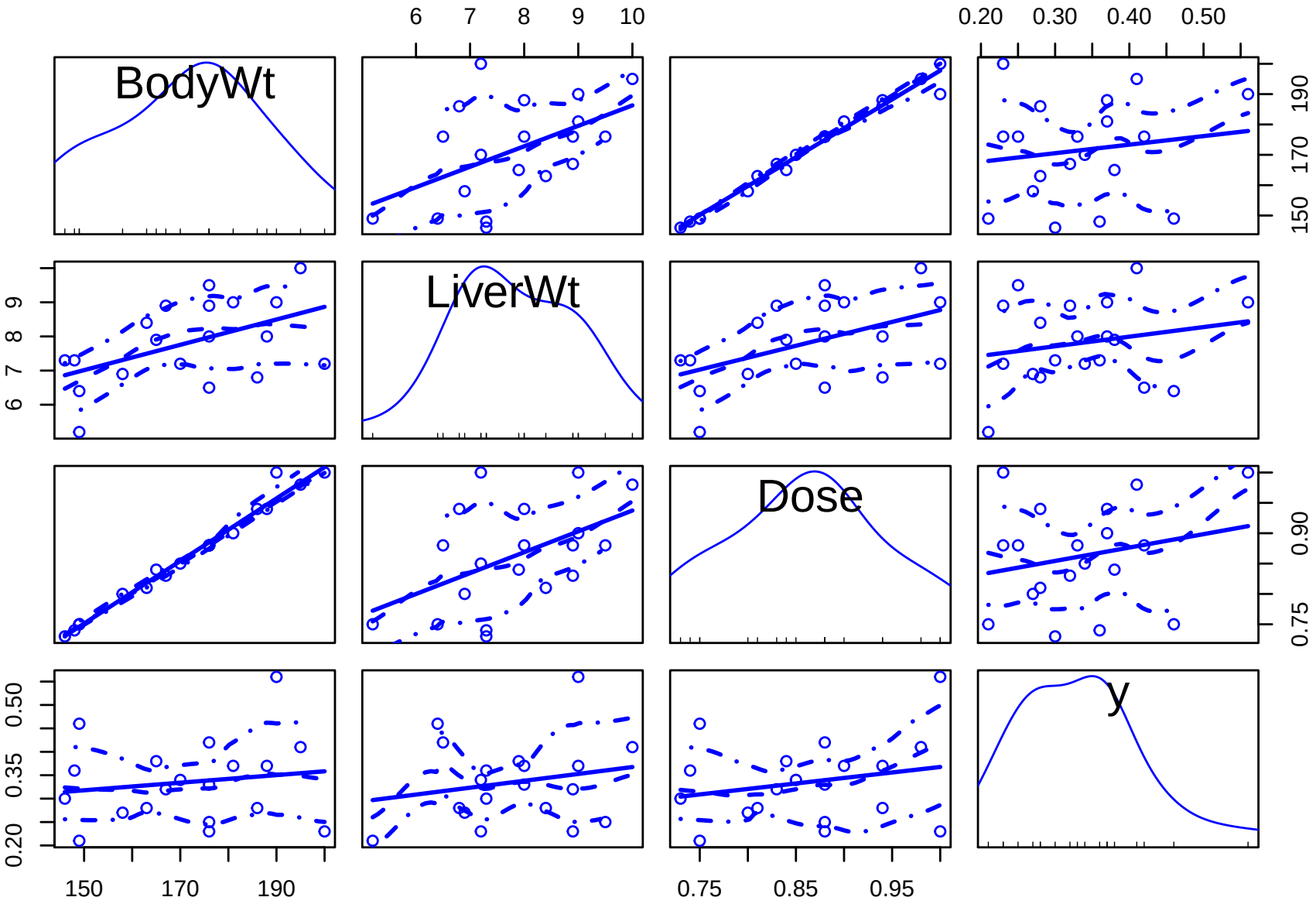
```
library(alr4)
library(car)
str(rat, vec.len = 2)
```

```
## 'data.frame':    19 obs. of  4 variables:
##  $ BodyWt : int  176 176 190 176 200 ...
##  $ LiverWt: num  6.5 9.5 9 8.9 7.2 ...
##  $ Dose   : num  0.88 0.88 1 0.88 1 ...
##  $ y      : num  0.42 0.25 0.56 0.23 0.23 ...
```

Start with a scatterplot matrix of `rat` using the `car` function `scatterplotMatrix`

Other Diagnostic Tools

```
scatterplotMatrix(rat)
```



Other Diagnostic Tools

From S. Weisberg **Applied Linear Regression**

'*An experiment was conducted to investigate the amount of a particular drug present in the liver of a rat. Nineteen rats were randomly selected, weighed, placed under light ether anesthesia and given an oral dose of the drug. Because large livers would absorb more of a given dose than smaller livers, the actual dose an animal received was approximately determined as 40 mg of the drug per kilogram of body weight. Liver weight is known to be strongly related to body weight.*'

Other Diagnostic Tools

Fit a full model

```
m1 <- lm(y ~ BodyWt + LiverWt + Dose, rat)
summary(m1)
```

```
##
## Call:
## lm(formula = y ~ BodyWt + LiverWt + Dose, data = rat)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.100557 -0.063233  0.007131  0.045971  0.134691
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.265922   0.194585   1.367   0.1919
## BodyWt      -0.021246   0.007974  -2.664   0.0177 *
## LiverWt      0.014298   0.017217   0.830   0.4193
## Dose         4.178111   1.522625   2.744   0.0151 *
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07729 on 15 degrees of freedom
## Multiple R-squared:  0.3639, Adjusted R-squared:  0.2367
## F-statistic:  2.86 on 3 and 15 DF,  p-value: 0.07197
```

Other Diagnostic Tools: predictorEffects

Drop LiverWt from the model

```
m2 <- update(m1, ~ . - LiverWt)
summary(m2)
```

```
##
## Call:
## lm(formula = y ~ BodyWt + Dose, data = rat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.12333 -0.07416  0.01238  0.04884  0.12668
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.285517   0.191267   1.493   0.1550
## BodyWt      -0.020444   0.007838  -2.608   0.0190 *
## Dose         4.125330   1.506472   2.738   0.0146 *
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07654 on 16 degrees of freedom
## Multiple R-squared:  0.3347, Adjusted R-squared:  0.2515
## F-statistic: 4.024 on 2 and 16 DF,  p-value: 0.0384
```

Other Diagnostic Tools

Going back to the scatterplot matrix, we see that `Dose` and `BodyWt` are almost perfectly aligned, so they carry the same information.

Let's try to drop one of them from the model

```
m3 <- update(m2, ~ . - Dose, rat)
S(m3)
```

```
## Call: lm(formula = y ~ BodyWt, data = rat)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.1962346  0.2215825   0.886    0.388
## BodyWt      0.0008105  0.0012862   0.630    0.537
##
## Residual standard deviation: 0.08999 on 17 degrees of freedom
## Multiple R-squared: 0.02283
## F-statistic: 0.3971 on 1 and 17 DF,  p-value: 0.537
##    AIC    BIC
## -33.70 -30.87
```

Other Diagnostic Tools

```
m4 <- update(m2, ~ . - BodyWt, rat)
S(m4)
```

```
## Call: lm(formula = y ~ Dose, data = rat)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.1330     0.2109   0.631    0.537
## Dose          0.2346     0.2435   0.963    0.349
##
## Residual standard deviation: 0.08864 on 17 degrees of freedom
## Multiple R-squared: 0.05178
## F-statistic: 0.9283 on 1 and 17 DF,  p-value: 0.3488
##    AIC    BIC
## -34.27 -31.44
```

Other Diagnostic Tools: Influence plots

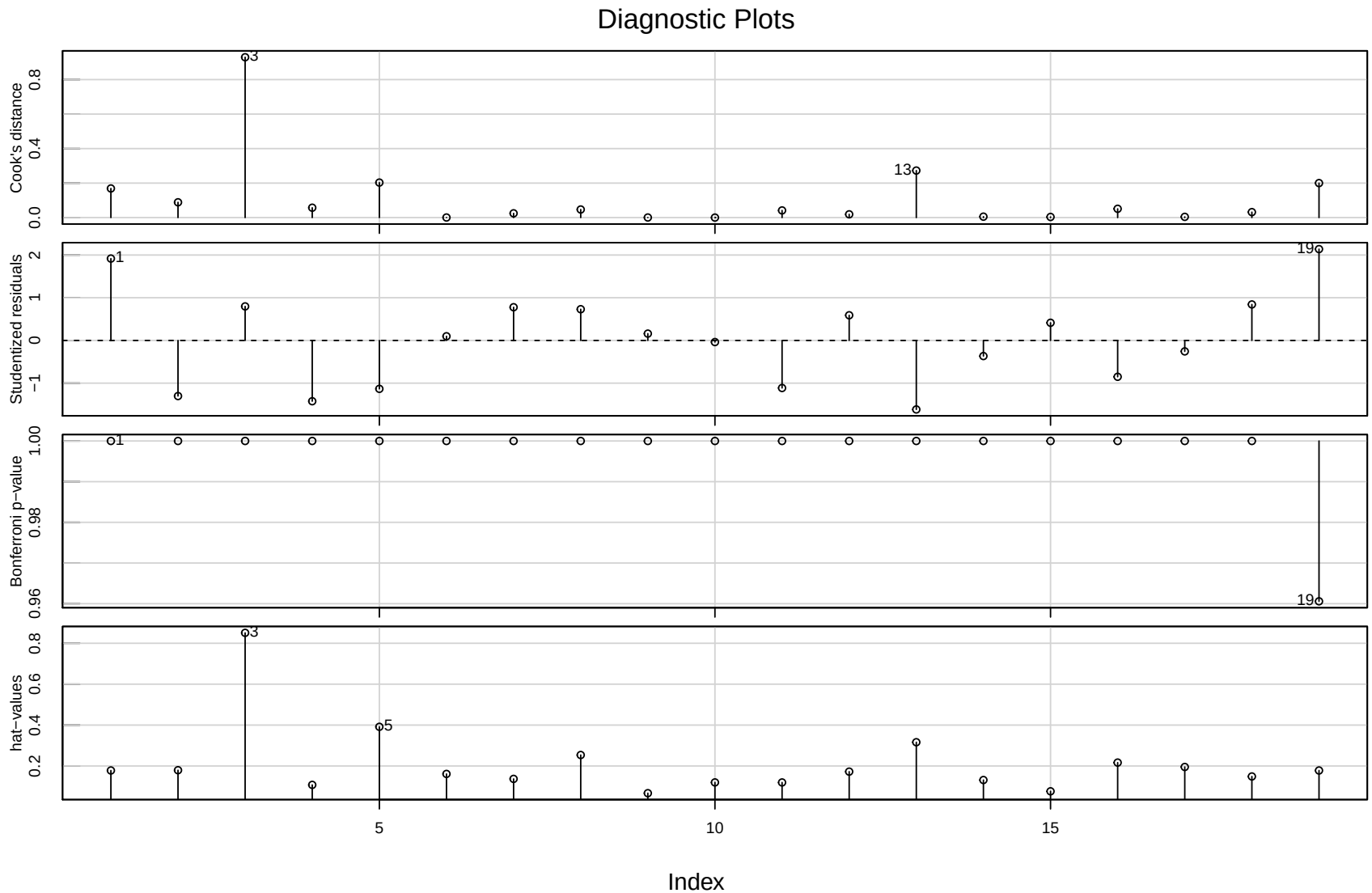
It seems that neither `Dose` nor `BodyWt` on its own is associated with the response, but when they are together, the story is different.

The function `influenceIndexPlot` plots

- Cook's distance,
- Studentized residuals,
- Bonferroni significance levels to testing each observation in turn to be an outlier, and
- Leverage values, or a subset of these, versus observation number

Other Diagnostic Tools: Influence plots

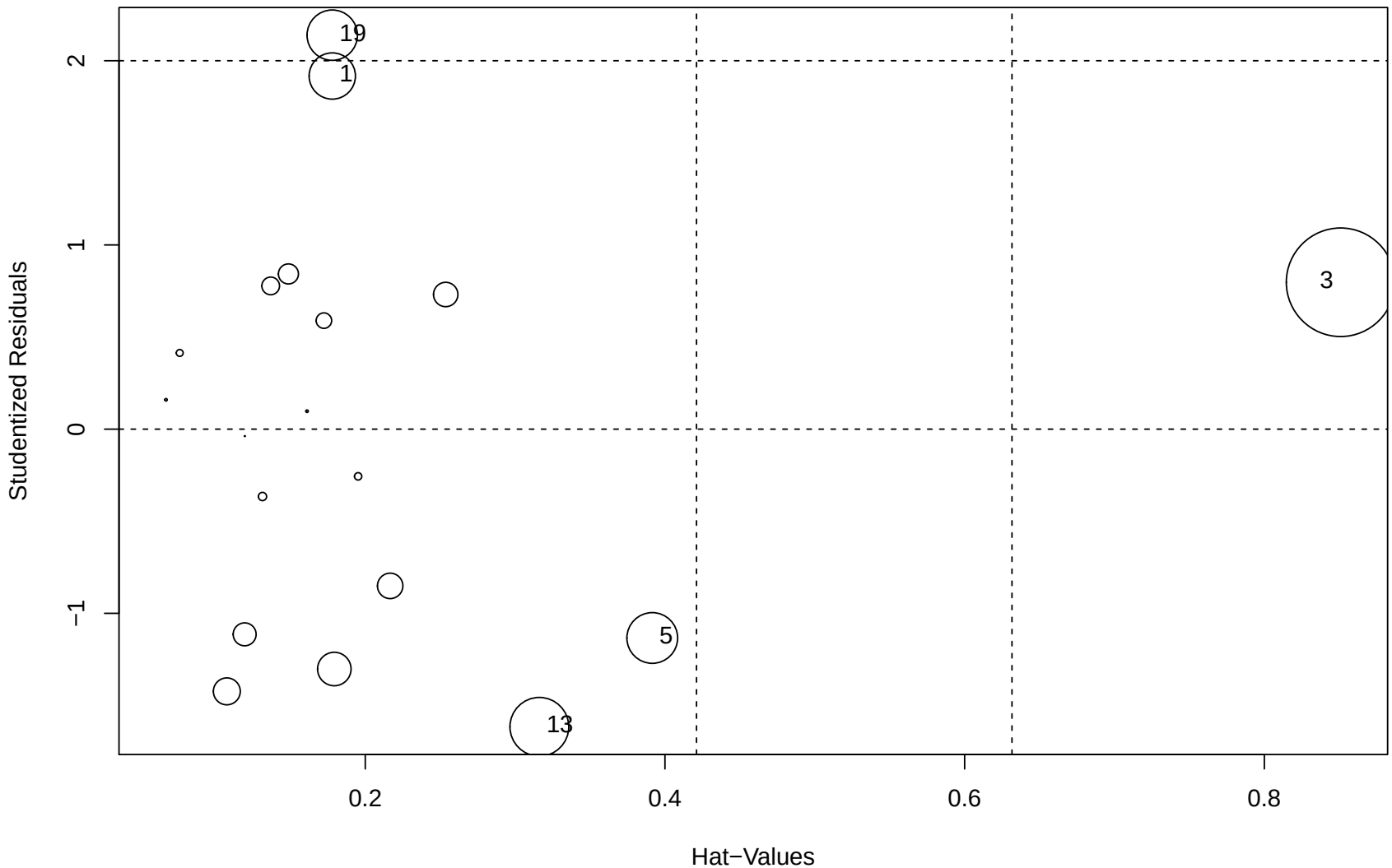
```
influenceIndexPlot(m1)
```



Other Diagnostic Tools: Influence plots

The function `influencePlot` graphs Studentized residuals against leverage values, and includes Cook's distance as the radius of circles.

```
influencePlot(m1)
```



Other Diagnostic Tools: Influence plots

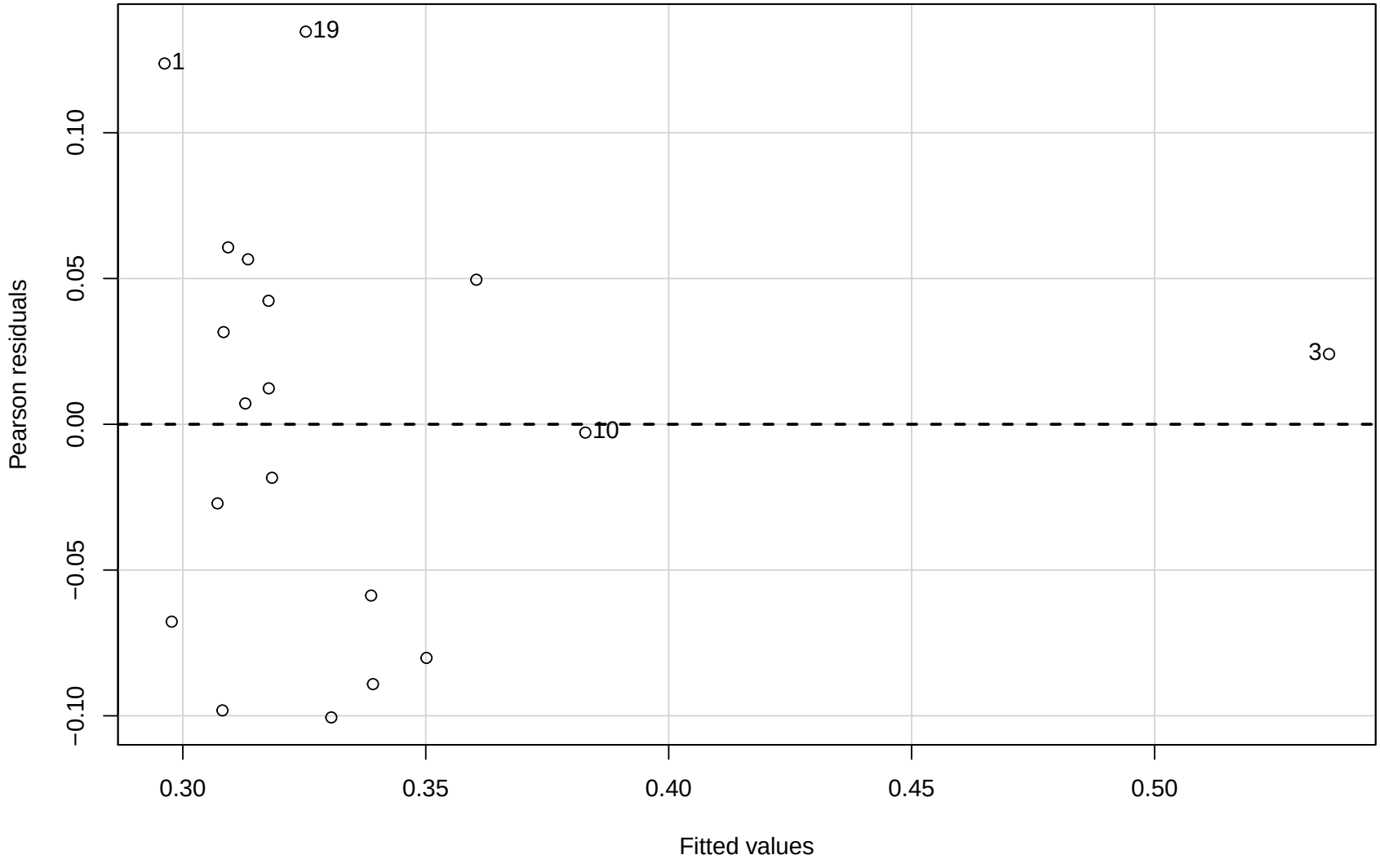
These plots point to case three, which has a large value for Cook's distance and high leverage.

In the next slide, we use `residualPlot()` from the package `car` that graphs residuals against fitted values.

There is also the command `residualPlots()` that graphs the usual residual plots with some extra features that will be described later.

Other Diagnostic Tools

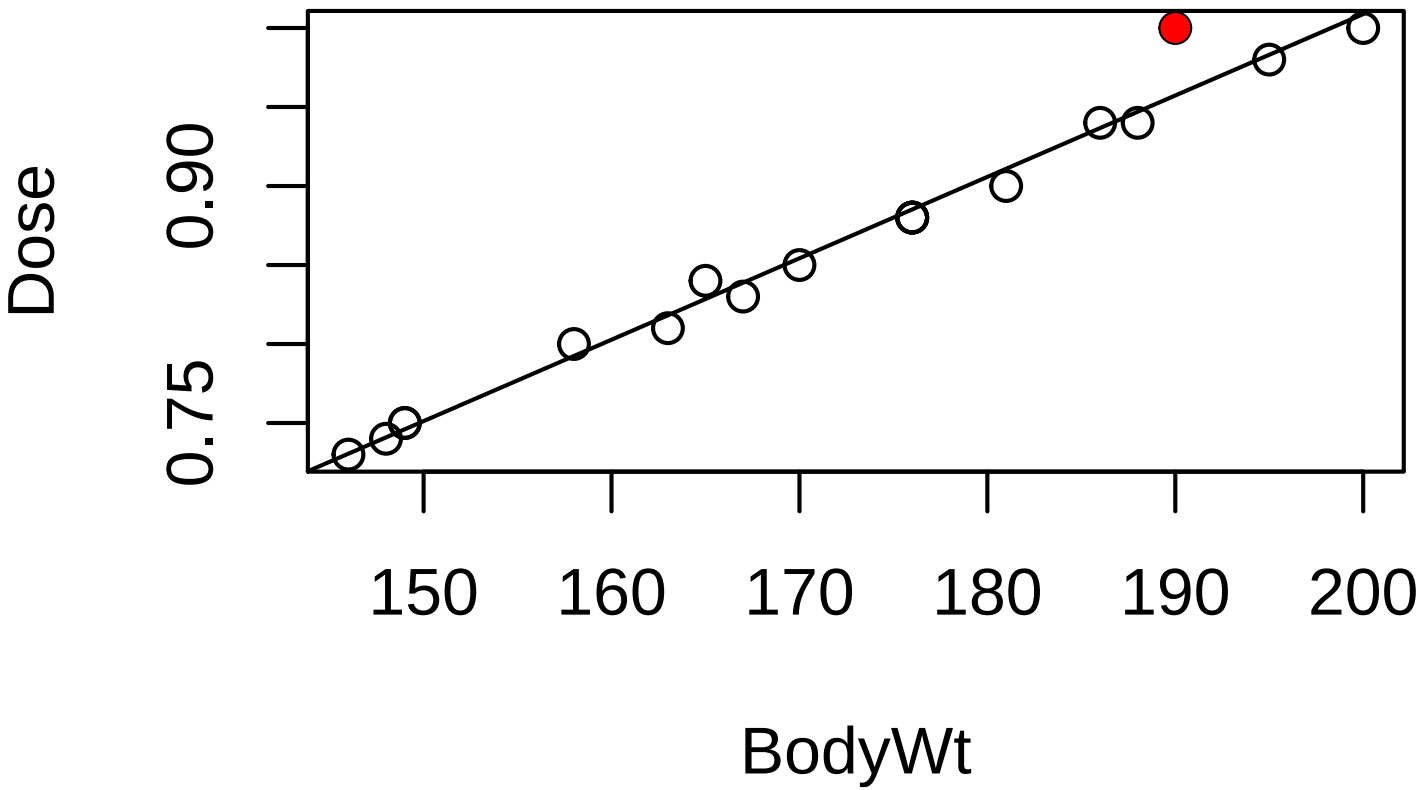
```
residualPlot(m1, id =list(method = list("x", "y"), n=2),
             quadratic = FALSE)
```



Other Diagnostic Tools

The previous residual plot shows that case number 3 has a very large fitted value, compared to the rest of the points.

The cause may be that rat number 3 got a larger dose than it should have received.



Other Diagnostic Tools

What to do next in a case like this depends on the problem.

One alternative that is sometimes advocated is to fit the model without the suspicious point. We do this as an illustration in the next slide.

However, this throws doubts on the whole experiment, and perhaps there is a need to collect further data, with dose determined with more precision or using a different method.

Other Diagnostic Tools

```
m1b <- lm(y ~ BodyWt + LiverWt + Dose, rat[-3,])
S(m1b)
```

```
## Call: lm(formula = y ~ BodyWt + LiverWt + Dose, data =
##           rat[-3, ])
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.311427   0.205094   1.518    0.151
## BodyWt      -0.007783   0.018717  -0.416    0.684
## LiverWt      0.008989   0.018659   0.482    0.637
## Dose         1.484877   3.713064   0.400    0.695
##
## Residual standard deviation: 0.07825 on 14 degrees of freedom
## Multiple R-squared: 0.02106
## F-statistic: 0.1004 on 3 and 14 DF,  p-value: 0.9585
##    AIC    BIC
## -35.17 -30.71
```

Other Diagnostic Tools:`residualPlots`

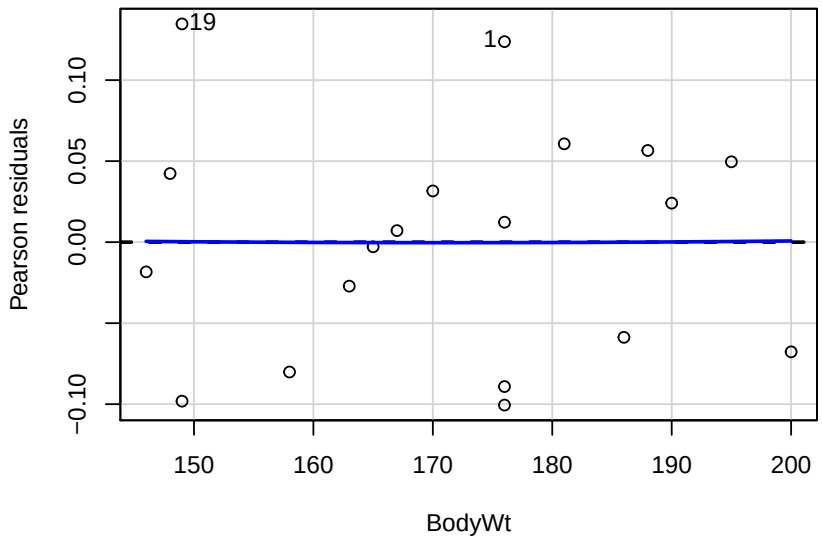
Let's go back to residual plots using the `car` functions.

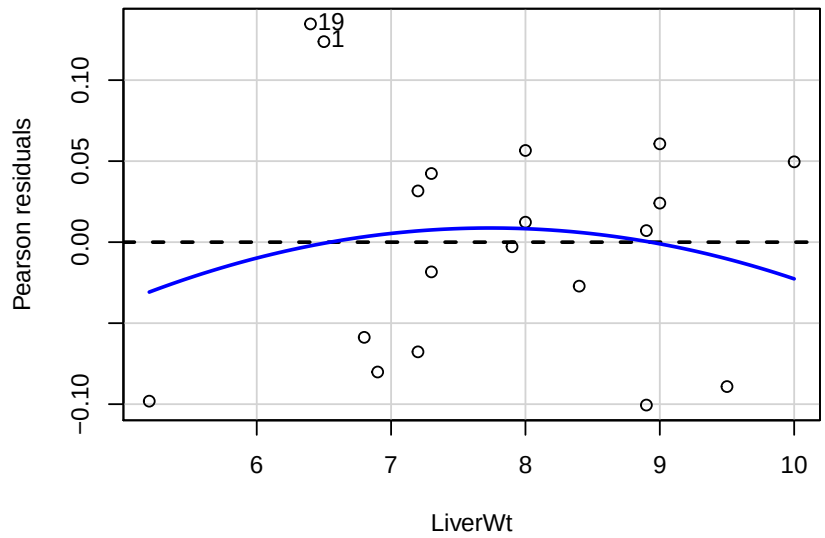
The command `residualPlots()` produces plots of residuals against all regressors and also against fitted values.

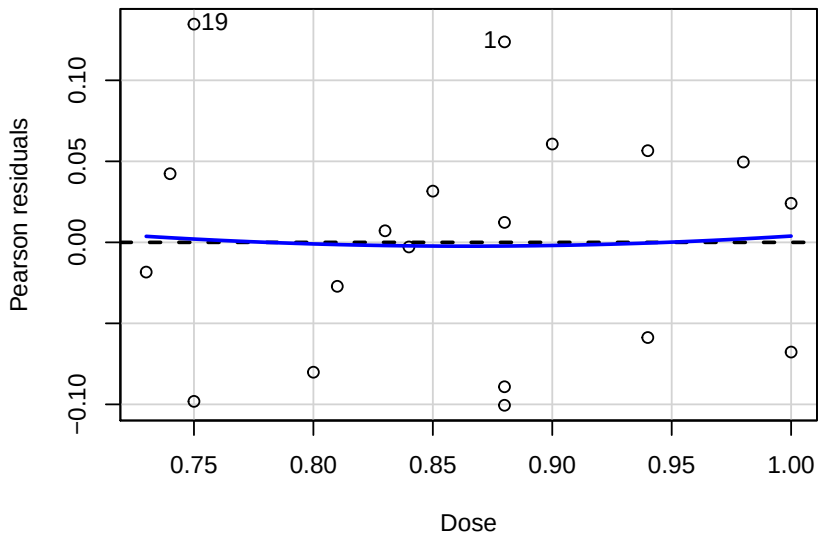
It also, by default, graphs quadratic regression terms and does curvature tests in all cases.

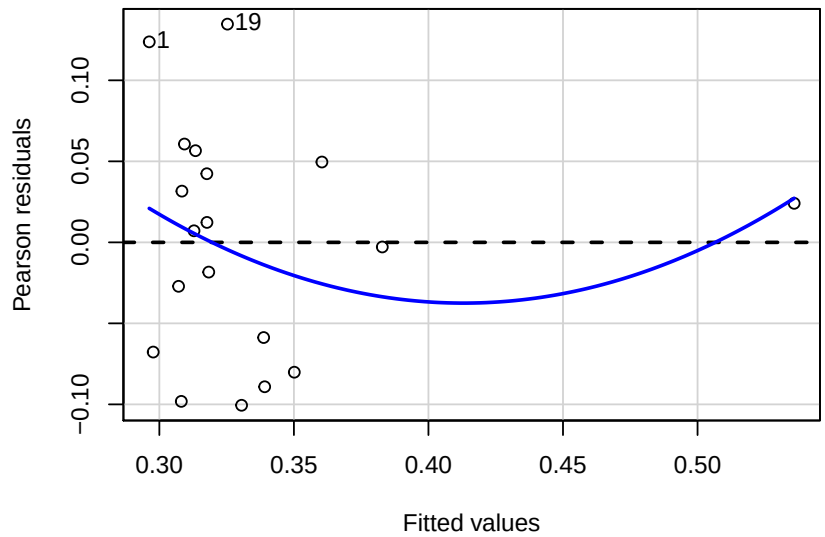
Other Diagnostic Tools:residualPlots

```
residualPlots(m1, id=TRUE)
```










```
##                Test stat Pr(>|Test stat|)
```

Other Diagnostic Tools:residualPlots

```
residualPlots(m1, id=TRUE, plot=FALSE)
```

```
##              Test stat Pr(>|Test stat|)
## BodyWt          0.0185            0.9855
## LiverWt        -0.5760            0.5737
## Dose            0.1444            0.8873
## Tukey test      0.9320            0.3513
```

Other Diagnostic Tools:`residualPlots`

What we see at the bottom are curvature tests.

These tests help decide whether the plots show the need for higher-order terms in the regression.

Suppose we have a plot of residuals against a regressor or a combination of regressors. The test for curvature refits the original model with an additional term for the square of the regressor.

The test for curvature is based on the t-test for the coefficient of the quadratic term under the null hypothesis that it is zero.

If the plot is against fitted values, which depend on the estimated parameters, the t statistic should be compared with the standard normal distribution. This is known as Tukey's test.

None of the tests in the example are significant.

Other Diagnostic Plots

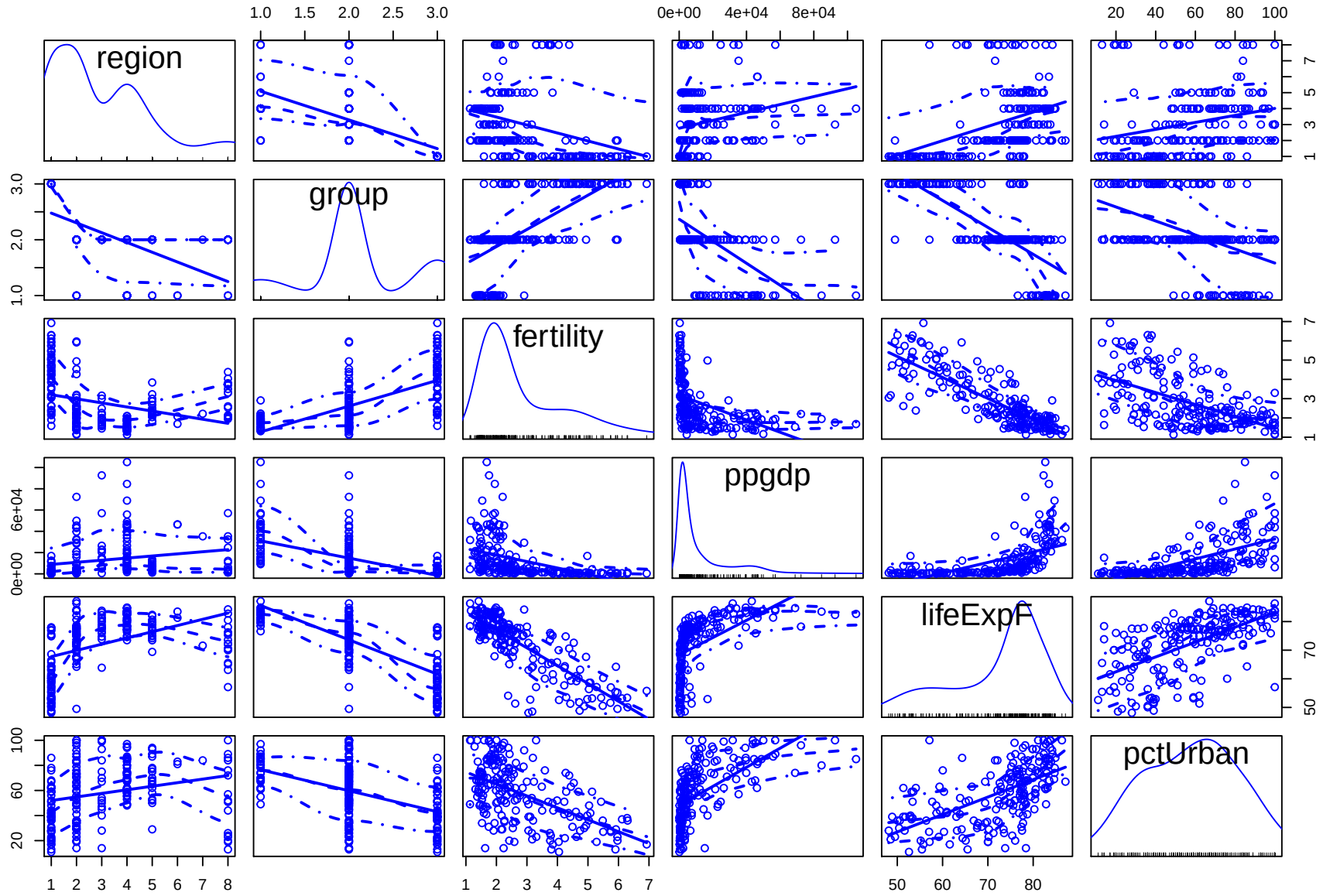
As a second example, consider the data set `UN11`, which has statistics on national health, education, and welfare for 210 places, mostly UN members.

```
str(UN11)
```

```
## 'data.frame':    199 obs. of  6 variables:
##  $ region   : Factor w/ 8 levels "Africa","Asia",..: 2 4 1 1 3 5 2 3 8 4 ...
##  $ group    : Factor w/ 3 levels "oecd","other",..: 2 2 3 3 2 2 2 2 1 1 ...
##  $ fertility: num  5.97 1.52 2.14 5.13 2 ...
##  $ ppgdp    : num  499 3677 4473 4322 13750 ...
##  $ lifeExpF : num  49.5 80.4 75 53.2 81.1 ...
##  $ pctUrban : num  23 53 67 59 100 93 64 47 89 68 ...
##  - attr(*, "na.action")= 'omit' Named int  4 5 8 28 41 67 68 72 79 83 ...
##   ..- attr(*, "names")= chr  "Am Samoa" "Andorra" "Antigua and Barbuda" "Br
```

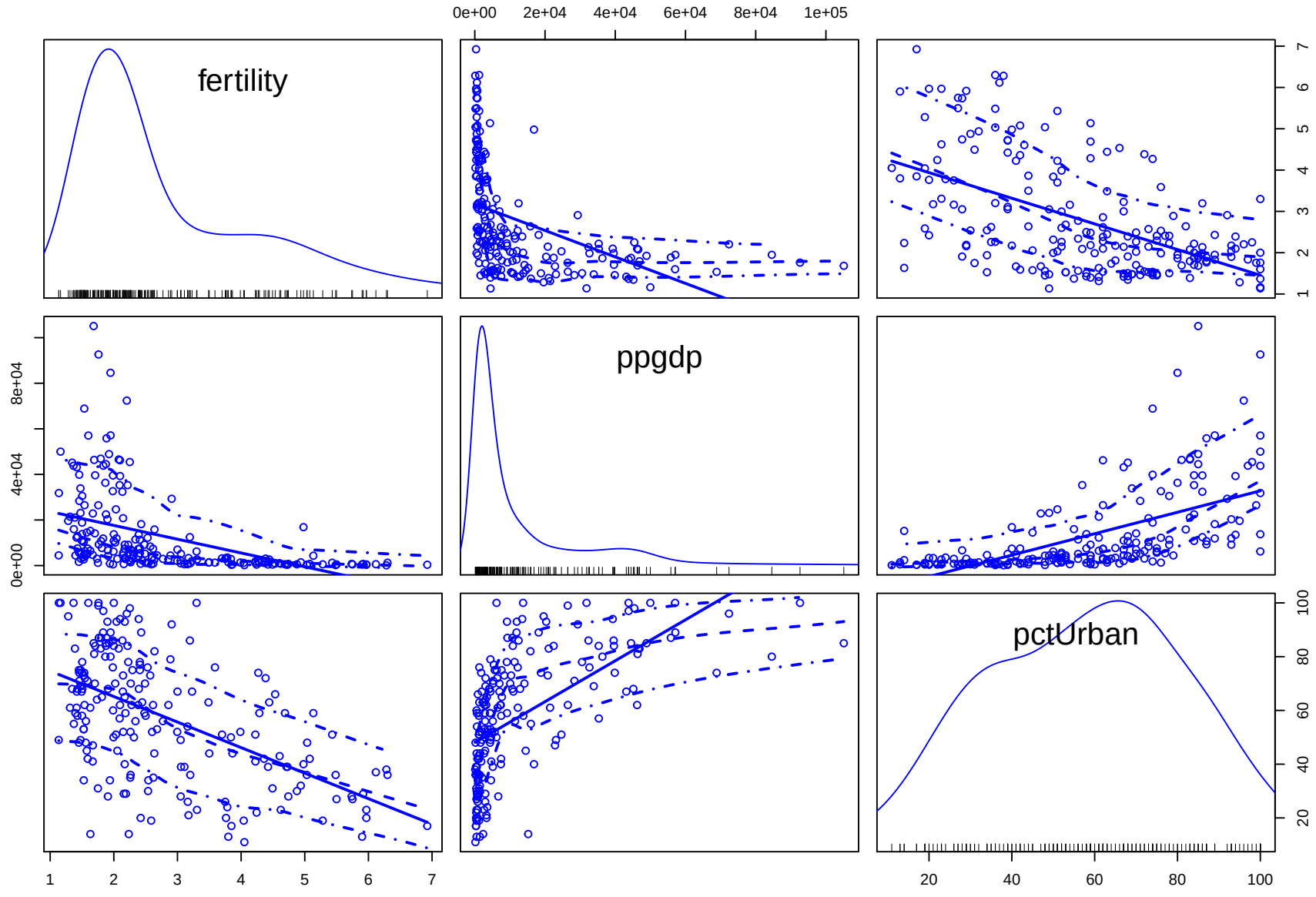
Other Diagnostic Plots

```
scatterplotMatrix(UN11)
```



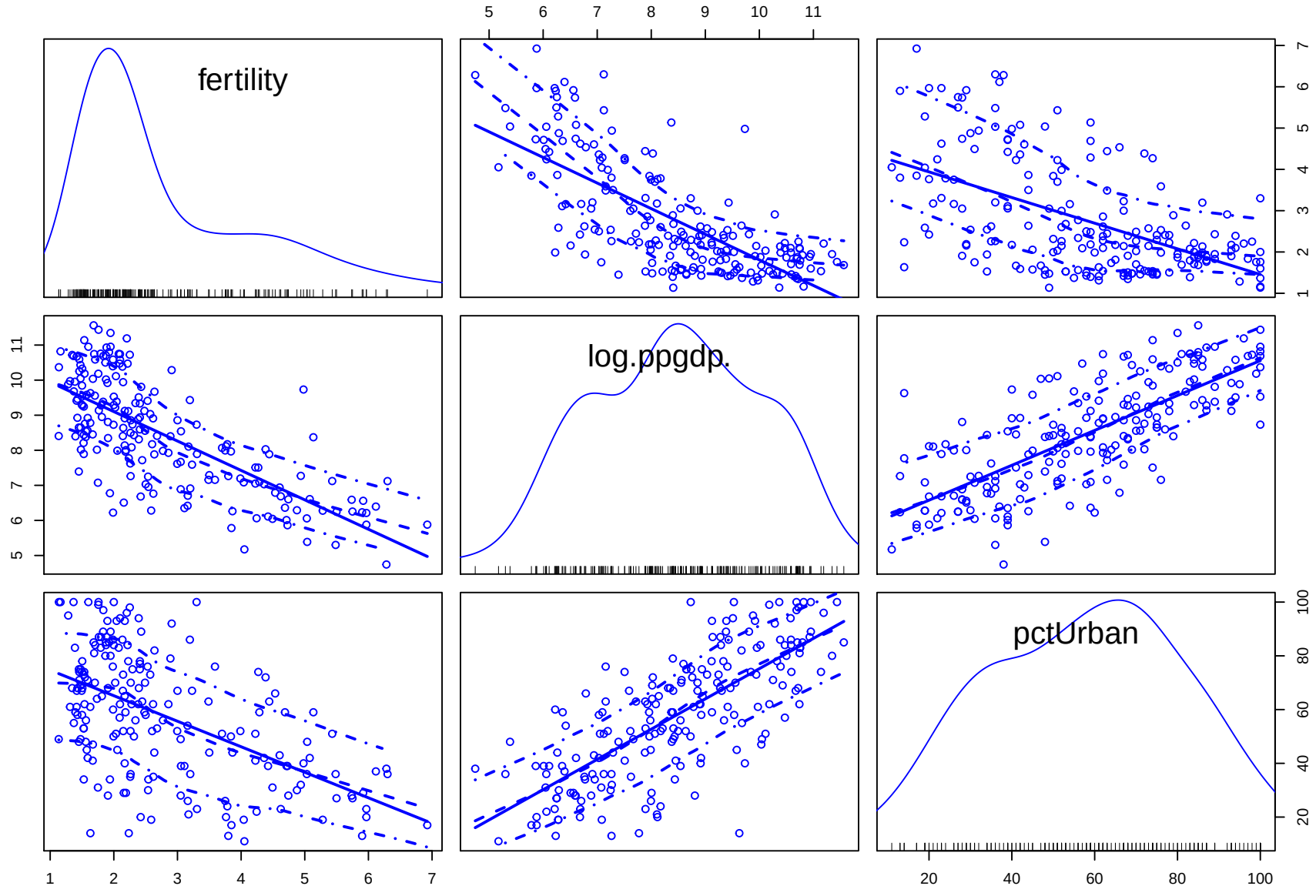
Other Diagnostic Plots

```
scatterplotMatrix(UN11[,c(3,4,6)])
```



Other Diagnostic Plots

```
scatterplotMatrix( ~ fertility + log(ppgdp) + pctUrban, data = UN11)
```



Other Diagnostic Plots

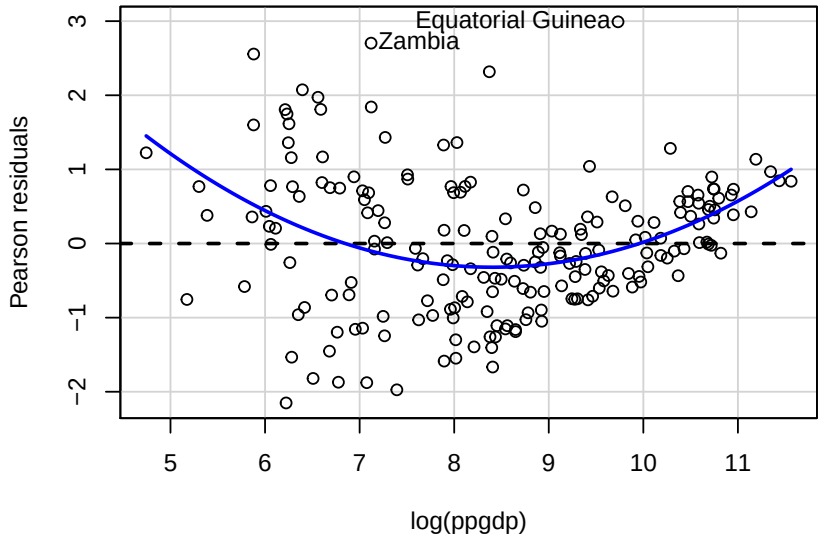
We fit a model for fertility against the log of per capita gross domestic product `log(ppgdp)` and percent urban population `pctUrban`.

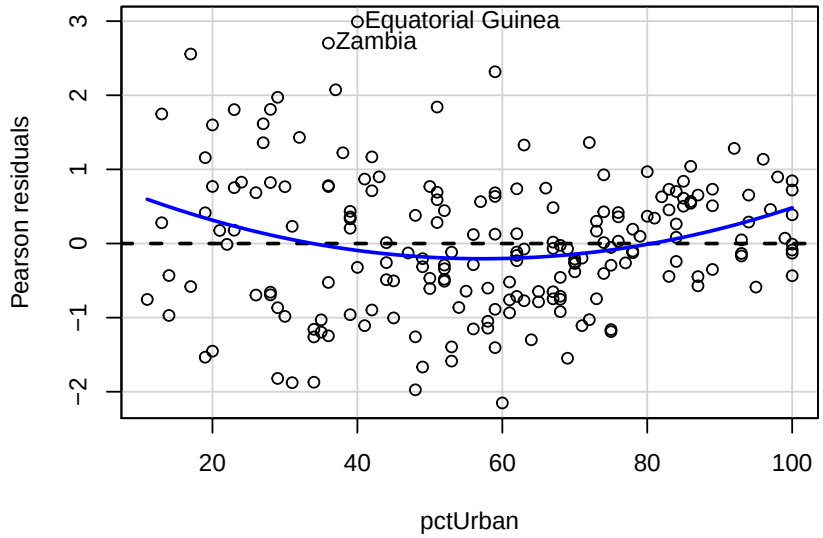
```
lm1 <- lm(fertility ~ log(ppgdp) + pctUrban, data = UN11)
summary(lm1)
```

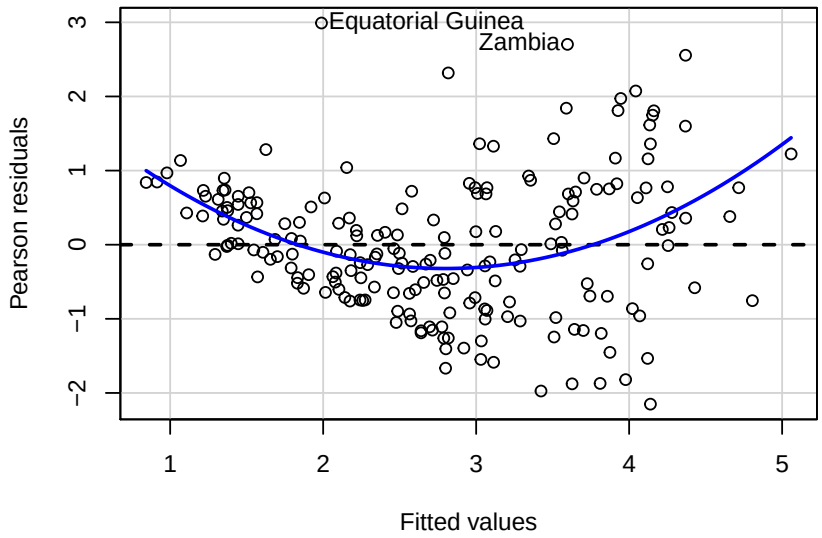
```
##
## Call:
## lm(formula = fertility ~ log(ppgdp) + pctUrban, data = UN11)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.15114 -0.64929 -0.06604  0.63253  2.99102
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.9932699  0.3993367  20.016   <2e-16 ***
## log(ppgdp)  -0.6151425  0.0641565  -9.588   <2e-16 ***
## pctUrban    -0.0004393  0.0042656  -0.103    0.918
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9328 on 196 degrees of freedom
## Multiple R-squared:   0.52,  Adjusted R-squared:  0.5151
## F-statistic: 106.2 on 2 and 196 DF,  p-value: < 2.2e-16
```

Other Diagnostic Plots:`residualPlots`

```
residualPlots(lm1, id=TRUE, tests = FALSE)
```







Other Diagnostic Plots:residualPlots

```
residualPlots(lm1, id=TRUE, plot = FALSE)
```

```
##              Test stat Pr(>|Test stat|)
## log(ppgdp)      5.4068         1.863e-07 ***
## pctUrban        3.2868          0.001202 **
## Tukey test      5.4198         5.966e-08 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In this example, all the tests are significant.