Residuals

Let's begin by considering some specific residuals from our data. Suppose we are interested in making a prediction about the student in row 12, a vegetarian whose favorite celebrity is Michelle Obama. Let's study that row of data:

```
require(mosaic)
# install.packages("googlesheets")
require(googlesheets)
url = gs_url("https://docs.google.com/spreadsheets/d/1tHg_Oo88GIS8E00-lf286-4AzR-wht7ZddUMFM1s5s0/")
ds = gs_read_csv(url)

ds %>% slice(12)
```

This student has 623 facebook friends and has had an account for 7 years. We could also look at row 10,

```
ds %>% slice(10)
```

This student has 320 friends and an account for 5 years. But, what are the predicted values for those students?

```
mod = lm(numFBFriends ~ accountLength, data=ds)
summary(mod)
xyplot(numFBFriends ~ accountLength, data=ds, type=c("p", "r"))
new_ds = data.frame(accountLength = c(7, 5))
predict(mod, newdata = new_ds)
```

So, it predicts that the first student would have 1006 friends, and the second student would have 837 friends. What are their residuals?

```
623 - 584
320 - 415
```

One is positive, one is negative. We can also look at the residuals for the entire class, or compute the Sum of Squared Residuals (SSR).

```
residuals(mod)
sum(residuals(mod)^2)
```

Finally, we could take the mean of the residuals

```
mean(residuals(mod))
```

(the sum is essentially 0)

Assessing conditions for regression

Let's try to assess the conditions for regression on this example. The easiest way to do this is to run the `plot()` command on your model object,

```
plot(mod, which=c(1,2))
```

We'll focus on the first two plots for now (if you don't specify `which`, you will get four plots). The first is the residual versus fitted values plot, which helps us assess linearity and equality of variance, and the second is a QQ plot (quantile-quantile plot) to help us assess normality. Another way to assess normality of the residuals is to make a histogram.

```
histogram(~residuals, data=mod, fit="normal")
```

Pathological examples

Because most real models are hard to assess, we want to examine some pathological examples of the LINE conditions for regression being violated.

For all these examples, we will generate simulated data to show the particular conditions being violated.

You'll notice that even as we're trying to simulate data to violate one condition, the others start to look bad, too. This is very common in the real world, as well!

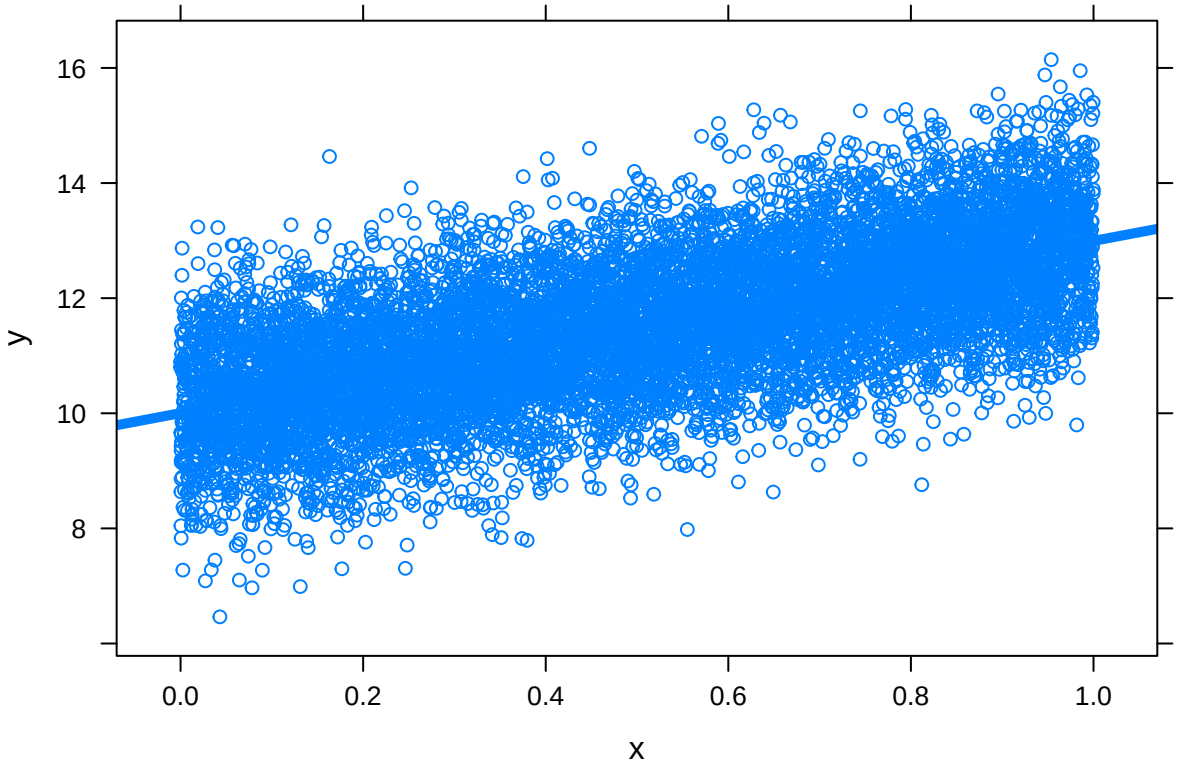
Good Model

First, a positive example! Let's make some good data.

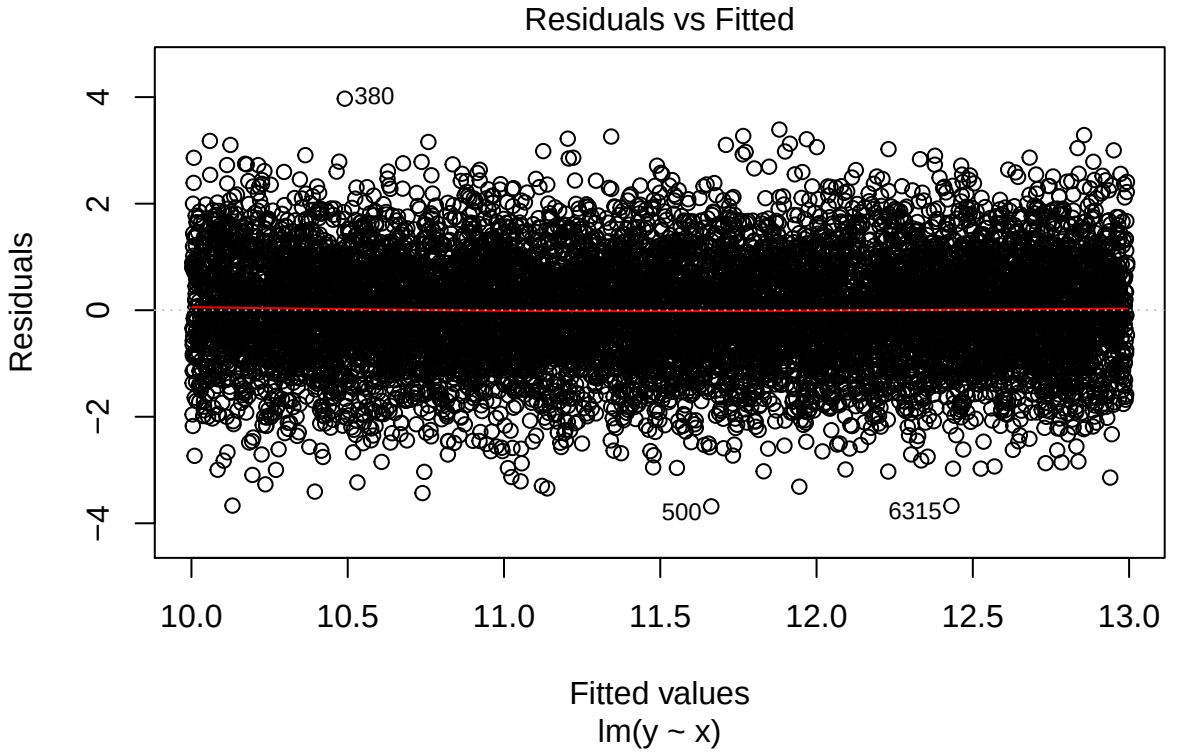
```
require(mosaic)
n = 10000
beta0 = 10
beta1 = 3
x = runif(n)
e = rnorm(n)
ds = data.frame(y = beta0 + beta1 * x + e)
```

Now, we can look at the relationship and check the conditions.

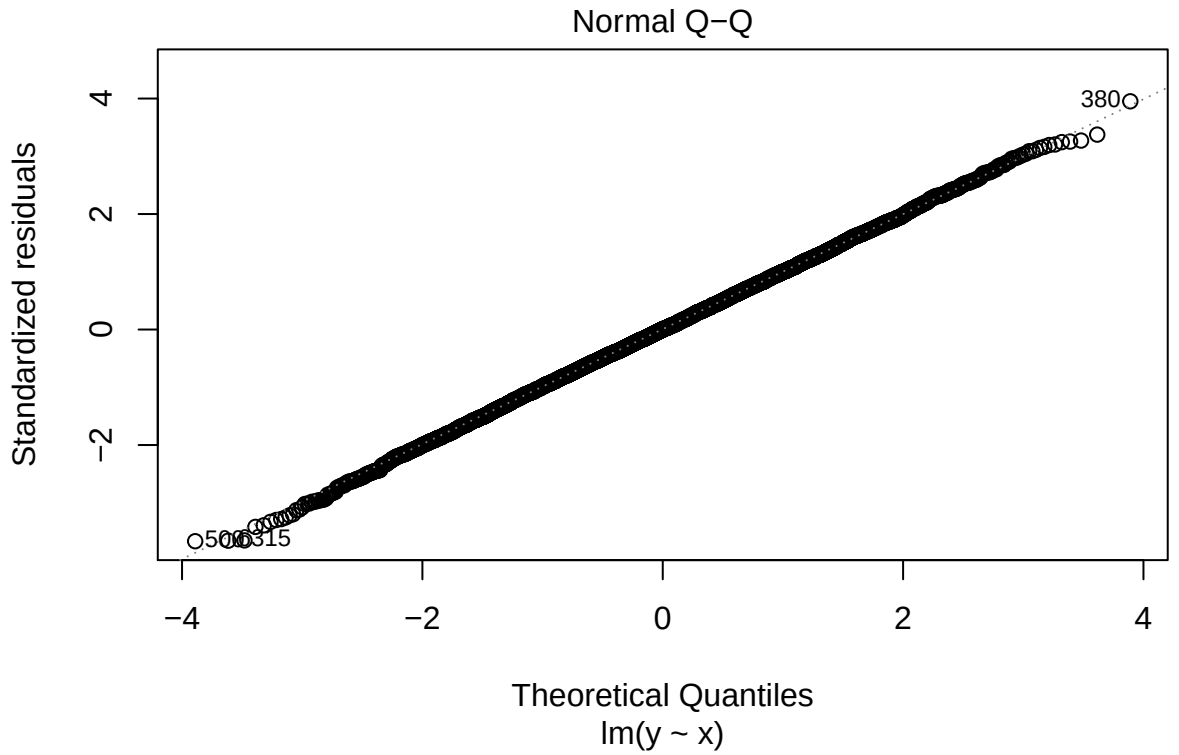
```
xyplot(y ~ x, data=ds, type=c("p", "r"), lwd=5)
```



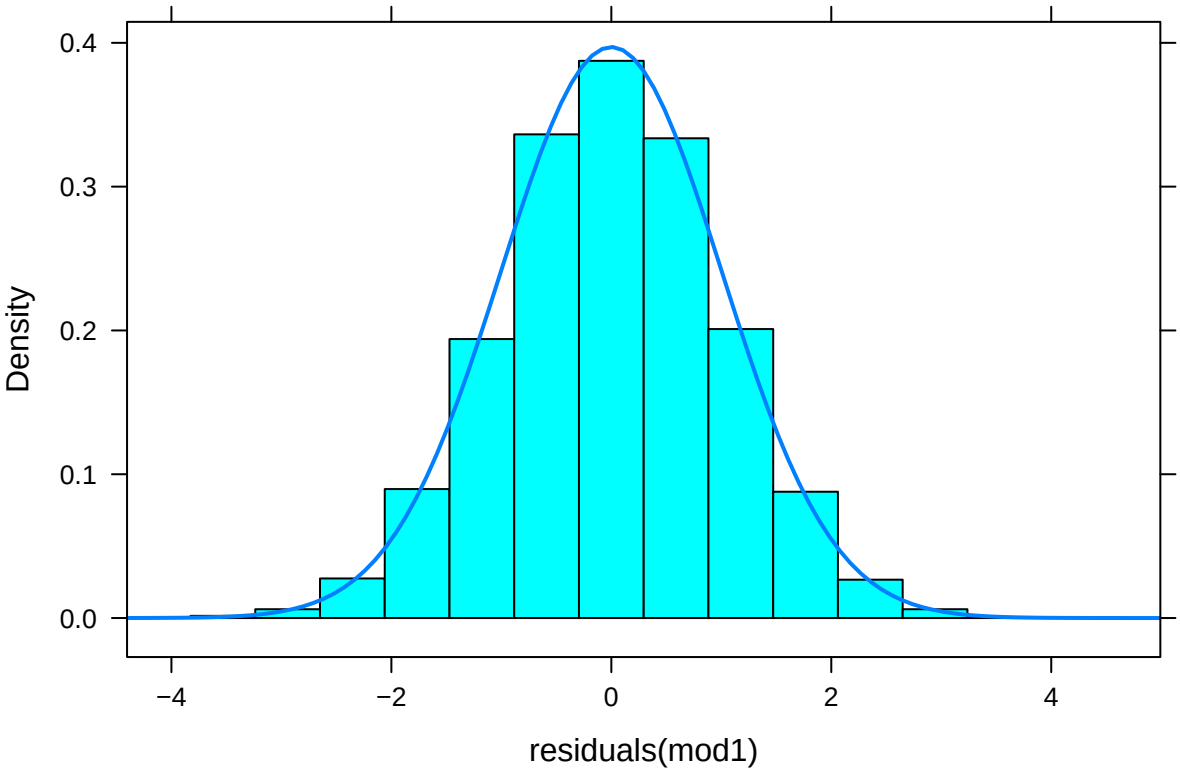

```
mod1 = lm(y ~ x, data=ds)
plot(mod1, which=1)
```



```r
plot(mod1, which=2)
```



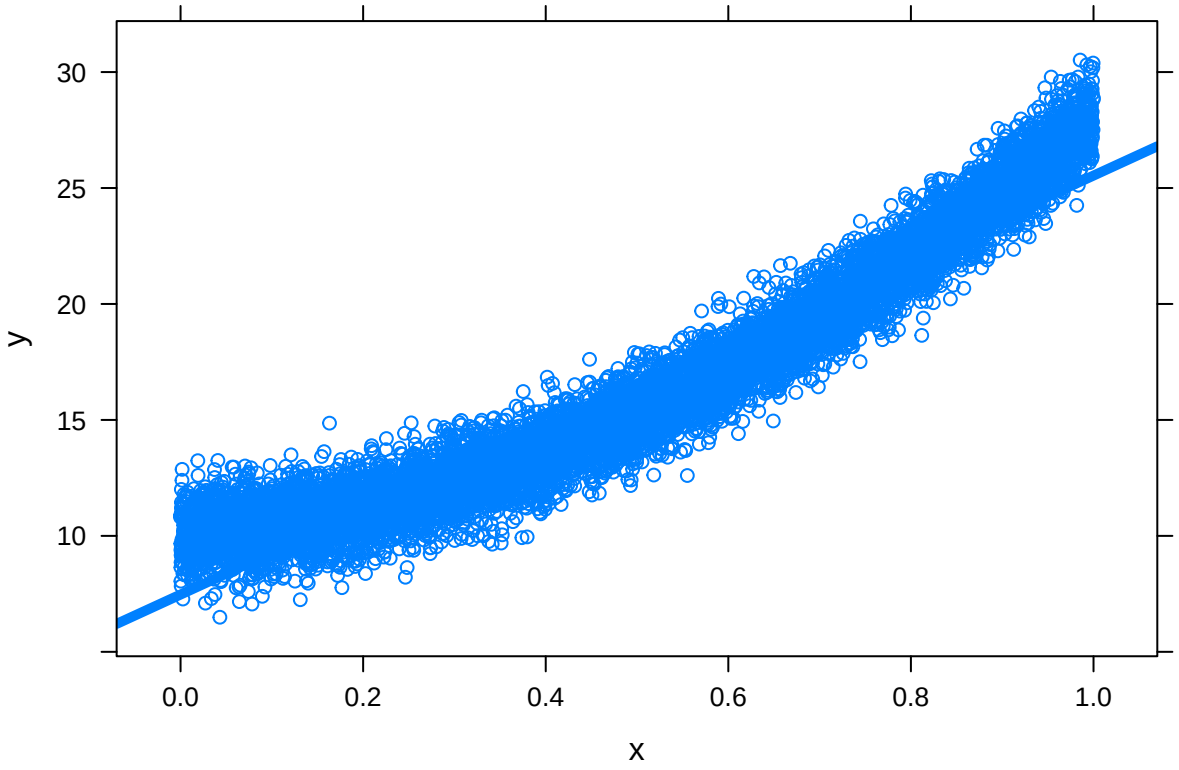

```r
histogram(~residuals(mod1), fit="normal")
```



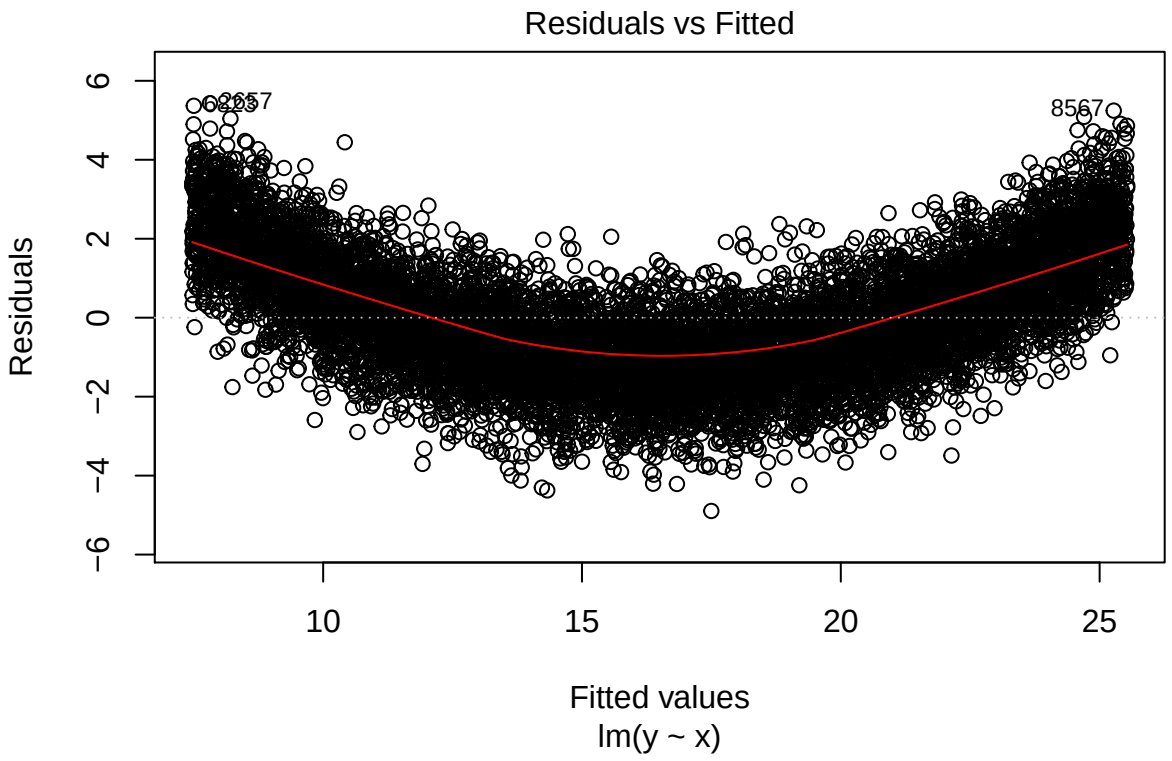
Linearity

```
ds = data.frame(y = beta0 + beta1 * x + 15*x^2 + e)
```

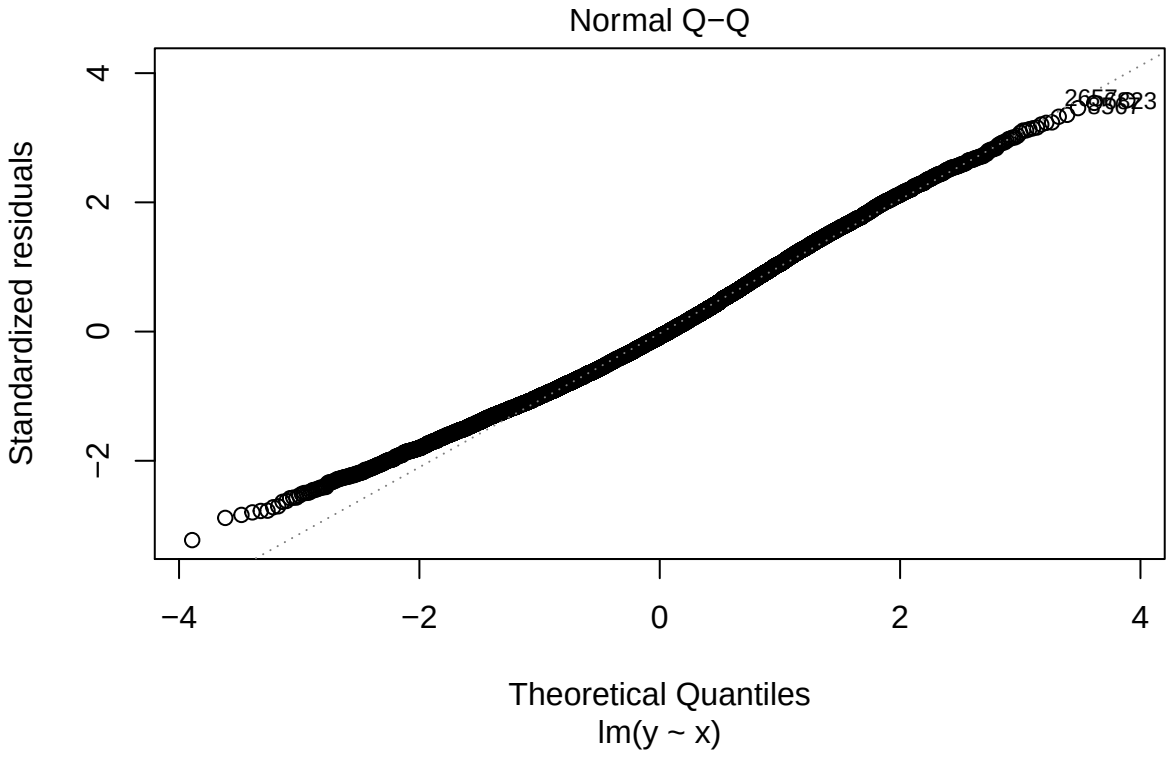
```
xyplot(y ~ x, data=ds, type=c("p", "r"), lwd=5)
```



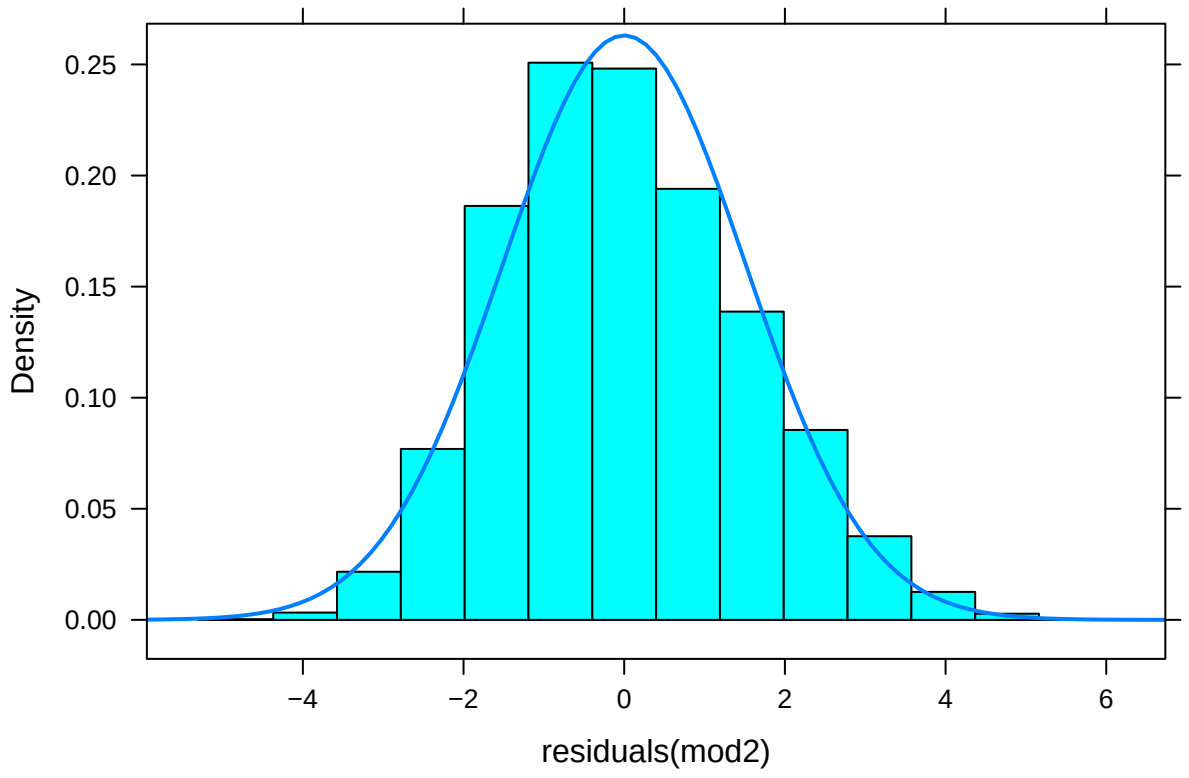

```
mod2 = lm(y ~ x, data=ds)
plot(mod2, which=1)
```




```
plot(mod2, which=2)
```



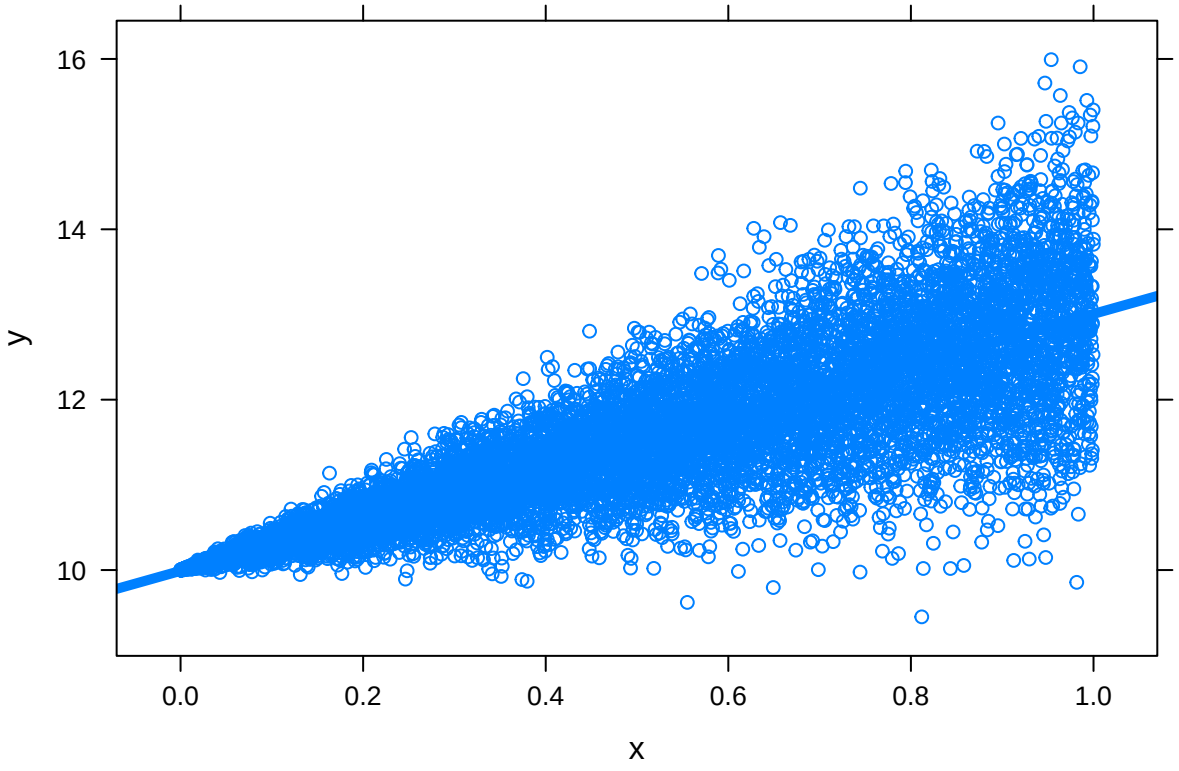

```
histogram(~residuals(mod2), fit="normal")
```



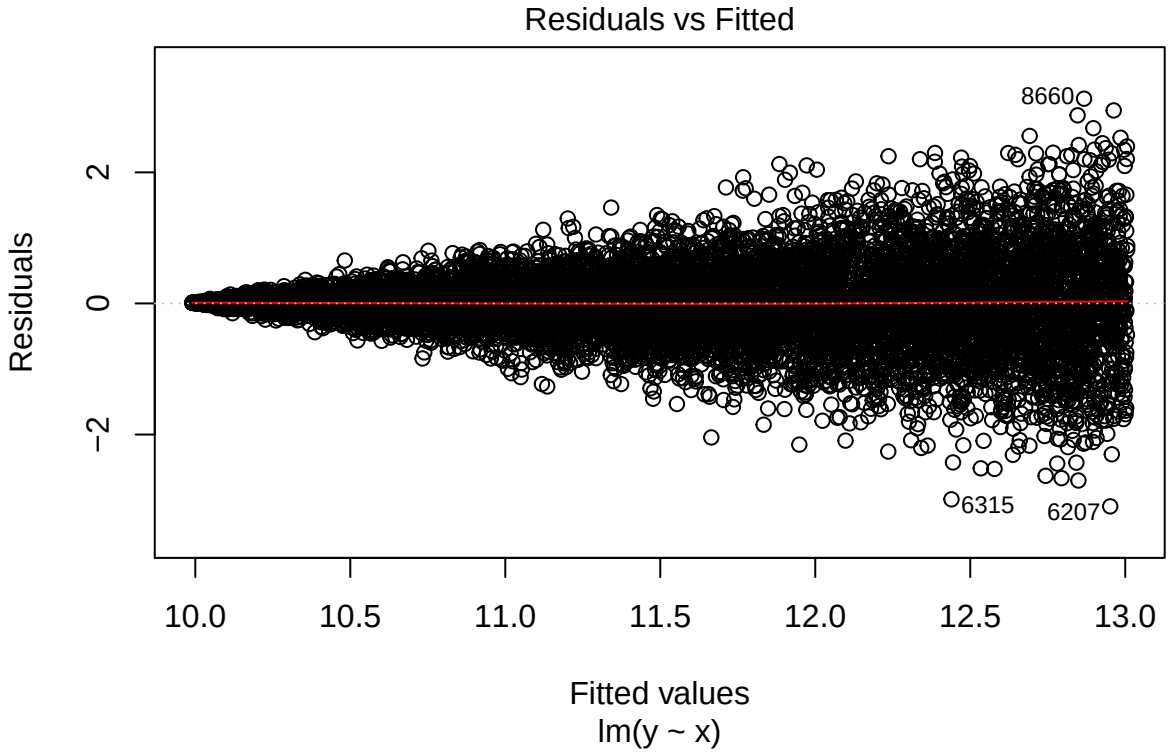
Constant Variance

```
ds = data.frame(y = beta0 + beta1 * x + e*x)
```

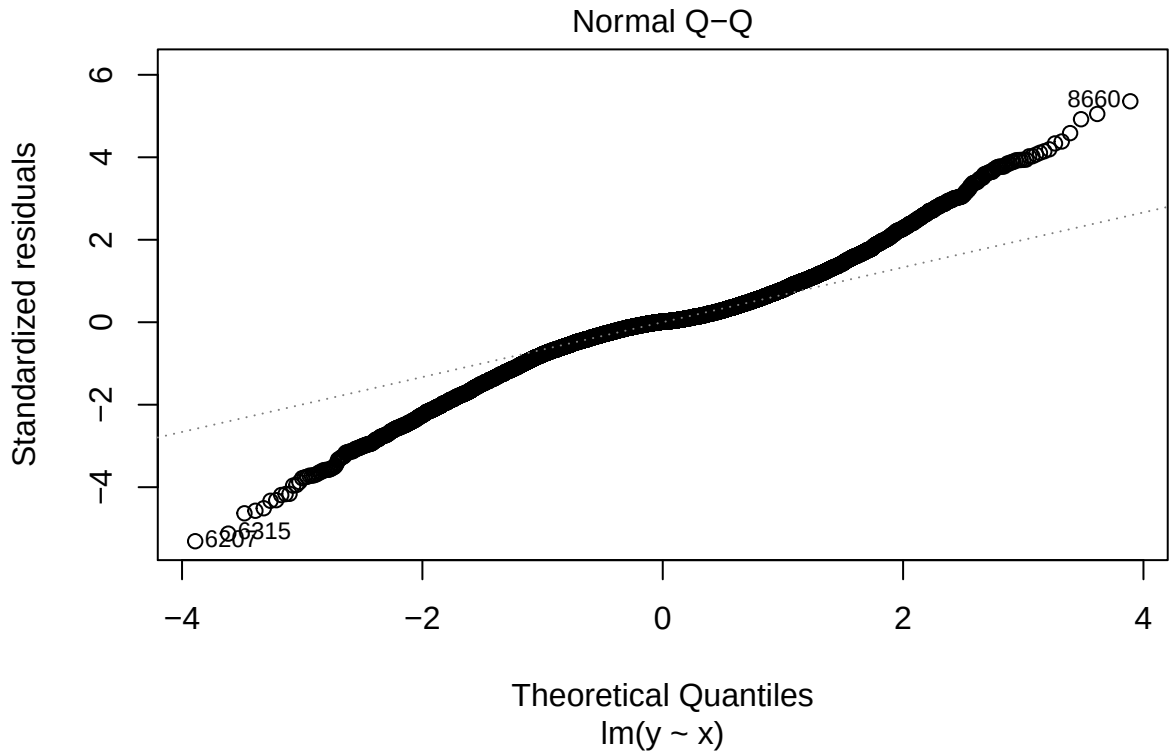
```
xyplot(y ~ x, data=ds, type=c("p", "r"), lwd=5)
```



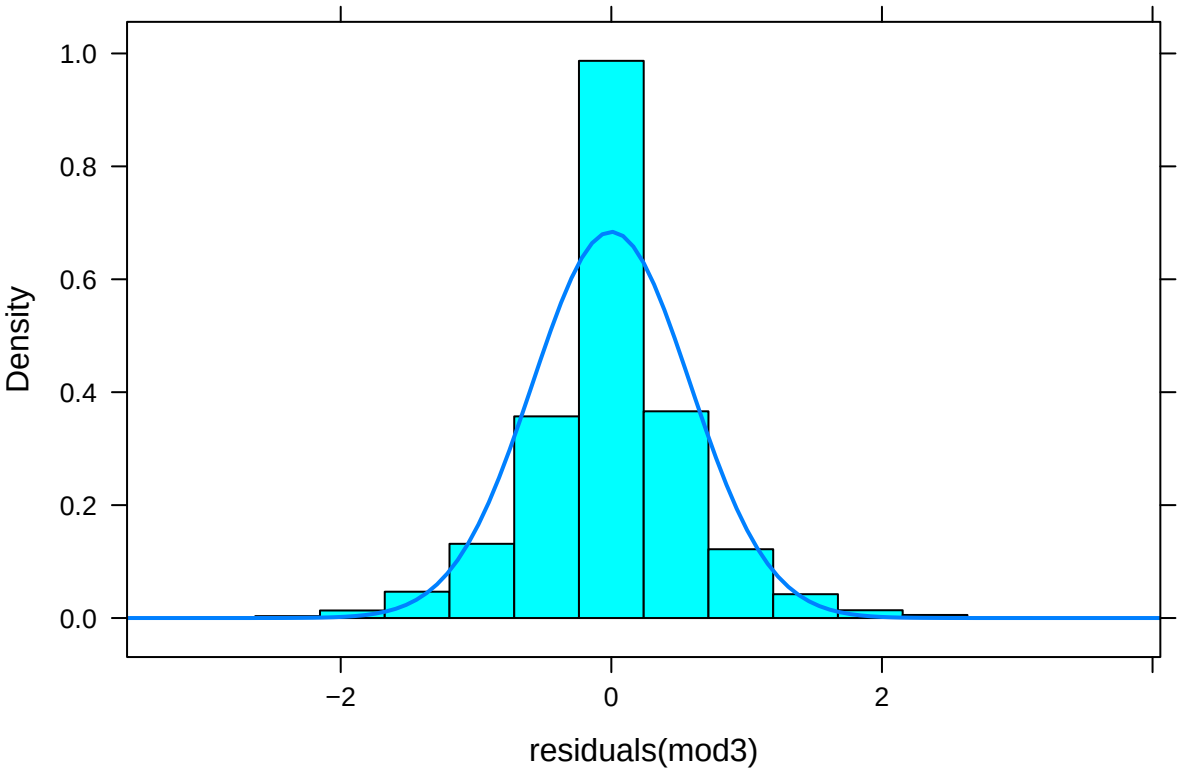

```
mod3 = lm(y ~ x, data=ds)
plot(mod3, which=1)
```



```
plot(mod3, which=2)
```



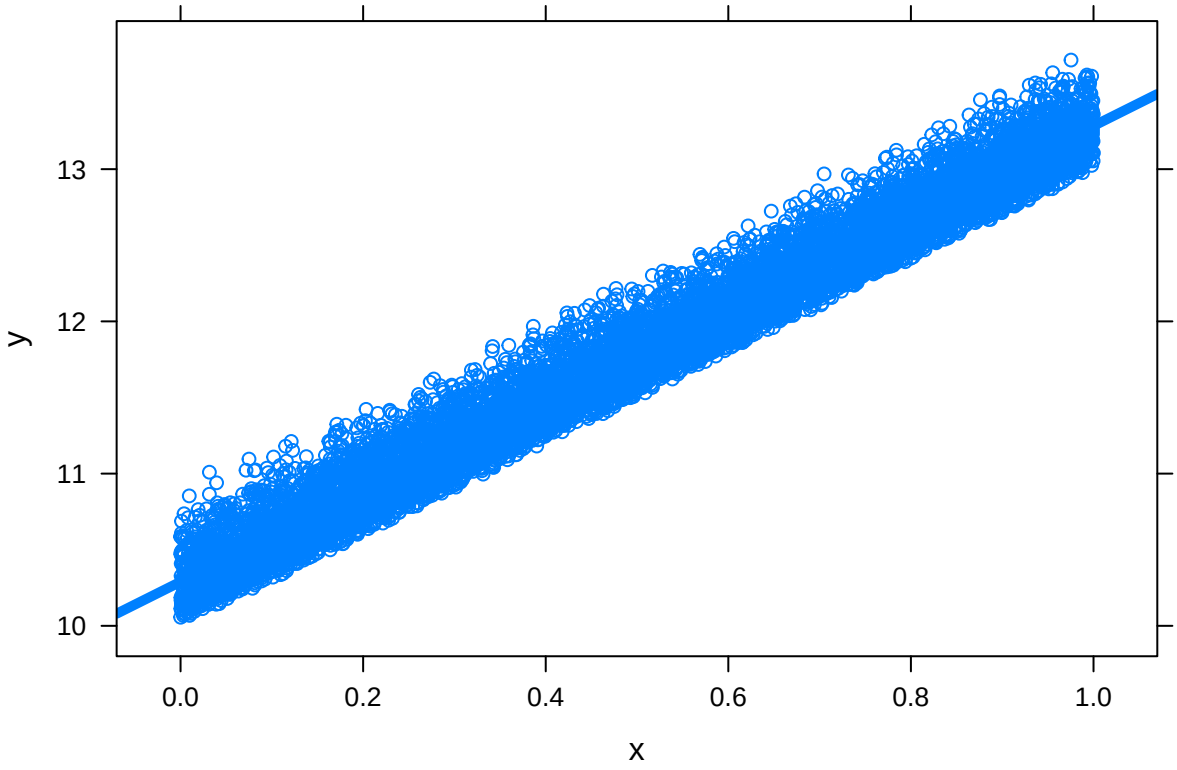

```
histogram(~residuals(mod3), fit="normal")
```



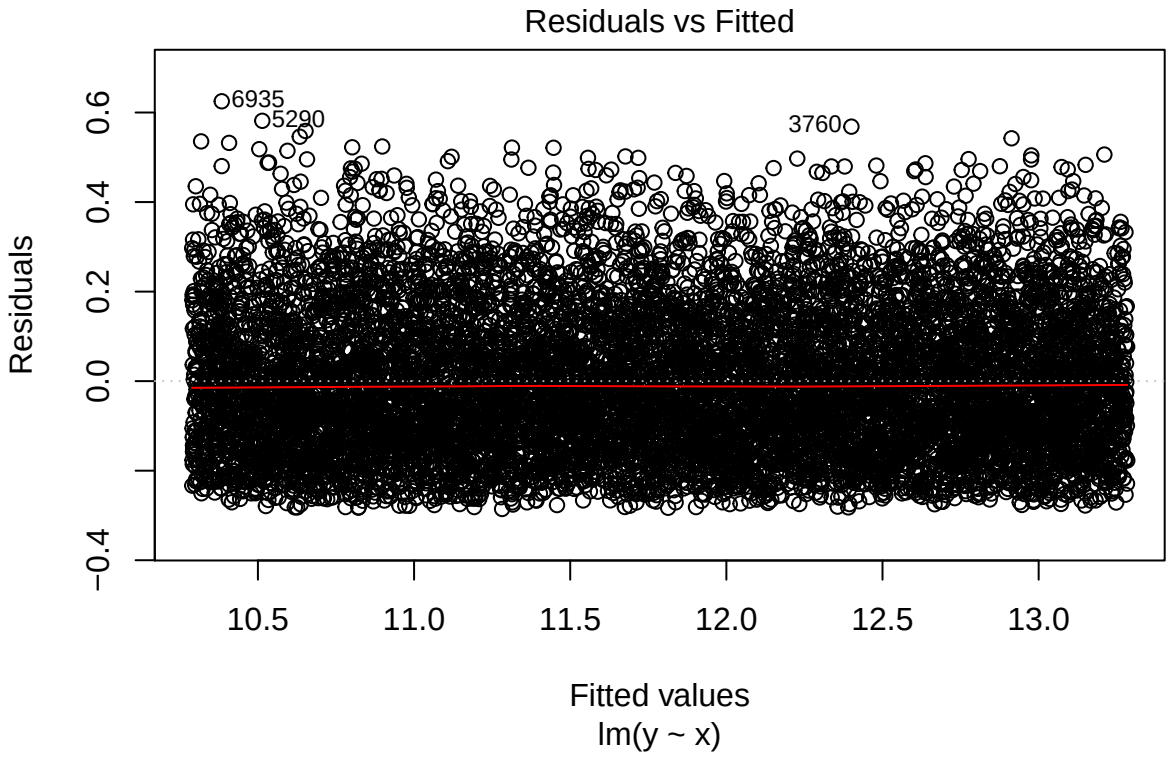
Normality

```
ds = data.frame(y = beta0 + beta1 * x + rbeta(n, shape1 = 2, shape2 = 5))
```

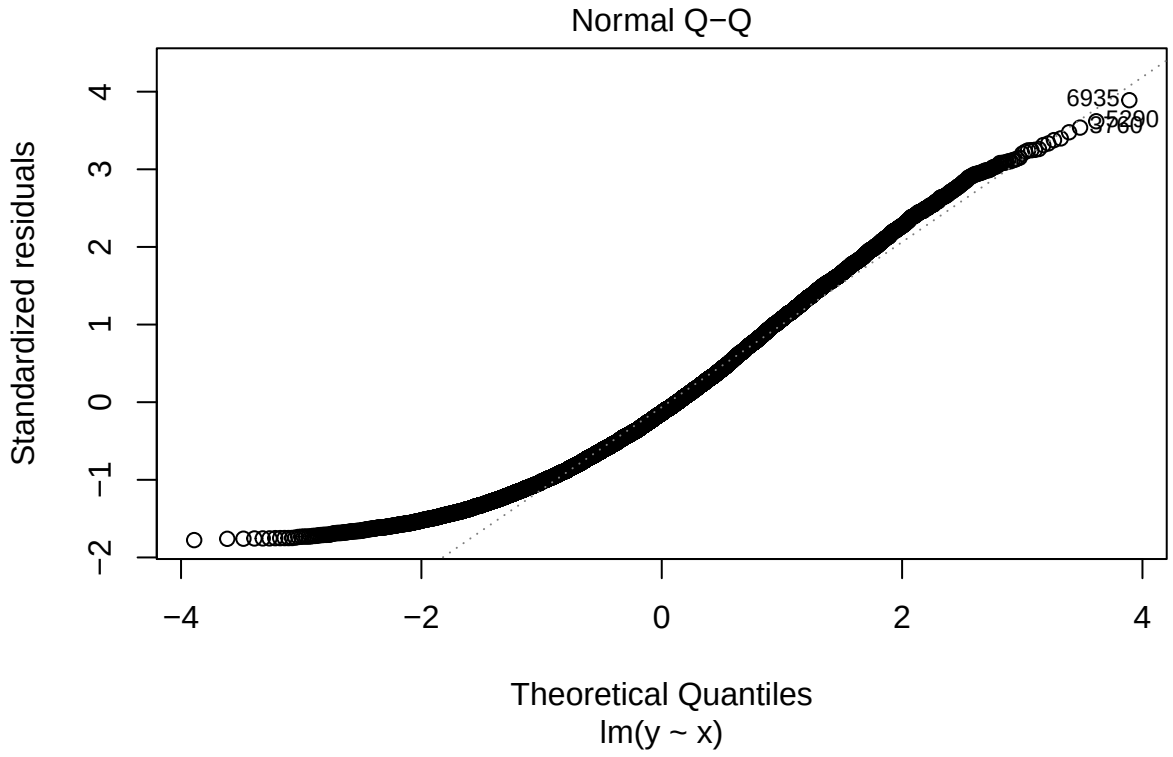
```
xyplot(y ~ x, data=ds, type=c("p", "r"), lwd=5)
```



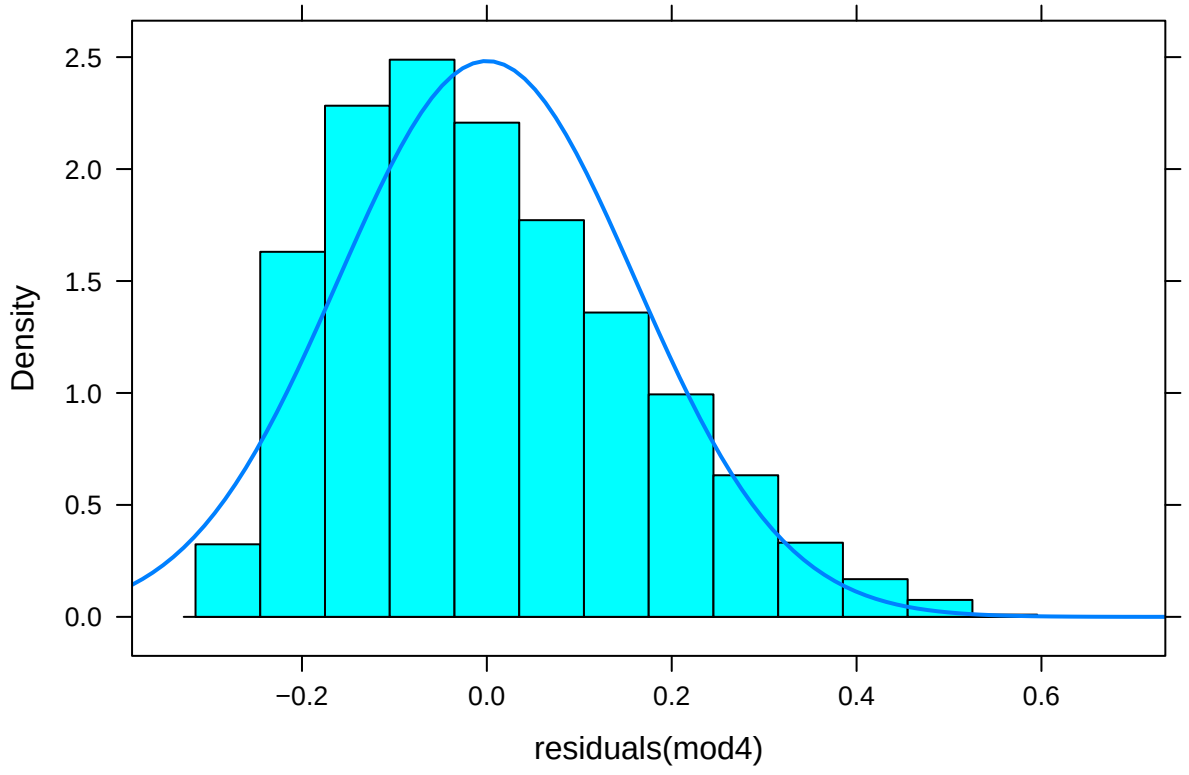

```
mod4 = lm(y ~ x, data=ds)
plot(mod4, which=1)
```




```
plot(mod4, which=2)
```




```
histogram(~residuals(mod4), fit="normal")
```

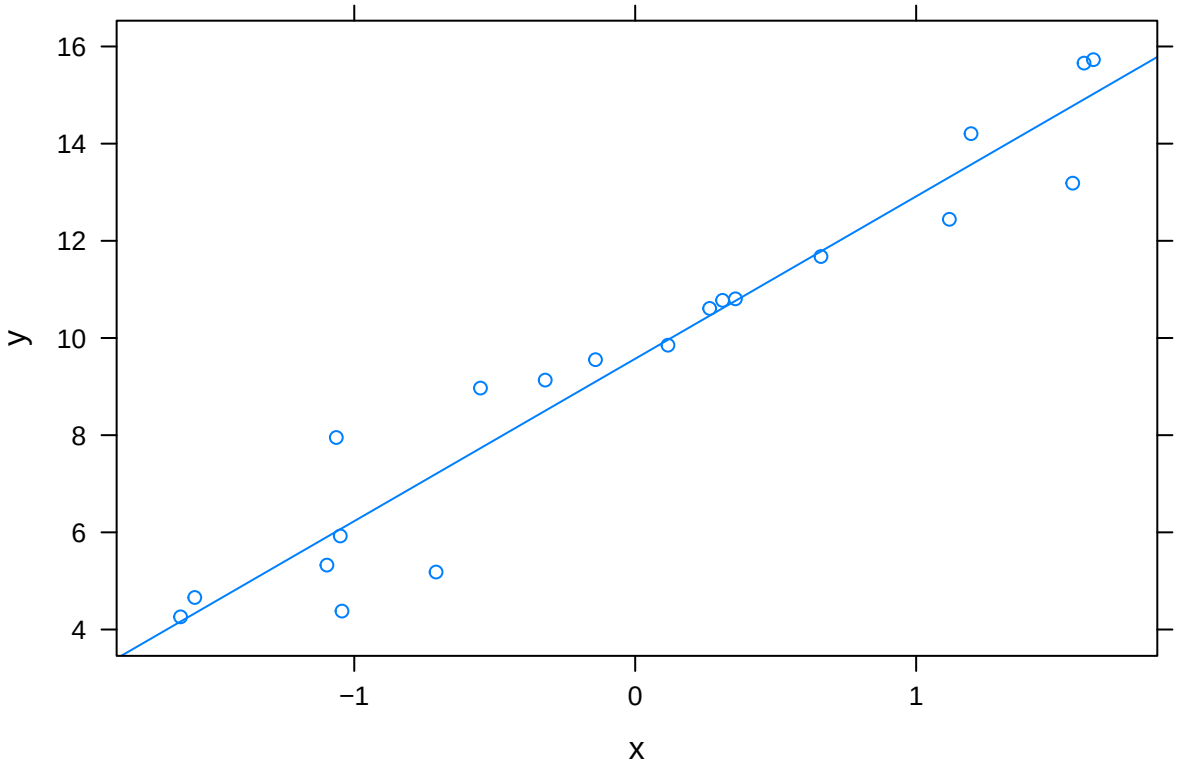


How much is too much?

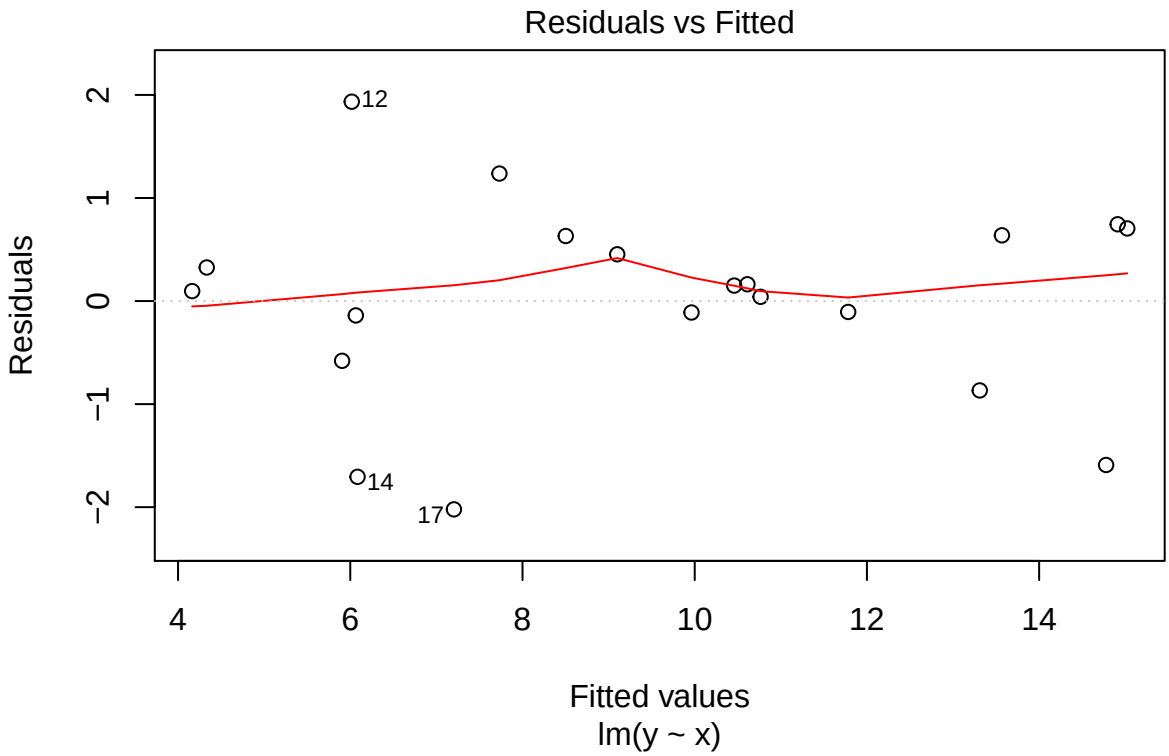
How loosely should we allow the conditions for regression to be violated? To get a feel for this, consider the following simulation. Note that, by construction, the data are generated from the model:

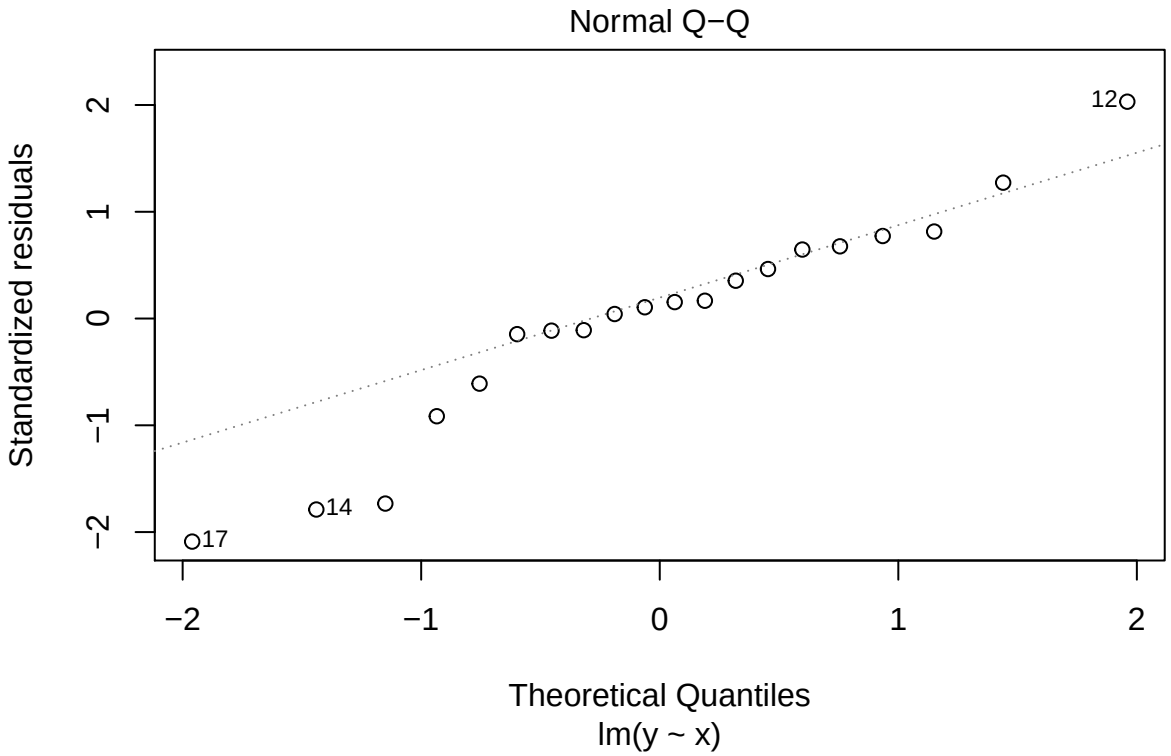
$$Y = 10 + 3 \cdot X + \epsilon, \qquad \epsilon \sim N(0, 1)$$

```
require(mosaic)
n = 20
ds = data.frame(x = rnorm(n))
ds = ds %>%
  mutate(y = 10 + 3 * x + rnorm(n))
xyplot(y ~ x, data=ds, type=c("p", "r"))
```




```
mod_ex <- lm(y ~x, data=ds)
plot(mod_ex, which=c(1,2))
```





Run this chunk several times, with several different choices of n. What do you notice?