Anscombe

Chris Parrish

December 10, 2016

Contents

Anscombe

references:
- Sheather, A Modern Approach to Regression with R, chapter 3, pp.45-61
- `lm.influence`

Load package.

```
library(ggplot2)
```

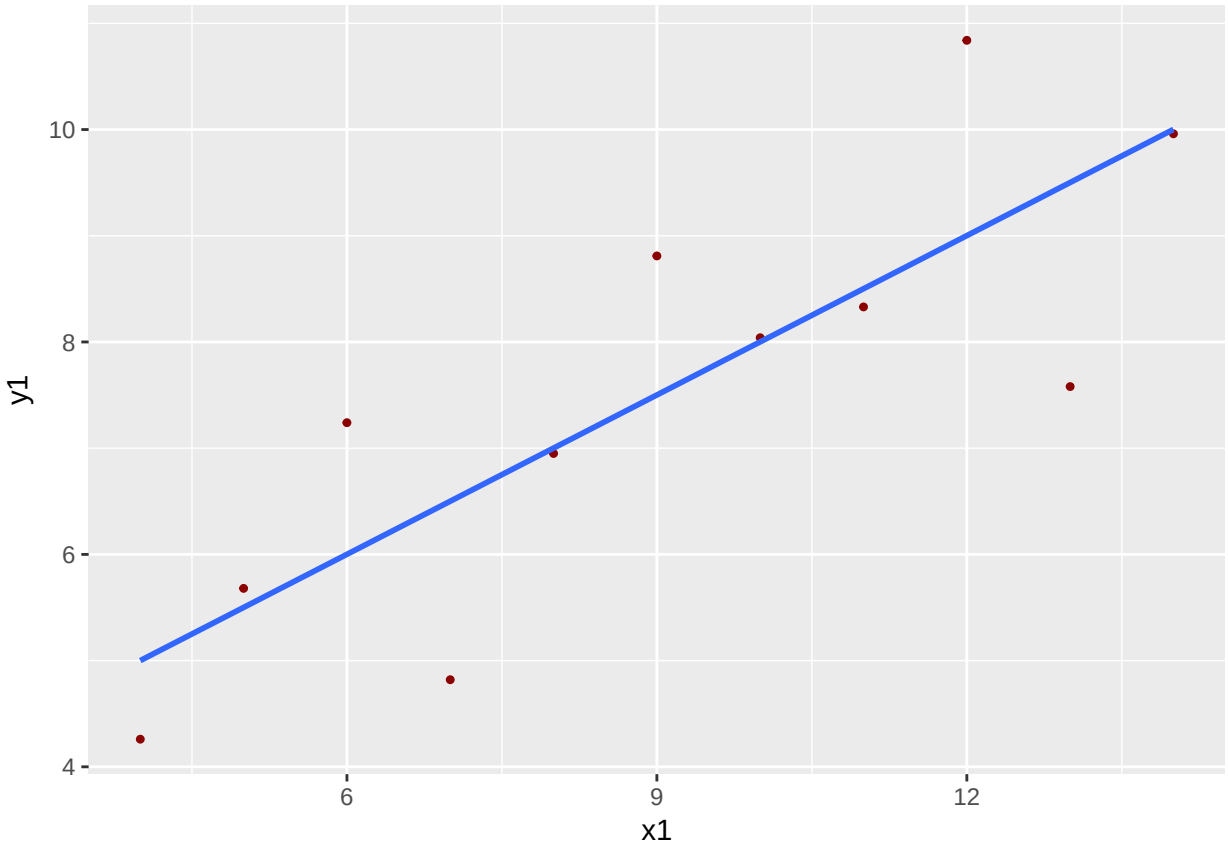
Import the data.

```
data <- read.table("anscombe.txt", header = TRUE)
head(data)
```

```
##   case x1 x2 x3 x4   y1   y2    y3   y4
## 1    1 10 10 10  8 8.04 9.14  7.46 6.58
## 2    2  8  8  8  8 6.95 8.14  6.77 5.76
## 3    3 13 13 13  8 7.58 8.74 12.74 7.71
## 4    4  9  9  9  8 8.81 8.77  7.11 8.84
## 5    5 11 11 11  8 8.33 9.26  7.81 8.47
## 6    6 14 14 14  8 9.96 8.10  8.84 7.04
```

Anscombe dataset 1

```
ggplot(data, aes(x1, y1)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm", se = FALSE)
```

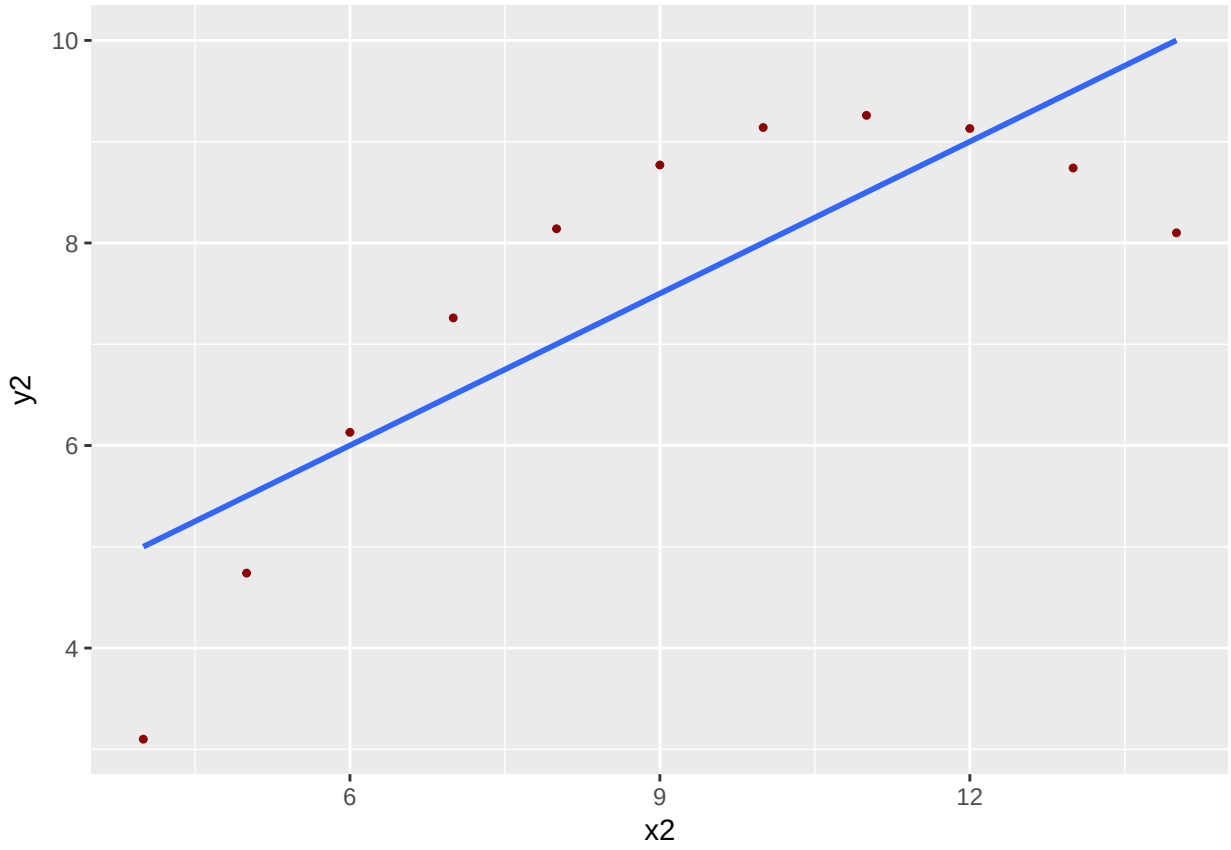



```
anscombe.lm1 <- lm(y1 ~ x1, data=data)
options(show.signif.stars = FALSE)
summary(anscombe.lm1)
```

```
##
## Call:
## lm(formula = y1 ~ x1, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.92127 -0.45577 -0.04136  0.70941  1.83882
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0001     1.1247   2.667  0.02573
## x1            0.5001     0.1179   4.241  0.00217
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6665, Adjusted R-squared:  0.6295
## F-statistic: 17.99 on 1 and 9 DF,  p-value: 0.00217
```

Anscombe dataset 2

```
ggplot(data, aes(x2, y2)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm", se = FALSE)
```

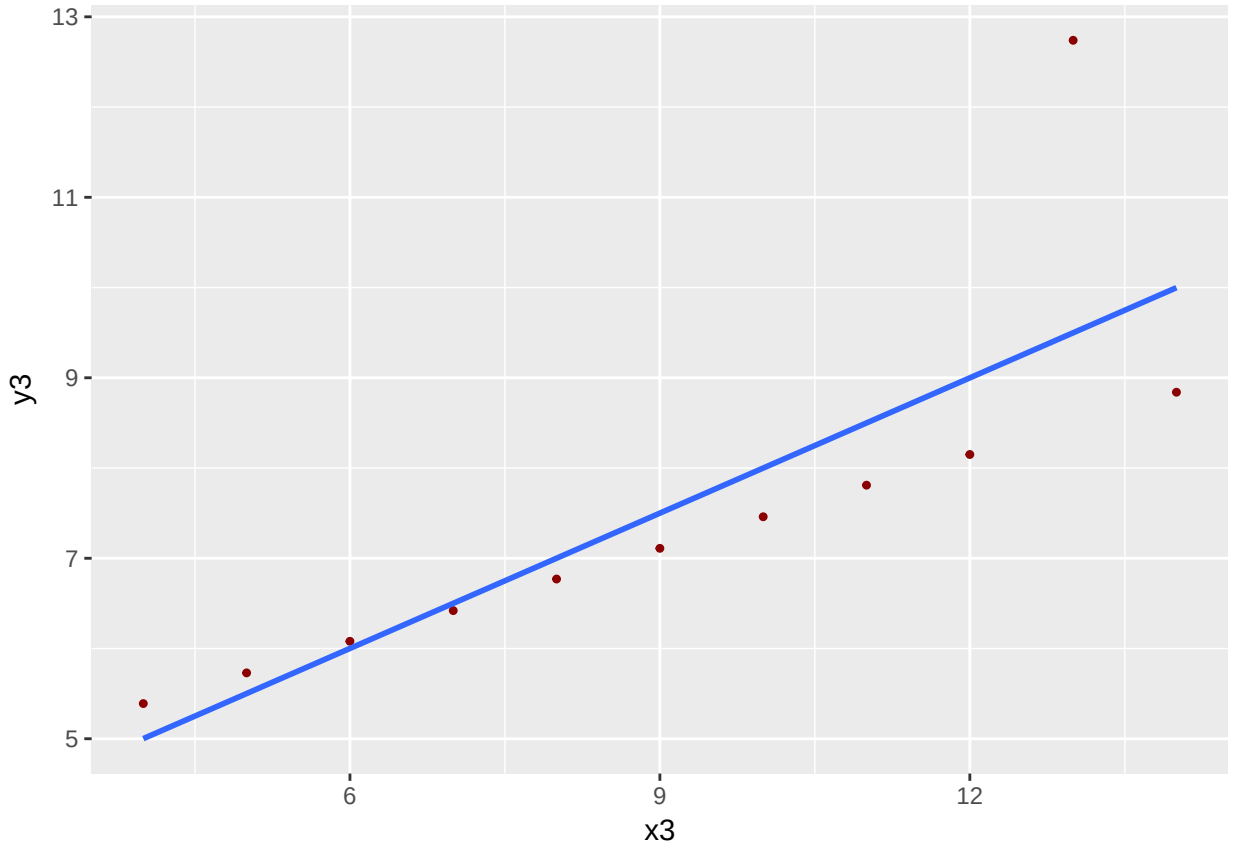



```
anscombe.lm2 <- lm(y2 ~ x2, data=data)
summary(anscombe.lm2)
```

```
##
## Call:
## lm(formula = y2 ~ x2, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9009 -0.7609  0.1291  0.9491  1.2691
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.001      1.125   2.667  0.02576
## x2             0.500      0.118   4.239  0.00218
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6662, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002179
```

Anscombe dataset 3

```
ggplot(data, aes(x3, y3)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm", se = FALSE)
```

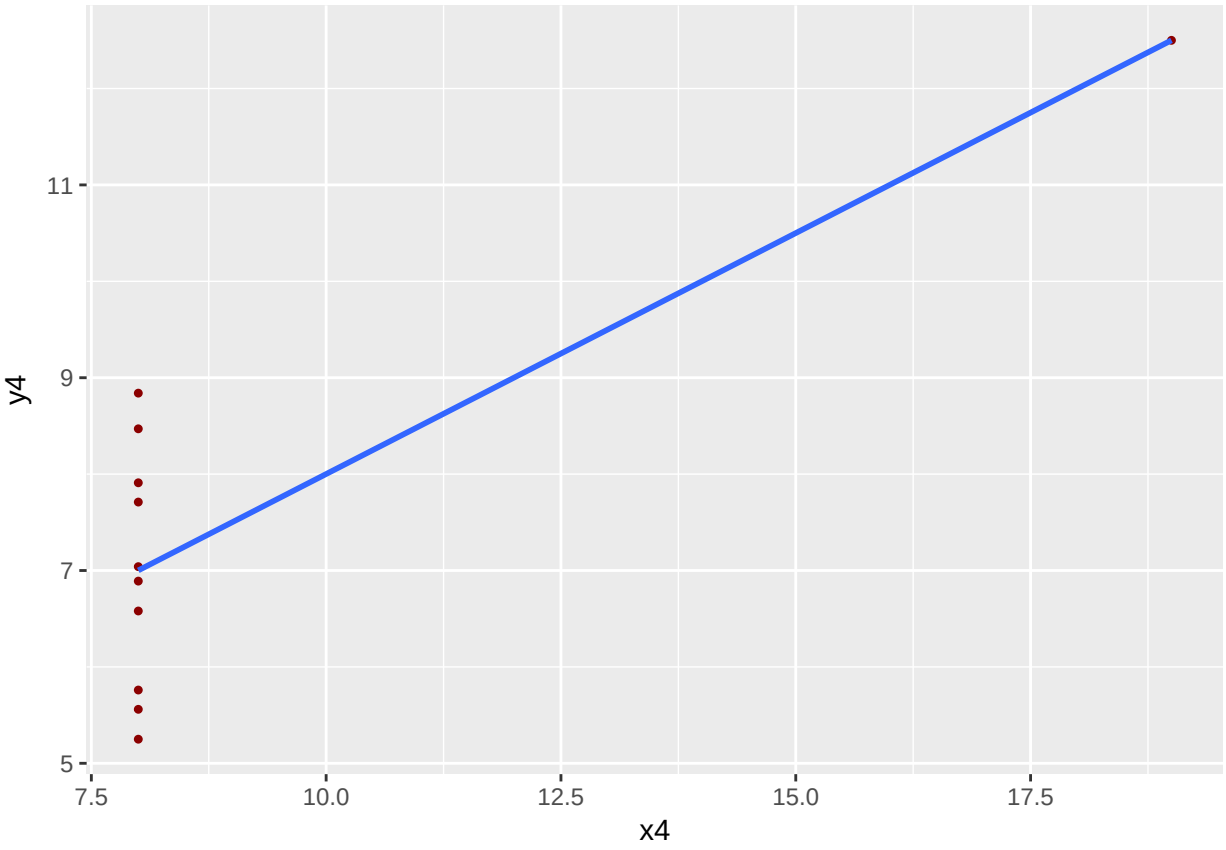



```
anscombe.lm3 <- lm(y3 ~ x3, data=data)
summary(anscombe.lm3)
```

```
##
## Call:
## lm(formula = y3 ~ x3, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1586 -0.6146 -0.2303  0.1540  3.2411
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0025     1.1245   2.670  0.02562
## x3            0.4997     0.1179   4.239  0.00218
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6663, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002176
```

Anscombe dataset 4

```
ggplot(data, aes(x4, y4)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm", se = FALSE)
```

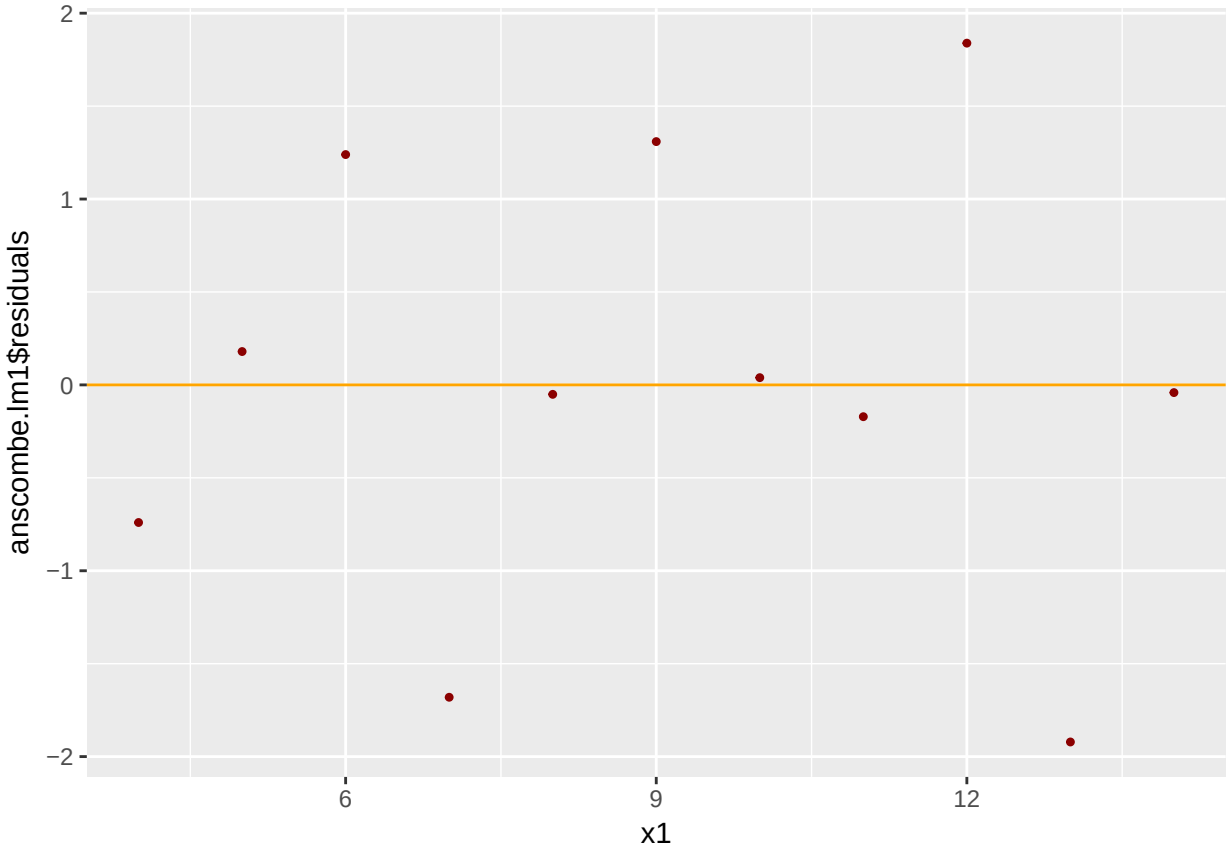



```
anscombe.lm4 <- lm(y4 ~ x4, data=data)
summary(anscombe.lm4)
```

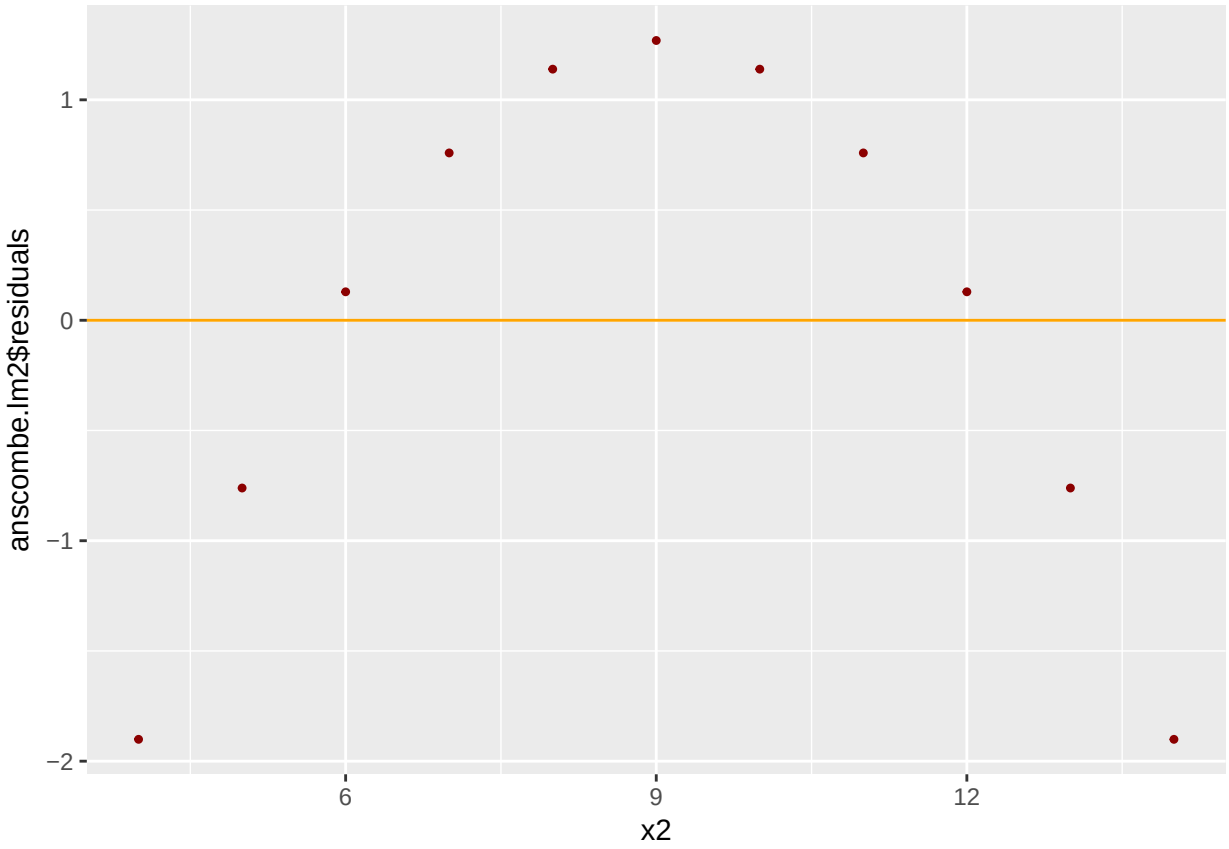
```
##
## Call:
## lm(formula = y4 ~ x4, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.751 -0.831  0.000  0.809  1.839
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0017     1.1239   2.671  0.02559
## x4            0.4999     0.1178   4.243  0.00216
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6667, Adjusted R-squared:  0.6297
## F-statistic:    18 on 1 and 9 DF,  p-value: 0.002165
```

residuals

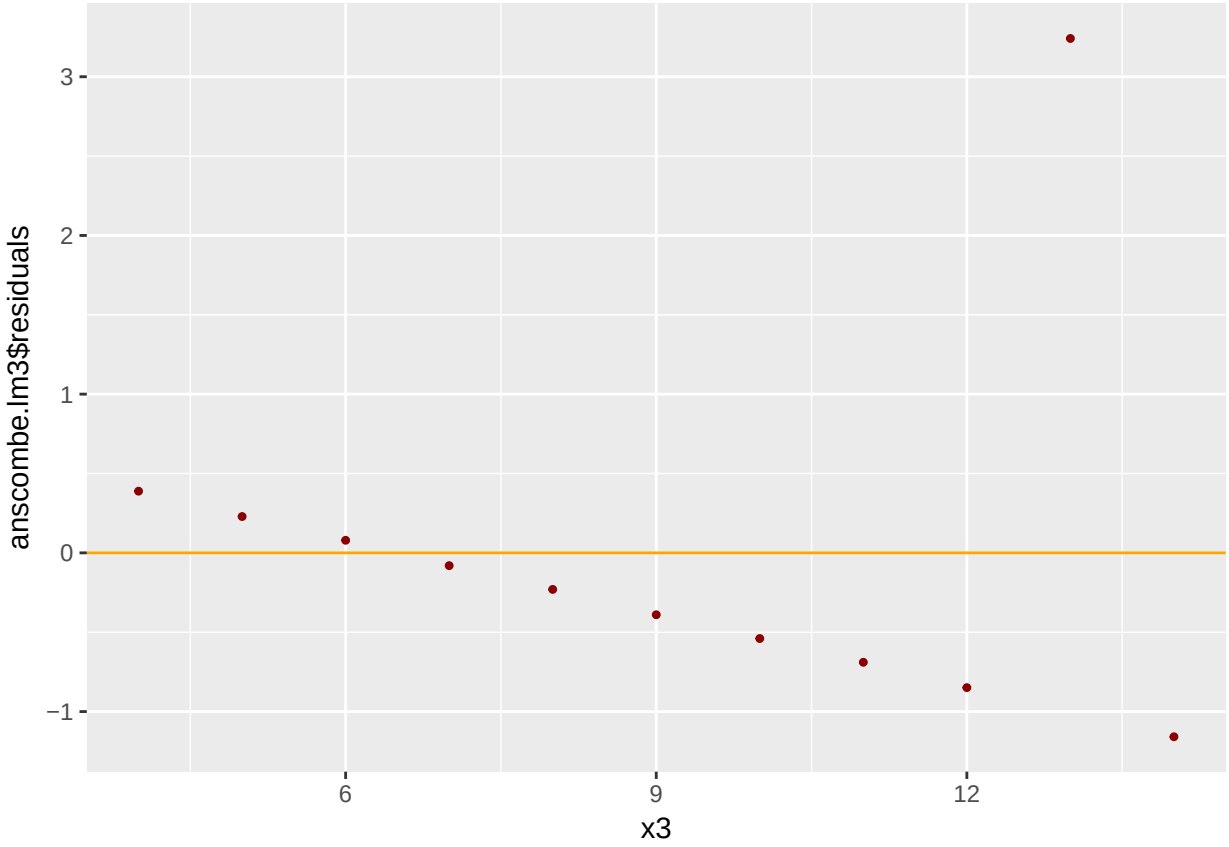
```
ggplot(data, aes(x1, anscombe.lm1$residuals)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange")
```



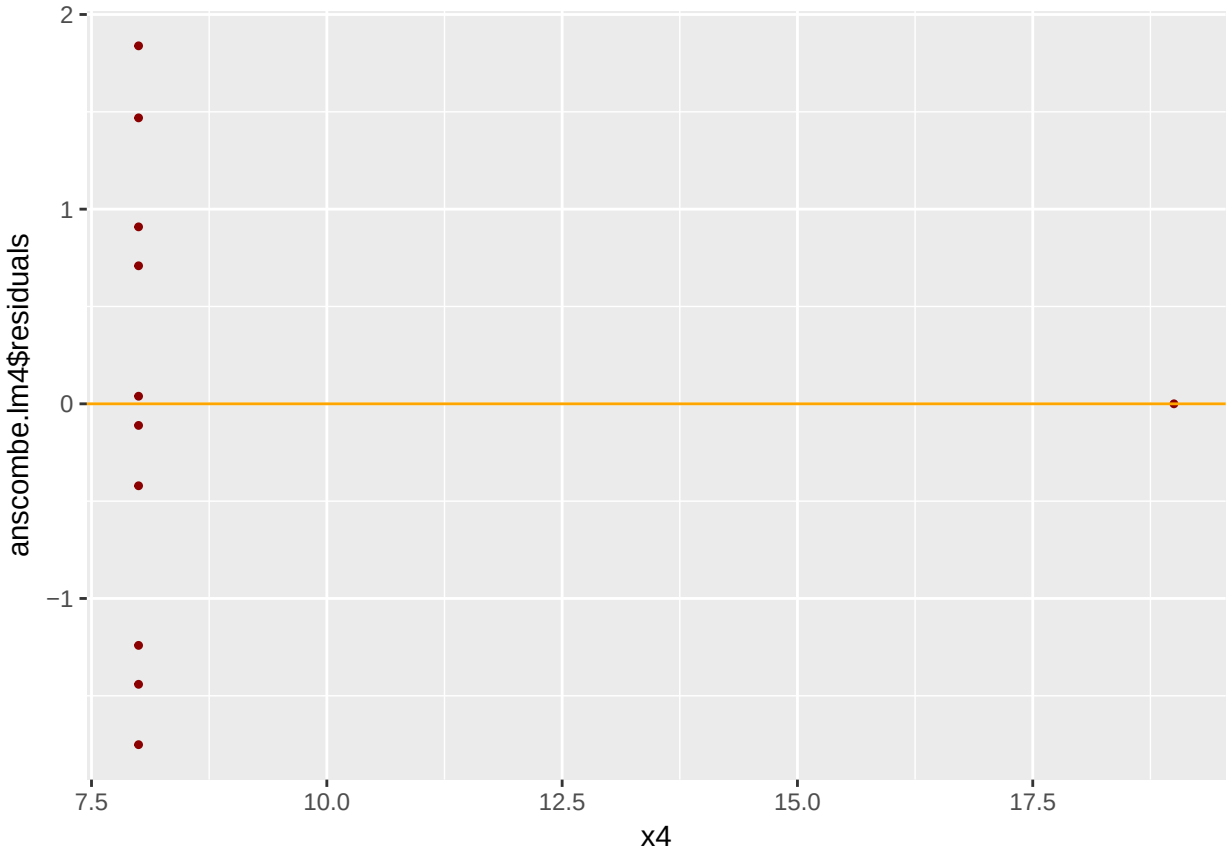
```
ggplot(data, aes(x2, anscombe.lm2$residuals)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange")
```



```
ggplot(data, aes(x3, anscombe.lm3$residuals)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange")
```



```
ggplot(data, aes(x4, anscombe.lm4$residuals)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange")
```

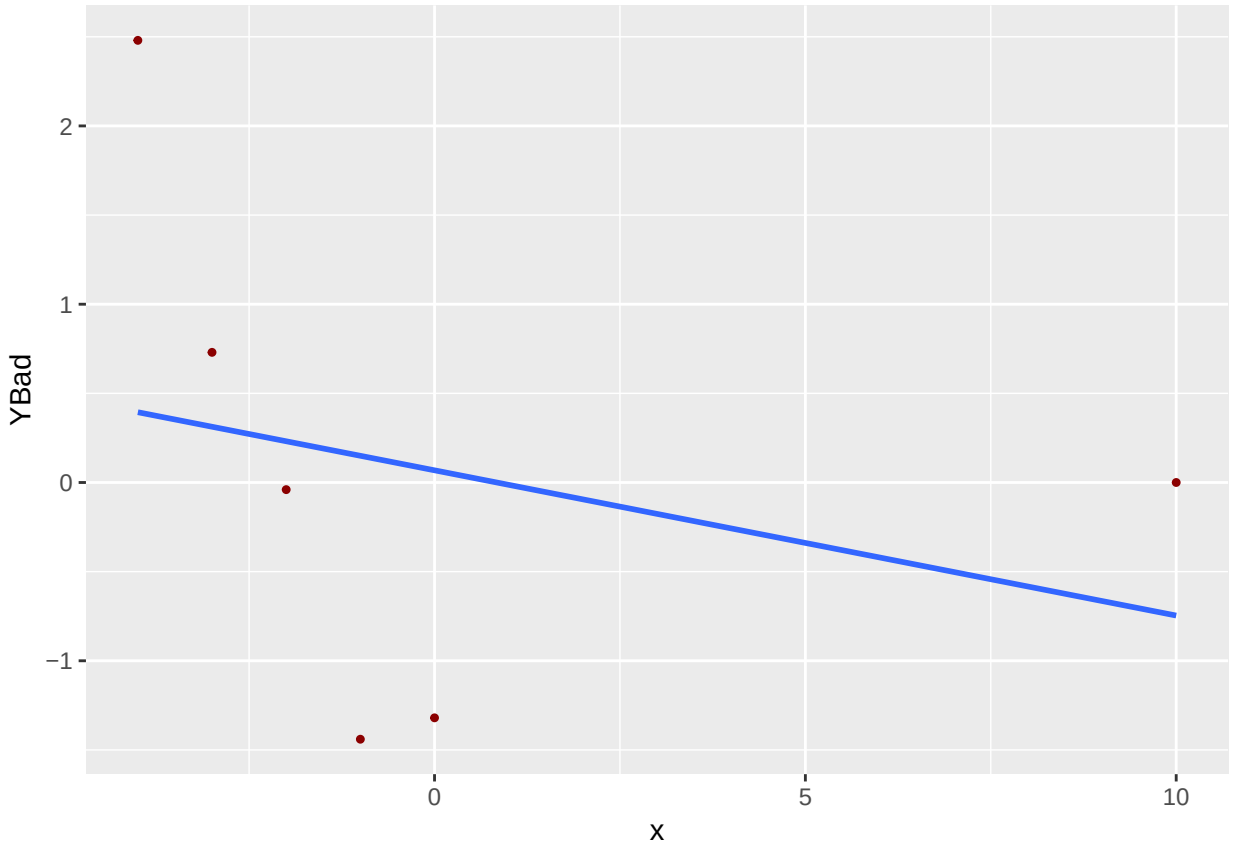


Huber dataset

```
data.Huber <- read.delim("huber.txt", header=TRUE)
data.Huber
```

```
##    x  YBad  YGood
## 1 -4  2.48   2.48
## 2 -3  0.73   0.73
## 3 -2 -0.04  -0.04
## 4 -1 -1.44  -1.44
## 5  0 -1.32  -1.32
## 6 10  0.00 -11.40
```

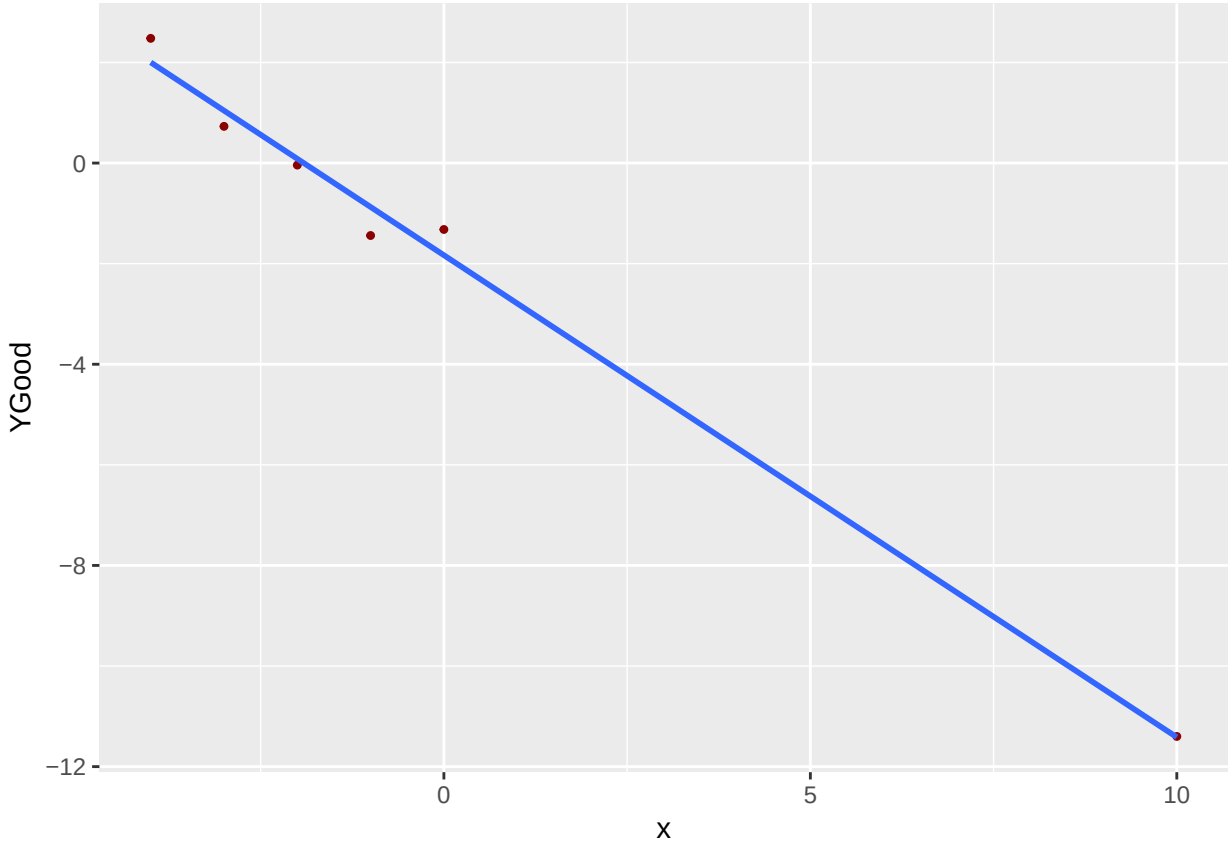
```
ggplot(data.Huber, aes(x, YBad)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm", se = FALSE)
```




```
huber.lm1 <- lm(YBad ~ x, data=data.Huber)
summary(huber.lm1)
```

```
##
## Call:
## lm(formula = YBad ~ x, data = data.Huber)
##
## Residuals:
##       1       2       3       4       5       6
##  2.0858  0.4173 -0.2713 -1.5898 -1.3883  0.7463
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.06833    0.63279   0.108    0.919
## x           -0.08146    0.13595  -0.599    0.581
##
## Residual standard error: 1.55 on 4 degrees of freedom
## Multiple R-squared:  0.08237,    Adjusted R-squared:  -0.147
## F-statistic: 0.3591 on 1 and 4 DF,  p-value: 0.5813
```

```
ggplot(data.Huber, aes(x, YGood)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm", se = FALSE)
```




```
huber.lm2 <- lm(YGood ~ x, data=data.Huber)
summary(huber.lm2)
```

```
##
## Call:
## lm(formula = YGood ~ x, data = data.Huber)
##
## Residuals:
##        1        2        3        4        5        6
##  0.47813 -0.31349 -0.12510 -0.56672  0.51167  0.01551
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.83167    0.19640  -9.326 0.000736
## x           -0.95838    0.04219 -22.714 2.23e-05
##
## Residual standard error: 0.4811 on 4 degrees of freedom
## Multiple R-squared:  0.9923, Adjusted R-squared:  0.9904
## F-statistic: 515.9 on 1 and 4 DF,  p-value: 2.225e-05
```

Influence.

```
tbl <- data.frame(lm.influence(huber.lm1)$hat,
                  lm.influence(huber.lm2)$hat)
```

```r
colnames(tbl) <- c("leverage.lm1", "leverage.lm2")
tbl
```

```
##   leverage.lm1 leverage.lm2
## 1    0.2897436    0.2897436
## 2    0.2358974    0.2358974
## 3    0.1974359    0.1974359
## 4    0.1743590    0.1743590
## 5    0.1666667    0.1666667
## 6    0.9358974    0.9358974
```

leverage point

A point x_i is a *leverage point* if

$$h_{i,i} > 2 * average.leverage = 2 * \frac{2}{n}$$

```r
average.leverage <- mean(tbl[ , 1])
average.leverage
```

```
## [1] 0.3333333
```

```r
h.66 <- tbl[6, 1]
h.66
```

```
## [1] 0.9358974
```

```r
leverage.point <- h.66 > 2 * average.leverage
leverage.point
```

```
## [1] TRUE
```

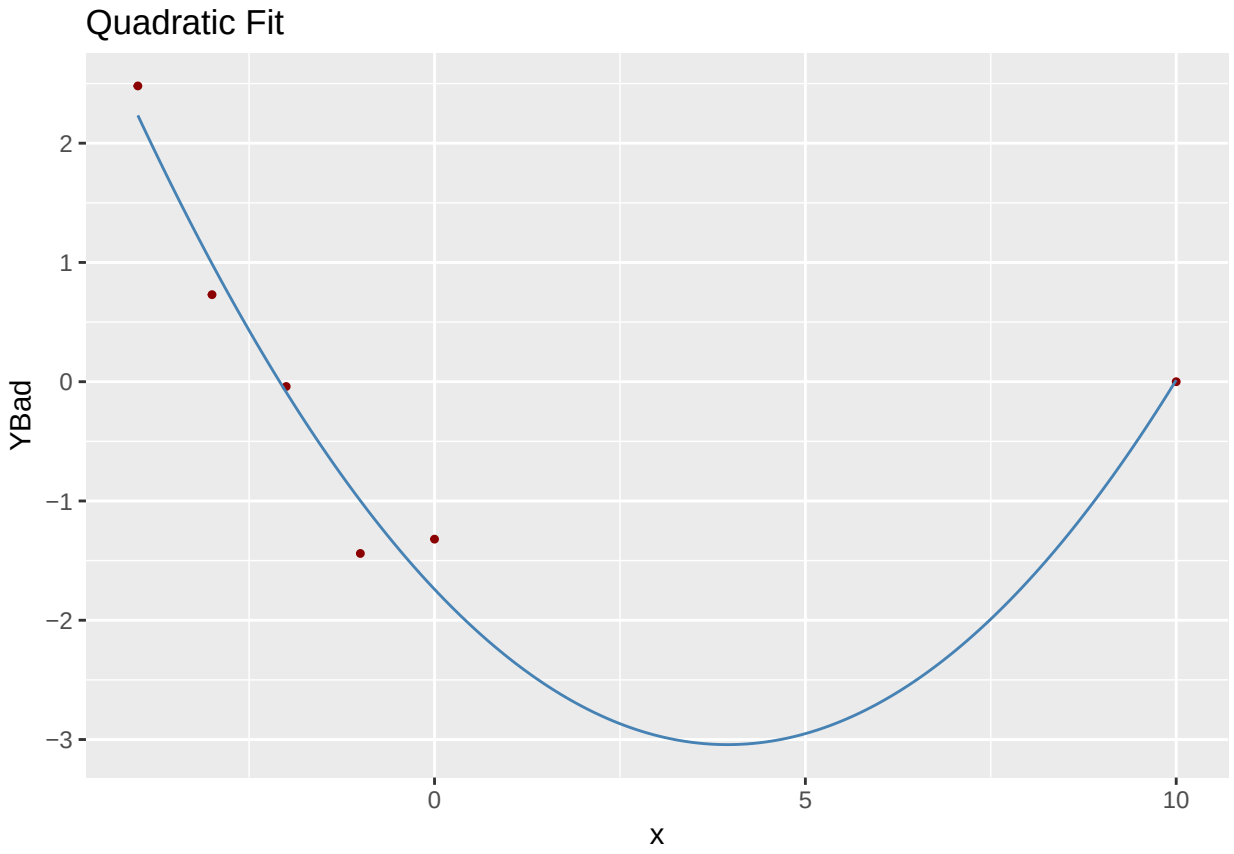
Quadratic model for YBad.

```r
huber.lm3 <- lm(YBad ~ x + I(x^2), data=data.Huber)
summary(huber.lm3)
```

```
##
## Call:
## lm(formula = YBad ~ x + I(x^2), data = data.Huber)
##
## Residuals:
##        1        2        3        4        5        6
##  0.24695 -0.25918  0.04771 -0.44237  0.42057 -0.01367
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.74057    0.29702  -5.860  0.00991
## x           -0.65945    0.08627  -7.644  0.00465
## I(x^2)       0.08349    0.01133   7.369  0.00517
##
## Residual standard error: 0.4096 on 3 degrees of freedom
## Multiple R-squared:  0.952,  Adjusted R-squared:  0.9199
## F-statistic: 29.72 on 2 and 3 DF,  p-value: 0.01053
```

quadratic fit for YBad

```
x = seq(-4, 10, 0.02)
new.data <- data.frame(x)
Y.hat <- predict(huber.lm3, new.data)
data.quadratic <- data.frame(x, Y.hat)
ggplot(data.Huber, aes(x, YBad)) +
  geom_point(shape = 20, color = "darkred") +
  geom_line(data = data.quadratic, aes(x, Y.hat), color = "steelblue") +
  ggtitle("Quadratic Fit")
```



leverage points and outliers

Regression lines can be influenced by points which are both leverage points and outliers.

The x coordinate of a point is used to determine if the point is a leverage point. A point x_i is a *leverage point* if

$$h_{i,i} > 2 * average.leverage = 2 * \frac{2}{n}$$

The y coordinate of a point, and in particular the residual $\hat{e} = y - \hat{y}$, determines if the point is an outlier in this context. A point is an *outlier* in the context of regression if the absolute value of its standardized residual is greater than 2 (or greater than 4 for large datasets).

$$|r_i| = \left| \frac{\hat{e}_i}{s\sqrt{1 - h_{i,i}}} \right| > 2$$

(Sheather, pp.56, 59).

points	leverage point	not leverage point
outlier	**BAD leverage point**	outlier
not outlier	leverage point	ordinary point