What data patterns can lie behind a correlation coefficient?

Jan Vanhove

(For the original blog post (21 November 2016), see http://janhove.github.io/teaching/2016/11/21/what-correlations-look-like.)

In this post, I want to, first, help you to improve your intuition of what data patterns correlation coefficients can represent and, second, hammer home the point that to sensibly interpret a correlation coefficient, you need the corresponding scatterplot.

The briefest of introductions to correlation coefficients

If you haven't dealt with correlation coefficients much, a correlation coefficient, abbreviated as r, is a number between -1 and 1 that captures the strength of the linear relationship between two numeric variables. For instance, say you've asked 30 people about their weight and body height and plot these 30 (weight, height)-pairs in a scatterplot.

- If all 30 data points fall perfectly on an increasing line, then the correlation between these two variables will be $r = 1$.
- If, however, the general shape of the (weight, height) relationship is an increasing one but the 30 data points don't fall perfectly on a single line, then r will be somewhere between 0 and 1; the closer the data points are to a straight line, the closer to 1 r will be.
- If the relationship is a decreasing one, then r will lie between 0 and -1,
- and if there's no linear relationship between weight and height at all, r will be 0.

You'll find plenty of examples below.

One correlation coefficient can represent any number of patterns

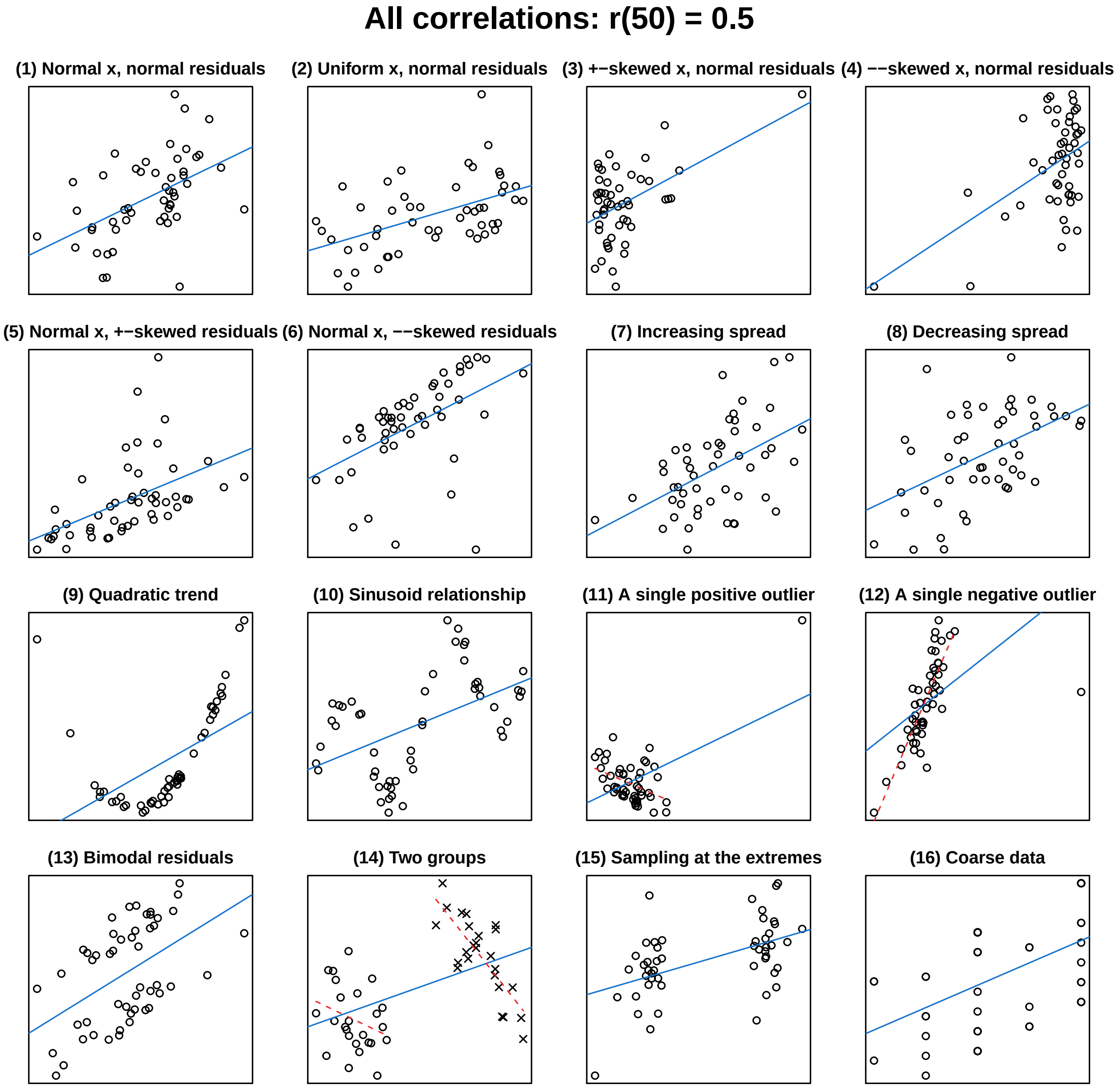
Correlation coefficients are popular among researchers because they allow them to summarise the relationship between two variables in a single number. However, a given correlation coefficient can represent any number of patterns between two variables, and without more information (ideally in the form of a scatterplot), the researchers themselves and their readers have no way of knowing which one.

To illustrate this, I've written an `R` function, `plot_r()`, that takes as its input a correlation coefficient and a sample size and outputs 16 quite different scatterplots that are all characterised by the same correlation coefficient. Below I first show and comment on how scatterplots for correlation coefficients of 0.5 and 0 based on 50 pairs might look like. Then, for those of you who don't use `R`, I provide scatterplots for a couple of other correlation coefficients so you can develop a sense of what the patterns underlying these correlation coefficients could look like.

What $r = 0.5$ might look like

First, `source()` the `plot_r()` function or download. Then, in `R`, you can draw scatterplots corresponding to a correlation coefficient of $r = 0.5$ and 50 observations like this. (You'll end up with a different set of scatterplots as the scatterplots are created randomly.)

```
source("http://janhove.github.io/RCode/plot_r.R")
plot_r(r = 0.5, n = 50)
```



Some comments

Top row

I suspect that when people think of a relationship with a correlation coefficient of 0.5, they have something like plots (1) and (2) in their mind's eye. For both plots, the underlying relationship between X and Y is linear, and the Y values are normally distributed about the best-fitting straight line. The minor difference between (1) and (2) is that for (1), X is normally distributed and for (2), X is uniformly distributed. These two plots represent the kind of relationship that *r* was meant to capture.

Plot (3) differs from (1) and (2) in that the X variable is now sampled from a skewed distribution. In this case, most X values are comparatively low but one X value is fairly large; if you run the code yourself, you may find that there are no outlying values or that there are more than one. Such a distribution may occur occur when X represents, for instance, participants' performance on a task that was too difficult (floor effect). In such a case, one or a couple of outlying but genuine X values may (not *will*) have 'high leverage', that is, they may unduly affect the correlation coefficient by pulling it up or down.

The problem in (4) is similar to the one in (3), but now most X values are comparatively large and a handful are fairly low, perhaps because X represents participants' performance on a task that was too easy (ceiling effects). Here, too, outlying points may have 'high leverage', i.e., they may unduly affect the correlation coefficient such that it doesn't accurately characterise the bulk of the data.

Second row

Plots (5) and (6) are variations on the same theme as in plots (3) and (4): The Y values aren't normally distributed about the regression line but are skewed. In such cases, too, some outlying but genuine Y values may (not *will*) have 'high leverage', i.e., they may pull the correlation coefficient up or down much more than ordinary data points.

Plots (7) and (8) are two examples where the variability of the Y values about the straight line increases and decreases, respectively, as X becomes larger, though admittedly, it isn't very clear in this example. This is known as heteroskedasticity. The main problems with blindly relying on correlation coefficients in the presence of heteroskedasticity, in my view, are that (a) 'r = 0.5' both _under_sells how well Y can be estimated from X for low (high) X values and _over_sells how well Y can be estimated from X for high (low) X values, and (b) by just reporting the correlation coefficient you gloss over an important aspect of the data. Additionally, heteroskedasticity may affect your inferential statistics.

Third row

Plot (9) illustrates that correlation coefficients express the strength of the *linear* relationship between two variables. If the relationship isn't linear, they're hardly informative. In this case, r = *0.5* seriously understates the strength of the XY relationship, which happens to be non-linear (quadratic in this case). The same goes for (10), where r = *0.5* understates the strength of the XY relationship and misses out on the cyclical nature of the relationship.

Plots (11) and (12) illustrate how a single outlying point, e.g., due to a technical error, can produce misleading correlation coefficients. In (11), a single outlying data point produces the significant positive correlation; had only the 49 data points on the left been considered, a negative relationship would've been observed (the dashed red line). Blindly going by r = *0.5* mischaracterises the bulk of the data. In (12), the relationship is considerably stronger than r = *0.5* suggests for the bulk of the data (the dashed red line); the outlier pulls the correlation coefficient down. Plots (11) and (12) differ from plots (3) and (4) in that in plots (3) and (4), the X values were all sampled from the same–but skewed–distribution and are, as such, genuine data points; in plots (11) and (12), the outliers were caused by a different mechanism from the other data points (e.g., a coding error or a technical glitch).

Fourth row

In (13), the Y values are bimodally distributed about the regression line. This suggests that we have overlooked an important aspect of the data, such as grouping factor: perhaps the datapoints above the regression line were sampled from a different population than those below the regression line.

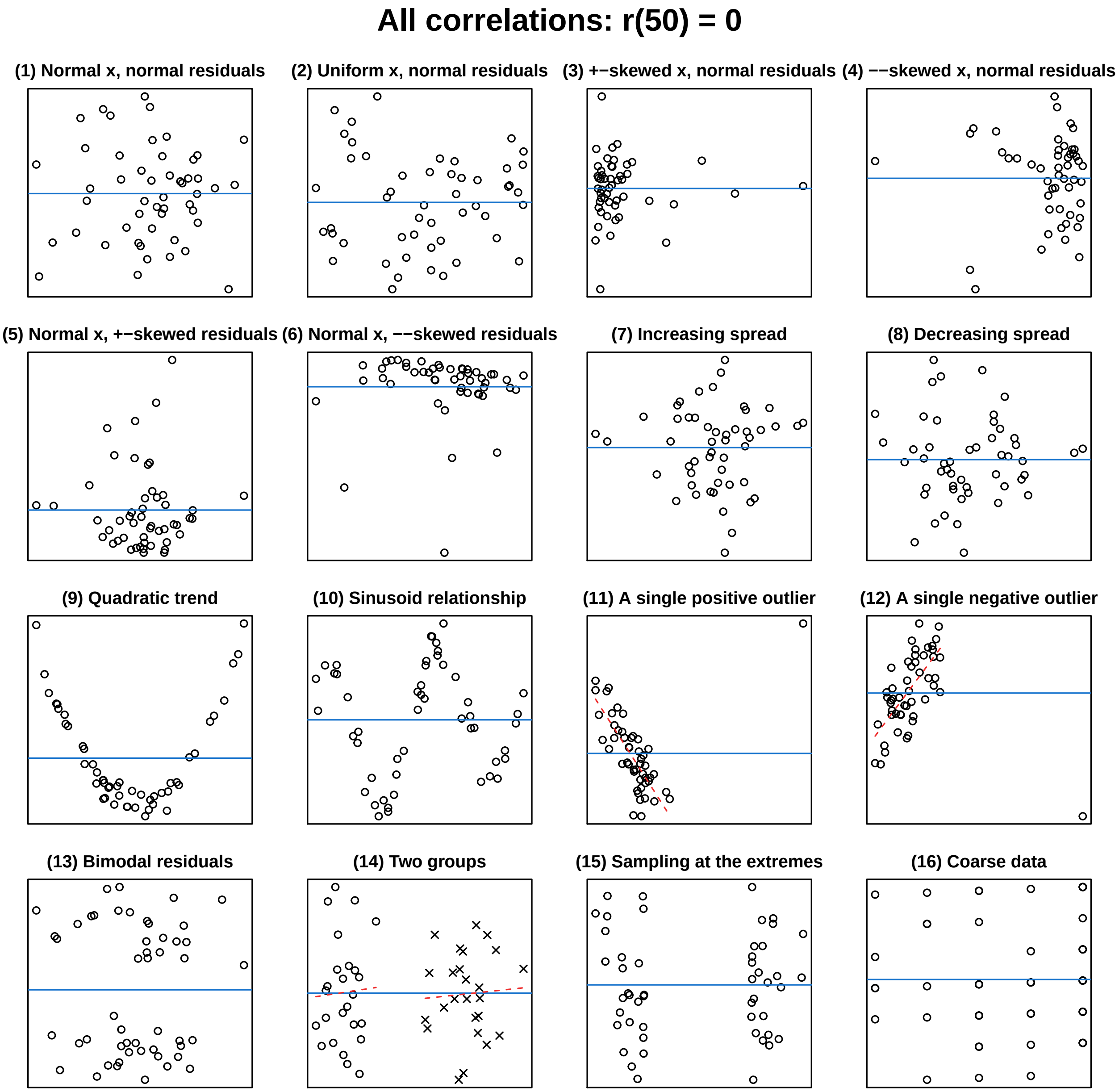
The situation in plot (14) is similar to but considerably worse than the one in (13): The dataset contains two groups, but unlike in (13), the overall trend captured by r = *0.5* betrays the fact that within each of these groups, the XY relationship is actually *negative.* Plot (14) will often, but not always, produce such a pattern, which is known as Simpson's paradox.

Plot (15) depicts a situation where the researchers, rather than investigating the XY relationship along the entire X range, only investigated the cases with the most extreme X values. Sampling at the extremes inflates correlation coefficients (see reason no. 2 why I don't particularly like correlation coefficients to begin with). In other words, if you took a sample of 150 XY cases and only looked at the 50 most extreme X observations, you'd end up with a correlation coefficient that is very likely to be larger than the one you'd observe if you looked at all 150 cases.

Plot (16), finally, is what I suspect many correlation coefficients actually represent. The X and Y data are lumpy, for instance, because they represent count data or responses to questionnaire data. I don't think correlation coefficients for such patterns are deceptive per se, but we're clearly talking about a different pattern than in plots (1) and (2).

What $r = 0$ might look like

```
plot_r(r = 0, n = 50)
```



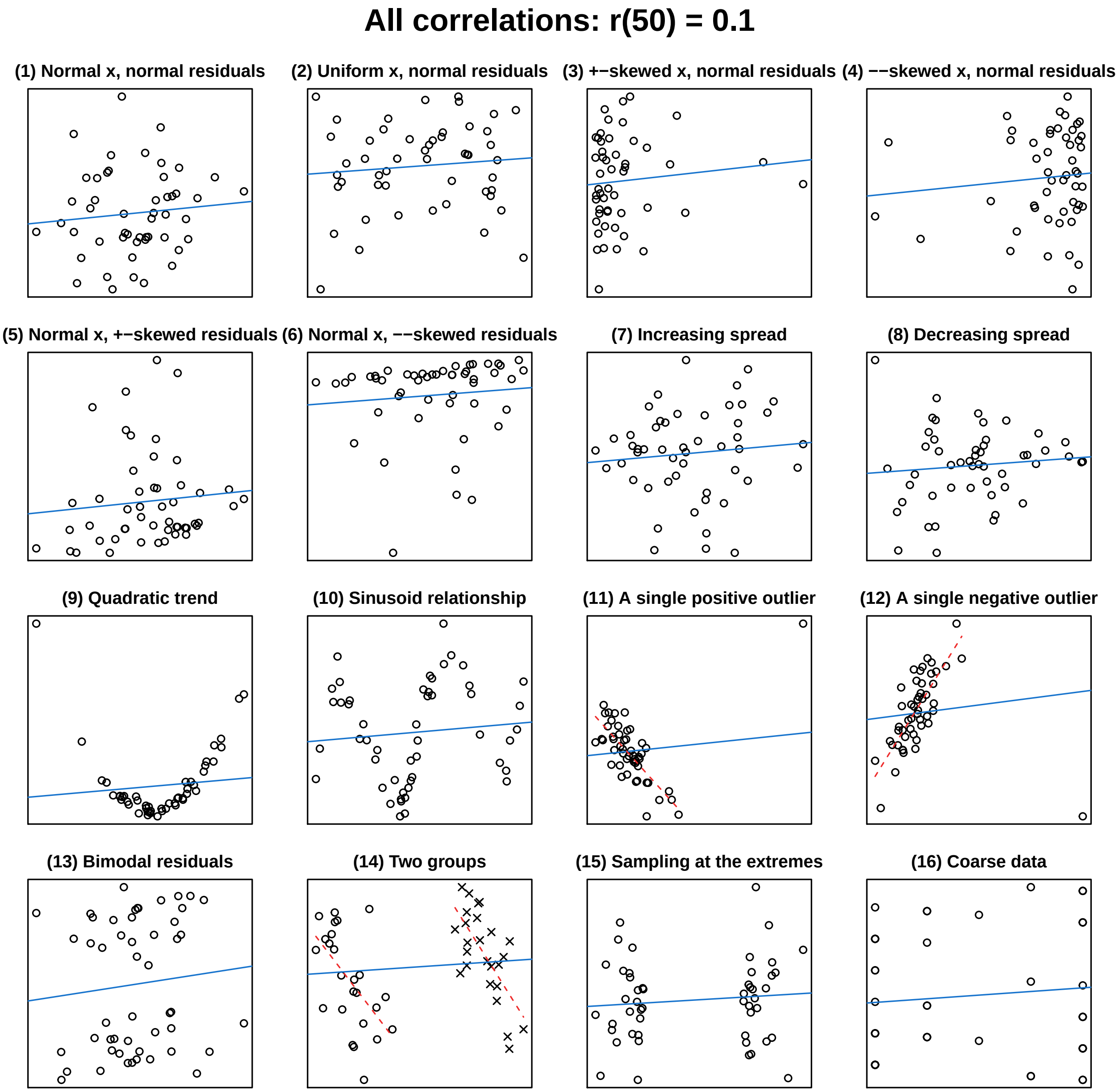
Some comments

The main point I want to make here is that $r = 0$ doesn't necessarily mean that there's no XY relationship. This is clear from plots (9) and (10), which evince strong *non-linear* relationships. Plots (11) and (12) similarly underscore this point: There exists a strong relationship for the bulk of the data, but this trend is cancelled out by a single outlying data point. Occasionally, in plot (14), a trend present in two subgroups may not be visible in an aggregated analysis; this doesn't seem to be the case in this example, though.

Two other examples

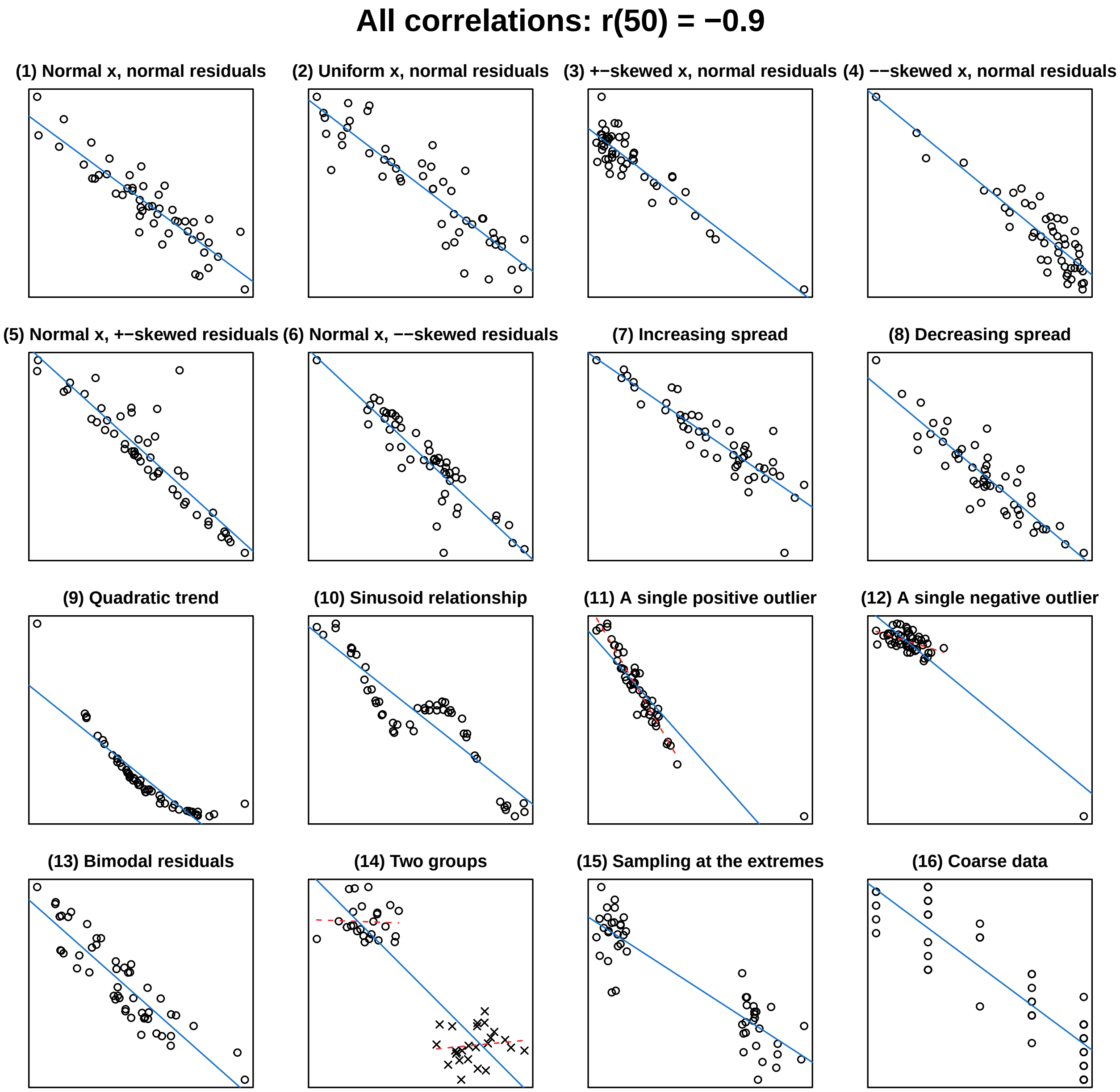
A weak positive correlation: $r = 0.1$

```
plot_r(r = 0.1, n = 50)
```



A strong negative correlation: $r =$ -0.9

```r
plot_r(r = -0.9, n = 50)
```



Storing the data

If you want to store the data underlying one or all of the plots, you can set the optional `showdata` parameter to either `all` or a number between 1 and 16 corresponding to one of the plots:

```r
# Show data for plot 11 (not run)
plot_r(r = 0.4, n = 10, showdata = 11)
# Show data for all plots (not run)
plot_r(r = 0.8, n = 15, showdata = "all")
```

Conclusions

I hope the `plot_r()` function helps you to develop a feel for what correlation coefficients may actually represent and that this post may convince more researchers to **draw scatterplots** before running any correlation analyses—or regression analyses, for that matter—and to actually **show them** when reporting correlations.