36-617: Applied Linear Models

Graphical Tools for Transformations

Brian Junker
132E Baker Hall
brian@stat.cmu.edu

Announcements

- Quiz 02 – I'm grading this afternoon
- HW01 – Grades should be out soon
- Reading
  - This week: Sheather Ch 6 (diagnostics & transformations)
    - (supplemental: ISLR 3.3.3; G&H Ch 4)
  - Next week:
    - For Monday: Sheather 6.4, 6.5, 6.6
    - For Wednesday: At least Sheather 7.1, 7.2 (not sure I'll get farther in one lecture…) [Quiz will focus on 7.1, 7.2]
- HW 03 out on Canvas
  - Due Mon 1159pm

Outline

- From last time:
  - Variance Stabilization for Y
  - Can residual plots distinguish $y^{(1)} = \beta_0 + \beta_1 x^2 + \varepsilon$, vs. $y^{(2)} = (\beta_0 + \beta_1 x + \varepsilon)^2$?
  - Inverse Response Plot for Y
- Added Variable Plots
- Marginal Model Plots
- Perspective and recommendations

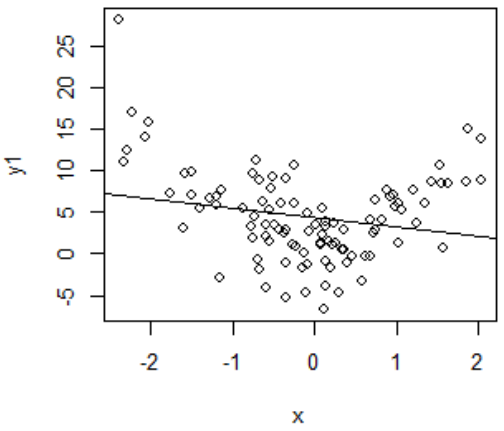
Can residual plots distinguish

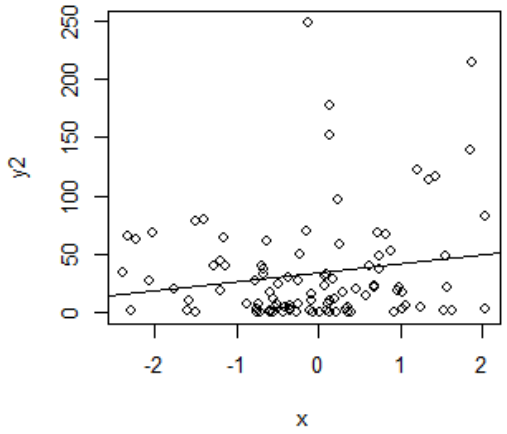
$y^{(1)} = \beta_0 + \beta_1 x^2 + \varepsilon$,

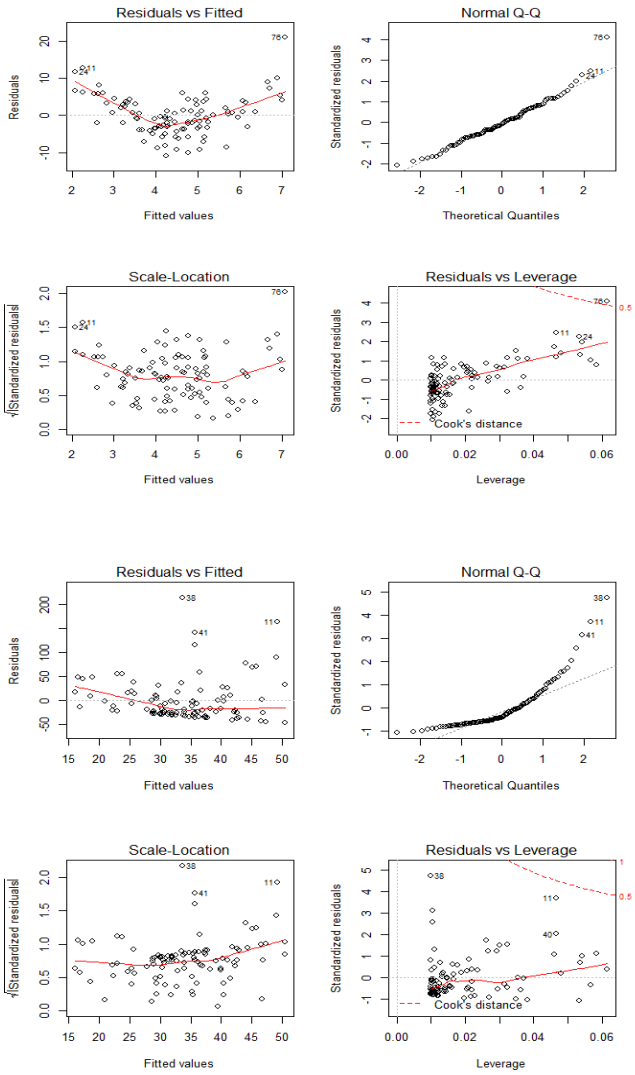
vs. $y^{(2)} = (\beta_0 + \beta_1 x + \varepsilon)^2$?

```
x <- rnorm(100,0,1)
y1 <- 1 + 3*x^2 + rnorm(100,0,4)
y2 <- (1 + 3*x + rnorm(100,0,4))^2
lm.1 <- lm(y1~x)
lm.2 <- lm(y2~x)
```

```
par(mfrow=c(2,2))
plot(x,y1); abline(lm.1)
plot(x,y2); abline(lm.2)
par(mfrow=c(2,2))
plot(lm.1)
plot(lm.2)
```







Functional form of Y: Inverse Response Plot

- Suppose

$$y_i = g(\beta_0 + \beta_1 x_i + \epsilon_i)$$

then of course

$$g^{-1}(y) = \beta_0 + \beta_1 x_i + \epsilon_i$$

- It turns out[1] that if x has an elliptically symmetric distribution, then g can be estimated from a plot of $\hat{y}_i$ vs y_i, where $\hat{y}_i$ are predicted values from

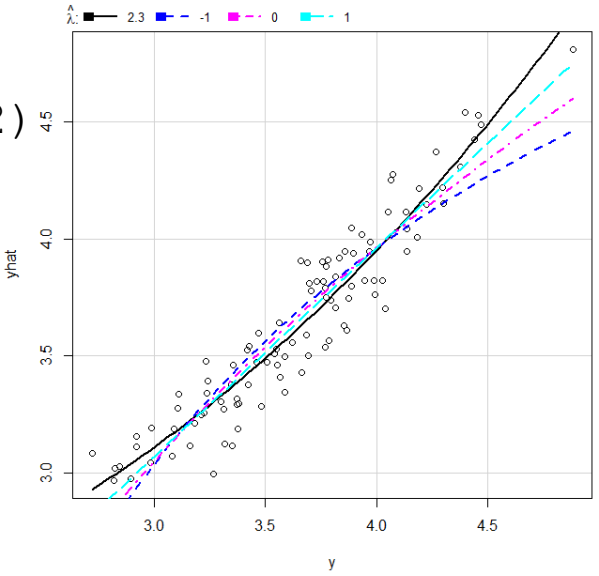
$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

[1]Li and Duan (1989), Cook and Weisberg (1994)

Implementing Inverse Response Plots In R

- `library(car)`
  ("Companion to Applied Regression")
  - `invResPlot()`: show inverse response plot ($\hat{y}_i$ vs.y_i) and calculate the power λ for y_i^{λ} by nonlinear least-squares(*)

```
> z <- rnorm(100,4,1)
> y <- (1 + 3*z + rnorm(100,0,.25))^(1/2)
> lm.1 <-  lm(y ~ z)
> invResPlot(lm.1)
  lambda       RSS
1  2.300377 1.711786  ← y' = y^(2.3)
2 -1.000000 2.711177  ← y' = 1/y
3  0.000000 2.211967  ← y' = log(y)
4  1.000000 1.874950  ← y' = y
```



(*) Specify particular lamdas to try with the lambda=c(…) argument.

Added-Variable Plots (add Z? or f(Z)?)

- Suppose the true model is

$$Y = X\beta + Z\gamma + \epsilon$$

- Let us fit the models

$$\begin{aligned} Y &= X\beta + \epsilon^{(1)} \text{ with residuals } \hat{e}^{(1)} = (I - H_X)Y \\ Z &= X\beta + \epsilon^{(2)} \text{ with residuals } \hat{e}^{(2)} = (I - H_X)Z \end{aligned}$$

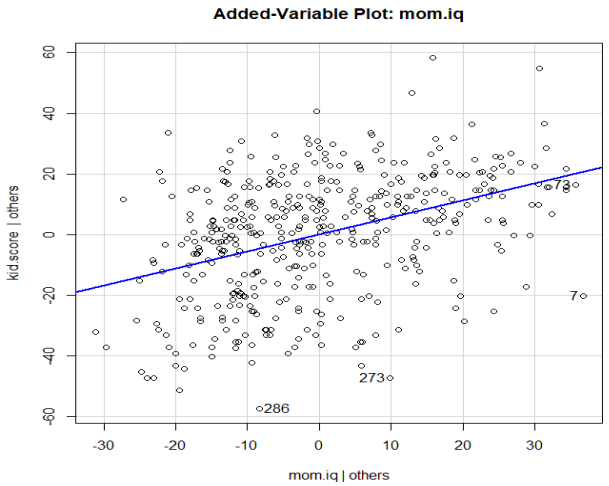
- If we multiply the true model by *(I-H_X)*, we get

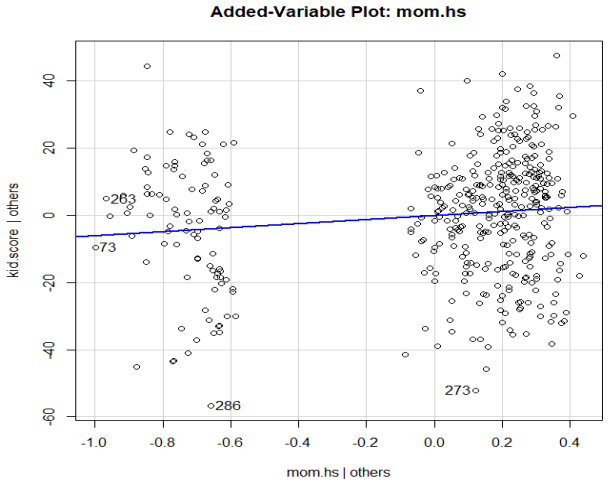
$$\begin{aligned} (I - H_X)Y &= (I - H_X)X\beta + (I - H_X)Z\gamma + (I - H_X)\epsilon \\ \hat{e}^{(1)} &= 0 + \hat{e}^{(2)}\gamma + \epsilon^* \end{aligned}$$

so, plotting (or regressing) $\hat{e}^{(1)}$ on $\hat{e}^{(2)}$ will reveal γ !

Added-Variable Plots – Example…

```
> library(car)
> lm.3

Call:
lm(formula = kid.score ~ mom.iq +
mom.hs, data = kidiq)

Coefficients:
(Intercept)    mom.iq    mom.hs
    25.7315    0.5639    5.9501

> avPlot(lm.3,"mom.iq")
> avPlot(lm.3,"mom.hs")
```



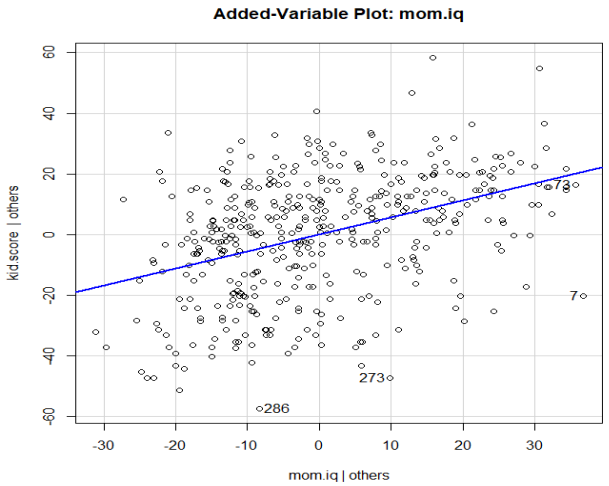


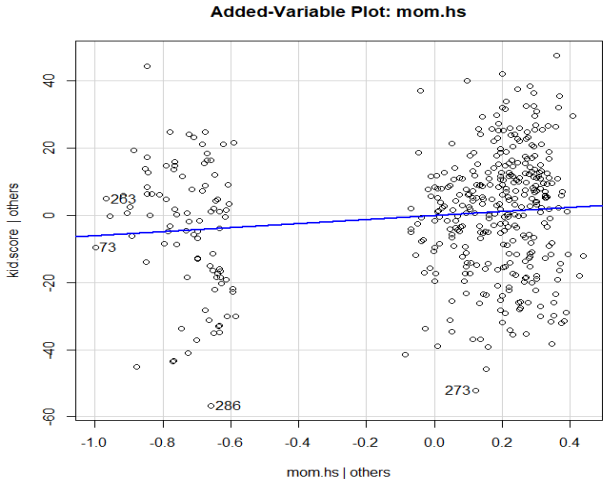
Added-Variable Plots – Interpretations

- Shows γ as the effect of *Z* after controlling for *X*, on *Y*, after controlling for *X*
- Allows you to visually assess the importance of γ , after controlling for all the other X's.
  - A visual form of the t-statistic!
- Also allows you to check for nonlinearity in predicting *Y* from *Z*, after controlling for *X*
- Another plot that allows us to assess nonlinearity is the "marginal model plot" – *later in this lecture*

Added-Variable Plots – Example…

```
> library(car)
> lm.3

Call:
lm(formula = kid.score ~ mom.iq +
mom.hs, data = kidiq)

Coefficients:
(Intercept)    mom.iq    mom.hs
    25.7315    0.5639    5.9501

> avPlot(lm.3,"mom.iq")
> avPlot(lm.3,"mom.hs")
```



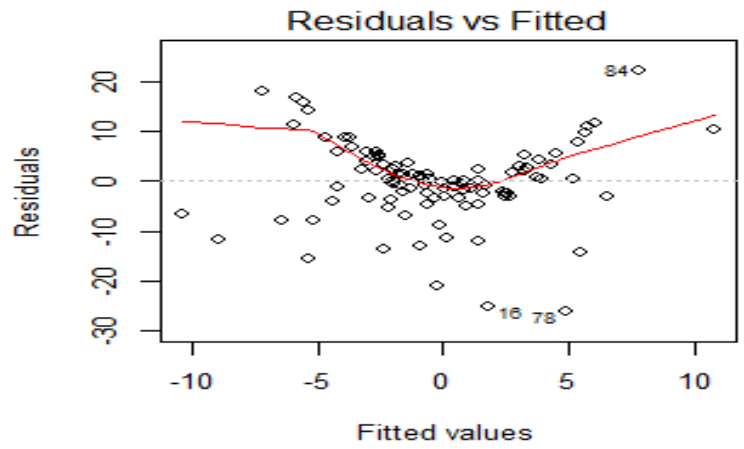


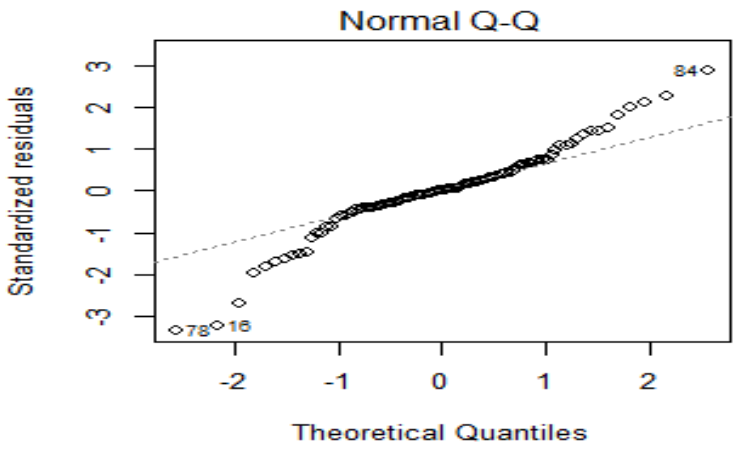
An example

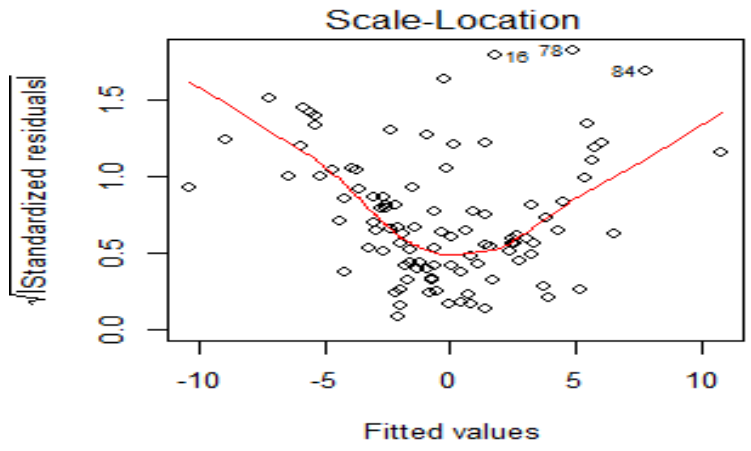
```
> library(car)
> x1 <- rnorm(100)
> x2 <- rnorm(100)
> y <- 1 + x1 + 2*x2 +
+ 10*x1*x2 + rnorm(100)
>
> lm.x1px2 <- lm(y ~ x1 + x2)
> lm.x1mx2 <- lm(y ~ x1 * x2)
>
> summary(lm.x1px2)

Call:
lm(formula = y ~ x1 + x2)

Coefficients:
            Est   SE     t p
(Int) -0.05 0.82 -0.06 0.95
x1     1.77 0.87  2.03 0.04 *
x2     3.44 0.82  4.20 0.00 ***
---

Residual standard error: 8.13 on
97 degrees of freedom
Multiple R-squared:  0.1722,
Adjusted R-squared:  0.1551
F-statistic: 10.09 on 2 and 97 DF,
p-value: 0.0001045
```

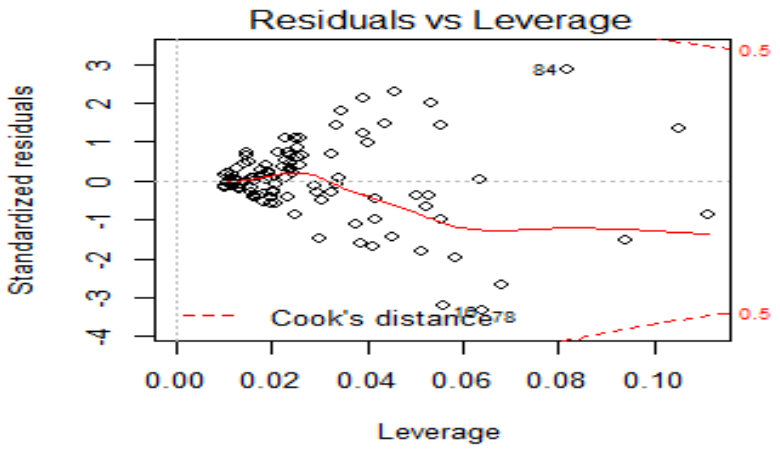
Casewise Diagnostic Plots

```
> par(mfrow=c(2,2))
> plot(lm.x1px2)
```



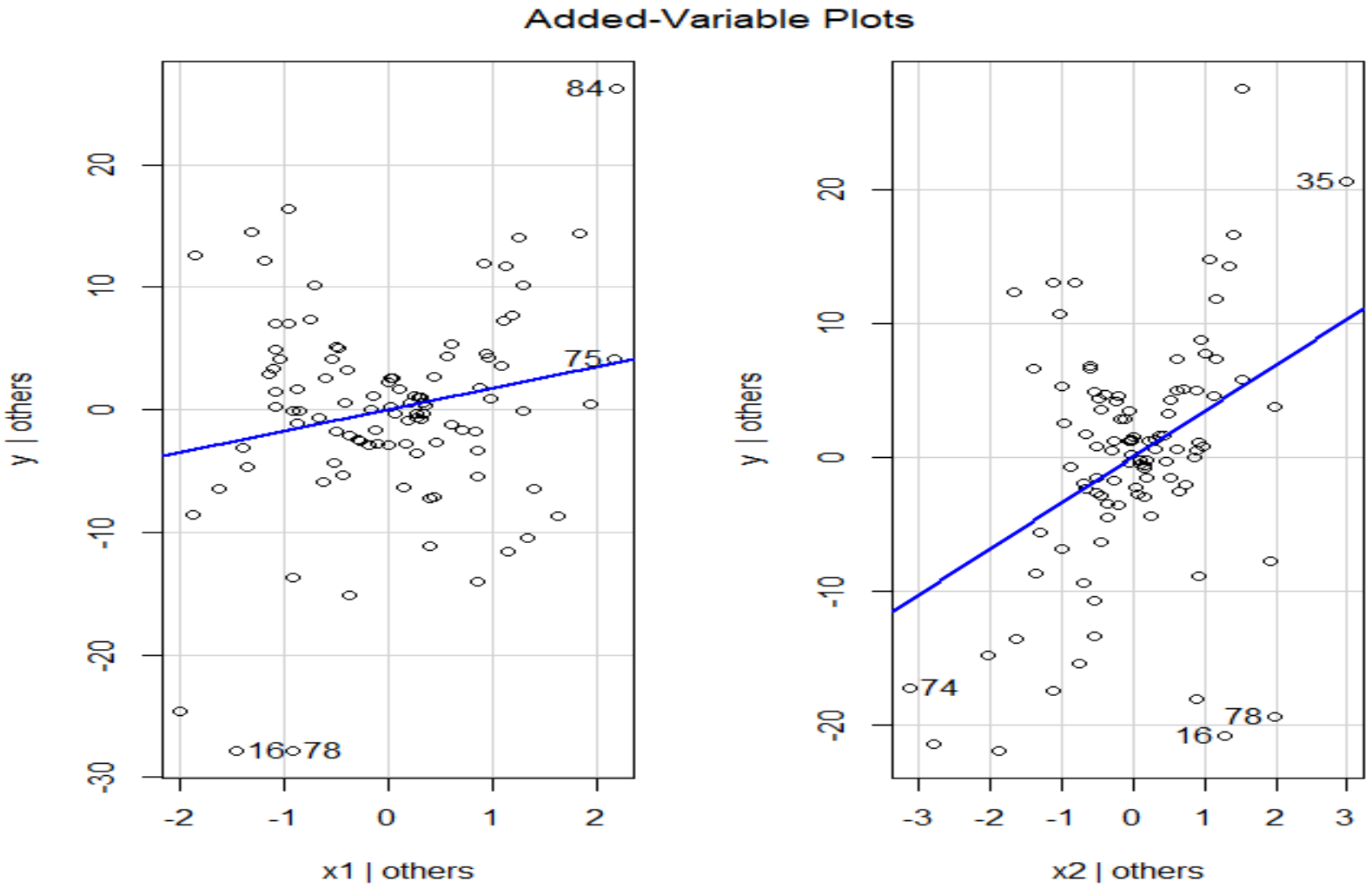






Added Variable Plots

> avPlots(lm.x1px2)

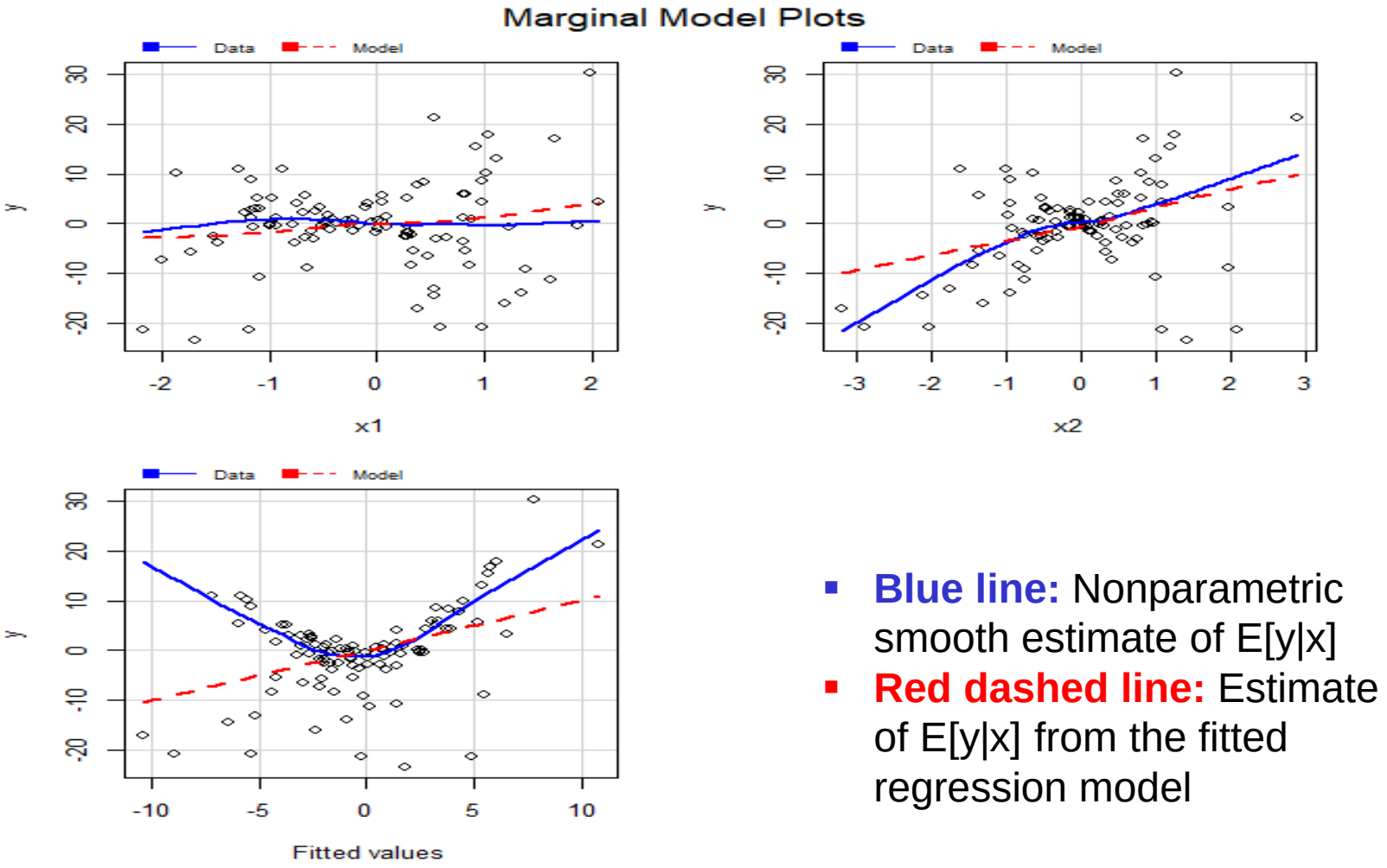


Marginal Model Plot

- The idea is very simple:
  - Plot y against a predictor (e.g. one of the x_j's or even $\hat{y}$); we'll call it x.
  - Use a nonparametric regression procedure (e.g. loess) to estimate E[y|x]
  - Use the fitted model to estimate E[y|x]
- The two should agree. If they do not,
  - x or y may need to be transformed
  - A term may be missing in the model
  - (or both!)

Marginal Model Plots

> mmps(lm.x1px2)



- **Blue line:** Nonparametric smooth estimate of E[y|x]
- **Red dashed line:** Estimate of E[y|x] from the fitted regression model

The “right” model (with interaction)

```
> summary(lm.x1mx2)

Call:
lm(formula = y ~ x1 * x2)


Coefficients:
        Est    SE      t     p
(Int)  0.77  0.11   7.06  0.00 ***
x1     0.69  0.12   5.93  0.00 ***
x2     2.03  0.11  18.33  0.00 ***
x1:x2  9.90  0.13  73.58  0.00 ***
---
```

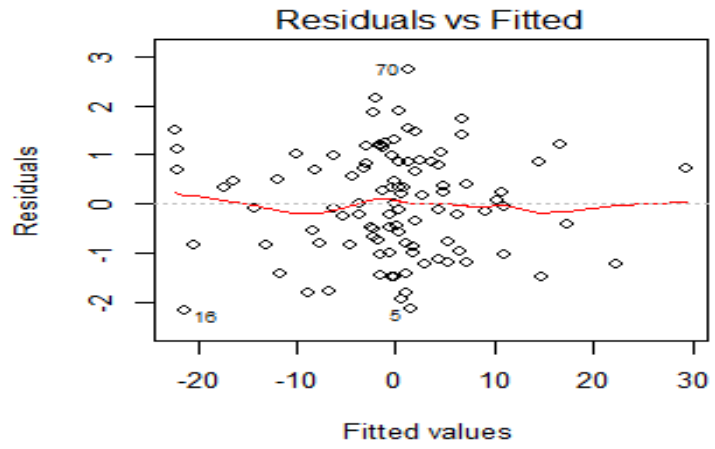
Residual standard error: 1.079 on 96 degrees of freedom

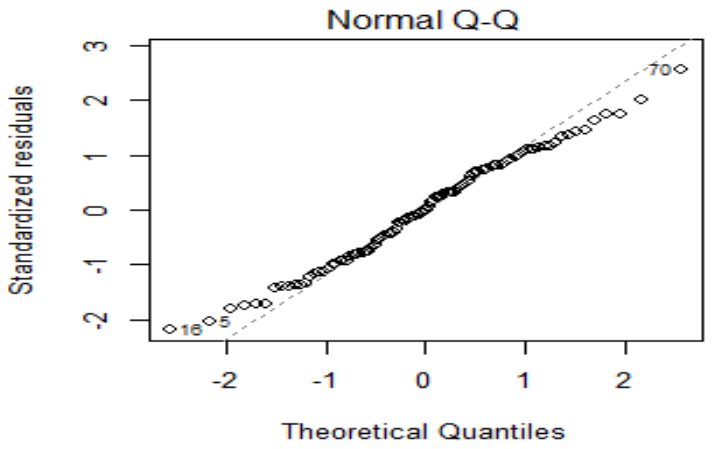
Multiple R-squared: 0.9856, Adjusted R-squared: 0.9851

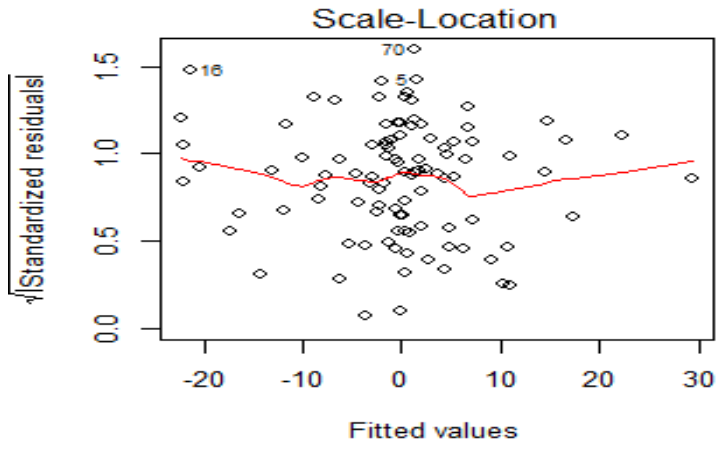
F-statistic: 2187 on 3 and 96 DF, p-value: < 2.2e-16

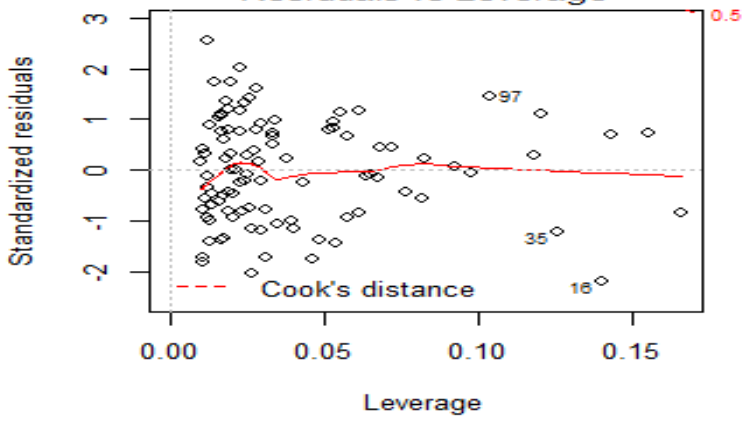
Casewise Diagnostic Plots

```
> par(mfrow=c(2,2))
> plot(lm.x1mx2)
```



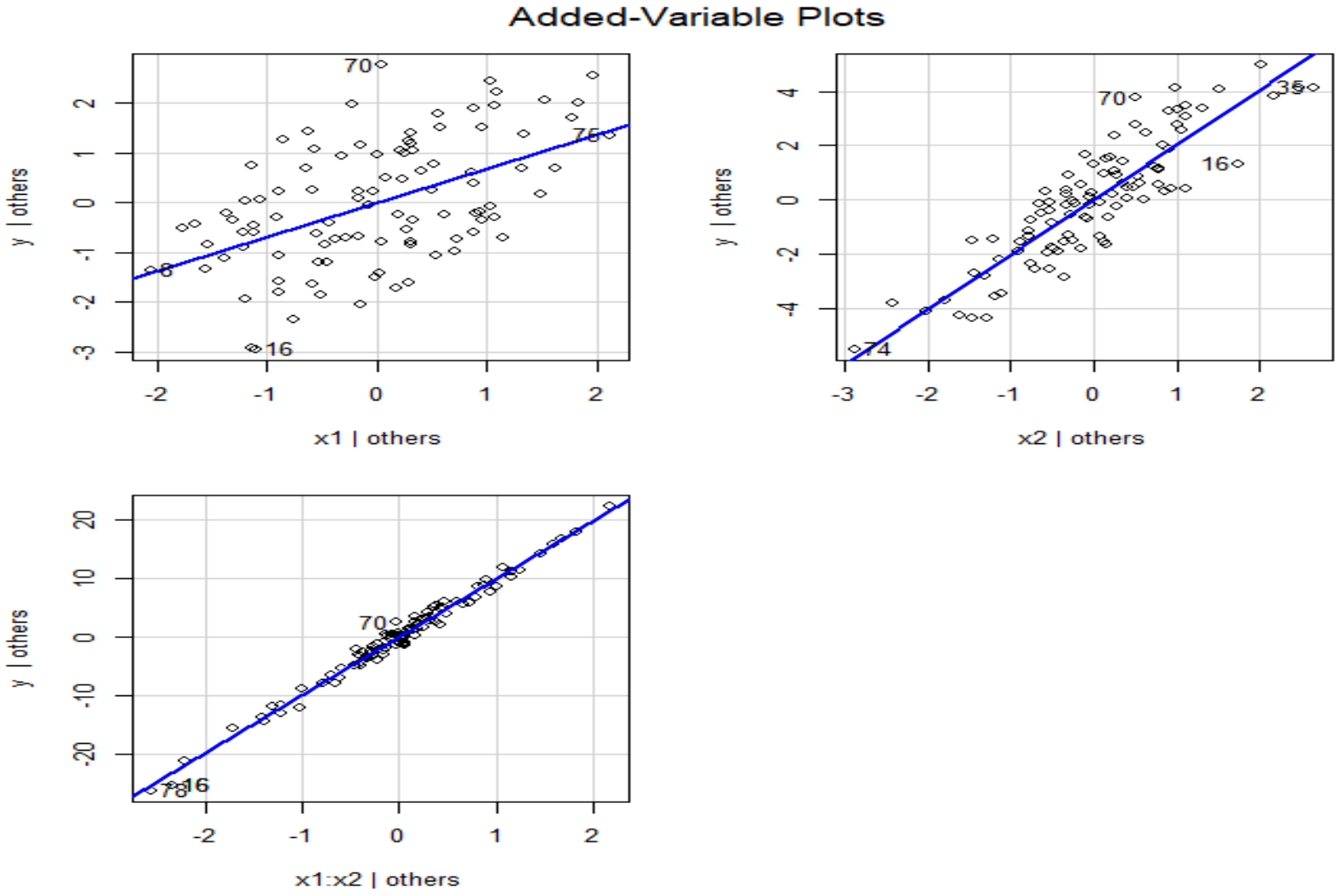






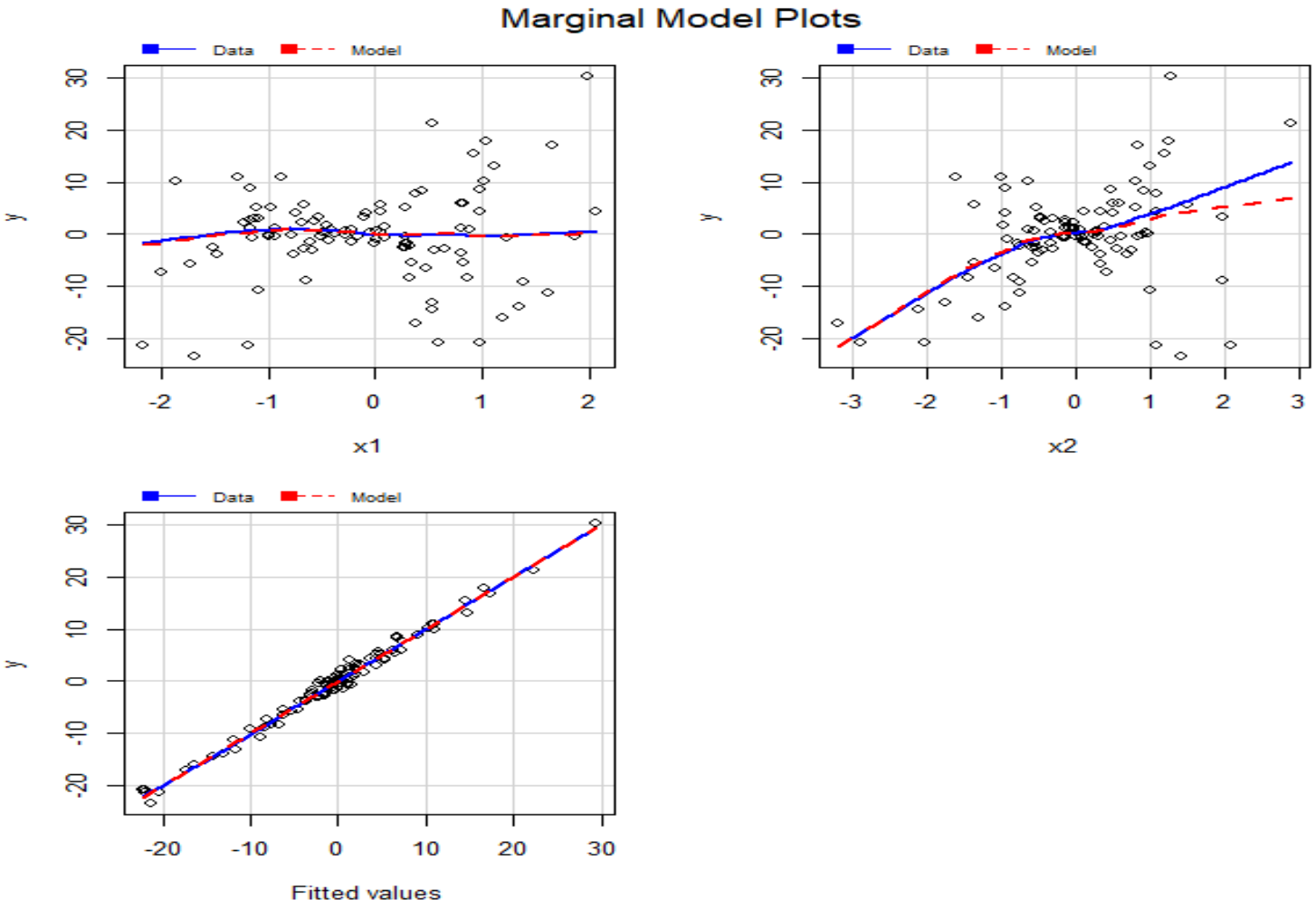
Added Variable Plots

> avPlots(lm.x1mx2)



Marginal Model Plots

> mmps(lm.x1mx2)

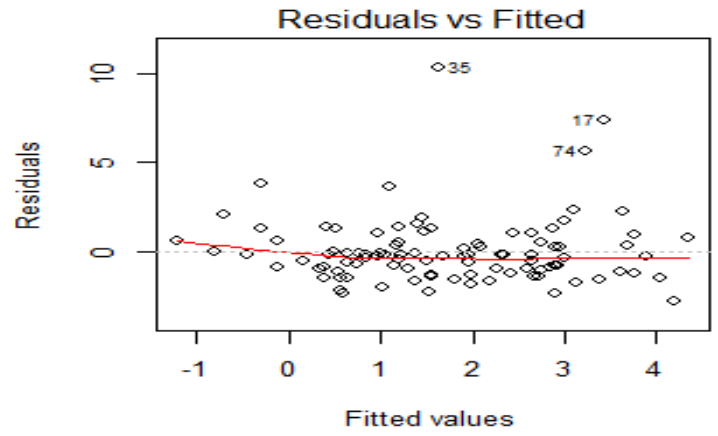


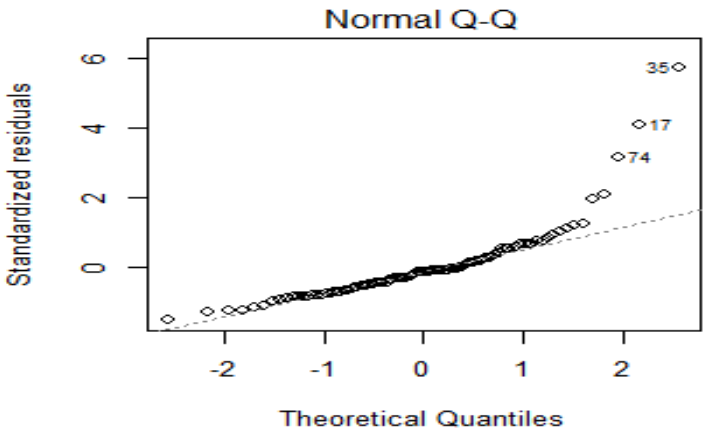
Another example

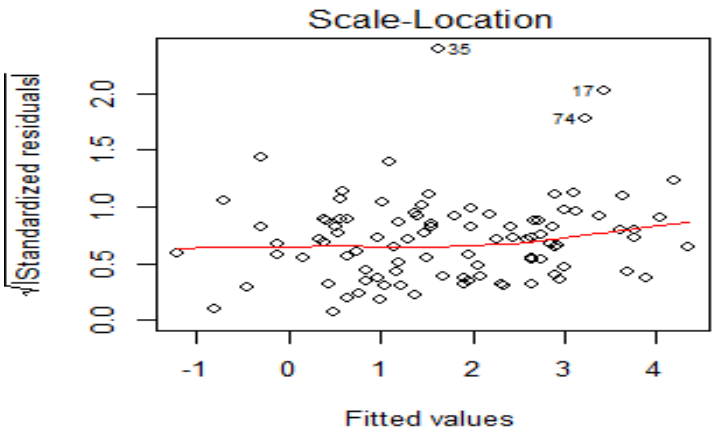
```
> y <- 1 + x1 + x2^2 +
+ rnorm(100)
>
> lm.x1px2 <- lm(y ~ x1 + x2)
> lm.x1mx2 <- lm(y ~ x1 * x2)
> lm.x1px2sq <- lm(y ~ x1 +
+ I(x2^2))
>
> summary(lm.x1px2)

Call:
lm(formula = y ~ x1 + x2)

Coefficients:
            Est   SE      t    p
(Int)  1.82 0.19  9.49 0.00 ***
x1     1.24 0.20  6.08 0.00 ***
x2    -0.30 0.19 -1.55 0.12
---
Residual standard error: 1.904
on 97 degrees of freedom
Multiple R-squared:  0.3014,
Adjusted R-squared:  0.287
F-statistic: 20.93 on 2 and 97
DF,  p-value: 2.779e-08
```

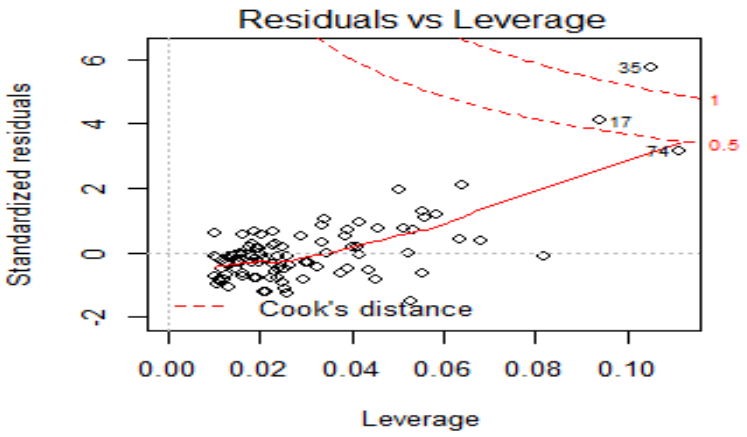
Casewise Diagnostic Plots

```
> par(mfrow=c(2,2))
> plot(lm.x1px2)
```



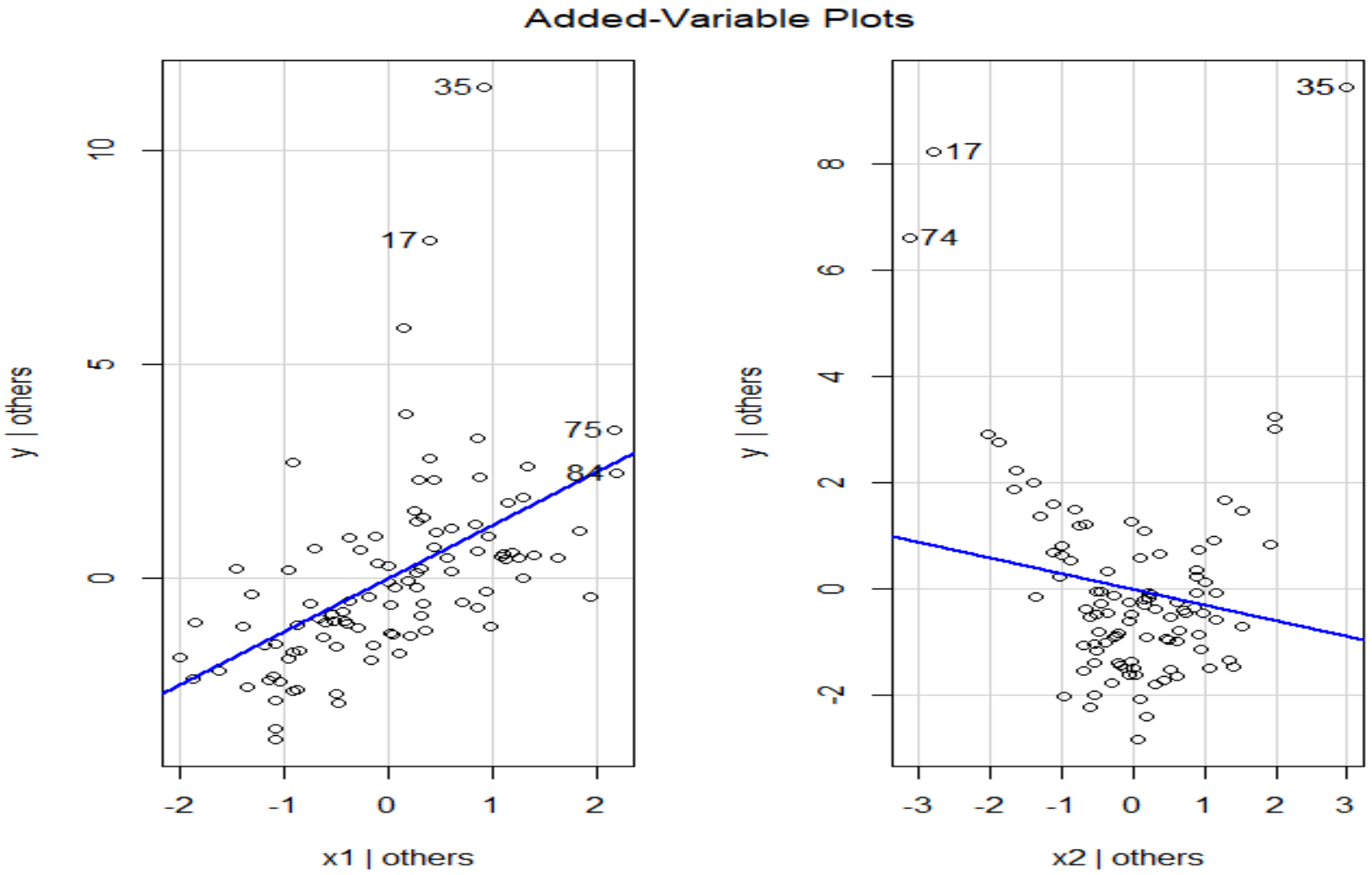






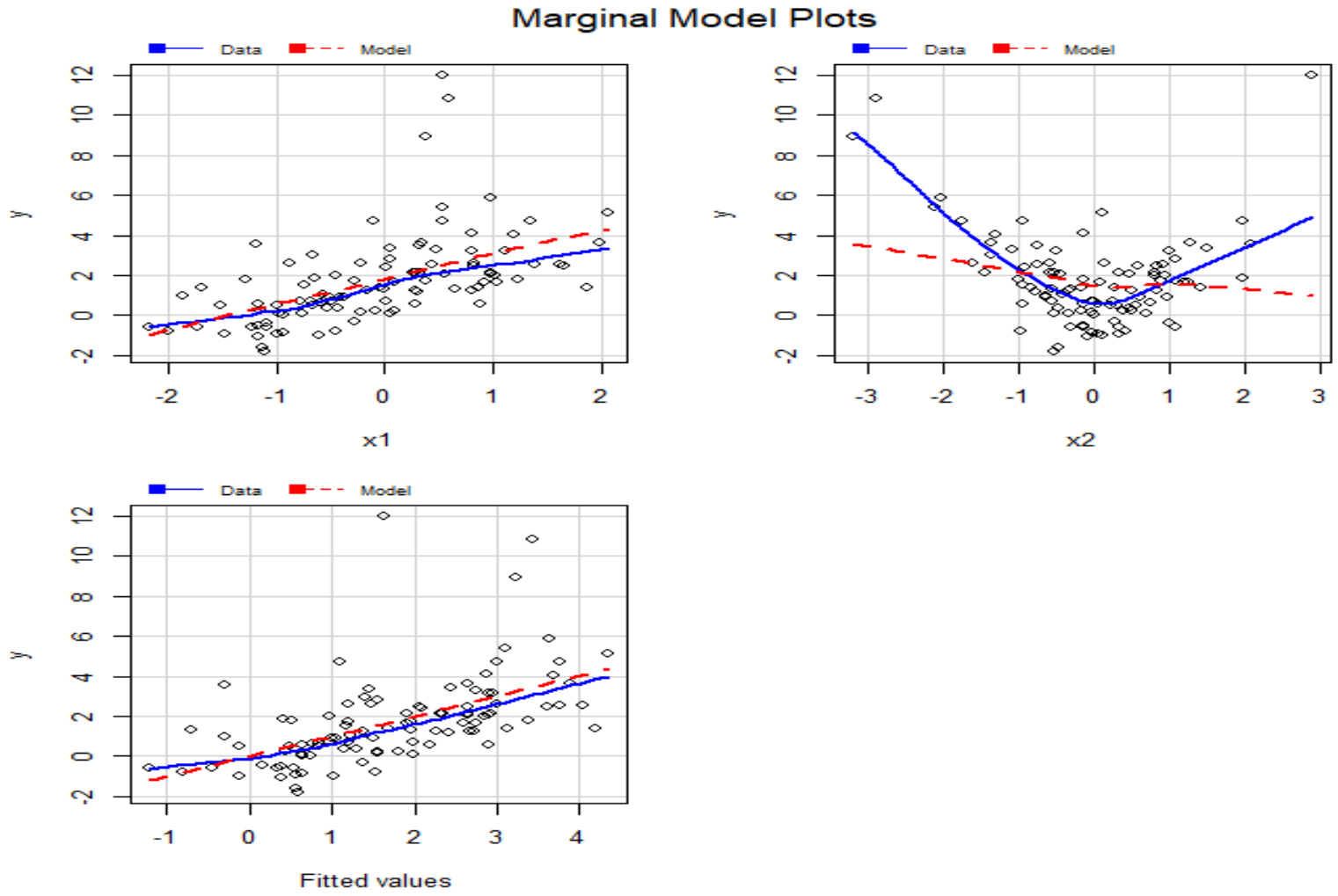
Added Variable Plots

> avPlots(lm.x1px2)



Marginal Model Plots

> mmps(lm.x1px2)

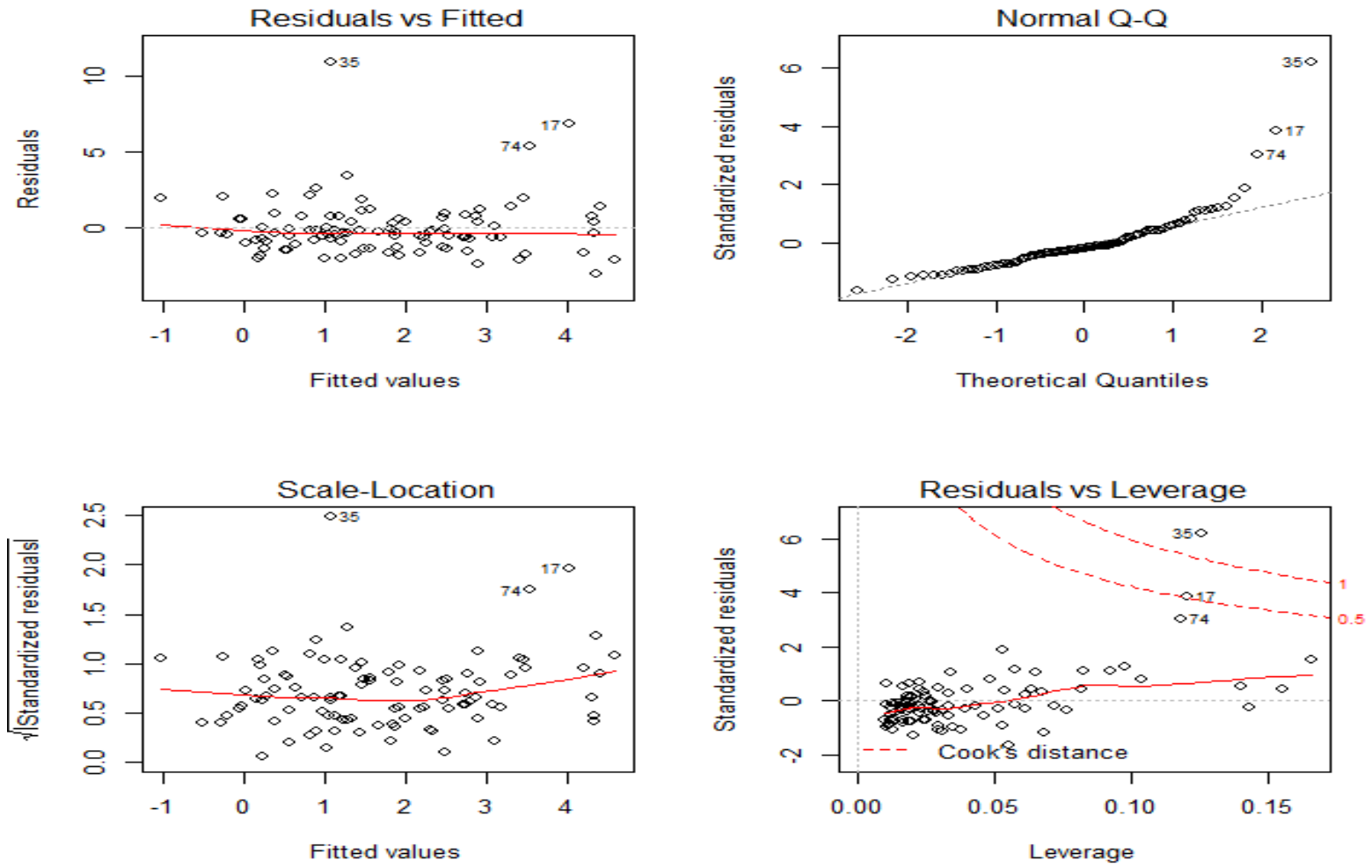


What if we think an interaction will fix it?

```
> summary(lm.x1mx2)

Call:
lm(formula = y ~ x1 * x2)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.7774     0.1895   9.380 3.19e-15 ***
x1            1.2891     0.2024   6.370 6.52e-09 ***
x2           -0.2321     0.1922  -1.208   0.2301
x1:x2        -0.4607     0.2340  -1.969   0.0518 .
---
Residual standard error: 1.877 on 96 degrees of freedom
Multiple R-squared:  0.3286,    Adjusted R-squared:  0.3076
F-statistic: 15.66 on 3 and 96 DF,  p-value: 2.29e-08
```

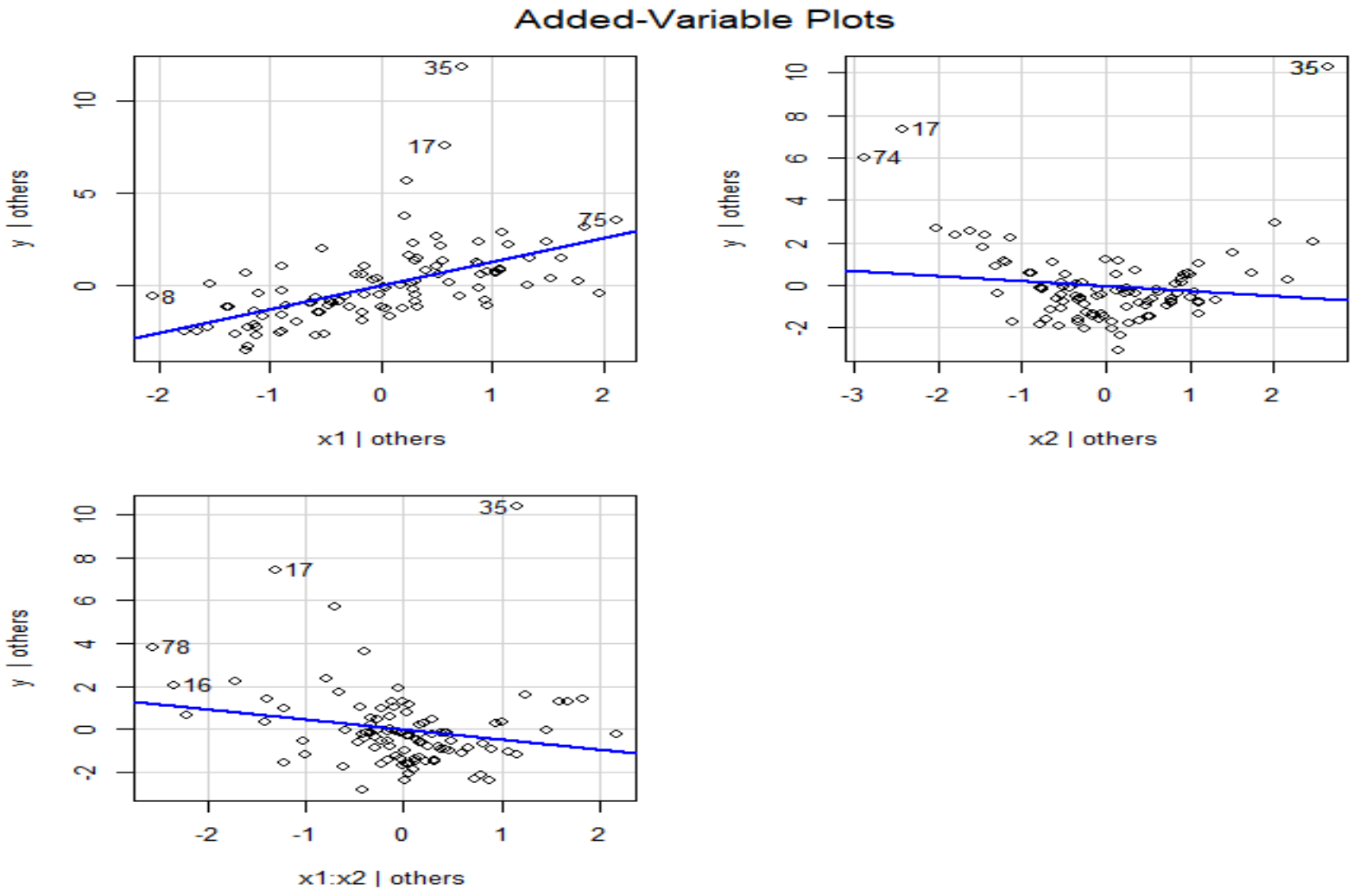
Casewise Diagnostic Plots

```
> par(mfrow=c(2,2))
> plot(lm.x1mx2)
```



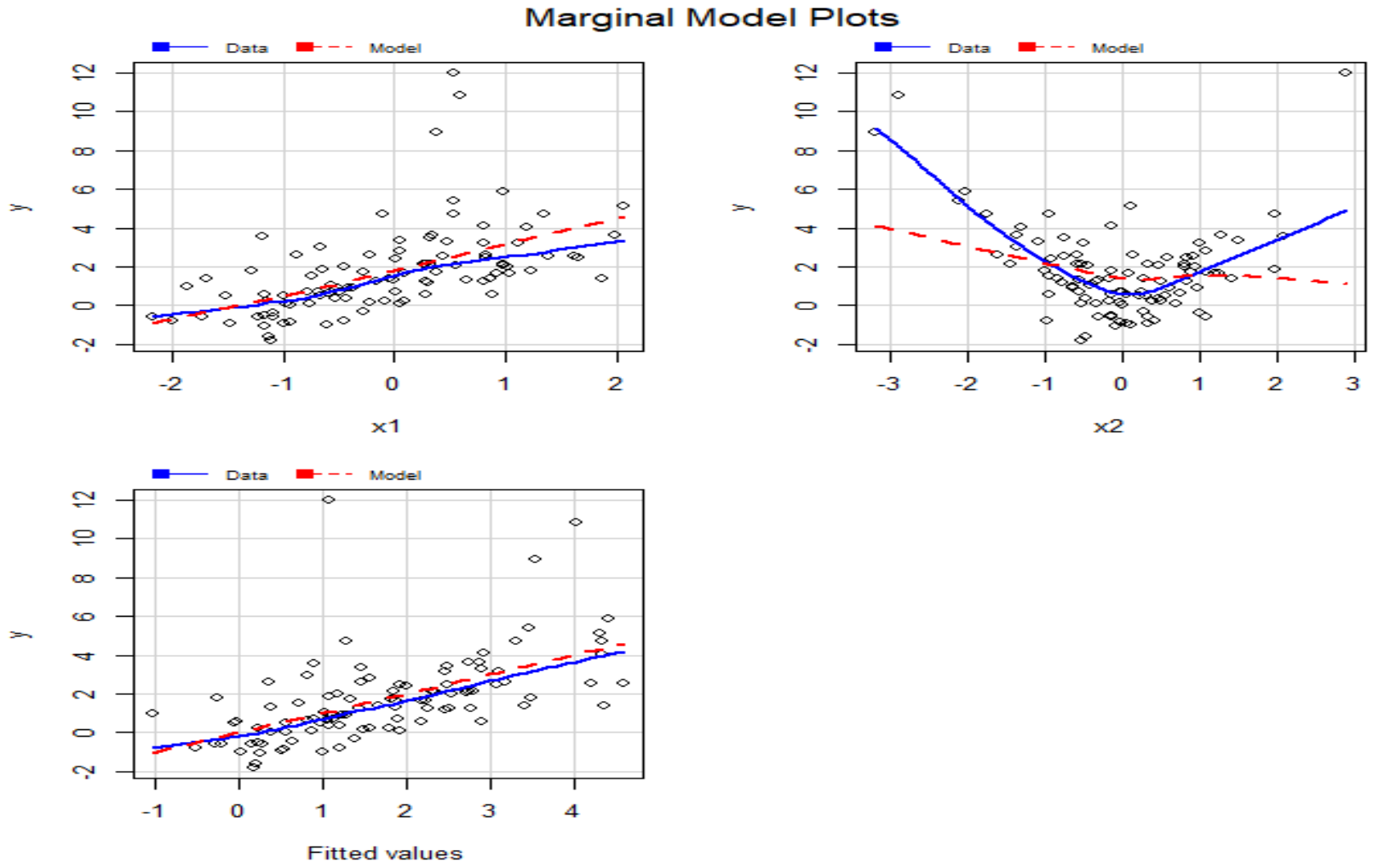
Added Variable Plots

> avPlots(lm.x1mx2)



Marginal Model Plots

> mmps(lm.x1mx2)

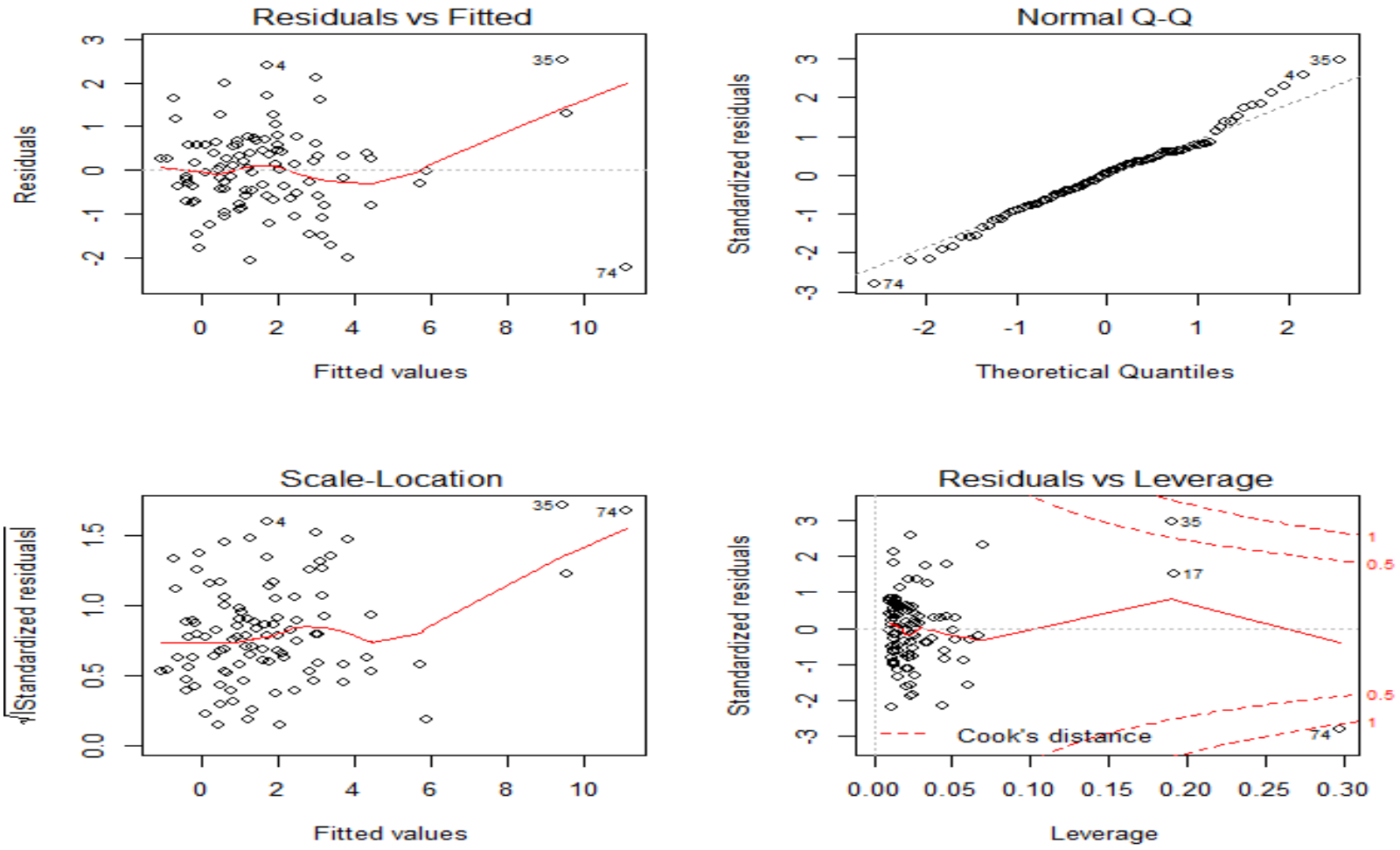


And now the correct model (with x2 squared term)..

```
> summary(lm.x1px2sq)
Call:
lm(formula = y ~ x1 + I(x2^2))

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.85241    0.11038   7.722 1.04e-11 ***
x1           1.04216    0.10171  10.247  < 2e-16 ***
I(x2^2)      0.96300    0.05522  17.438  < 2e-16 ***
---
Residual standard error: 0.9481 on 97 degrees of freedom
Multiple R-squared:  0.8269,    Adjusted R-squared:
0.8233
F-statistic: 231.6 on 2 and 97 DF,  p-value: < 2.2e-16
```

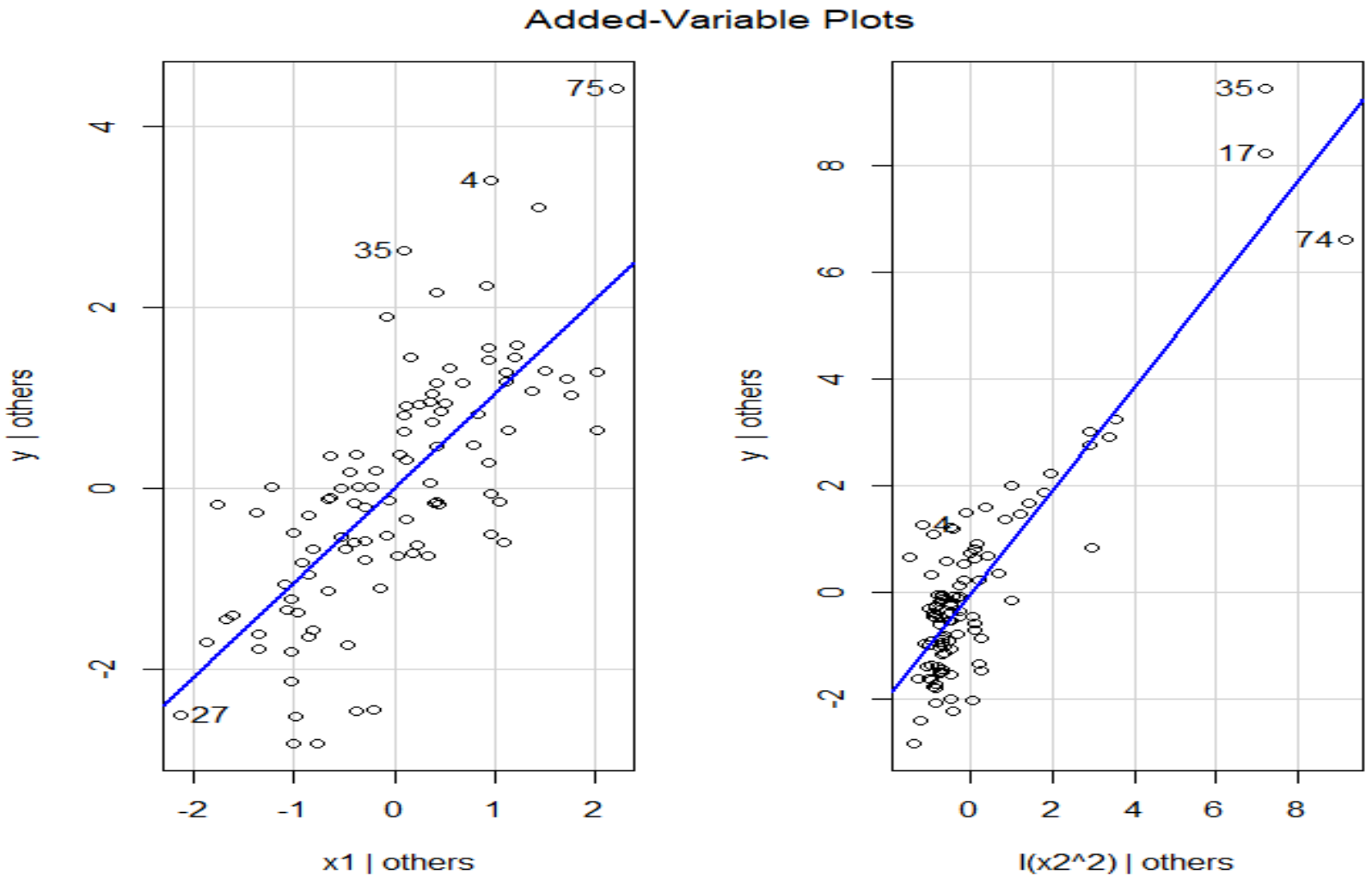
Casewise Diagnostic Plots

```
> par(mfrow=c(2,2))
> plot(lm.x1px2sq)
```



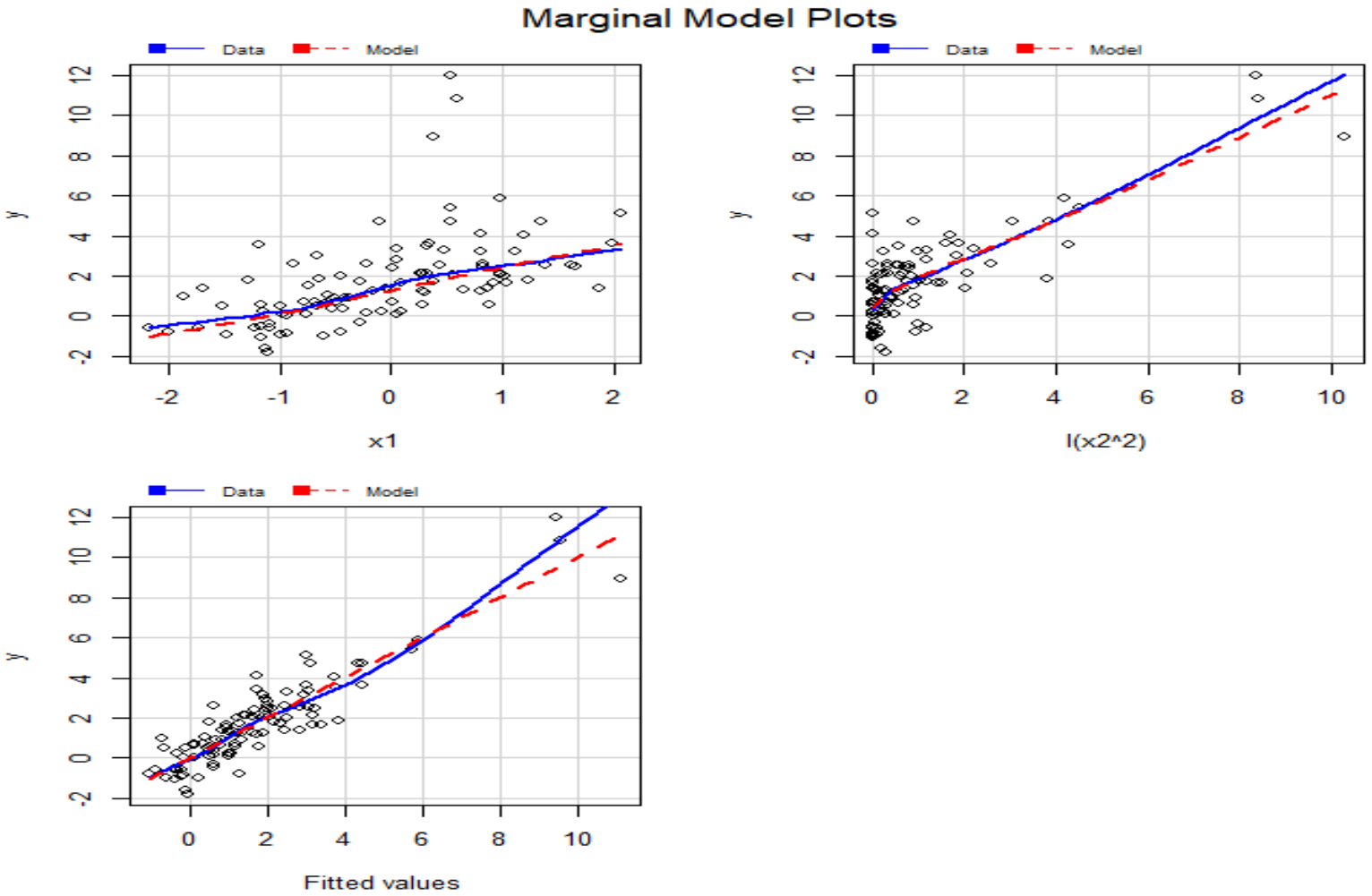
Added Variable Plots

> avPlots(lm.x1px2sq)



Marginal Model Plots

> mmps(lm.x1px2sq)



Moral of the Story

- Nonlinearity can show up in lots of ways, in lots of graphs
  - In casewise diagnostic plots
    - As nonlinearity
    - As nonconstant variance
    - As Non-normality (!!!)
  - In added-variable and marginal model plots
    - Nonlinearity shows up more clearly
    - Not always obvious what the right transformation would be.

Perspectives and Recommendations

- Substantive (investigator-driven) considerations *always come first*
- Power transforms of X *to reduce leverage* &
  Power transforms of Y *to improve distribution of* ϵ_i
  - By hand, or Box-Cox rounded to a simple power
- Inverse response plot for power transform of Y
  - Visually appealing, but Box-Cox probably better (directly addresses distribution of ϵ_i)
- Added Variable Plots and Marginal Model Plots can indicate nonlinearity, missing interactions, etc.
- There does not always exist a "perfect" transform!

Summary

- From last time:
  - Variance Stabilization for Y
  - Can residual plots distinguish $y^{(1)} = \beta_0 + \beta_1 x^2 + \varepsilon$, vs. $y^{(2)} = (\beta_0 + \beta_1 x + \varepsilon)^2$?
  - Inverse Response Plot for Y
- Added Variable Plots
- Marginal Model Plots
- Perspective and recommendations