Chapter 4

Model Adequacy Checking

Chapter 4a

Chapter 4a

1. Model assumptions

   The multiple linear regression model for a response variable y and regressor variables $x_1, x_2, ..., x_k$ can be written as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_k x_{ik} + \epsilon_i, \; i = 1, 2, ..., n.$$

   Major assumptions of multiple linear regression:

   (1) The relationship between the response y and the regressors is linear.

   (2) $E(\epsilon_i) = 0$

   (3) $Var(\epsilon_i) = \sigma^2$

   (4) ϵ_i's are independent.

   (5) ϵ_i's are normally distributed

2. Model adequacy checking using residual analysis

   The departures from these assumptions cannot be detected using standard summary statistics such as the t, F or R^2.

   Residual analysis is the main approach for checking the validity of these assumptions.

3. Residual and normal probability plots

Residual $e_i = y_i - \hat{y}_i$, $i = 1, 2, ..., n$

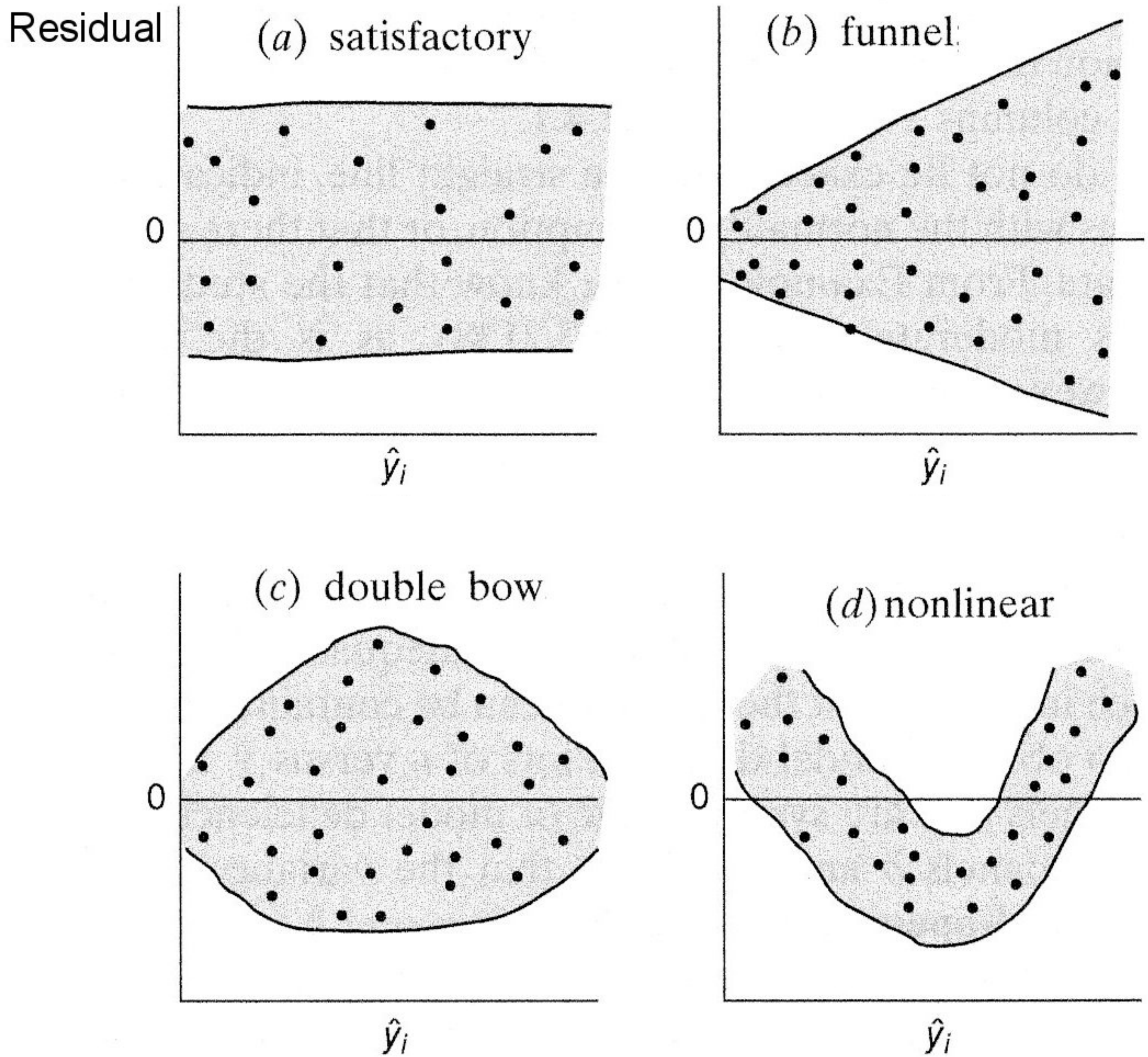
(1) Plot of residual e_i against the fitted value $\hat{y}_i$.

(2) Plot of residual e_i against any of the regressor variables.

(3) Plot of residual against the time.

For checking the normality assumption, we use normal probability plots.

Residual analysis



σ^2 depends on i

model is incorrect

4. Residual $e_i = y_i - \hat{y}_i$, $i = 1, 2, ..., n$

Let SS_{Res} be the residual sum of squares.

$SS_{Res} = \boldsymbol{y}'\boldsymbol{y} - \hat{\boldsymbol{\beta}}'\boldsymbol{X}'\boldsymbol{y} = \sum_{i=1}^{n} e_i^2$ $e_1, e_2, ..., e_n$ are not indep

all correlated

$\hat{\sigma}^2 = MS_{Res} \equiv \frac{SS_{Res}}{n-p}$

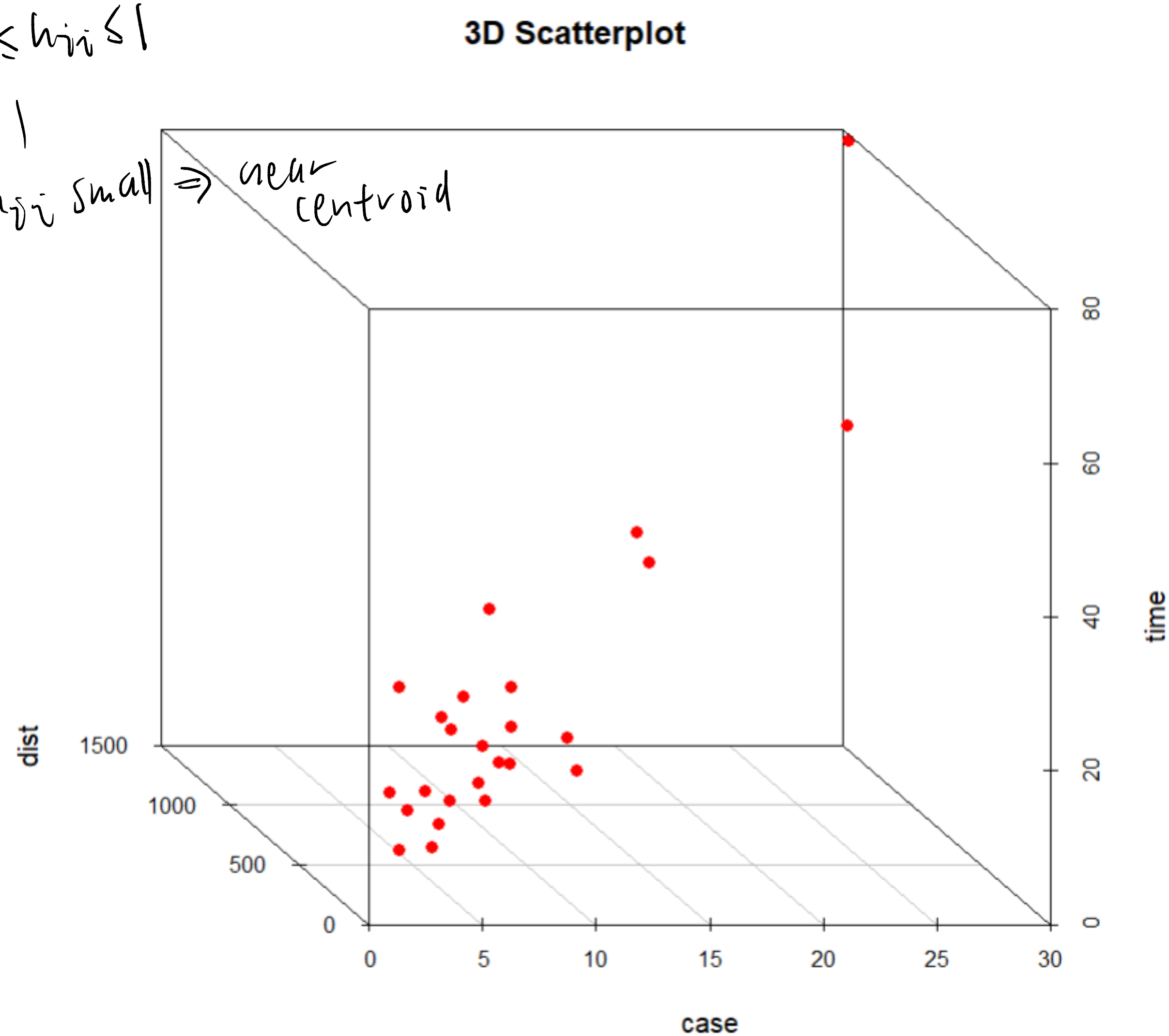
The residuals are not independent because the n residuals have only $n - p$ degrees of freedom associated with them.

Example - Delivery time data

$0 \le h_{ii} \le 1$

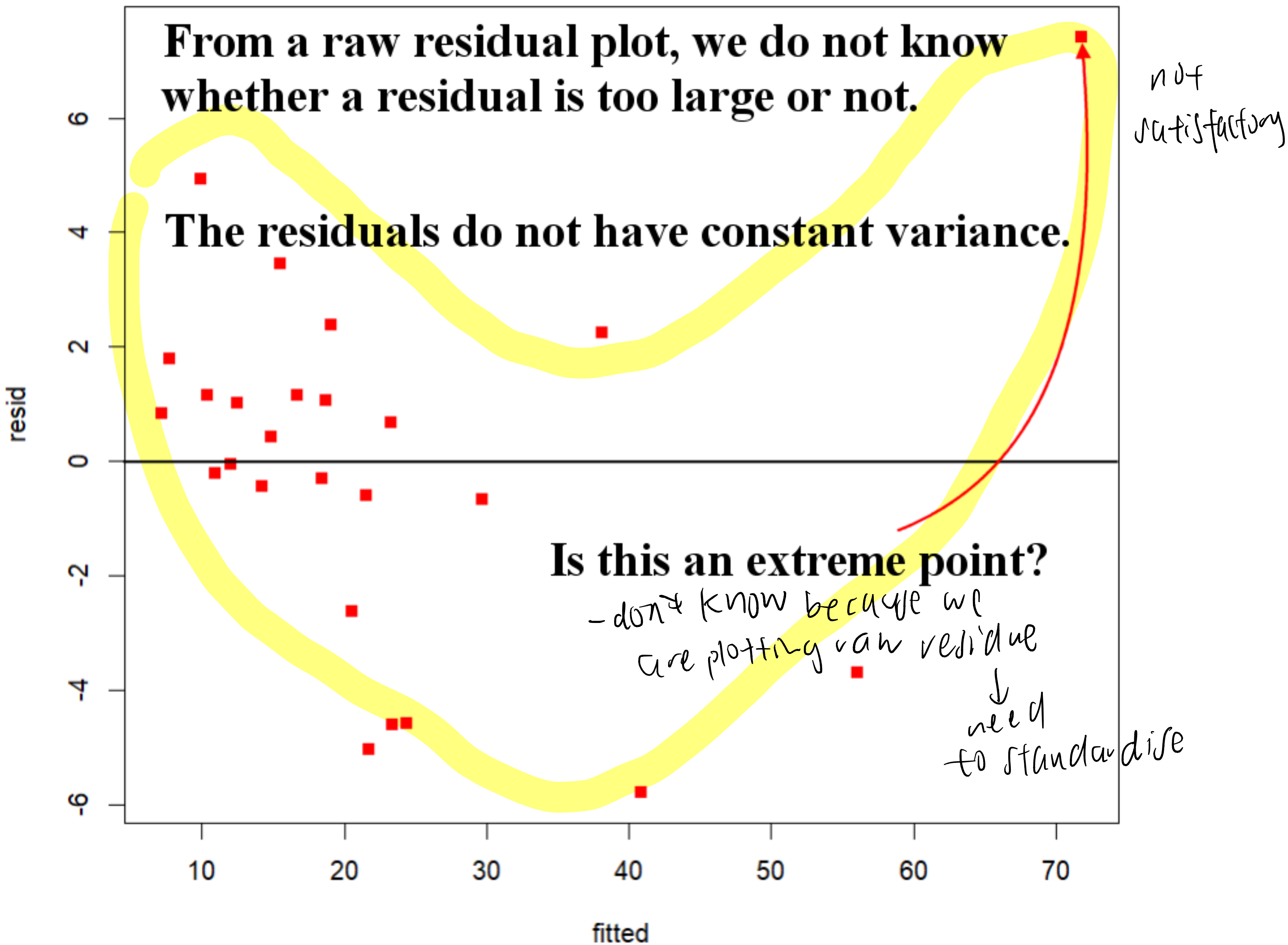
h_{ii} small $\Rightarrow$ near centroid

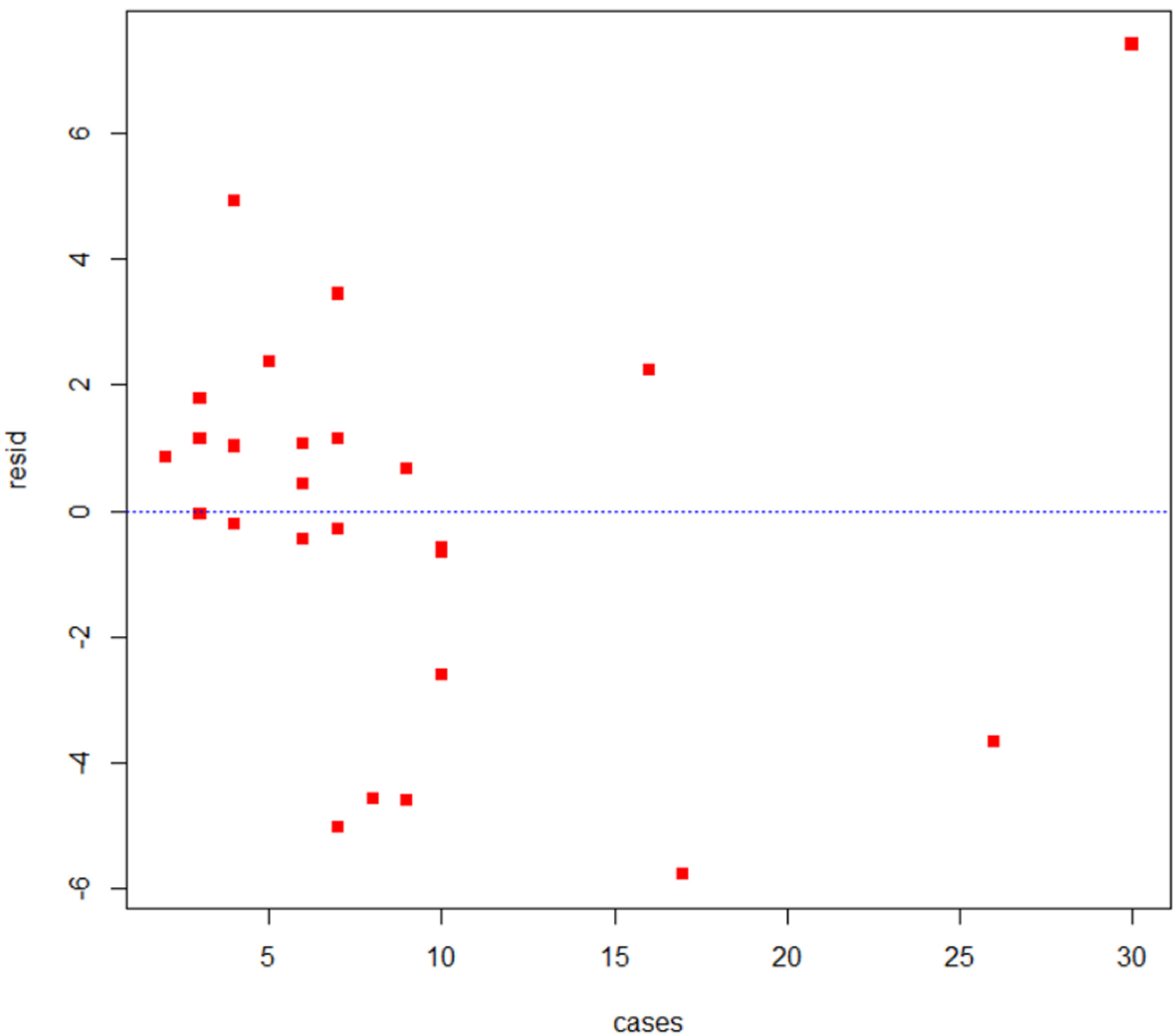
	Obs	time	case	dist
1	1	16.68	7	560
2	2	11.50	3	220
3	3	12.03	3	340
4	4	14.88	4	80
5	5	13.75	6	150
6	6	18.11	7	330
7	7	8.00	2	110
8	8	17.83	7	210
9	9	79.24	30	1460
10	10	21.50	5	605
11	11	40.33	16	688
12	12	21.00	10	215
13	13	13.50	4	255
14	14	19.75	6	462
15	15	24.00	9	448
16	16	29.00	10	776
17	17	15.35	6	200
18	18	19.00	7	132
19	19	9.50	3	36
20	20	35.10	17	770
21	21	17.90	10	140
22	22	52.32	26	810
23	23	18.75	9	450
24	24	19.83	8	635
25	25	10.75	4	150



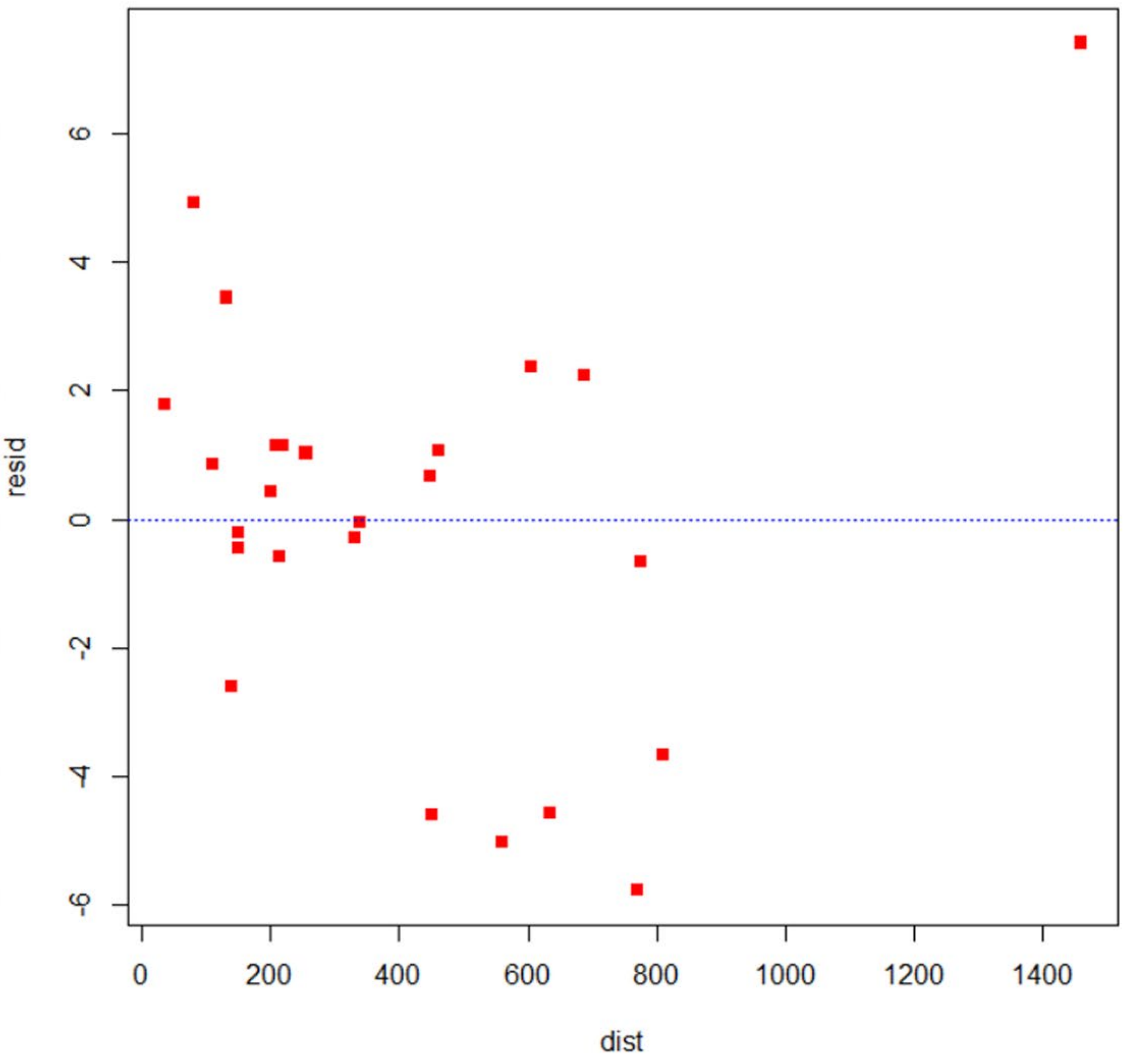
Example - Delivery time data

```
# fit multiple linear regression model
fitted.model <- lm(time ~ case + dist)
summary(fitted.model)

# calculate predicted values
fitted <- predict(fitted.model)
fitted

# calculate raw residuals
resid <- time - fitted
resid
# alternative approach
resid(fitted.model)

# construct residual plots
plot(fitted,resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(case,resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(dist,resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
```


From a raw residual plot, we do not know whether a residual is too large or not.
The residuals do not have constant variance.
Is this an extreme point?
- don't know because we are plotting raw residue
need to standardise
not satisfactory
resid
fitted
6
4
2
0
-2
-4
-6
10
20
30
40
50
60
70



resid
6
4
2
0
-2
-4
-6
5
10
15
20
25
30
cases



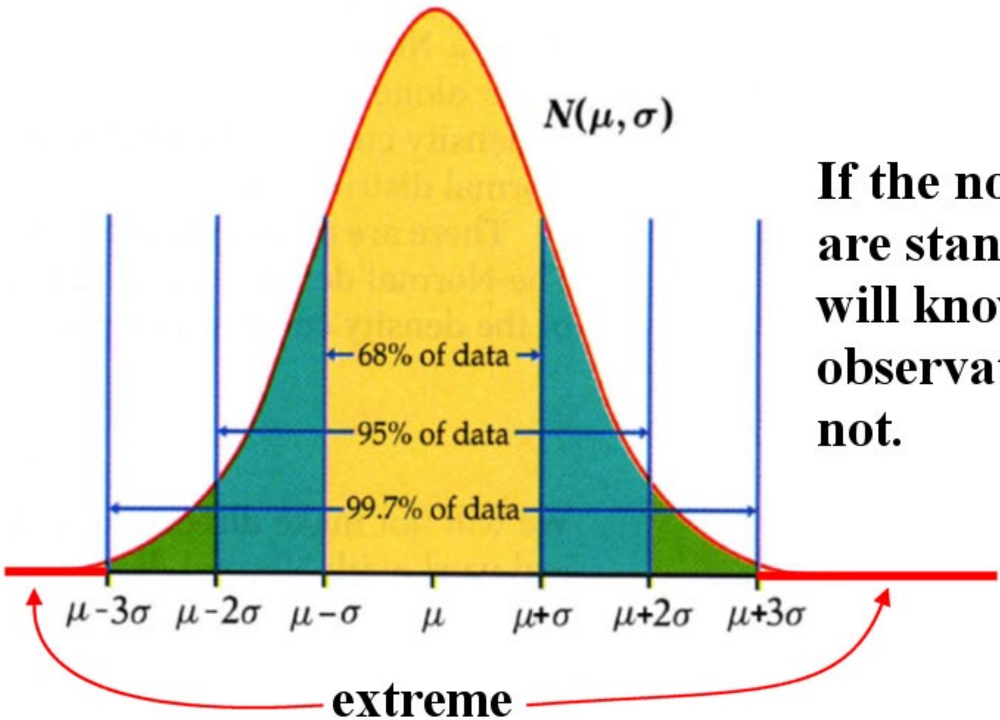
resid
6
4
2
0
-2
-4
-6
0
200
400
600
800
1000
1200
1400
dist

ST1131

THE 68–95–99.7 RULE

In the Normal distribution with mean μ and standard deviation σ:

- Approximately **68%** of the observations fall within σ of the mean μ.
- Approximately **95%** of the observations fall within 2σ of μ.
- Approximately **99.7%** of the observations fall within 3σ of μ.



If the normal observations are standardized, then we will know whether a given observation is extreme or not.

5. Methods for scaling residuals

(1) Standardized residuals

(2) Studentized residuals

(3) PRESS residuals

(4) R-student

6. Standardized residuals

The multiple linear regression model for a response variable y and regressor variables $x_1, x_2, ..., x_k$ can be written as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_k x_{ik} + \epsilon_i,\ i = 1, 2, ..., n.$$

where $E(\epsilon_i) = 0$, $Var(\epsilon_i) = \sigma^2,\ i = 1, 2, ..., n$ and ϵ_i's are independent.

Residual $e_i = y_i - \hat{y}_i$, $i = 1, 2, ..., n$.

Note that e_i is an realisation of ϵ_i.

$$
\begin{aligned}
E(e_i) &= E(y_i - \hat{y}_i) \\
&= E(y_i) - E(\hat{y}_i) \\
&= \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} - E(\hat{\beta}_0 + \sum_{j=1}^{k} \hat{\beta}_j x_{ij}) \\
&= \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} - \left[\beta_0 + \sum_{j=1}^{k} \beta_j x_{ij}\right] \\
&= 0
\end{aligned}
$$

estimate of σ^2 (cannot use in real life as it is an unknown value)

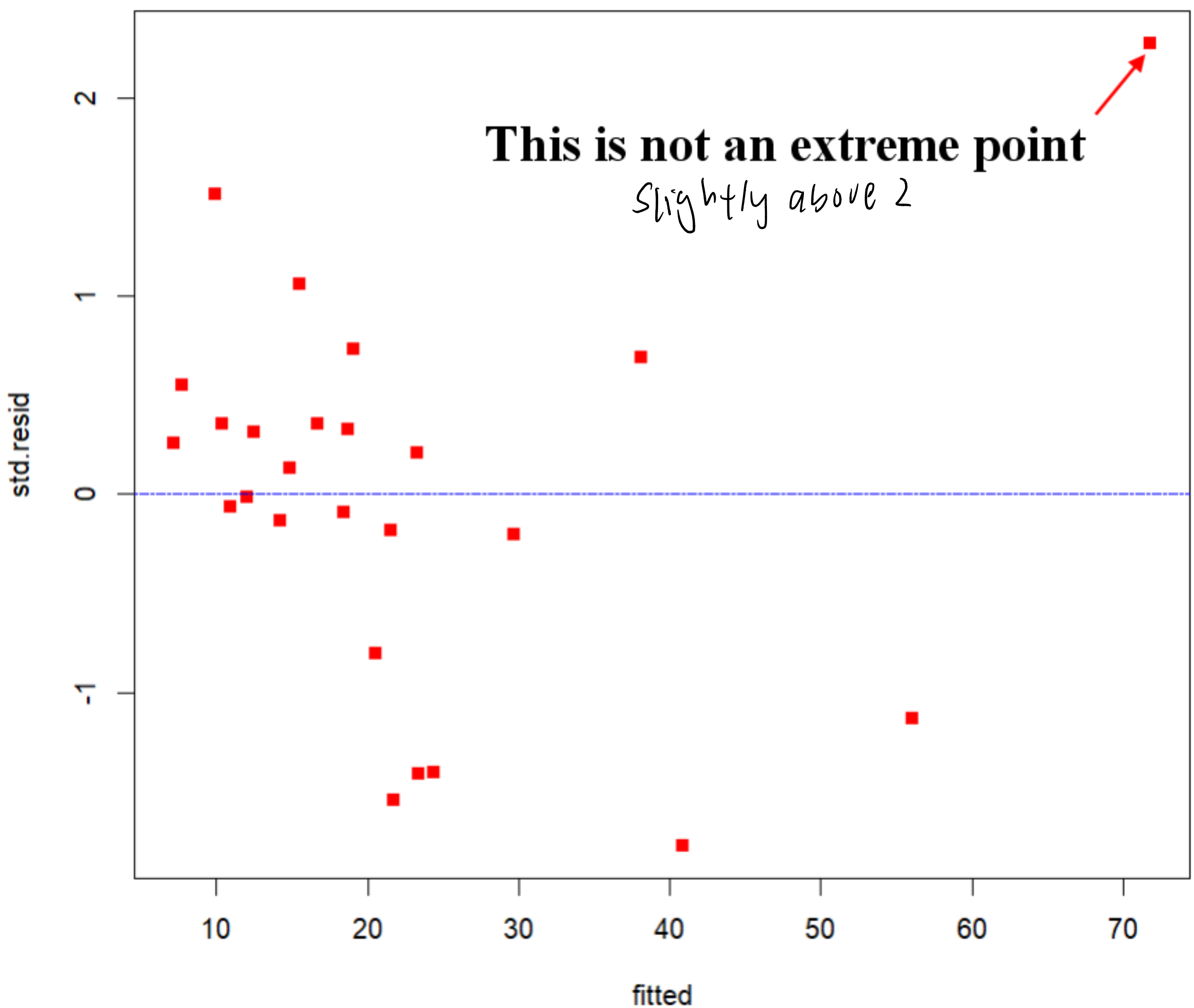
$$
\widehat{Var(\epsilon_i)} = \hat{\sigma}^2 = MS_{Res}
$$

Standardized residual is obtained as

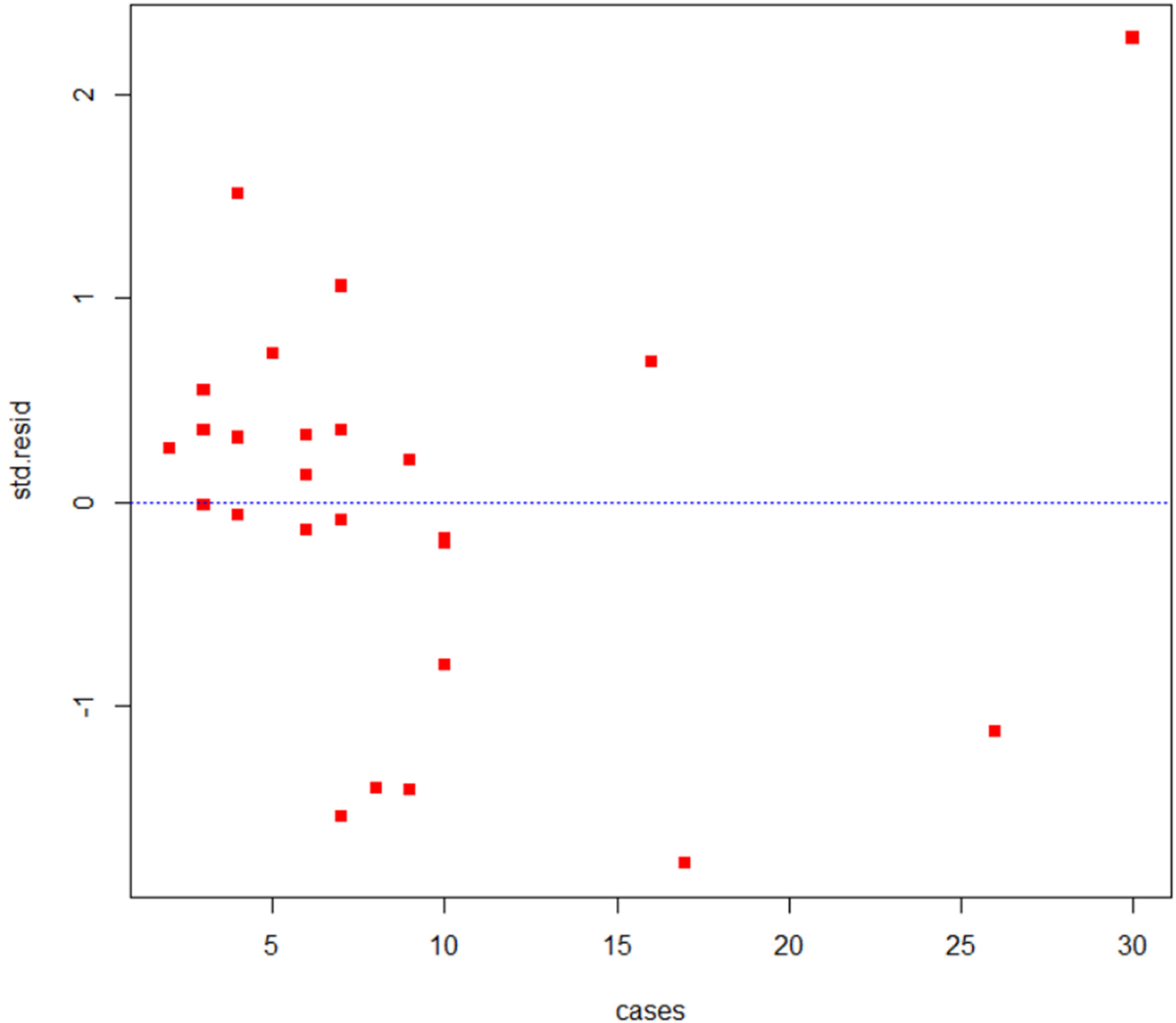
$$
d_i = \frac{e_i - 0}{\sqrt{MS_{Res}}}, \quad i = 1, 2, ..., n.
$$

Example - Delivery time data

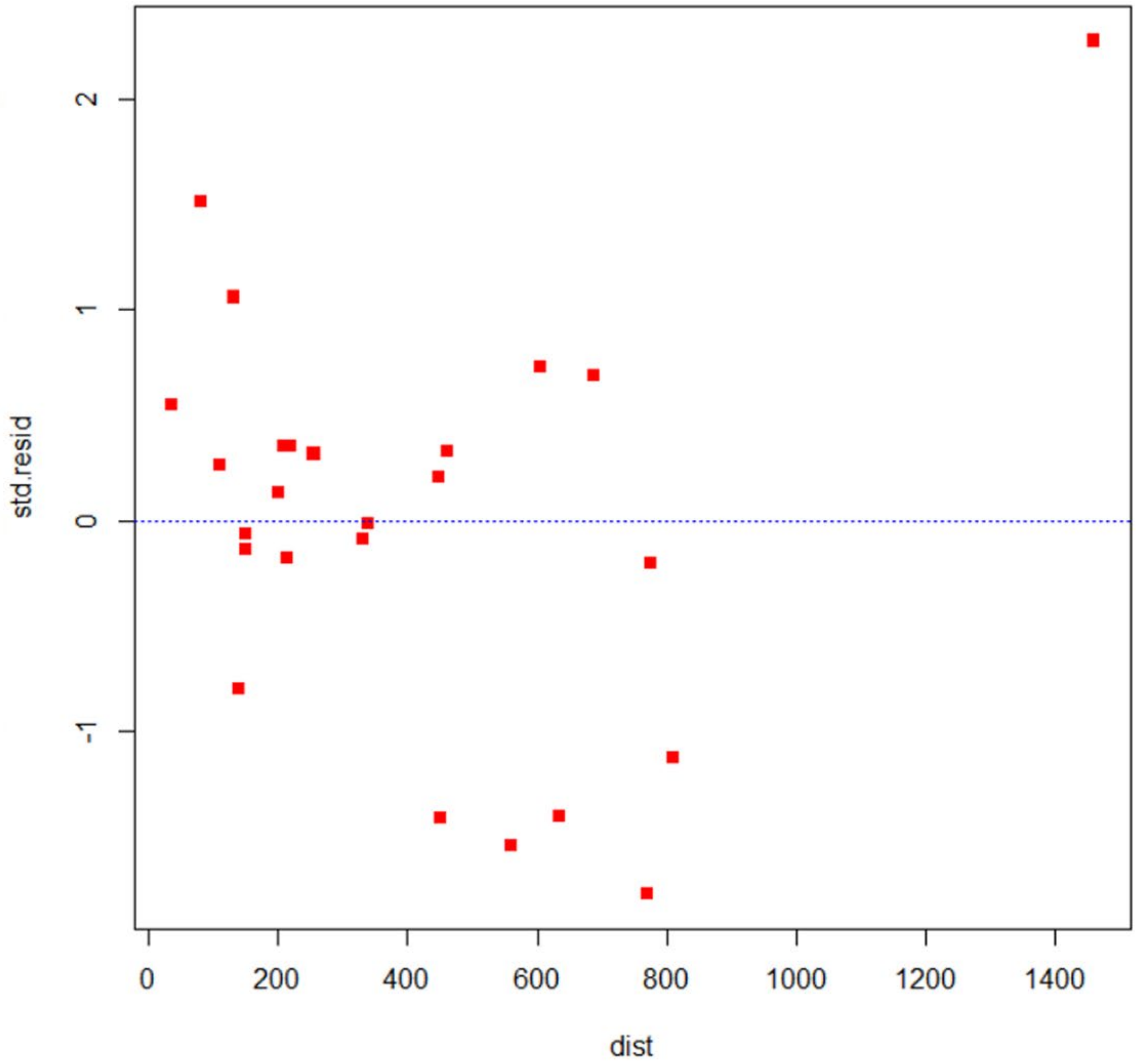
```
# calculate standardized residuals
MSRes <- (summary(fitted.model)$sigma)^2
MSRes
std.resid <- resid/sqrt(MSRes)
std.resid
# alternative approach
rstandard(fitted.model)

# construct standardized residual plots
plot(fitted,std.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(case,std.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(dist,std.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
```



This is not an extreme point
Slightly above 2
std.resid
fitted
2
1
0
-1
10
20
30
40
50
60
70



std.resid
cases
-1
0
1
2
5
10
15
20
25
30


std.resid
2
1
0
-1
0
200
400
600
800
1000
1200
1400
dist

7. Studentized residuals

Multiple regression model:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \text{ where } \boldsymbol{E}(\boldsymbol{\epsilon}) = 0 \text{ and } \boldsymbol{Var}(\boldsymbol{\epsilon}) = \sigma^2 \boldsymbol{I}$$

$$\begin{aligned}
\boldsymbol{e} &= \boldsymbol{y} - \hat{\boldsymbol{y}} \\
&= \boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}} \\
&= \boldsymbol{y} - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y} \\
&= (\boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}')\boldsymbol{y} \\
&= (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{y} \quad \text{where } \boldsymbol{H} = \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'
\end{aligned}$$

from $X'X\hat{\beta} = X'y$

$$\begin{aligned}
\boldsymbol{e} &= (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{y} \\
&= (\boldsymbol{I} - \boldsymbol{H})(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}) \\
&= (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{X}\boldsymbol{\beta} + (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{\epsilon} \\
&= (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{\epsilon} \quad \because (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{X} = 0
\end{aligned}$$

$$
\begin{aligned}
\boldsymbol{E}(\boldsymbol{e}) &= (\boldsymbol{I}-\boldsymbol{H})\boldsymbol{E}(\boldsymbol{\epsilon}) = \boldsymbol{0} \\
\boldsymbol{Var}(\boldsymbol{e}) &= (\boldsymbol{I}-\boldsymbol{H})\boldsymbol{Var}(\boldsymbol{\epsilon})(\boldsymbol{I}-\boldsymbol{H})' \\
&= (\boldsymbol{I}-\boldsymbol{H})\boldsymbol{Var}(\boldsymbol{\epsilon})(\boldsymbol{I}-\boldsymbol{H}) \because (\boldsymbol{I}-\boldsymbol{H}) \text{ symmetric} \\
&= (\boldsymbol{I}-\boldsymbol{H})\sigma^2\boldsymbol{I}(\boldsymbol{I}-\boldsymbol{H}) \quad \because (\boldsymbol{I}-\boldsymbol{H}) \text{ symmetric} \\
&= (\boldsymbol{I}-\boldsymbol{H})\sigma^2 \quad \because (\boldsymbol{I}-\boldsymbol{H}) \text{ is idempotent}
\end{aligned}
$$

standardised: $\dfrac{e_i - 0}{\sqrt{MS_{Res}}}$

$$
\boldsymbol{H} = \begin{bmatrix}
h_{11} & h_{12} & h_{13} & \dots & h_{1n} \\
h_{21} & h_{22} & h_{23} & \dots & h_{2n} \\
. & . & . & & . \\
. & . & . & & . \\
. & . & . & & . \\
h_{n1} & h_{n1} & h_{n2} & \dots & h_{nn}
\end{bmatrix}_{n\times n}
$$

studentised: $\dfrac{e_i - 0}{\sqrt{MS_{Res}(1-h_{ii})}}$

$$Var(e_i) = \sigma^2(1-h_{ii})$$

$$Cov(e_i, e_j) = -\sigma^2 h_{ij}, \quad i \neq j$$

not 0 so they are correlated

Note: $0 \leq h_{ii} \leq 1$

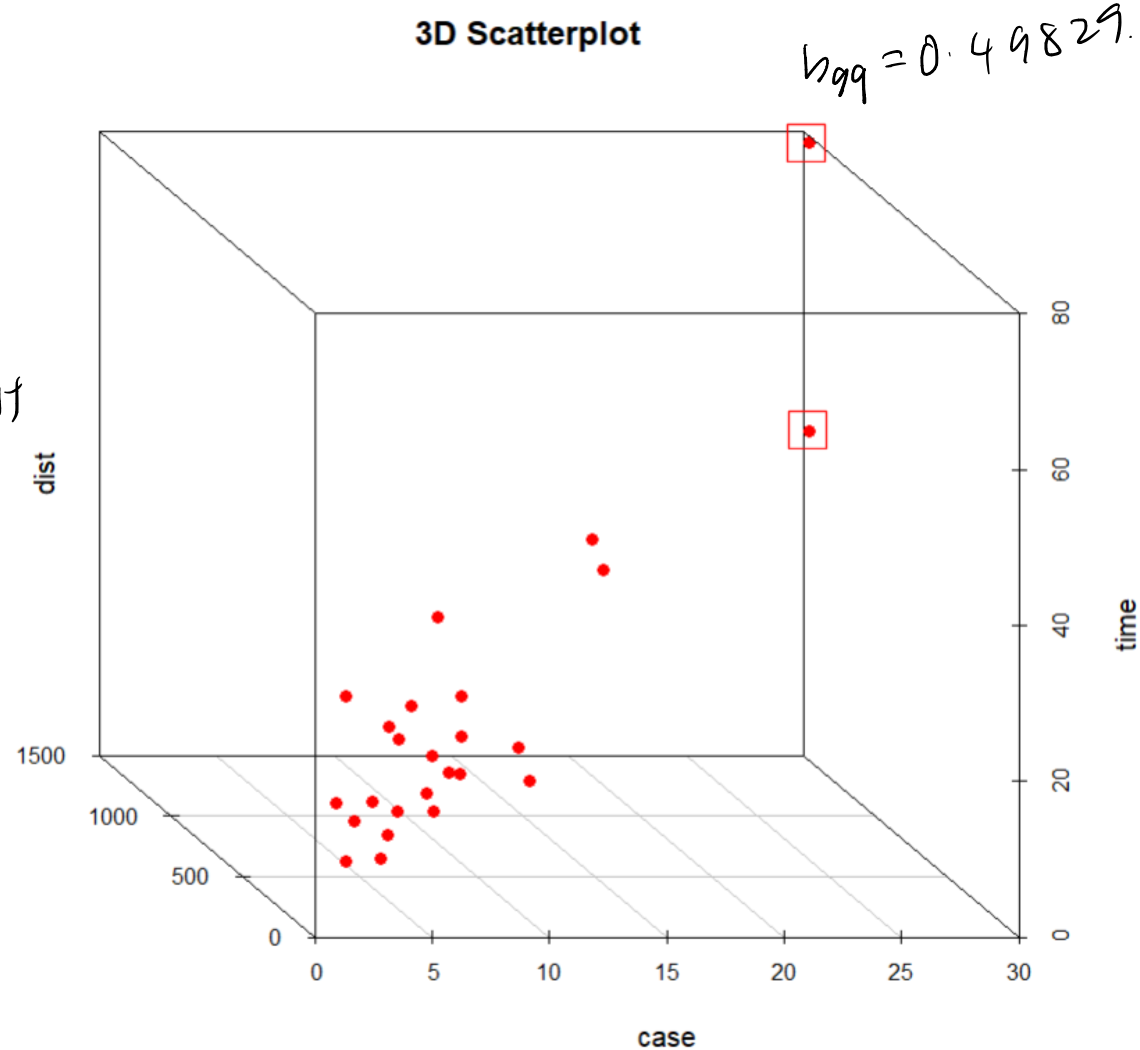
Find h_ii using R

```
# calculate h_ii
one <- rep(1,length(time))
X <- array(c(one, case,dist), dim=c(length(time),3))
XPX <-t(X) %*% X
XPXI <- solve(XPX)
H <- X %*% XPXI %*% t(X)
Hii <- diag(H)
Hii
#alternative approach
Hii <- hatvalues(fitted.model, fullHatMatrix = FALSE)

# display data with diagonal elements of H
data.frame(time,case,dist,Hii)

# 3d scatter plot
scatterplot3d(case,dist,time,main="3D Scatterplot",
              pch=16,color="red",angle=145)
```

```
> # display data with diagonal
    elements of H
> data.frame(time,case,dist,Hii)
    time case dist        Hii
1  16.68    7  560 0.10180178
2  11.50    3  220 0.07070164
3  12.03    3  340 0.09873476
4  14.88    4   80 0.08537479
5  13.75    6  150 0.07501050
6  18.11    7  330 0.04286693
7   8.00    2  110 0.08179867
8  17.83    7  210 0.06372559
9  79.24   30 1460 0.49829216
10 21.50    5  605 0.19629595
11 40.33   16  688 0.08613260
12 21.00   10  215 0.11365570
13 13.50    4  255 0.06112463
14 19.75    6  462 0.07824332
15 24.00    9  448 0.04111077
16 29.00   10  776 0.16594043
17 15.35    6  200 0.05943202
18 19.00    7  132 0.09626046
19  9.50    3   36 0.09644857
20 35.10   17  770 0.10168486
21 17.90   10  140 0.16527689
22 52.32   26  810 0.39157522
23 18.75    9  450 0.04126005
24 19.83    8  635 0.12060826
25 10.75    4  150 0.06664345
```

largest hii

$h_{99} = 0.49829..$



Studentized residual is obtained as

big when
↑ h_{ii} is big

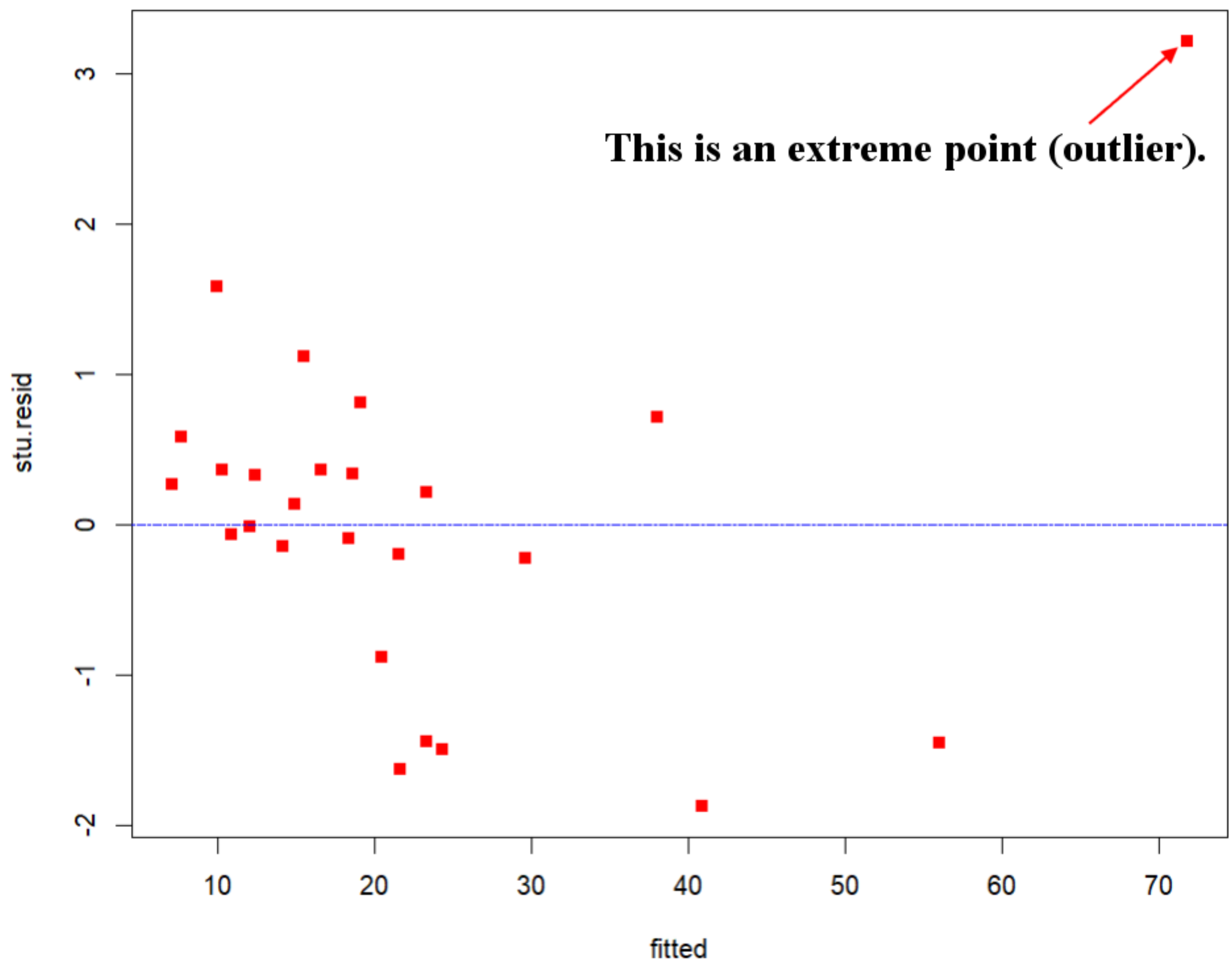
$$r_i = \frac{e_i - 0}{\sqrt{MS_{Res}(1 - h_{ii})}}, \quad i = 1, 2, ..., n.$$

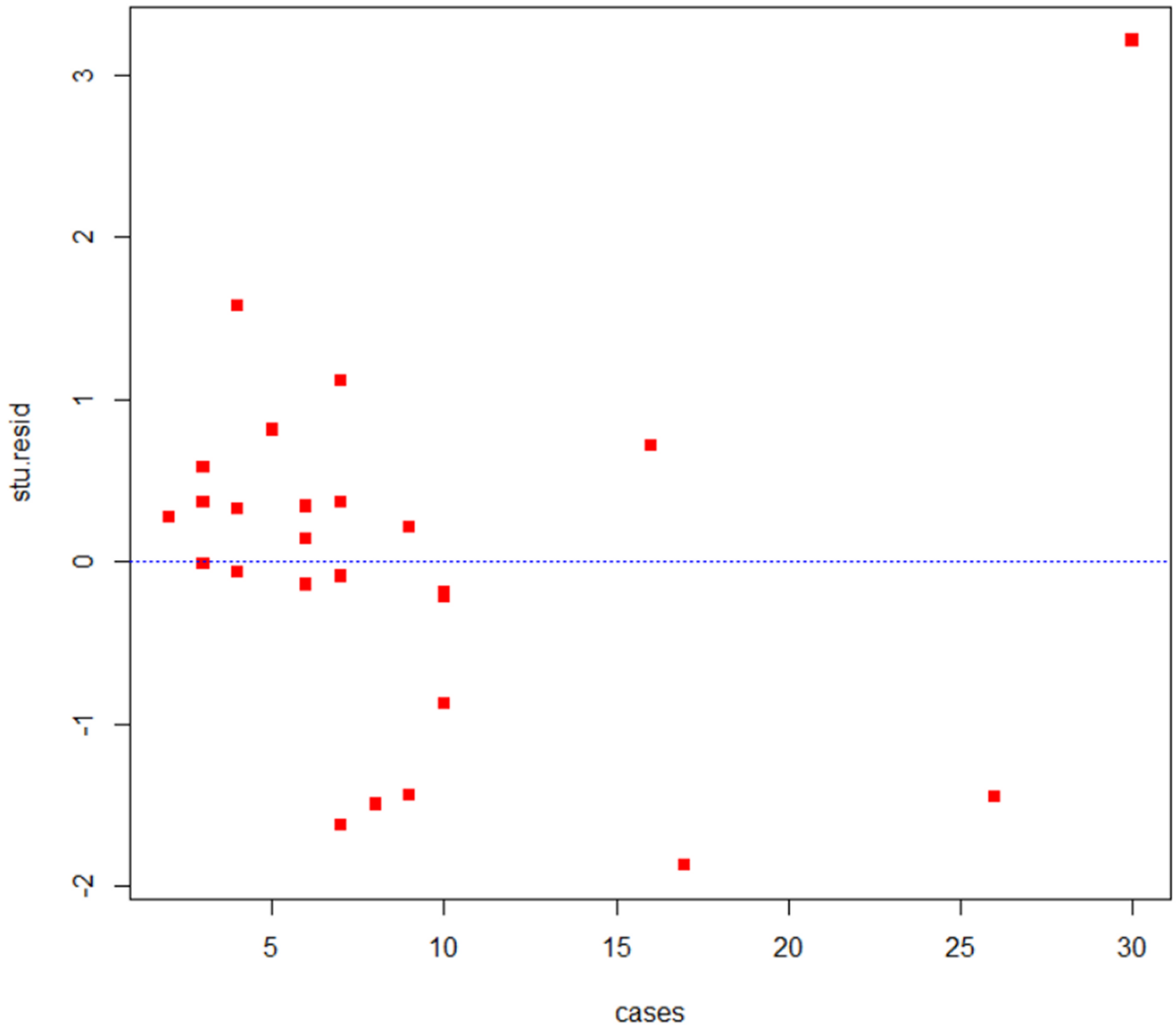
Note: In studentized residuals, we use a more accurate estimate of $Var(e_i)$.

Note: The value of h_{ii} is a measure of the Euclidean distance of the ith observation $\boldsymbol{x}_i' = [1, x_{i1}, x_{i1}, ..., x_{ik}]$ from the centroid of the data. Hence an outlier which is a point far from the centroid will result in a large studentized residual. Examination of studentized residuals is generally recommended.

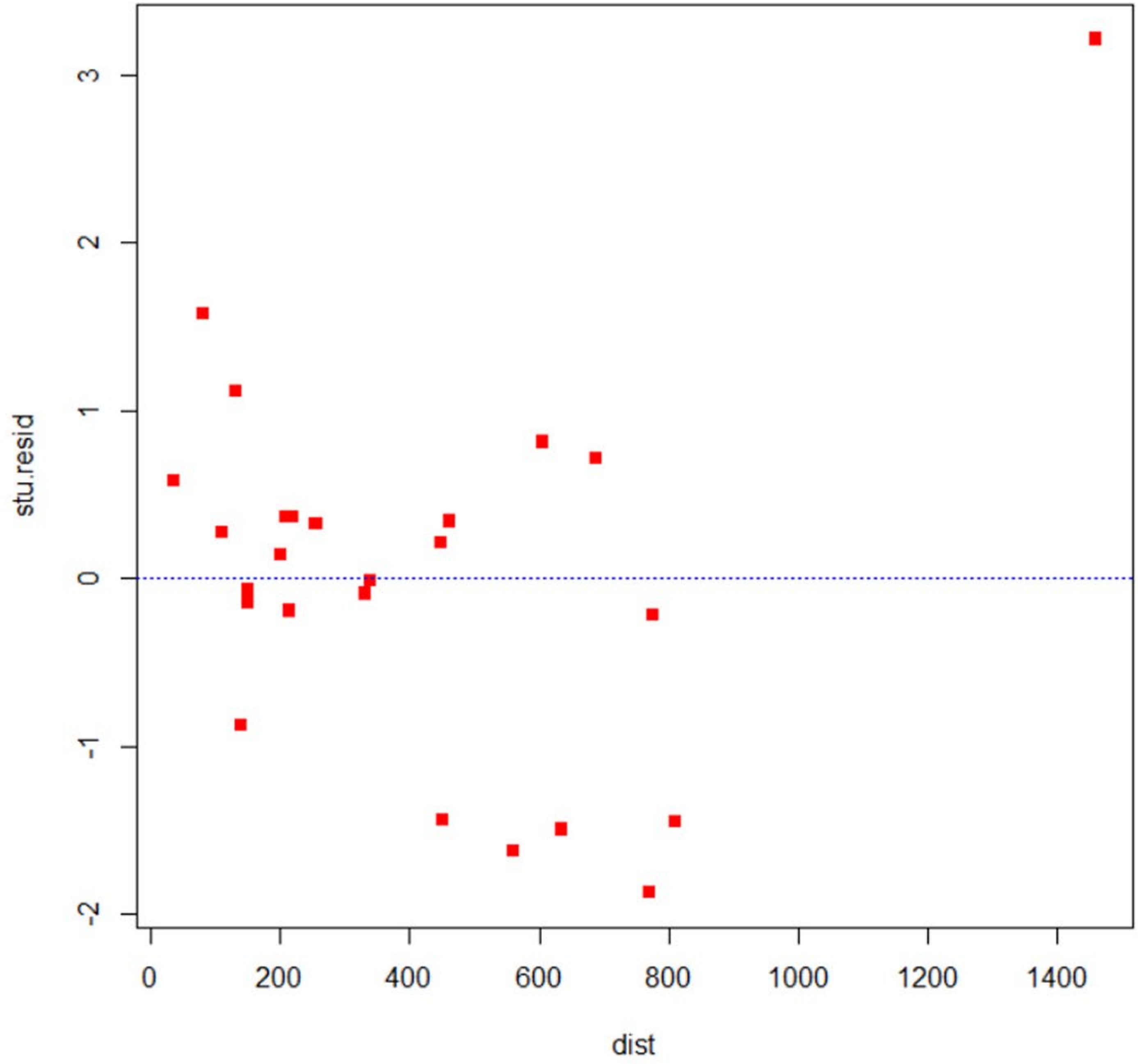
Example - Delivery time data

```
# calculate studentized residuals
MSRes <- (summary(fitted.model)$sigma)^2
MSRes
stu.resid <- resid/sqrt(MSRes*(1-Hii))
stu.resid
# alternative approach
stu.resid <- studres(fitted.model)

# construct studentized residual plots
plot(fitted,stu.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(case,stu.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(dist,stu.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
```


This is an extreme point (outlier).
stu.resid
-2
-1
0
1
2
3
10
20
30
40
50
60
70
fitted



stu.resid
3
2
1
0
-1
-2
5
10
15
20
25
30
cases


stu.resid
3
2
1
0
-1
-2
0
200
400
600
800
1000
1200
1400
dist

8. PRESS residuals similar role to r^2

The PRESS residual is defined as

$$e_{(i)} \equiv y_i - \hat{y}_{(i)}, \quad i = 1, 2, ..., n,$$

where $\hat{y}_{(i)}$ is the fitted value of the ith response based on all observations except the ith one.

remove observation i then fit model

It can be shown that

press residue

$$e_{(i)} = \frac{e_i}{1 - h_{ii}}, \quad i = 1, 2, ..., n.$$

raw residue

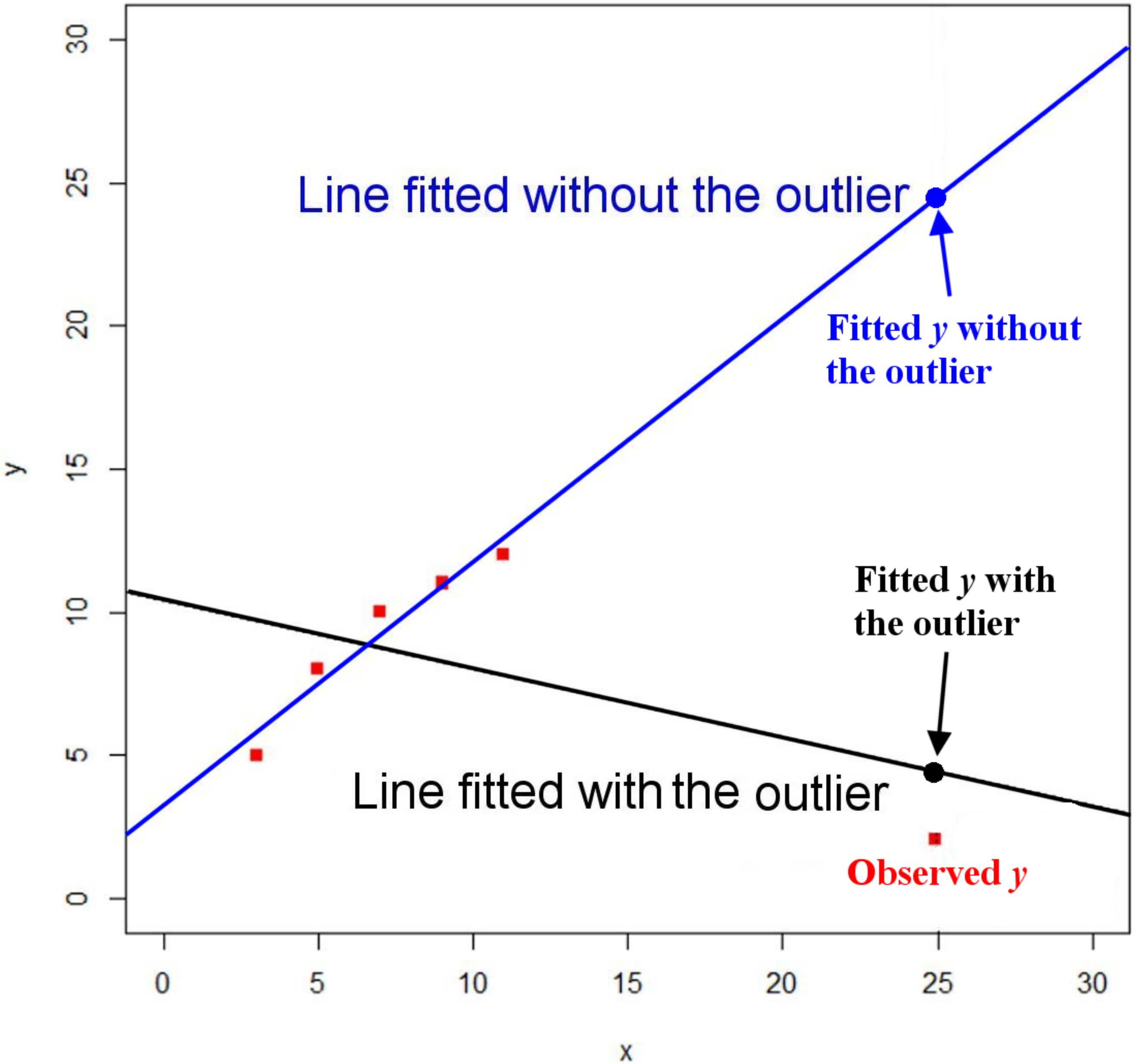
→ both still not standardised

Logic behind PRESS residuals

Suppose the ith observation is an outlier that is influential. It may produce a fitted response $\hat{y}_i$ that is similar to the observed response y_i, making detection of the outlier difficult using the raw residual $e_i = y_i - \hat{y}_i$.

If the outlier is removed, the fitted response $\hat{y}_{(i)}$ can no longer be influenced by the outlier. The resulting residual $y_i - \hat{y}_{(i)}$ should be likely to indicate the presence of the outlier.

Logic behind PRESS residuals



Variance of the ith PRESS residual

$$
\begin{aligned}
Var(e_{(i)}) &= Var\left(\frac{e_i}{1-h_{ii}}\right) \\
&= \frac{Var(e_i)}{(1-h_{ii})^2} \\
&= \frac{\sigma^2(1-h_{ii})}{(1-h_{ii})^2} \\
&= \frac{\sigma^2}{1-h_{ii}}
\end{aligned}
$$

Standardized PRESS residual

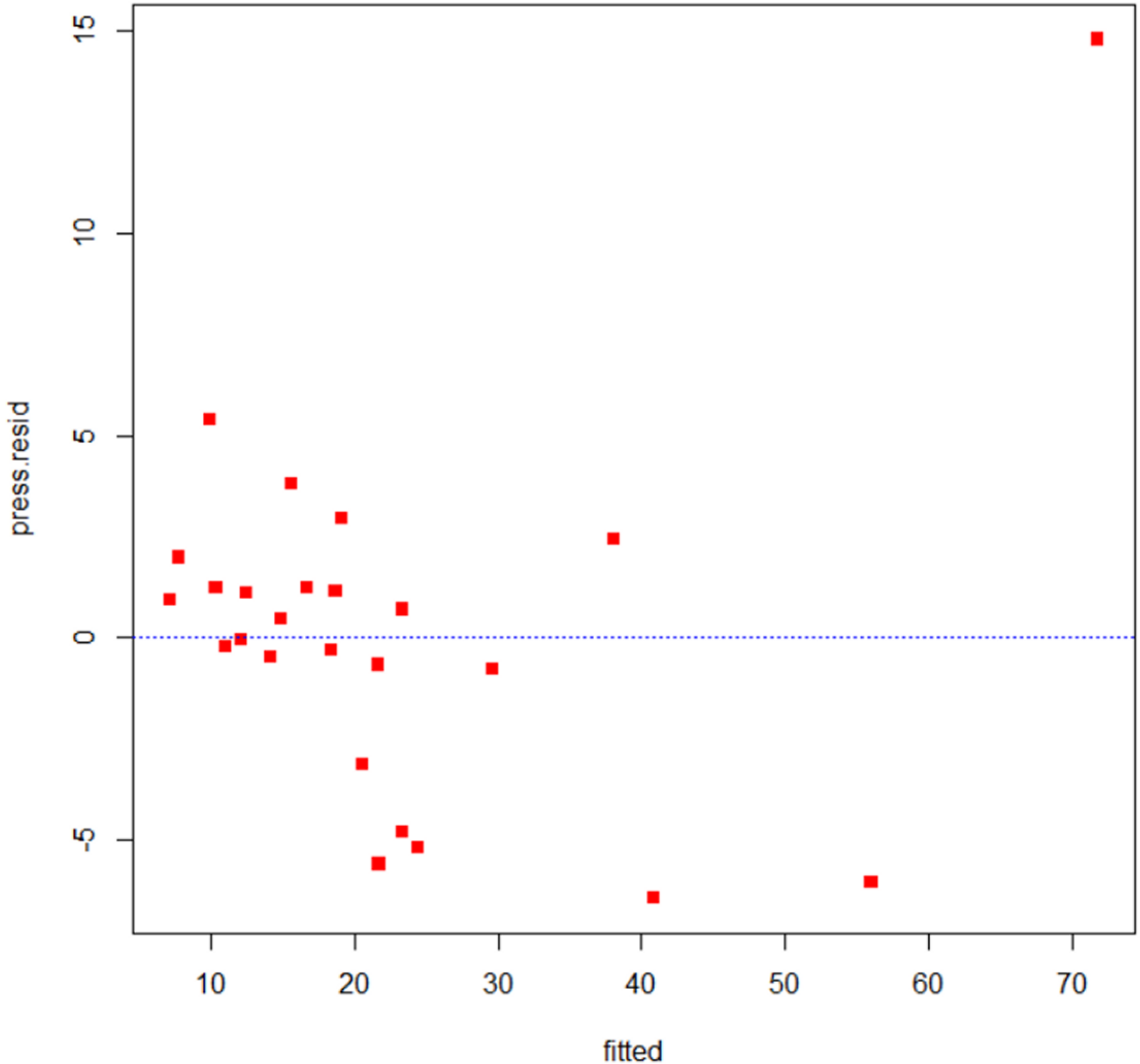
$$
\begin{aligned}
\frac{e_{(i)}-0}{\sqrt{Var(e_{(i)})}} &= \frac{e_i/(1-h_{ii})}{\sqrt{\sigma^2/(1-h_{ii})}} \\
&= \frac{e_i}{\sqrt{\sigma^2(1-h_{ii})}}
\end{aligned}
$$

Note: Studentized residuals

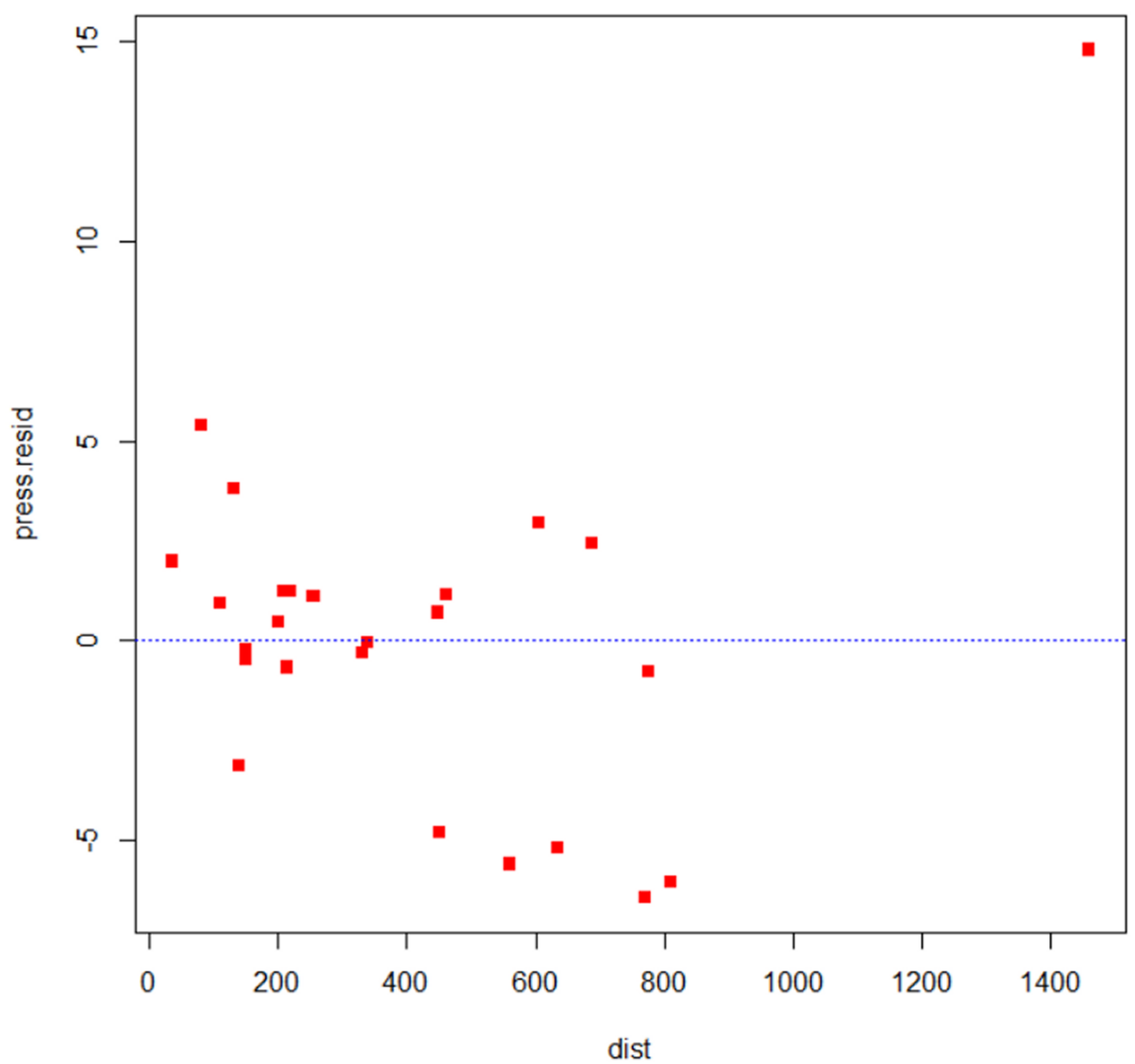
$$
r_i = \frac{e_i - 0}{\sqrt{MS_{Res}(1-h_{ii})}}, \quad i = 1, 2, ..., n.
$$

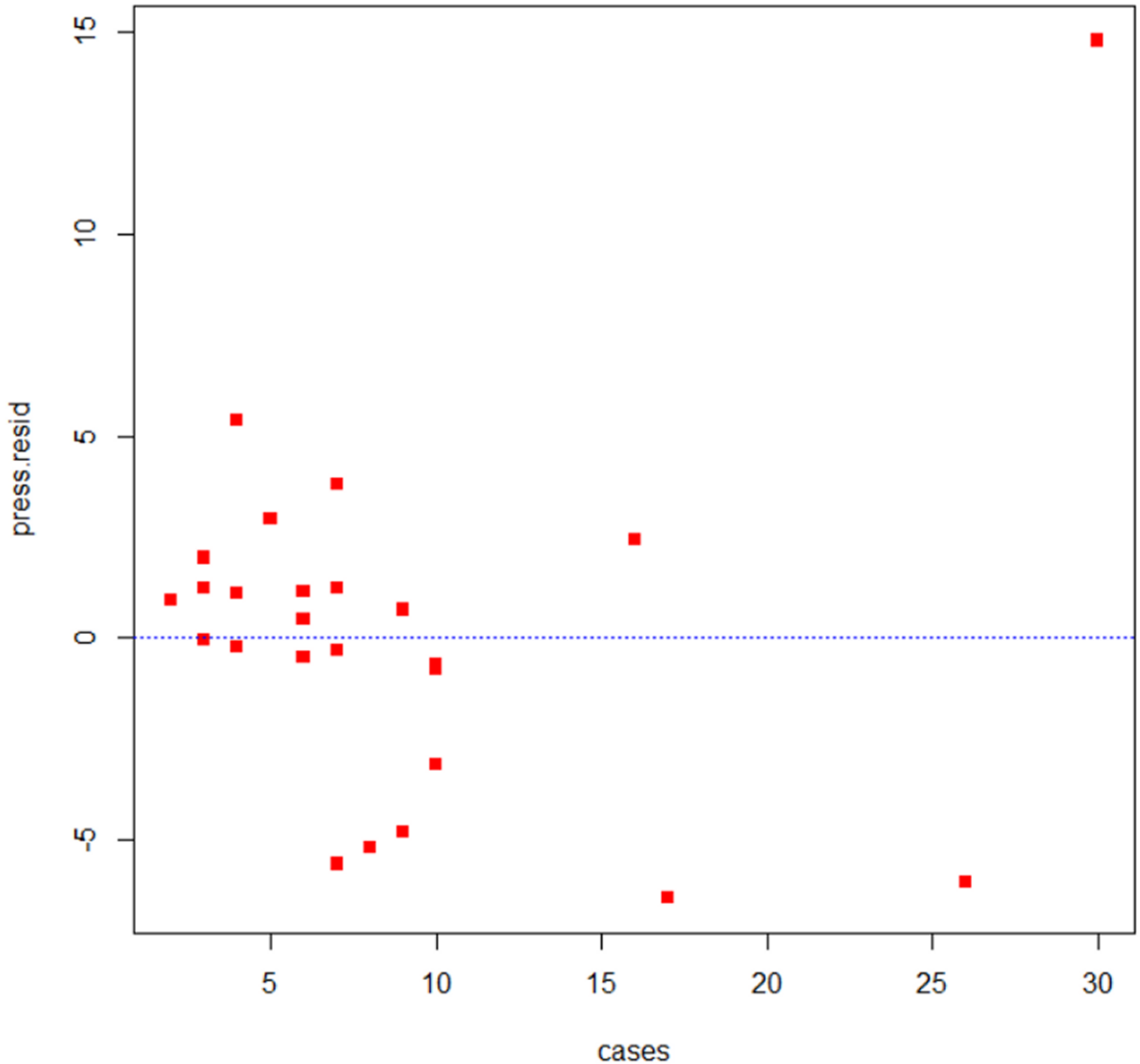
Example - Delivery time data

```
# calculate PRESS residuals
MSRes <- (summary(fitted.model)$sigma)^2
MSRes
press.resid <- resid/(1-Hii)
press.resid

# construct PRESS residual plots
plot(fitted,press.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(case,press.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(dist,press.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
```



15
10
5
0
-5
press.resid
10
20
30
40
50
60
70
fitted


press.resid
dist
15
10
5
0
-5
0
200
400
600
800
1000
1200
1400



press.resid
cases
15
10
5
0
-5
5
10
15
20
25
30

9. R-Student residuals

The R-Student residual is defined as

$$t_i \equiv \frac{e_i - 0}{\sqrt{S^2_{(i)}(1 - h_{ii})}}, \quad i = 1, 2, ..., n.$$

where $S^2_{(i)}$ is the estimate of σ^2 based on a data set with the ith observation removed.

It can be shown that

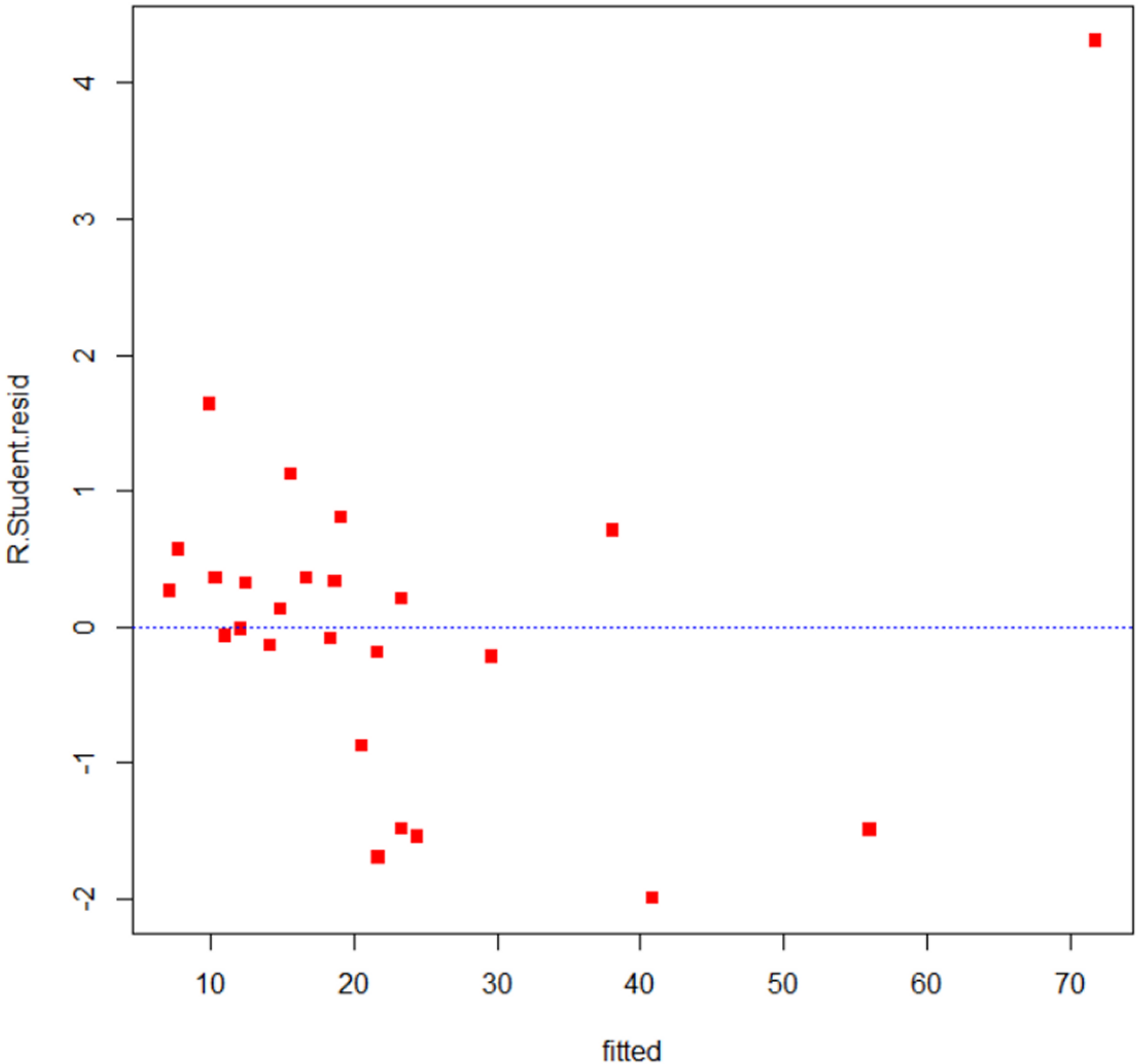
$$S^2_{(i)} = \frac{(n-p)MS_{Res} - e_i^2/(1 - h_{ii})}{n - p - 1}.$$

Note: Studentized residuals

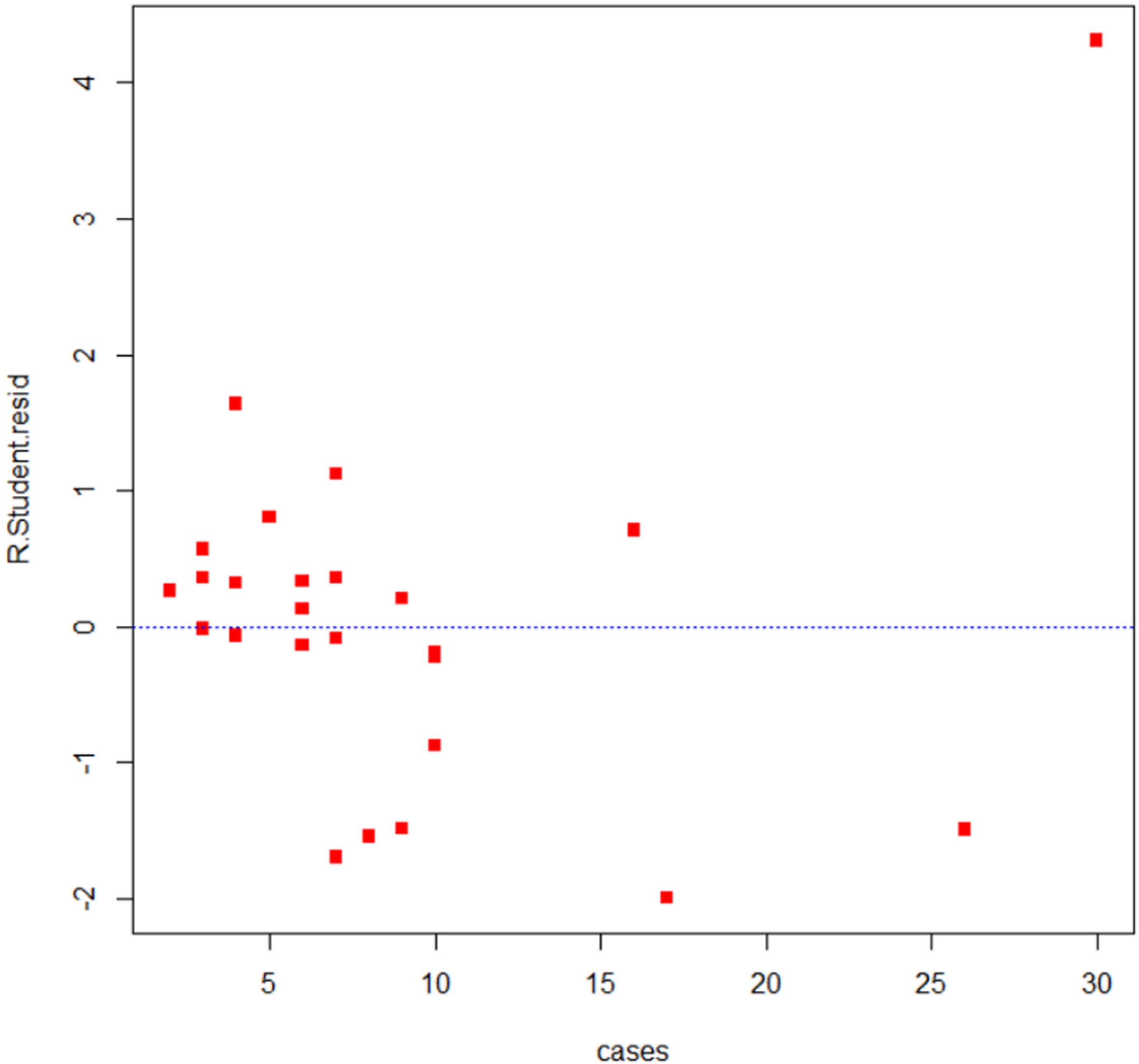
$$r_i = \frac{e_i - 0}{\sqrt{MS_{Res}(1 - h_{ii})}}, \quad i = 1, 2, ..., n.$$

Example - Delivery time data

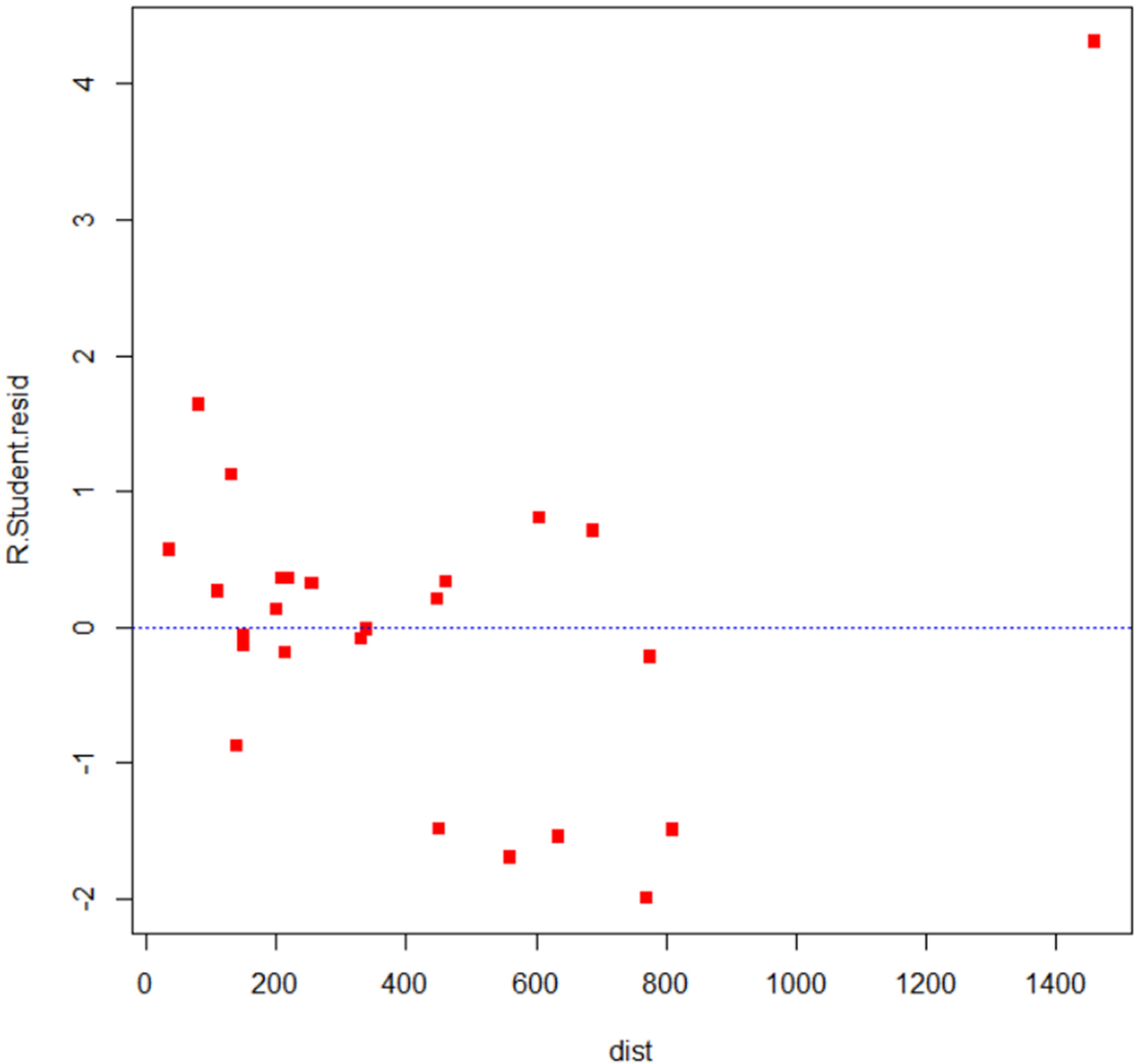
```
# calculate R-Student residuals
MSRes <- (summary(fitted.model)$sigma)^2
MSRes
S2 <- ((length(time)-3)*MSRes - resid^2/(1-Hii))/(length(time)-4)
R.Student.resid <- resid/sqrt(S2*(1-Hii))
R.Student.resid

# construct PRESS residual plots
plot(fitted,R.Student.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(case,R.Student.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
plot(dist,R.Student.resid,pch=15,col="red")
abline(h=0,col="blue",lty=6)
```



R.Student.resid
fitted
4
3
2
1
0
-1
-2
10
20
30
40
50
60
70



R.Student.resid
-2
-1
0
1
2
3
4
5
10
15
20
25
30
cases



R.Student.resid
dist
-2
-1
0
1
2
3
4
0
200
400
600
800
1000
1200
1400

10. Comparisons of resiuduals

Raw residual: $e_i = y_i - \hat{y}_i$

Standardized residual: $d_i = \dfrac{e_i}{\sqrt{MS_{Res}}}$

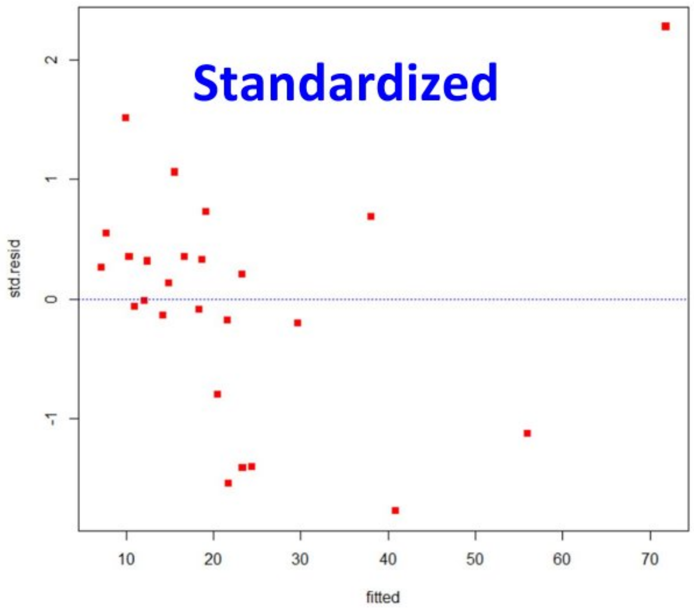
Studentized residual: $r_i = \dfrac{e_i}{\sqrt{MS_{Res}(1 - h_{ii})}}$

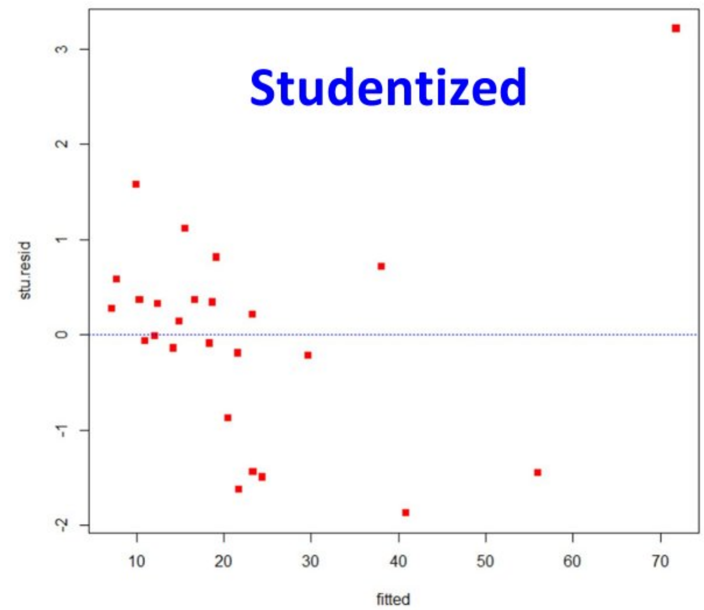
PRESS residuals: $e_{(i)} = \dfrac{e_i}{1 - h_{ii}}$.

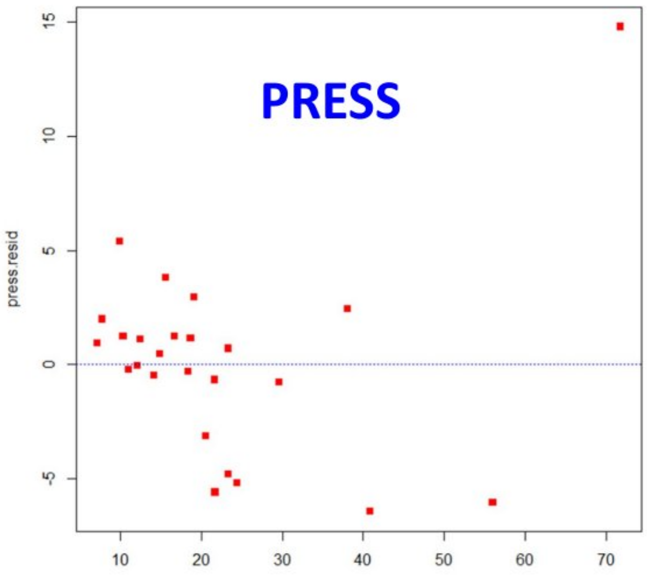
Standardized PRESS residual: $\dfrac{e_i}{\sqrt{\sigma^2(1 - h_{ii})}}$

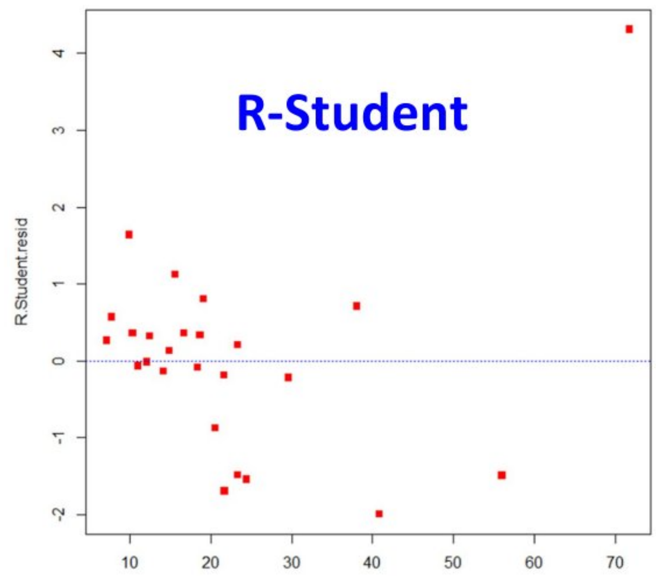
R-Student residual: $t_i \equiv \dfrac{e_i}{\sqrt{S_{(i)}^2(1 - h_{ii})}}$

Comparisons of residual plots





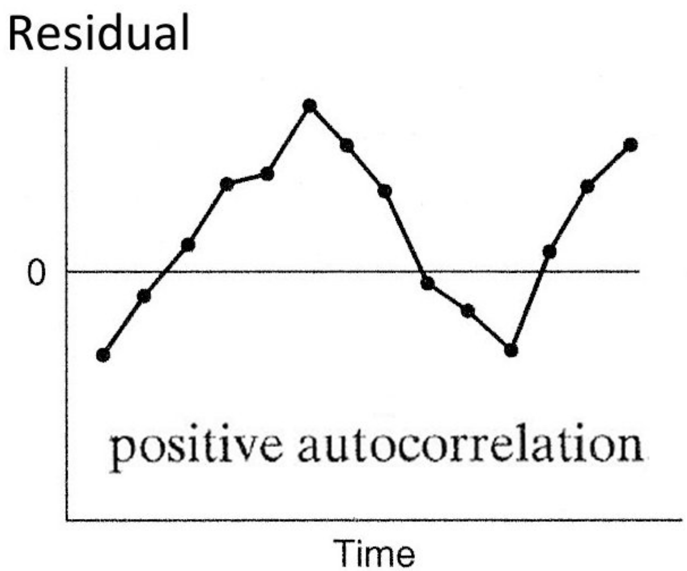


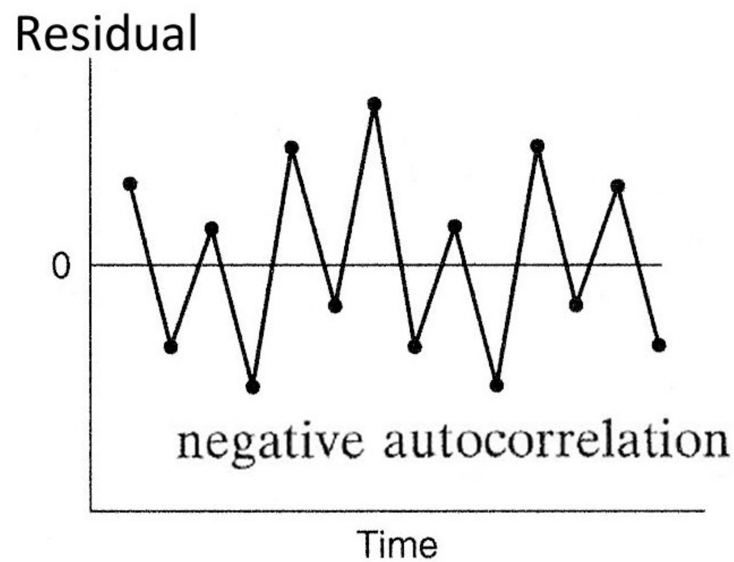


Plot of residuals in time sequence

(1) One key assumption of linear regression is that the responses are independent but in practice, the responses are often correlated. The correlation between model errors at different time periods is called autocorrelation.

(2) By plotting the residual against time will allow us to detect such correlation.





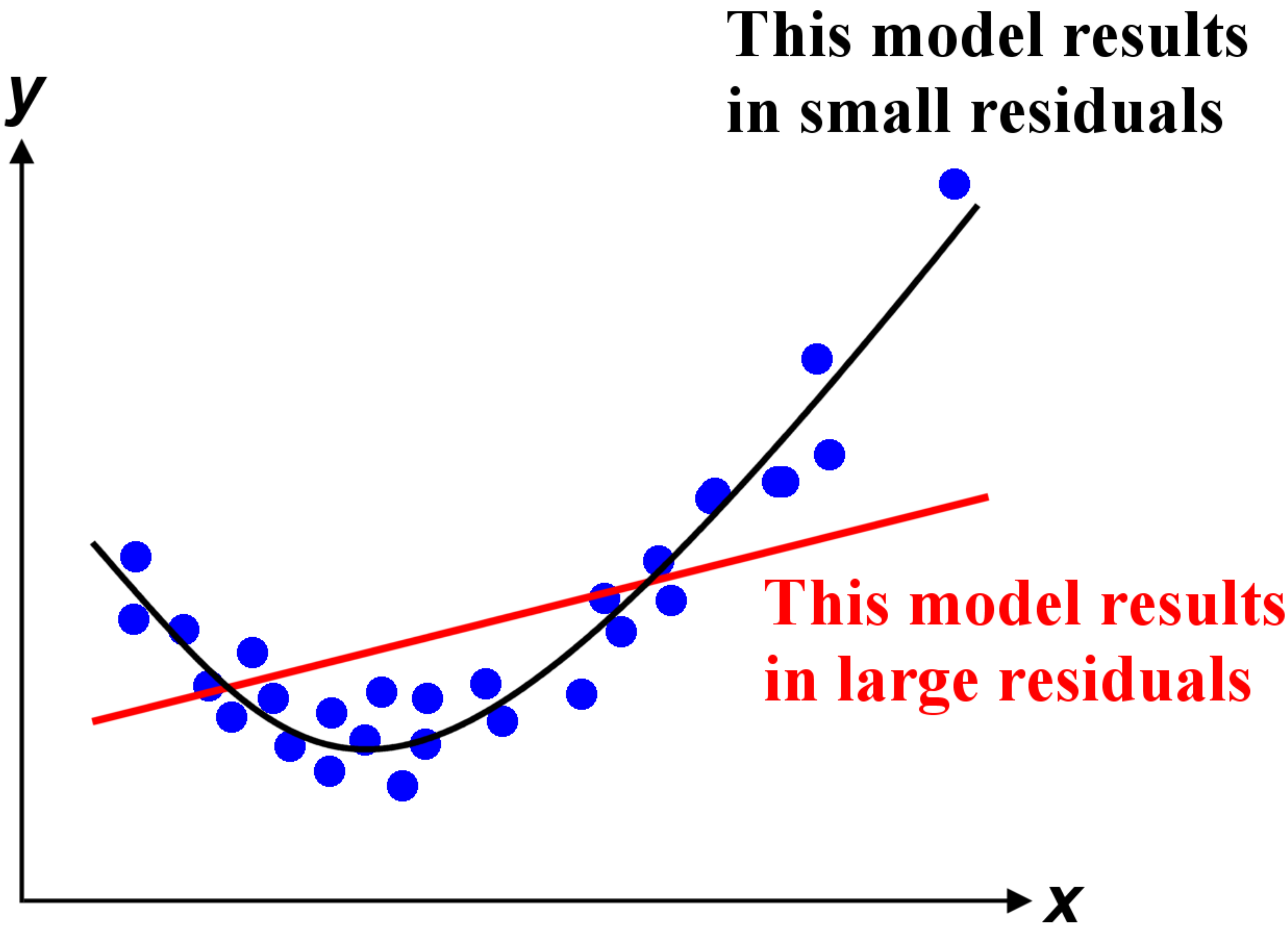
12. Detection and treatment of outliers

Residual plots and probability plots are useful in detecting outliers. Why do we need to worry about outliers and what should we do when outliers are found?

(1) Outliers often indicate certain inadequacy of our model.

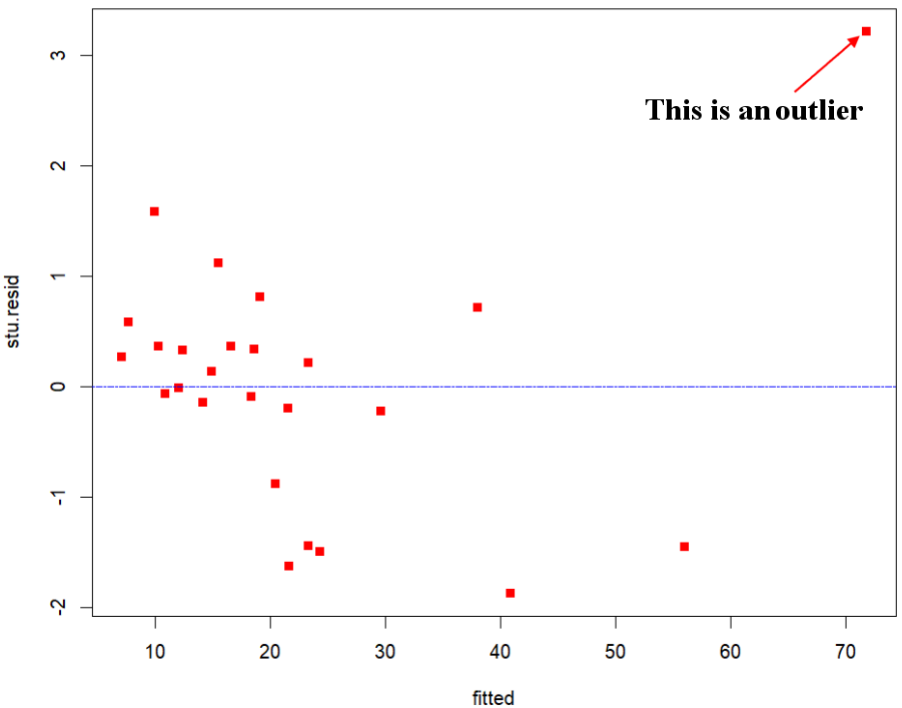
(2) Check to see if the observations were properly collected and recorded.

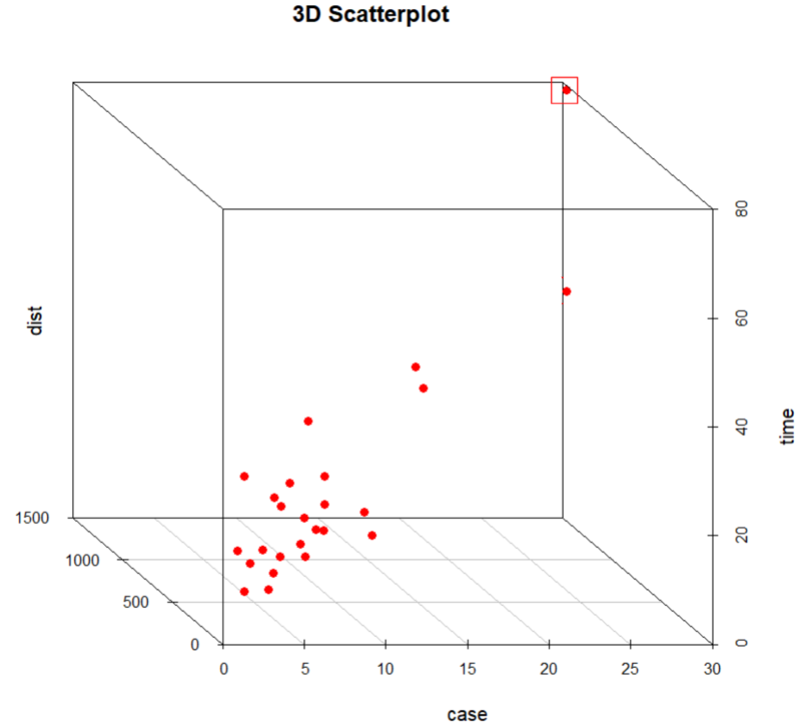
(3) The effect of outliers on the model may be easily checked by fitting the model without these observations.



y
This model results in small residuals
This model results in large residuals
x

$$time = \beta_0 + \beta_1\ case + \beta_2\ dist + \epsilon$$





A model with quadratic components in *case* and *time* could be better.

13. PRESS statistic

PRESS residuals: $e_{(i)} = \dfrac{e_i}{1 - h_{ii}}$

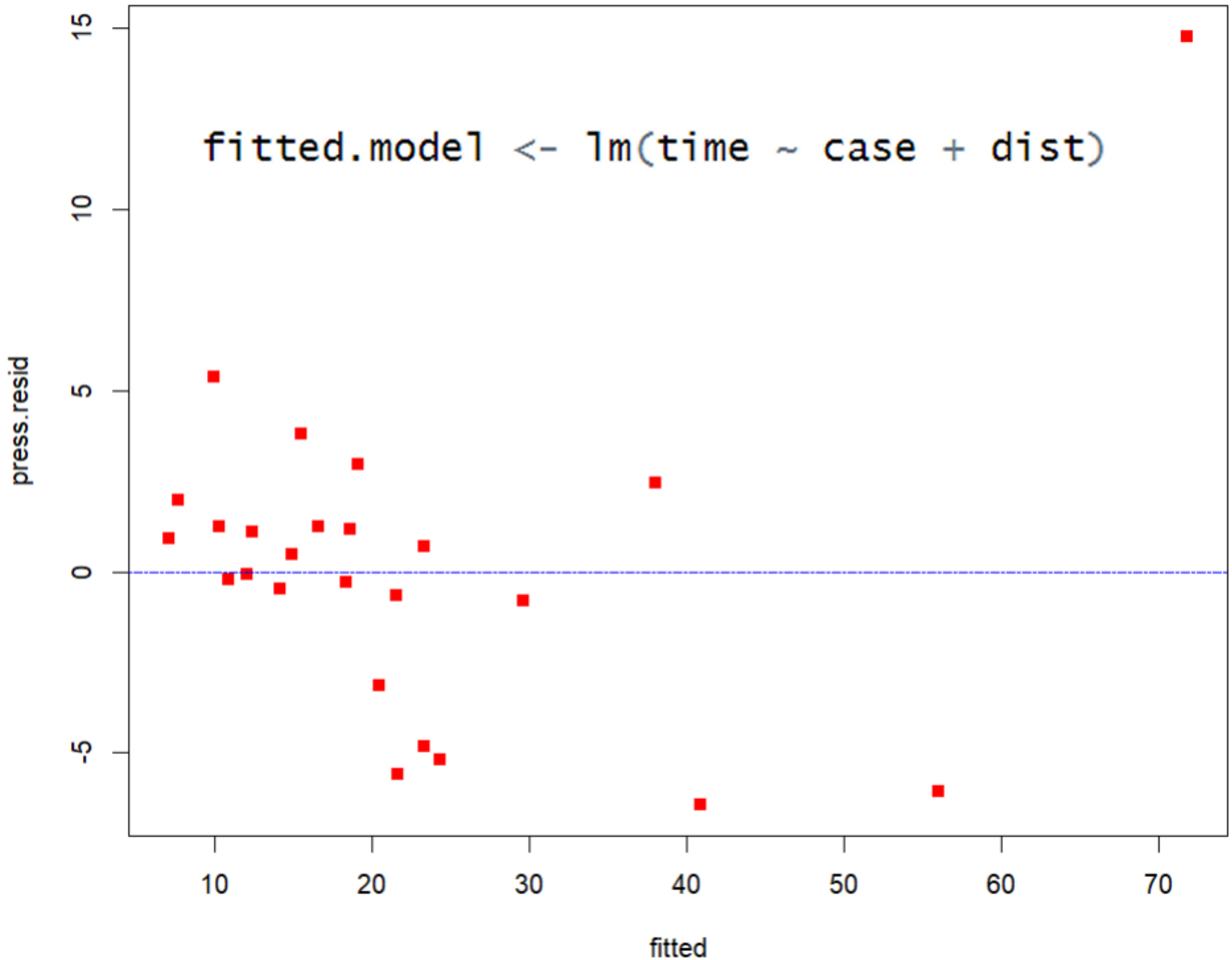
PRESS statistic:

$$\text{PRESS} = \sum_{i=1}^{n} e_{(i)}^2 = \sum_{i=1}^{n} \left(\frac{e_i}{1 - h_{ii}} \right)^2$$

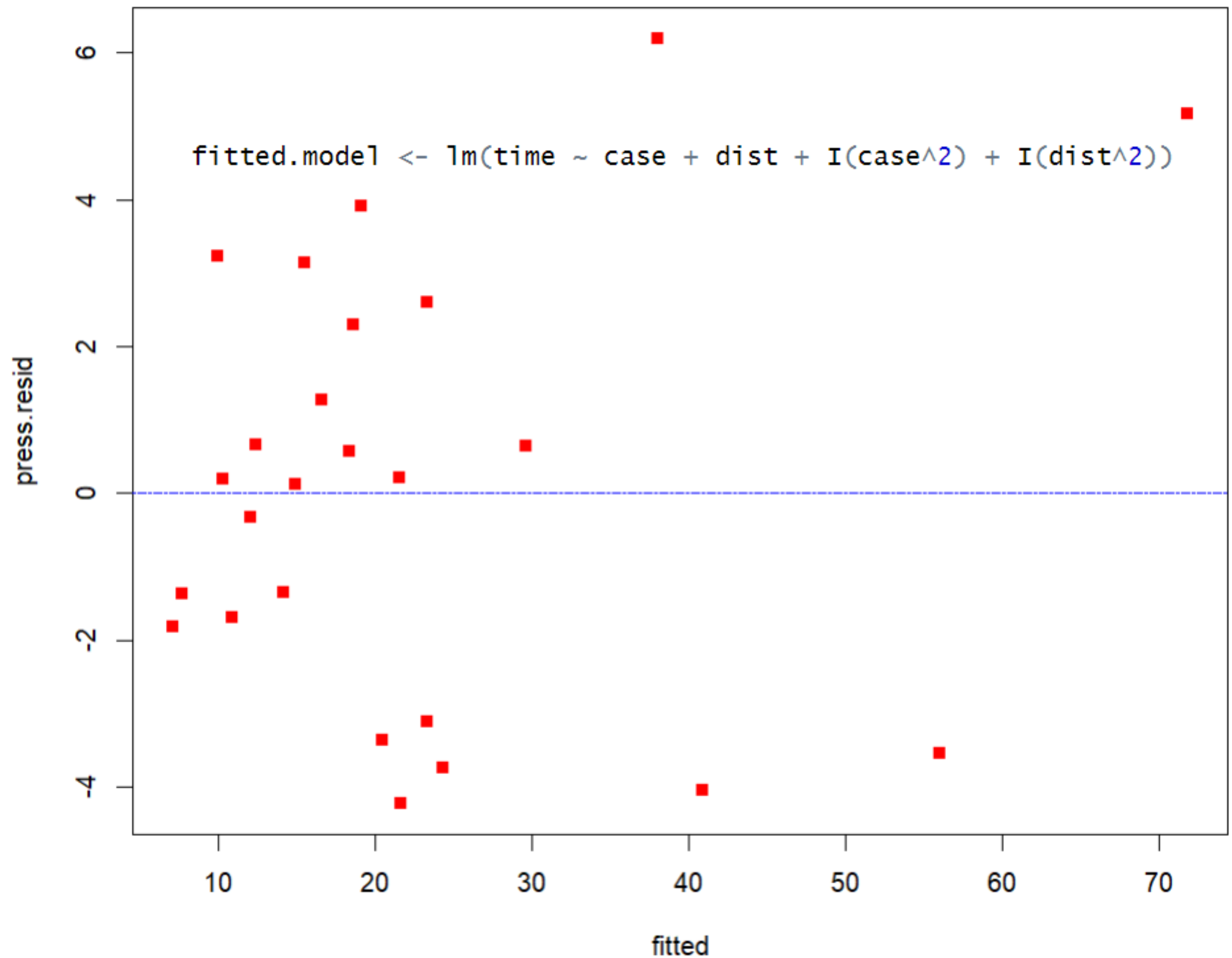
The PRESS statistic is often used as a measure of model quality. A small value of PRESS is desirable. One important use of the PRESS statistic is in comparing models.

Example - Delivery time data

```
> # PRESS statistic
> fitted.model <- lm(time ~ case + dist)
> resid <- resid(fitted.model)
> Hii <- hatvalues(fitted.model, fullHatMatrix = FALSE)
> press.resid <- resid/(1-Hii)
> PRESS <- sum(press.resid*press.resid)
> PRESS
[1] 459.0393
>
> # PRESS statistic
> fitted.model <- lm(time ~ case + dist + I(case^2) + I(dist^2))
> resid <- resid(fitted.model)
> Hii <- hatvalues(fitted.model, fullHatMatrix = FALSE)
> press.resid <- resid/(1-Hii)
> PRESS <- sum(press.resid*press.resid)
> PRESS
[1] 207.2597
```



fitted.model <- lm(time ~ case + dist)
press.resid
fitted
15
10
5
0
-5
10
20
30
40
50
60
70


fitted.model <- lm(time ~ case + dist + I(case^2) + I(dist^2))
press.resid
fitted
6
4
2
0
-2
-4
10
20
30
40
50
60
70

The End