1. Correlation and linear regression

The goal in this chapter is to introduce **correlation** and **linear regression**. These are the standard tools that statisticians rely on when analysing the relationship between continuous predictors and continuous outcomes.

1.1 Correlations

In this section we'll talk about how to describe the relationships *between* variables in the data. To do that, we want to talk mostly about the **correlation** between variables. But first, we need some data.

1.1.1 The data

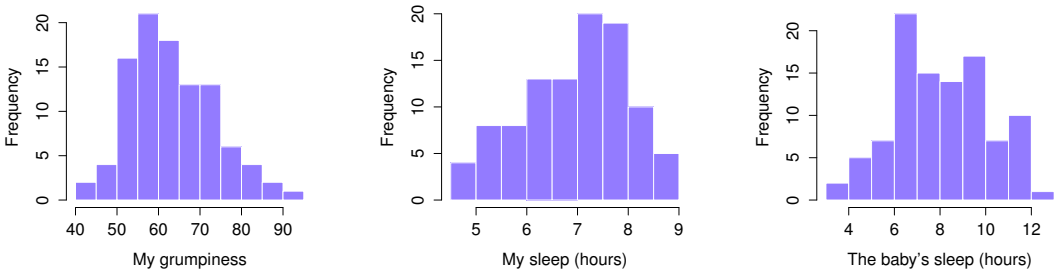
Table1.1 Descriptive statistics for the parenthood data.

variable	min	max	mean	median	std. dev	IQR
Dan's grumpiness	41	91	63.71	62	10.05	14
Dan's hours slept	4.84	9.00	6.97	7.03	1.02	1.45
Dan's son's hours slept	3.25	12.07	8.05	7.95	2.07	3.21

Let's turn to a topic close to every parent's heart: sleep. The data set we'll use is fictitious, but based on real events. Suppose I'm curious to find out how much my infant son's sleeping habits affect my mood. Let's say that I can rate my grumpiness very precisely, on a scale from 0 (not at

all grumpy) to 100 (grumpy as a very, very grumpy old man or woman). And lets also assume that I've been measuring my grumpiness, my sleeping patterns and my son's sleeping patterns for quite some time now. Let's say, for 100 days. And, being a nerd, I've saved the data as a file called `parenthood.csv`. If we load the data into JASP we can see that the file contains four variables `dan.sleep`, `baby.sleep`, `dan.grump` and `day`. Note that when you first load this data set JASP may not have guessed the data type for each variable correctly, in which case you should fix it: `dan.sleep`, `baby.sleep`, `dan.grump` and `day` can be specified as continuous variables, and `ID` is a nominal(integer) variable.

Next, I'll take a look at some basic descriptive statistics and, to give a graphical depiction of what each of the three interesting variables looks like, Figure **??** plots histograms. One thing to note: just because JASP can calculate dozens of different statistics doesn't mean you should report all of them. If I were writing this up for a report, I'd probably pick out those statistics that are of most interest to me (and to my readership), and then put them into a nice, simple table like the one in Table **??**.[*1] Notice that when I put it into a table, I gave everything "human readable" names. This is always good practice. Notice also that I'm not getting enough sleep. This isn't good practice, but other parents tell me that it's pretty standard.



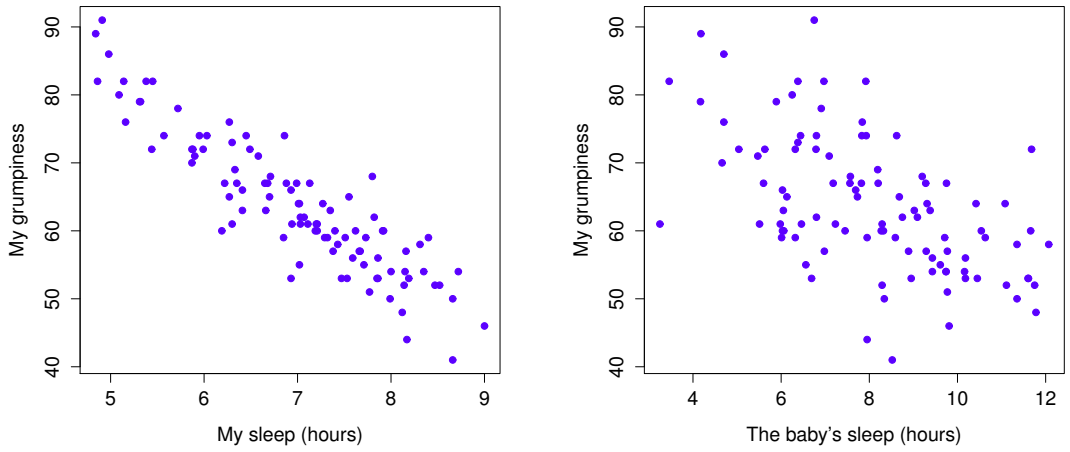
(a) (b) (c)

Figure1.1 Histograms for the three interesting variables in the `parenthood` data set.

1.1.2 The strength and direction of a relationship

We can draw scatterplots to give us a general sense of how closely related two variables are. Ideally

[*1] Actually, even that table is more than I'd bother with. In practice most people pick *one* measure of central tendency, and *one* measure of variability only.



(a) (b)

Figure1.2 Scatterplots showing the relationship between `dan.sleep` and `dan.grump` (left) and the relationship between `baby.sleep` and `dan.grump` (right).

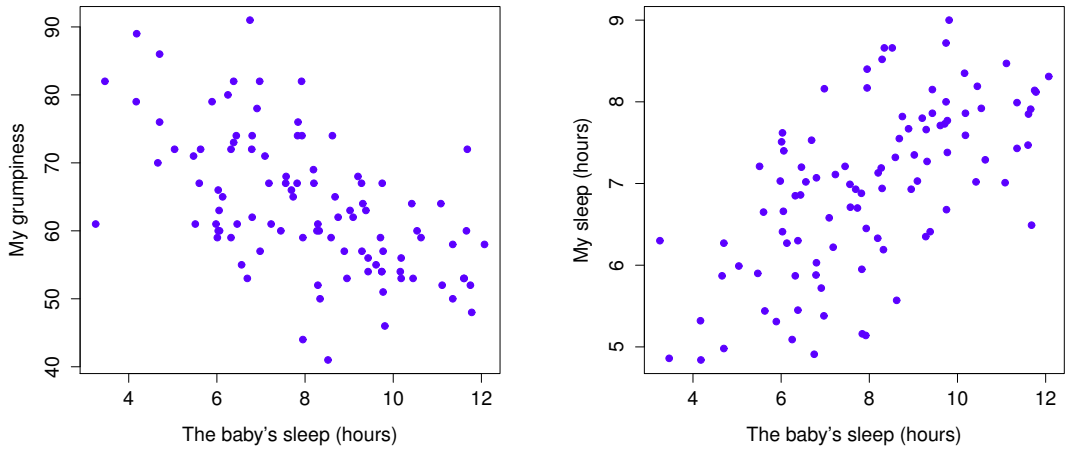
. . . . . .

though, we might want to say a bit more about it than that. For instance, let's compare the relationship between `dan.sleep` and `dan.grump` (Figure **??**, left) with that between `baby.sleep` and `dan.grump` (Figure **??**, right). When looking at these two plots side by side, it's clear that the relationship is *qualitatively* the same in both cases: more sleep equals less grump! However, it's also pretty obvious that the relationship between `dan.sleep` and `dan.grump` is *stronger* than the relationship between `baby.sleep` and `dan.grump`. The plot on the left is "neater" than the one on the right. What it feels like is that if you want to predict what my mood is, it'd help you a little bit to know how many hours my son slept, but it'd be more helpful to know how many hours I slept.

In contrast, let's consider the two scatterplots shown in Figure **??**. If we compare the scatterplot of "`baby.sleep` v `dan.grump`" (left) to the scatterplot of "'`baby.sleep` v `dan.sleep`" (right), the overall strength of the relationship is the same, but the direction is different. That is, if my son sleeps more, I get *more* sleep (positive relationship, right hand side), but if he sleeps more then I get *less* grumpy (negative relationship, left hand side).

1.1.3 The correlation coefficient

We can make these ideas a bit more explicit by introducing the idea of a **correlation coefficient** (or, more specifically, Pearson's correlation coefficient), which is traditionally denoted as r. The



(a) (b)

Figure1.3 Scatterplots showing the relationship between `baby.sleep` and `dan.grump` (left), as compared to the relationship between `baby.sleep` and `dan.sleep` (right).

correlation coefficient between two variables X and Y (sometimes denoted r_{XY}), which we'll define more precisely in the next section, is a measure that varies from -1 to 1. When $r = -1$ it means that we have a perfect negative relationship, and when $r = 1$ it means we have a perfect positive relationship. When $r = 0$, there's no relationship at all. If you look at Figure **??**, you can see several plots showing what different correlations look like.

The formula for the Pearson's correlation coefficient can be written in several different ways. I think the simplest way to write down the formula is to break it into two steps. Firstly, let's introduce the idea of a **covariance**. The covariance between two variables X and Y is a generalisation of the notion of the variance amd is a mathematically simple way of describing the relationship between

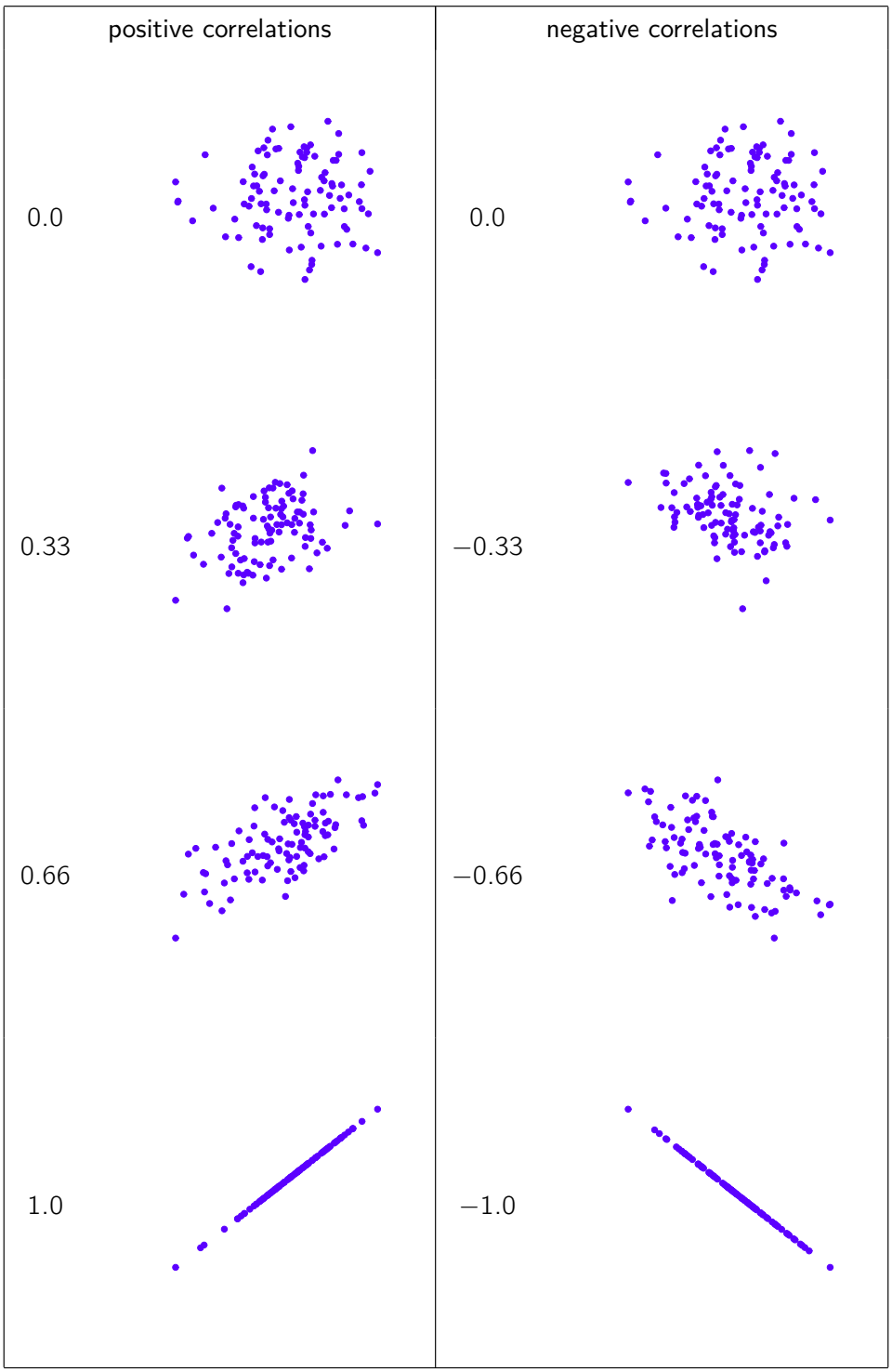


Figure1.4 Illustration of the effect of varying the strength and direction of a correlation. In the left hand column, the correlations are 0, .33, .66 and 1. In the right hand column, the correlations are 0, -.33, -.66 and -1.

two variables that isn't terribly informative to humans

$$\text{Cov}(X, Y) = \frac{1}{N-1} \sum_{i=1}^{N} \left(X_i - \bar{X}\right) \left(Y_i - \bar{Y}\right)$$

Because we're multiplying (i.e., taking the "product" of) a quantity that depends on X by a quantity that depends on Y and then averaging[a], you can think of the formula for the covariance as an "average cross product" between X and Y.

The covariance has the nice property that, if X and Y are entirely unrelated, then the covariance is exactly zero. If the relationship between them is positive (in the sense shown in Figure **??**) then the covariance is also positive, and if the relationship is negative then the covariance is also negative. In other words, the covariance captures the basic qualitative idea of correlation. Unfortunately, the raw magnitude of the covariance isn't easy to interpret as it depends on the units in which X and Y are expressed and, worse yet, the actual units that the covariance itself is expressed in are really weird. For instance, if X refers to the `dan.sleep` variable (units: hours) and Y refers to the `dan.grump` variable (units: grumps), then the units for their covariance are "hours $\times$ grumps". And I have no freaking idea what that would even mean.

The Pearson correlation coefficient r fixes this interpretation problem by standardising the covariance, in pretty much the exact same way that the z-score standardises a raw score, by dividing by the standard deviation. However, because we have two variables that contribute to the covariance, the standardisation only works if we divide by both standard deviations.[b] In other words, the correlation between X and Y can be written as follows:

$$r_{XY} = \frac{\text{Cov}(X, Y)}{\hat{\sigma}_X \, \hat{\sigma}_Y}$$

[a] Just like we saw with the variance and the standard deviation, in practice we divide by $N - 1$ rather than N.
[b] This is an oversimplification, but it'll do for our purposes.

By standardising the covariance, not only do we keep all of the nice properties of the covariance discussed earlier, but the actual values of r are on a meaningful scale: $r = 1$ implies a perfect positive relationship and $r = -1$ implies a perfect negative relationship. I'll expand a little more on this point later, in Section **??**. But before I do, let's look at how to calculate correlations in JASP.

1.1.4 Calculating correlations in JASP

Calculating correlations in JASP can be done by clicking on the 'Regression' - 'Correlation Matrix' button. Transfer all four continuous variables across into the box on the right to get the output in Figure **??**. Notice that each correlation (denoted 'Pearson's r') is paired with a p-value. Clearly,

something is being tested here, but ignore it for now. We'll talk more about that soon!

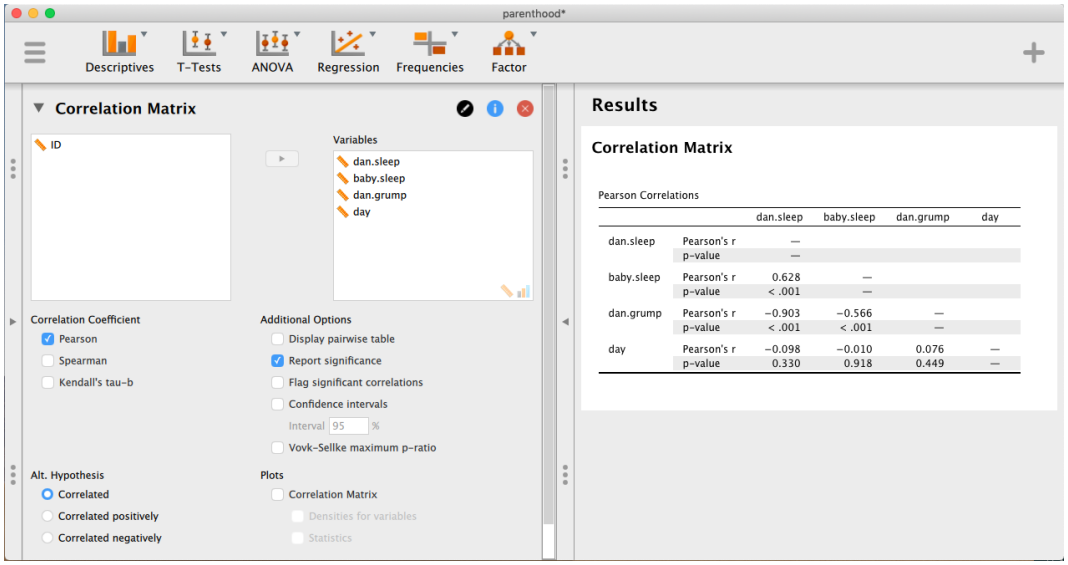


Figure1.5 A JASP screenshot showing correlations between variables in the `parenthood.csv` file

1.1.5 Interpreting a correlation

Naturally, in real life you don't see many correlations of 1. So how should you interpret a correlation of, say, $r = .4$? The honest answer is that it really depends on what you want to use the data for, and on how strong the correlations in your field tend to be. A friend of mine in engineering once argued that any correlation less than .95 is completely useless (I think he was exaggerating, even for engineering). On the other hand, there are real cases, even in psychology, where you should really expect correlations that strong. For instance, one of the benchmark data sets used to test theories of how people judge similarities is so clean that any theory that can't achieve a correlation of at least .9 really isn't deemed to be successful. However, when looking for (say) elementary correlates of intelligence (e.g., inspection time, response time), if you get a correlation above .3 you're doing very very well. In short, the interpretation of a correlation depends a lot on the context. That said, the rough guide in Table **??** is pretty typical.

However, something that can never be stressed enough is that you should *always* look at the scatterplot before attaching any interpretation to the data. A correlation might not mean what

Table1.2 A rough guide to interpreting correlations. Note that I say a *rough* guide. There aren't hard and fast rules for what counts as strong or weak relationships. It depends on the context.

Correlation	Strength	Direction
-1.0 to -0.9	Very strong	Negative
-0.9 to -0.7	Strong	Negative
-0.7 to -0.4	Moderate	Negative
-0.4 to -0.2	Weak	Negative
-0.2 to 0	Negligible	Negative
0 to 0.2	Negligible	Positive
0.2 to 0.4	Weak	Positive
0.4 to 0.7	Moderate	Positive
0.7 to 0.9	Strong	Positive
0.9 to 1.0	Very strong	Positive

you think it means. The classic illustration of this is "Anscombe's Quartet" (**Anscombe1973**), a collection of four data sets. Each data set has two variables, an X and a Y. For all four data sets the mean value for X is 9 and the mean for Y is 7.5. The standard deviations for all X variables are almost identical, as are those for the Y variables. And in each case the correlation between X and Y is $r = 0.816$. You can verify this yourself, since I happen to have saved it in a file called `anscombe.csv`.

You'd think that these four data sets would look pretty similar to one another. They do not. If we draw scatterplots of X against Y for all four variables, as shown in Figure **??**, we see that all four of these are *spectacularly* different to each other. The lesson here, which so very many people seem to forget in real life, is " *always graph your raw data*" (Chapter **??**).

1.1.6 Spearman's rank correlations

The Pearson correlation coefficient is useful for a lot of things, but it does have shortcomings. One issue in particular stands out: what it actually measures is the strength of the *linear* relationship between two variables. In other words, what it gives you is a measure of the extent to which the data all tend to fall on a single, perfectly straight line. Often, this is a pretty good approximation to what we mean when we say "relationship", and so the Pearson correlation is a good thing to calculate. Sometimes though, it isn't.

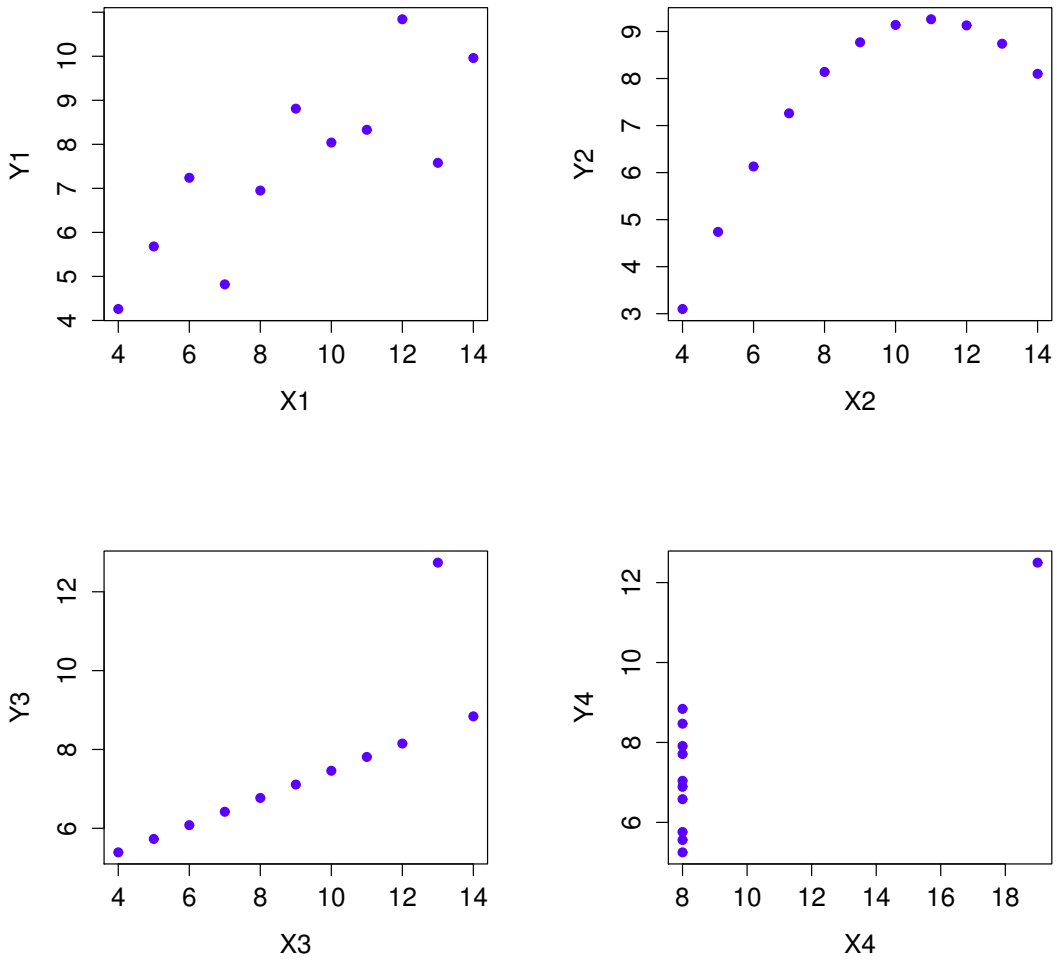


Figure1.6 Anscombe's quartet. All four of these data sets have a Pearson correlation of $r = .816$, but they are qualitatively different from one another.

One very common situation where the Pearson correlation isn't quite the right thing to use arises when an increase in one variable X really is reflected in an increase in another variable Y, but the nature of the relationship isn't necessarily linear. An example of this might be the relationship between effort and reward when studying for an exam. If you put zero effort (X) into learning a subject then you should expect a grade of 0% (Y). However, a little bit of effort will cause a *massive* improvement. Just turning up to lectures means that you learn a fair bit, and if you just turn up to classes and scribble a few things down your grade might rise to 35%, all without a lot of effort.

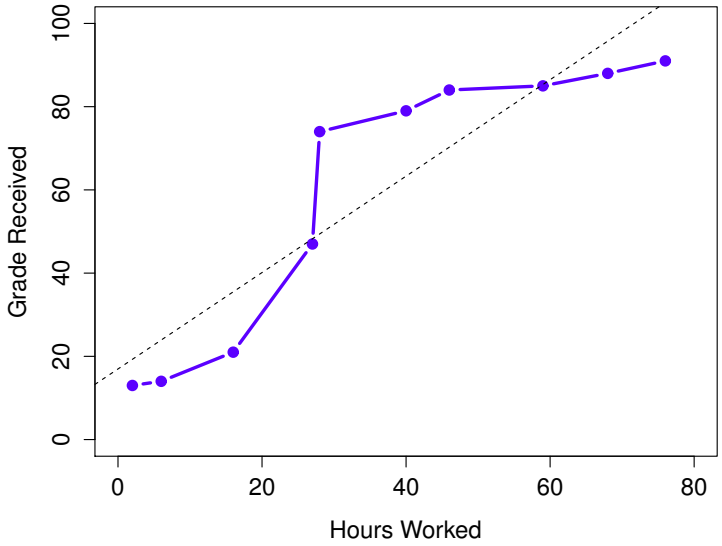


Figure1.7 The relationship between hours worked and grade received for a toy data set consisting of only 10 students (each circle corresponds to one student). The dashed line through the middle shows the linear relationship between the two variables. This produces a strong Pearson correlation of $r = .91$. However, the interesting thing to note here is that there's actually a perfect monotonic relationship between the two variables. In this toy example, increasing the hours worked always increases the grade received, as illustrated by the solid line. This is reflected in a Spearman correlation of $\rho = 1$. With such a small data set, however, it's an open question as to which version better describes the actual relationship involved.

However, you just don't get the same effect at the other end of the scale. As everyone knows, it takes *a lot* more effort to get a grade of 90% than it takes to get a grade of 55%. What this means is that, if I've got data looking at study effort and grades, there's a pretty good chance that Pearson correlations will be misleading.

To illustrate, consider the data plotted in Figure **??**, showing the relationship between hours worked and grade received for 10 students taking some class. The curious thing about this (highly fictitious) data set is that increasing your effort *always* increases your grade. It might be by a lot or it might be by a little, but increasing effort will never decrease your grade. If we run a standard Pearson correlation, it shows a strong relationship between hours worked and grade received, with a correlation coefficient of `0.91`. However, this doesn't actually capture the observation that increasing hours worked *always* increases the grade. There's a sense here in which we want to be able to say

that the correlation is *perfect* but for a somewhat different notion of what a "relationship" is. What we're looking for is something that captures the fact that there is a perfect **ordinal relationship** here. That is, if student 1 works more hours than student 2, then we can guarantee that student 1 will get the better grade. That's not what a correlation of $r = .91$ says at all.

How should we address this? Actually, it's really easy. If we're looking for ordinal relationships all we have to do is treat the data as if it were ordinal scale! So, instead of measuring effort in terms of "hours worked", lets rank all 10 of our students in order of hours worked. That is, student 1 did the least work out of anyone (2 hours) so they get the lowest rank (rank = 1). Student 4 was the next laziest, putting in only 6 hours of work over the whole semester, so they get the next lowest rank (rank = 2). Notice that I'm using "rank =1" to mean "low rank". Sometimes in everyday language we talk about "rank = 1" to mean "top rank" rather than "bottom rank". So be careful, you can rank "from smallest value to largest value" (i.e., small equals rank 1) or you can rank "from largest value to smallest value" (i.e., large equals rank 1). In this case, I'm ranking from smallest to largest, but as it's really easy to forget which way you set things up you have to put a bit of effort into remembering!

Okay, so let's have a look at our students when we rank them from worst to best in terms of effort and reward:

	rank (hours worked)	rank (grade received)
student 1	1	1
student 2	10	10
student 3	6	6
student 4	2	2
student 5	3	3
student 6	5	5
student 7	4	4
student 8	8	8
student 9	7	7
student 10	9	9

Hmm. These are *identical*. The student who put in the most effort got the best grade, the student with the least effort got the worst grade, etc. As the table above shows, these two rankings are identical, so if we now correlate them we get a perfect relationship, with a correlation of `1.0`.

What we've just re-invented is **Spearman's rank order correlation**, usually denoted ρ to distinguish it from the Pearson correlation r. We can calculate Spearman's ρ using JASP simply by clicking the 'Spearman' check box in the 'Correlation Matrix' screen.

1.2

Scatterplots

Scatterplots are a simple but effective tool for visualising the relationship between *two* variables, like we saw with the figures in the section on correlation (Section **??**). It's this latter application that we usually have in mind when we use the term "scatterplot". In this kind of plot each observation corresponds to one dot. The horizontal location of the dot plots the value of the observation on one variable, and the vertical location displays its value on the other variable. In many situations you don't really have a clear opinions about what the *causal* relationship is (e.g., does A cause B, or does B cause A, or does some other variable C control both A and B). If that's the case, it doesn't really matter which variable you plot on the x-axis and which one you plot on the y-axis. However, in many situations you do have a pretty strong idea which variable you think is most likely to be causal, or at least you have some suspicions in that direction. If so, then it's conventional to plot the cause variable on the x-axis, and the effect variable on the y-axis. With that in mind, let's look at how to draw scatterplots in JASP, using the same `parenthood` data set (i.e. `parenthood.csv`) that I used when introducing correlations.

Suppose my goal is to draw a scatterplot displaying the relationship between the amount of sleep that I get (`dan.sleep`) and how grumpy I am the next day (`dan.grump`). The way in which we can use JASP to get this plot is to use the 'Plots' option under the 'Regression' - 'Correlation Matrix' button, giving us the output shown in Figure **??**. Note that JASP draws a line through the points, we'll come onto this a bit later in Section (**??**). Plotting a scatterplot in this way also allow you to specify 'Densities for variables', which adds a histogram and density curve showing how the data in each variable is distributed. You can also specify the 'Statistics' option, which provides an estimate of the correlation along with a 95% confidence interval.

1.3

What is a linear regression model?

Stripped to its bare essentials, linear regression models are basically a slightly fancier version of the Pearson correlation (Section **??**), though as we'll see regression models are much more powerful tools.

Since the basic ideas in regression are closely tied to correlation, we'll return to the `parenthood.csv` file that we were using to illustrate how correlations work. Recall that, in this data

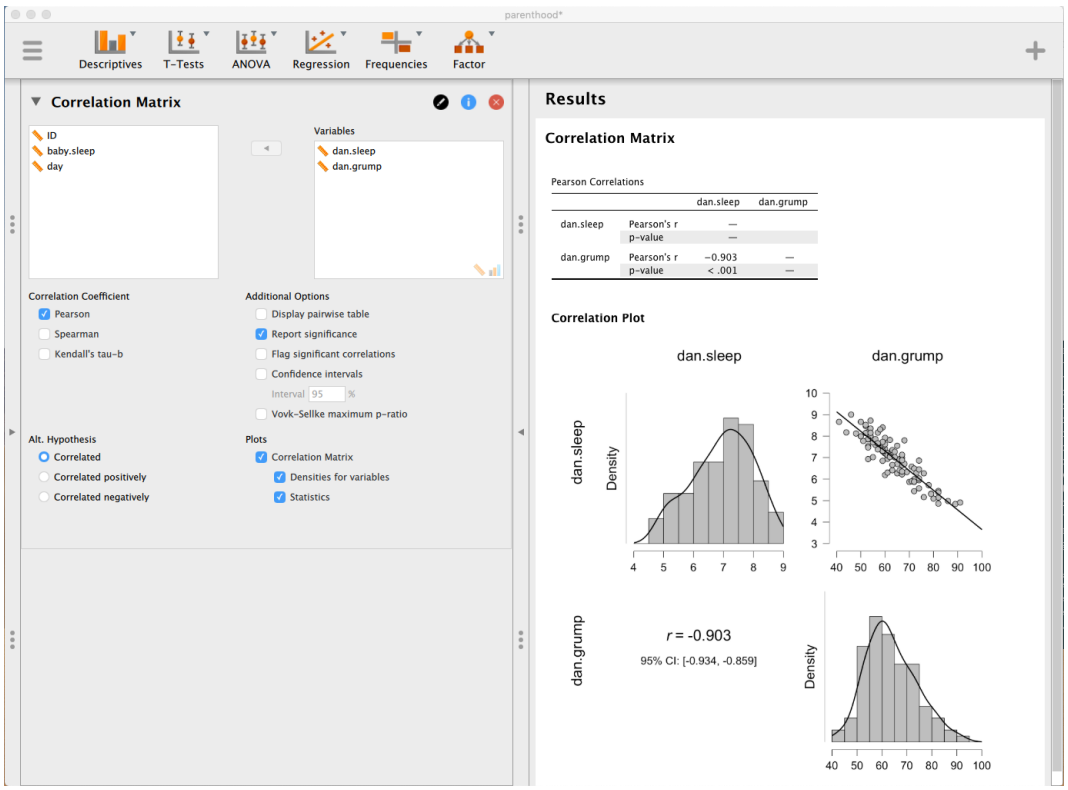
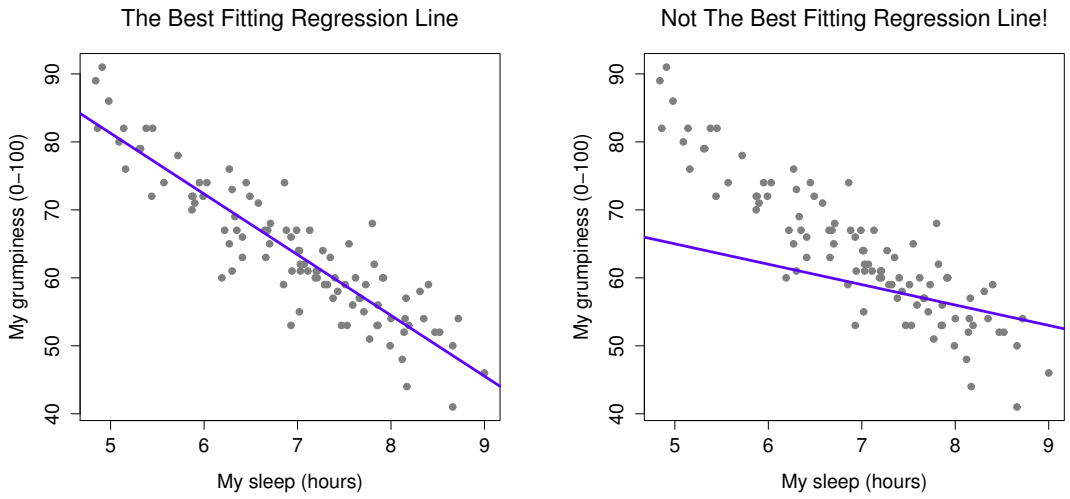


Figure1.8 Scatterplot via the 'Correlation Matrix' method in JASP

set we were trying to find out why Dan is so very grumpy all the time and our working hypothesis was that I'm not getting enough sleep. We drew some scatterplots to help us examine the relationship between the amount of sleep I get and my grumpiness the following day, as in Figure **??**, and as we saw previously this corresponds to a correlation of $r = -.90$, but what we find ourselves secretly imagining is something that looks closer to Figure **??**a. That is, we mentally draw a straight line through the middle of the data. In statistics, this line that we're drawing is called a **regression line**. Notice that, since we're not idiots, the regression line goes through the middle of the data. We don't find ourselves imagining anything like the rather silly plot shown in Figure **??**b.

This is not highly surprising. The line that I've drawn in Figure **??**b doesn't "fit" the data very well, so it doesn't make a lot of sense to propose it as a way of summarising the data, right? This is a very simple observation to make, but it turns out to be very powerful when we start trying to wrap just a little bit of maths around it. To do so, let's start with a refresher of some high school maths. The formula for a straight line is usually written like this

$$y = a + bx$$



(a) (b)

Figure1.9 Panel a shows the sleep-grumpiness scatterplot from Figure **??** with the best fitting regression line drawn over the top. Not surprisingly, the line goes through the middle of the data. In contrast, panel b shows the same data, but with a very poor choice of regression line drawn over the top.

. . . . . .

Or, at least, that's what it was when I went to high school all those years ago. The two *variables* are x and y, and we have two *coefficients*, a and b.[*2] The coefficient a represents the *y-intercept* of the line, and coefficient b represents the *slope* of the line. Digging further back into our decaying memories of high school (sorry, for some of us high school was a long time ago), we remember that the intercept is interpreted as "the value of y that you get when $x = 0$". Similarly, a slope of b means that if you increase the x-value by 1 unit, then the y-value goes up by b units, and a negative slope means that the y-value would go down rather than up. Ah yes, it's all coming back to me now. Now that we've remembered that it should come as no surprise to discover that we use the exact same formula for a regression line. If Y is the outcome variable (the DV) and X is the predictor variable (the IV), then the formula that describes our regression is written like this

$$\hat{Y}_i = b_0 + b_1 X_i$$

Hmm. Looks like the same formula, but there's some extra frilly bits in this version. Let's make sure we understand them. Firstly, notice that I've written X_i and Y_i rather than just plain old X and Y. This is because we want to remember that we're dealing with actual data. In this equation,

[*2] Also sometimes written as $y = mx + b$ where m is the slope coefficient and b is the intercept (constant) coefficient.

X_i is the value of predictor variable for the ith observation (i.e., the number of hours of sleep that I got on day i of my little study), and Y_i is the corresponding value of the outcome variable (i.e., my grumpiness on that day). And although I haven't said so explicitly in the equation, what we're assuming is that this formula works for all observations in the data set (i.e., for all i). Secondly, notice that I wrote $\hat{Y}_i$ and not Y_i. This is because we want to make the distinction between the *actual data* Y_i, and the *estimate* $\hat{Y}_i$ (i.e., the prediction that our regression line is making). Thirdly, I changed the letters used to describe the coefficients from a and b to b_0 and b_1. That's just the way that statisticians like to refer to the coefficients in a regression model. I've no idea why they chose b, but that's what they did. In any case b_0 always refers to the intercept term, and b_1 refers to the slope.

Excellent, excellent. Next, I can't help but notice that, regardless of whether we're talking about the good regression line or the bad one, the data don't fall perfectly on the line. Or, to say it another way, the data Y_i are not identical to the predictions of the regression model $\hat{Y}_i$. Since statisticians love to attach letters, names and numbers to everything, let's refer to the difference between the model prediction and that actual data point as a *residual*, and we'll refer to it as ε_i.[*3] Written using mathematics, the residuals are defined as

$$\varepsilon_i = Y_i - \hat{Y}_i$$

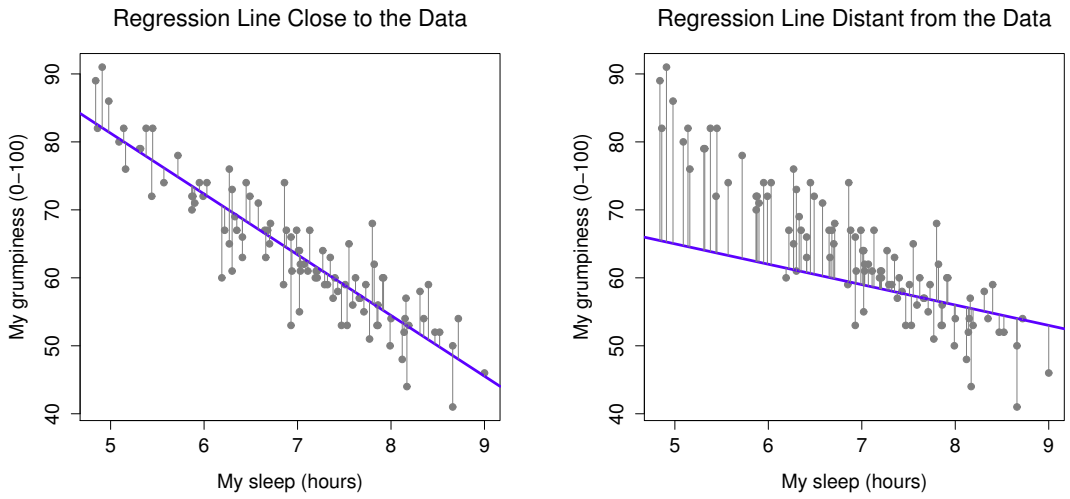
which in turn means that we can write down the complete linear regression model as

$$Y_i = b_0 + b_1 X_i + \varepsilon_i$$

1.4 Estimating a linear regression model

Okay, now let's redraw our pictures but this time I'll add some lines to show the size of the residual for all observations. When the regression line is good, our residuals (the lengths of the solid black lines) all look pretty small, as shown in Figure **??**a, but when the regression line is a bad one the residuals are a lot larger, as you can see from looking at Figure **??**b. Hmm. Maybe what we "want" in a regression model is *small* residuals. Yes, that does seem to make sense. In fact, I think I'll go so far as to say that the "best fitting" regression line is the one that has the smallest residuals. Or, better yet, since statisticians seem to like to take squares of everything why not say that:

[*3] The ε symbol is the Greek letter epsilon. It's traditional to use ε_i or e_i to denote a residual.



(a) (b)

Figure1.10 A depiction of the residuals associated with the best fitting regression line (panel a), and the residuals associated with a poor regression line (panel b). The residuals are much smaller for the good regression line. Again, this is no surprise given that the good line is the one that goes right through the middle of the data.

. . . . . .

> The estimated regression coefficients, $\hat{b}_0$ and $\hat{b}_1$, are those that minimise the sum of the squared residuals, which we could either write as $\sum_i (Y_i - \hat{Y}_i)^2$ or as $\sum_i {\epsilon_i}^2$.

Yes, yes that sounds even better. And since I've indented it like that, it probably means that this is the right answer. And since this is the right answer, it's probably worth making a note of the fact that our regression coefficients are *estimates* (we're trying to guess the parameters that describe a population!), which is why I've added the little hats, so that we get $\hat{b}_0$ and $\hat{b}_1$ rather than b_0 and b_1. Finally, I should also note that, since there's actually more than one way to estimate a regression model, the more technical name for this estimation process is **ordinary least squares (OLS) regression**.

At this point, we now have a concrete definition for what counts as our "best" choice of regression coefficients, $\hat{b}_0$ and $\hat{b}_1$. The natural question to ask next is, if our optimal regression coefficients are those that minimise the sum squared residuals, how do we *find* these wonderful numbers? The actual answer to this question is complicated and doesn't help you understand the logic of

regression.[*4] This time I'm going to let you off the hook. Instead of showing you the long and tedious way first and then "revealing" the wonderful shortcut that JASP provides, let's cut straight to the chase and just use JASP to do all the heavy lifting.

1.4.1 Linear regression in JASP

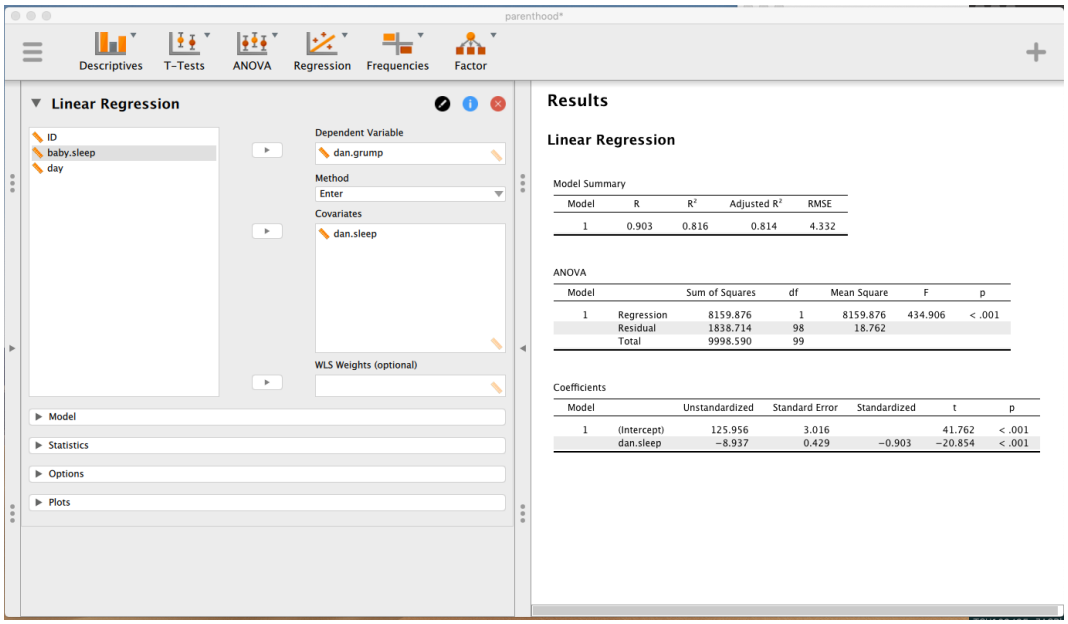


Figure1.11 A JASP screenshot showing a simple linear regression analysis.

To run my linear regression, open up the 'Regression' - 'Linear Regression' analysis in JASP, using the `parenthood.csv` data file. Then specify `dan.grump` as the 'Dependent Variable' and `dan.sleep` as the variable entered in the 'Covariates' box. This gives the results shown in Figure **??**, showing an intercept $\hat{b}_0 = 125.956$ and the slope $\hat{b}_1 = -8.937$. In other words, the best-fitting regression

[*4] Or at least, I'm assuming that it doesn't help most people. But on the off chance that someone reading this is a proper kung fu master of linear algebra (and to be fair, I always have a few of these people in my intro stats class), it *will* help *you* to know that the solution to the estimation problem turns out to be $\hat{\boldsymbol{b}} = (\mathrm{X}'\mathrm{X})^{-1}\mathrm{X}'\boldsymbol{y}$, where $\hat{\boldsymbol{b}}$ is a vector containing the estimated regression coefficients, X is the "design matrix" that contains the predictor variables (plus an additional column containing all ones; strictly X is a matrix of the regressors, but I haven't discussed the distinction yet), and $\boldsymbol{y}$ is a vector containing the outcome variable. For everyone else, this isn't exactly helpful and can be downright scary. However, since quite a few things in linear regression can be written in linear algebra terms, you'll see a bunch of footnotes like this one in this chapter. If you can follow the maths in them, great. If not, ignore it.

line that I plotted in Figure **??** has this formula:

$$\hat{Y}_i = 125.956 + (-8.937\ X_i)$$

1.4.2 Interpreting the estimated model

The most important thing to be able to understand is how to interpret these coefficients. Let's start with $\hat{b}_1$, the slope. If we remember the definition of the slope, a regression coefficient of $\hat{b}_1 = -8.94$ means that if I increase X_i by 1, then I'm decreasing Y_i by 8.94. That is, each additional hour of sleep that I gain will improve my mood, reducing my grumpiness by 8.94 grumpiness points. What about the intercept? Well, since $\hat{b}_0$ corresponds to "the expected value of Y_i when X_i equals 0", it's pretty straightforward. It implies that if I get zero hours of sleep ($X_i = 0$) then my grumpiness will go off the scale, to an insane value of ($Y_i = 125.96$). Best to be avoided, I think.

1.5 Multiple linear regression

The simple linear regression model that we've discussed up to this point assumes that there's a single predictor variable that you're interested in, in this case `dan.sleep`. In fact, up to this point *every* statistical tool that we've talked about has assumed that your analysis uses one predictor variable and one outcome variable. However, in many (perhaps most) research projects you actually have multiple predictors that you want to examine. If so, it would be nice to be able to extend the linear regression framework to be able to include multiple predictors. Perhaps some kind of **multiple regression** model would be in order?

Multiple regression is conceptually very simple. All we do is add more terms to our regression equation. Let's suppose that we've got two variables that we're interested in; perhaps we want to use both `dan.sleep` and `baby.sleep` to predict the `dan.grump` variable. As before, we let Y_i refer to my grumpiness on the i-th day. But now we have two X variables: the first corresponding to the amount of sleep I got and the second corresponding to the amount of sleep my son got. So we'll let X_{i1} refer to the hours I slept on the i-th day and X_{i2} refers to the hours that the baby slept on that day. If so, then we can write our regression model like this:

$$Y_i = b_0 + b_1 X_{i1} + b_2 X_{i2} + \varepsilon_i$$

As before, ε_i is the residual associated with the i-th observation, $\varepsilon_i = Y_i - \hat{Y}_i$. In this model, we now have three coefficients that need to be estimated: b_0 is the intercept, b_1 is the coefficient associated

with my sleep, and b_2 is the coefficient associated with my son's sleep. However, although the number of coefficients that need to be estimated has changed, the basic idea of how the estimation works is unchanged: our estimated coefficients $\hat{b}_0$, $\hat{b}_1$ and $\hat{b}_2$ are those that minimise the sum squared residuals.

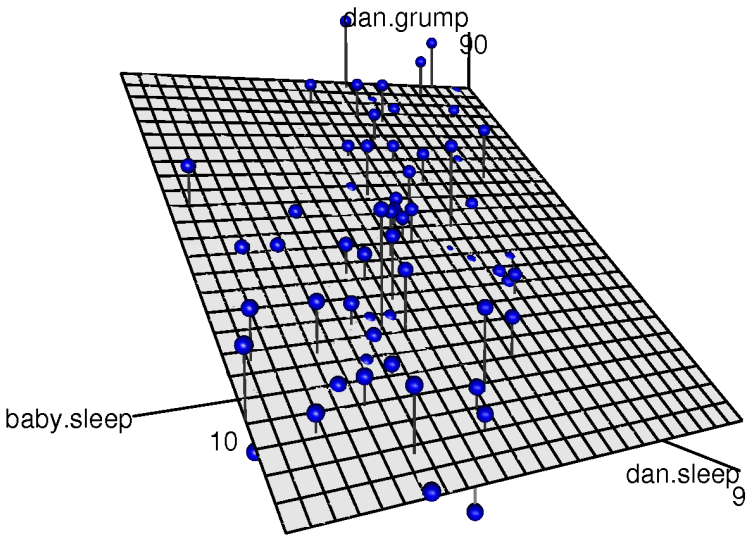


Figure1.12 A 3D visualisation of a multiple regression model. There are two predictors in the model, `dan.sleep` and `baby.sleep` and the outcome variable is `dan.grump`. Together, these three variables form a 3D space. Each observation (dot) is a point in this space. In much the same way that a simple linear regression model forms a line in 2D space, this multiple regression model forms a plane in 3D space. When we estimate the regression coefficients what we're trying to do is find a plane that is as close to all the blue dots as possible.

1.5.1 Doing it in JASP

Multiple regression in JASP is no different to simple regression. All we have to do is add additional variables to the 'Covariates' box in JASP. For example, if we want to use both `dan.sleep` and `baby.sleep` as predictors in our attempt to explain why I'm so grumpy, then move `baby.sleep` across into the 'Covariates' box alongside `dan.sleep`. By default, JASP assumes that the model should include an intercept. The coefficients we get this time are:

```
(Intercept)    dan.sleep   baby.sleep
    125.966       -8.950        0.011
```

The coefficient associated with `dan.sleep` is quite large, suggesting that every hour of sleep I lose makes me a lot grumpier. However, the coefficient for `baby.sleep` is very small, suggesting that it doesn't really matter how much sleep my son gets. What matters as far as my grumpiness goes is how much sleep *I* get. To get a sense of what this multiple regression model looks like, Figure **??** shows a 3D plot that plots all three variables, along with the regression model itself.

1.5.2 Formula for the general case

The equation that I gave above shows you what a multiple regression model looks like when you include two predictors. Not surprisingly, then, if you want more than two predictors all you have to do is add more X terms and more b coefficients. In other words, if you have K predictor variables in the model then the regression equation looks like this

$$Y_i = b_0 + \left(\sum_{k=1}^{K} b_k X_{ik} \right) + \varepsilon_i$$

1.6 Quantifying the fit of the regression model

So we now know how to estimate the coefficients of a linear regression model. The problem is, we don't yet know if this regression model is any good. For example, the regression model that we constructed in section **??** *claims* that every hour of sleep will improve my mood by quite a lot, but it might just be rubbish. Remember, the regression model only produces a prediction $\hat{Y}_i$ about what

my mood is like, but my actual mood is Y_i. If these two are very close, then the regression model has done a good job. If they are very different, then it has done a bad job.

1.6.1 The R^2 value

Once again, let's wrap a little bit of mathematics around this. Firstly, we've got the sum of the squared residuals

$$\text{SS}_{res} = \sum_i (Y_i - \hat{Y}_i)^2$$

which we would hope to be pretty small. Specifically, what we'd like is for it to be very small in comparison to the total variability in the outcome variable

$$\text{SS}_{tot} = \sum_i (Y_i - \bar{Y})^2$$

While we're here, let's calculate these values ourselves, not by hand though. I have constructed a JASP file called `parenthood_rsquared.jasp`, which you can open from the book's data folder. You'll notice that this data file has 5 variables; two of them are the original `dan.sleep` and `dan.grump` variables that we've already been using. The other three are *calculated* variables:

1. `Y.pred` is the predicted value of grumpiness using the regression equation. It is calculated using the formula '`125.97 + (-8.94 * dan.sleep)`'.

2. `resid` is a measure of the residual error $\epsilon_i = Y_i - \hat{Y}_i$, which represents the difference between our *predicted* value of grumpiness and our *actual* value of grumpiness. It is calculated using the formula '`dan.grump - Y.pred`'.

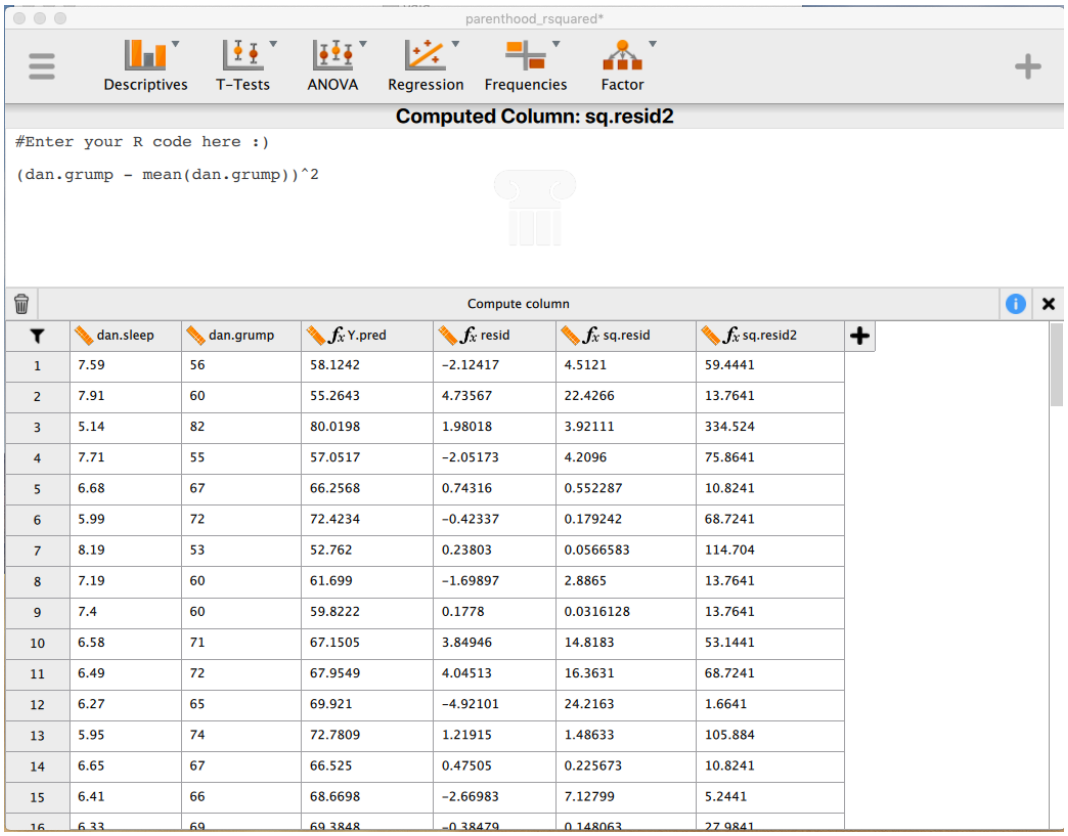
3. `sq.resid` is the square of the residual, and is calculated using the formula '`resid2`'.

Since SS_{res} is the sum of these squared residuals, we can use JASP to find the sum of the `sq.resid` column. Simply click 'Descriptives' - 'Descriptive Statistics' and move `sq.resid` to the 'Variables' box. You'll then need to select 'Sum' from the 'Statistics' options below. This should give you a value of '`1838.714`'.

Wonderful. A big number that doesn't mean very much. Still, let's forge boldly onwards anyway and calculate the total sum of squares as well. That's also pretty simple. Let's calculate SS_{tot} similarly. This time, you'll need to create a new computed column. Click the "+" symbol to start. For 'Name', let's type '`sq.resid2`' (you'll see why in a minute). Be sure to select the "R" button, then click 'Create Column'. For your R code, type the following (see Figure **??**):

```
(dan.grump - mean(dan.grump))^2
```

Then click 'Compute column'. This will produce a column of values that are themselves residuals, but they are residuals (errors) against a *really bad* predictive model; that is, the model that just predicts grumpiness using the *mean* of all grumpiness values. To find SS_{tot}, we need to compute the sum of `sq.resid2`, just like we did above.



	dan.sleep	dan.grump	Y.pred	resid	sq.resid	sq.resid2
1	7.59	56	58.1242	-2.12417	4.5121	59.4441
2	7.91	60	55.2643	4.73567	22.4266	13.7641
3	5.14	82	80.0198	1.98018	3.92111	334.524
4	7.71	55	57.0517	-2.05173	4.2096	75.8641
5	6.68	67	66.2568	0.74316	0.552287	10.8241
6	5.99	72	72.4234	-0.42337	0.179242	68.7241
7	8.19	53	52.762	0.23803	0.0566583	114.704
8	7.19	60	61.699	-1.69897	2.8865	13.7641
9	7.4	60	59.8222	0.1778	0.0316128	13.7641
10	6.58	71	67.1505	3.84946	14.8183	53.1441
11	6.49	72	67.9549	4.04513	16.3631	68.7241
12	6.27	65	69.921	-4.92101	24.2163	1.6641
13	5.95	74	72.7809	1.21915	1.48633	105.884
14	6.65	67	66.525	0.47505	0.225673	10.8241
15	6.41	66	68.6698	-2.66983	7.12799	5.2441
16	6.33	69	69.3848	-0.38479	0.148063	27.9841

Figure1.13 A JASP screenshot showing a residual computation in R.

This should give you a value of '`9998.590`'. Hmm. Well, it's a much bigger number than the last one, so this does suggest that our regression model was making good predictions (that is, it has greatly reduced the residual error compared to the model that uses the mean as a single predictor). But it's not very interpretable.

Perhaps we can fix this. What we'd like to do is to convert these two fairly meaningless numbers into one number. A nice, interpretable number, which for no particular reason we'll call R^2. What we would like is for the value of R^2 to be equal to 1 if the regression model makes no errors in predicting the data. In other words, if it turns out that the residual errors are zero. That is, if $\text{SS}_{res} = 0$ then we expect $R^2 = 1$. Similarly, if the model is completely useless, we would like R^2

to be equal to 0. What do I mean by "useless"? Tempting as it is to demand that the regression model move out of the house, cut its hair and get a real job, I'm probably going to have to pick a more practical definition. In this case, all I mean is that the residual sum of squares is no smaller than the total sum of squares, $\text{SS}_{res} = \text{SS}_{tot}$. Wait, why don't we do exactly that? The formula that provides us with our R^2 value is pretty simple to write down,

$$R^2 = 1 - \frac{\text{SS}_{res}}{\text{SS}_{tot}}$$

and equally simple to calculate by hand:

$$\begin{aligned} R^2 &= 1 - \frac{SS_{res}}{SS_{tot}} \\ &= 1 - \frac{1838.714}{9998.590} \\ &= 1 - 0.1839 \\ &= 0.8161. \end{aligned}$$

The R^2 value, sometimes called the **coefficient of determination**[*5] has a simple interpretation: it is the *proportion* of the variance in the outcome variable that can be accounted for by the predictor. So, in this case the fact that we have obtained $R^2 = .8161$ means that the predictor (`dan.sleep`) explains 81.61% of the variance in the outcome (`dan.grump`).

Naturally, you don't actually need to do all these computations by hand if you want to obtain the R^2 value for your regression model. It turns out that JASP gives you this by default! Take a look at Figure **??** again; notice that in the top table labeled 'Model Summary', the value of R^2 is already there!

1.6.2 The relationship between regression and correlation

At this point we can revisit my earlier claim that regression, in this very simple form that I've discussed so far, is basically the same thing as a correlation. Previously, we used the symbol r to denote a Pearson correlation. Might there be some relationship between the value of the correlation coefficient r and the R^2 value from linear regression? Of course there is: the squared correlation r^2 is identical to the R^2 value for a linear regression with only a single predictor. In other words, running a Pearson correlation is more or less equivalent to running a linear regression model that

[*5] And by "sometimes" I mean "almost never". In practice everyone just calls it "R-squared".

uses only one predictor variable.

1.6.3 The adjusted R^2 value

One final thing to point out before moving on. It's quite common for people to report a slightly different measure of model performance, known as "adjusted R^2". The motivation behind calculating the adjusted R^2 value is the observation that adding more predictors into the model will *always* cause the R^2 value to increase (or at least not decrease).

The adjusted R^2 value introduces a slight change to the calculation, as follows. For a regression model with K predictors, fit to a data set containing N observations, the adjusted R^2 is:

$$\text{adj. } R^2 = 1 - \left(\frac{\text{SS}_{res}}{\text{SS}_{tot}} \times \frac{N-1}{N-K-1} \right)$$

This adjustment is an attempt to take the degrees of freedom into account. The big advantage of the adjusted R^2 value is that when you add more predictors to the model, the adjusted R^2 value will only increase if the new variables improve the model performance more than you'd expect by chance. The big disadvantage is that the adjusted R^2 value *can't* be interpreted in the elegant way that R^2 can. R^2 has a simple interpretation as the proportion of variance in the outcome variable that is explained by the regression model. To my knowledge, no equivalent interpretation exists for adjusted R^2.

An obvious question then is whether you should report R^2 or adjusted R^2. This is probably a matter of personal preference. If you care more about interpretability, then R^2 is better. If you care more about correcting for bias, then adjusted R^2 is probably better. Speaking just for myself, I prefer R^2. My feeling is that it's more important to be able to interpret your measure of model performance. Besides, as we'll see in Section **??**, if you're worried that the improvement in R^2 that you get by adding a predictor is just due to chance and not because it's a better model, well we've got hypothesis tests for that.

1.7 Hypothesis tests for regression models

So far we've talked about what a regression model is, how the coefficients of a regression model are estimated, and how we quantify the performance of the model (the last of these, incidentally, is basically our measure of effect size). The next thing we need to talk about is hypothesis tests.

There are two different (but related) kinds of hypothesis tests that we need to talk about: those in which we test whether the regression model as a whole is performing significantly better than a null model, and those in which we test whether a particular regression coefficient is significantly different from zero.

1.7.1 Testing the model as a whole

Okay, suppose you've estimated your regression model. The first hypothesis test you might try is the null hypothesis that there is *no relationship* between the predictors and the outcome, and the alternative hypothesis that *the data are distributed in exactly the way that the regression model predicts*.

Formally, our "null model" corresponds to the fairly trivial "regression" model in which we include 0 predictors and only include the intercept term b_0:

$$H_0 : Y_i = b_0 + \varepsilon_i$$

If our regression model has K predictors, the "alternative model" is described using the usual formula for a multiple regression model:

$$H_1 : Y_i = b_0 + \left(\sum_{k=1}^{K} b_k X_{ik} \right) + \varepsilon_i$$

How can we test these two hypotheses against each other? The trick is to understand that it's possible to divide up the total variance SS_{tot} into the sum of the residual variance SS_{res} and the regression model variance SS_{mod}. I'll skip over the technicalities, since we'll get to that later when we look at ANOVA in Chapter **??**. But just note that

$$\text{SS}_{mod} = \text{SS}_{tot} - \text{SS}_{res}$$

And we can convert the sums of squares into mean squares by dividing by the degrees of freedom.

$$\text{MS}_{mod} = \frac{\text{SS}_{mod}}{df_{mod}}$$

$$\text{MS}_{res} = \frac{\text{SS}_{res}}{df_{res}}$$

So, how many degrees of freedom do we have? As you might expect the df associated with the model is closely tied to the number of predictors that we've included. In fact, it turns out that $df_{mod} = K$. For the residuals the total degrees of freedom is $df_{res} = N - K - 1$.

Now that we've got our mean square values we can calculate an F-statistic like this

$$F = \frac{\text{MS}_{mod}}{\text{MS}_{res}}$$

and the degrees of freedom associated with this are K and $N - K - 1$.

We'll see much more of the F statistic in Chapter **??**, but for now just know that we can interpret large F values as indicating that the null hypothesis is performing poorly in comparison to the alternative hypothesis. In a moment I'll show you how to do the test in JASP the easy way, but first let's have a look at the tests for the individual regression coefficients.

1.7.2 Tests for individual coefficients

The F-test that we've just introduced is useful for checking that the model as a whole is performing better than chance. If your regression model doesn't produce a significant result for the F-test then you probably don't have a very good regression model (or, quite possibly, you don't have very good data). However, while failing this test is a pretty strong indicator that the model has problems, *passing* the test (i.e., rejecting the null) doesn't imply that the model is good! Why is that, you might be wondering? The answer to that can be found by looking at the coefficients for the multiple regression model we have already looked at in section **??** above, where the coefficients we got were:

```
(Intercept)    dan.sleep   baby.sleep
    125.966       -8.950        0.011
```

I can't help but notice that the estimated regression coefficient for the `baby.sleep` variable is tiny (0.011), relative to the value that we get for `dan.sleep` (-8.950). Given that these two variables are absolutely on the same scale (they're both measured in "hours slept"), I find this illuminating. In fact, I'm beginning to suspect that it's really only the amount of sleep that *I* get that matters in order to predict my grumpiness.

We can re-use a hypothesis test that we discussed earlier, the t-test. The test that we're interested in has a null hypothesis that the true regression coefficient is zero ($b = 0$), which is to be tested against the alternative hypothesis that it isn't ($b \neq 0$). That is:

$$\begin{aligned} H_0 &: \quad b = 0 \\ H_1 &: \quad b \neq 0 \end{aligned}$$

How can we test this? Well, if the central limit theorem is kind to us we might be able to guess that the sampling distribution of $\hat{b}$, the estimated regression coefficient, is a normal distribution with mean centred on b. What that would mean is that if the null hypothesis were true, then the sampling distribution of $\hat{b}$ has mean zero and unknown standard deviation. Assuming that we can come up with a good estimate for the standard error of the regression coefficient, $\mathrm{SE}(\hat{b})$, then we're in luck. That's *exactly* the situation for which we introduced the one-sample t-test way back in Chapter **??**. So let's define a t-statistic like this

$$t = \frac{\hat{b}}{\mathrm{SE}(\hat{b})}$$

I'll skip over the reasons why, but our degrees of freedom in this case are $df = N-K-1$. Irritatingly, the estimate of the standard error of the regression coefficient, $\mathrm{SE}(\hat{b})$, is not as easy to calculate as the standard error of the mean that we used for the simpler t-tests in Chapter **??**. In fact, the formula is somewhat ugly, and not terribly helpful to look at.[*6] For our purposes it's sufficient to point out that the standard error of the estimated regression coefficient depends on both the predictor and outcome variables, and it is somewhat sensitive to violations of the homogeneity of variance assumption (discussed shortly).

In any case, this t-statistic can be interpreted in the same way as the t-statistics that we discussed in Chapter **??**. Assuming that you have a two-sided alternative (i.e., you don't really care if $b > 0$ or $b < 0$), then it's the extreme values of t (i.e., a lot less than zero or a lot greater than zero) that suggest that you should reject the null hypothesis.

1.7.3 Running the hypothesis tests in JASP

To compute all of the statistics that we have talked about so far, all you need to do is make sure the relevant options are checked in JASP and then run the regression. Fortunately, these options are usually selected by default. As you can see in Figure **??**, we get a whole bunch of useful output.

The 'Coefficients' at the bottom of the JASP analysis results shown in **??** provides the coefficients of the regression model. Each row in this table refers to one of the coefficients in the regression model. The first row is the intercept term, and the later ones look at each of the predictors. The columns give you all of the relevant information. The first column (labeled 'Unstandardized') is the actual estimate of b (e.g., 125.966 for the intercept, and -8.950 for the `dan.sleep` predictor).

[*6] For advanced readers only. The vector of residuals is $\boldsymbol{\varepsilon} = \boldsymbol{y} - \mathrm{X}\hat{\boldsymbol{b}}$. For K predictors plus the intercept, the estimated residual variance is $\hat{\sigma}^2 = \boldsymbol{\varepsilon}'\boldsymbol{\varepsilon}/(N-K-1)$. The estimated covariance matrix of the coefficients is $\hat{\sigma}^2(\mathrm{X}'\mathrm{X})^{-1}$, the main diagonal of which is $\mathrm{SE}(\hat{\boldsymbol{b}})$, our estimated standard errors.

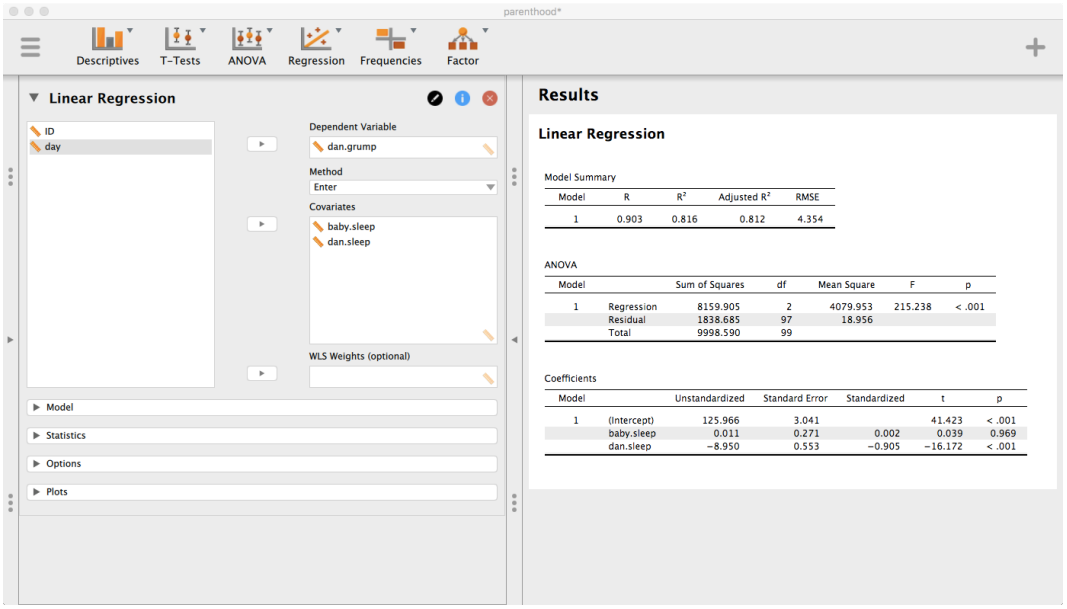


Figure1.14 A JASP screenshot showing a multiple linear regression analysis, including relevant hypothesis tests.

The second column is the standard error estimate $\hat{\sigma}_b$. The third column provides a 'Standardized' regression coefficient; more about this in Section **??**. The fourth column gives you the t-statistic, and it's worth noticing that in this table $t = \hat{b}/\mathrm{SE}(\hat{b})$ every time. Finally, the last column gives you the actual p-value for each of these tests.[*7]

The only thing that the coefficients table itself doesn't list is the degrees of freedom used in the t-test, which is always $N - K - 1$ and is listed in the table in the middle of the output, labelled 'ANOVA'. We can see from this table that the model performs significantly better than you'd expect by chance ($F(2, 97) = 215.238$, $p < .001$), which isn't all that surprising: the $R^2 = .816$ value indicate that the regression model accounts for 81.6% of the variability in the outcome measure. However, when we look back up at the t-tests for each of the individual coefficients, we can see that the `baby.sleep` variable seems to have no significant effect. All the work in this model is being done by the `dan.sleep` variable. Taken together, these results suggest that this regression model is actually the wrong model for the data. You'd probably be better off dropping the `baby.sleep` predictor entirely. In other words, the simple regression model that we started with is likely the

[*7] Note that, although JASP has done multiple tests here, it hasn't done any sort of correction for multiple comparisons. These are standard one-sample t-tests with a two-sided alternative. If you want to make corrections for multiple tests, you need to do that yourself.

better model.

1.8

Regarding regression coefficients

Before moving on to discuss the assumptions underlying linear regression and what you can do to check if they're being met, there's two more topics I want to briefly discuss, both of which relate to the regression coefficients. The first thing to talk about is calculating confidence intervals for the coefficients. After that, I'll discuss the somewhat murky question of how to determine which predictor is most important.

1.8.1 Confidence intervals for the coefficients

Like any population parameter, the regression coefficients b cannot be estimated with complete precision from a sample of data; that's part of why we need hypothesis tests. Given this, it's quite useful to be able to report confidence intervals that capture our uncertainty about the true value of b. This is especially useful when the research question focuses heavily on an attempt to find out *how* strongly variable X is related to variable Y, since in those situations the interest is primarily in the regression weight b.

Fortunately, confidence intervals for the regression weights can be constructed in the usual fashion

$$\text{CI}(b) = \hat{b} \pm \left(t_{crit} \times \text{SE}(\hat{b}) \right)$$

where $\text{SE}(\hat{b})$ is the standard error of the regression coefficient, and t_{crit} is the relevant critical value of the appropriate t distribution. For instance, if it's a 95% confidence interval that we want, then the critical value is the 97.5th quantile of a t distribution with $N - K - 1$ degrees of freedom. In other words, this is basically the same approach to calculating confidence intervals that we've used throughout.

In JASP we can display confidence intervals by selecting 'Confidence intervals' from the 'Statistics' menu in our regression model dialog. The default is 95% CI, but we could easily choose something different, say 99%, if that is what we decided on.

1.8.2 Calculating standardised regression coefficients

One more thing that you might want to do is to calculate "standardised" regression coefficients, often denoted β. The rationale behind standardised coefficients goes like this. In a lot of situations, your variables are on fundamentally different scales. Suppose, for example, my regression model aims to predict people's IQ scores using their educational attainment (number of years of education) and their income as predictors. Obviously, educational attainment and income are not on the same scales. The number of years of schooling might only vary by 10s of years, whereas income can vary by 10,000s of dollars (or more). The units of measurement have a big influence on the regression coefficients. The b coefficients only make sense when interpreted in light of the units, both of the predictor variables and the outcome variable. This makes it very difficult to compare the coefficients of different predictors. Yet there are situations where you really do want to make comparisons between different coefficients. Specifically, you might want some kind of standard measure of which predictors have the strongest relationship to the outcome. This is what **standardised coefficients** aim to do.

The basic idea is quite simple; the standardised coefficients are the coefficients that you would have obtained if you'd converted all the variables to z-scores before running the regression.[*8] The idea here is that, by converting all the predictors to z-scores, they all go into the regression on the same scale, thereby removing the problem of having variables on different scales. Regardless of what the original variables were, a β value of 1 means that an increase in the predictor of 1 standard deviation will produce a corresponding 1 standard deviation increase in the outcome variable. Therefore, if variable A has a larger absolute value of β than variable B, it is deemed to have a stronger relationship with the outcome. Or at least that's the idea. It's worth being a little cautious here, since this does rely very heavily on the assumption that "a 1 standard deviation change" is fundamentally the same kind of thing for all variables. It's not always obvious that this is true.

Leaving aside the interpretation issues, let's look at how it's calculated. What you could do is standardise all the variables yourself and then run a regression, but there's a much simpler way to do it. As it turns out, the β coefficient for a predictor X and outcome Y has a very simple formula, namely

$$\beta_X = b_X \times \frac{\sigma_X}{\sigma_Y}$$

[*8] Strictly, you standardise all the *regressors*. That is, every "thing" that has a regression coefficient associated with it in the model. For the regression models that I've talked about so far, each predictor variable maps onto exactly one regressor, and vice versa. However, that's not actually true in general and we'll see some examples of this in Chapter **??**. But, for now we don't need to care too much about this distinction.

where σ_X is the standard deviation of the predictor, and σ_Y is the standard deviation of the outcome variable Y. This makes matters a lot simpler.

To make things even simpler, JASP computes the β coefficients by default, as you can see the third column of the 'Coefficients' table in Figure **??**. This clearly shows that the `dan.sleep` variable has a much stronger effect than the `baby.sleep` variable. However, this is a perfect example of a situation where it would probably make sense to use the original coefficients b rather than the standardised coefficients β. After all, my sleep and the baby's sleep are *already* on the same scale: number of hours slept. Why complicate matters by converting these to z-scores?

1.9 Assumptions of regression

The linear regression model that I've been discussing relies on several assumptions. In Section **??** we'll talk a lot more about how to check that these assumptions are being met, but first let's have a look at each of them.

- *Normality*. Like many of the models in statistics, basic simple or multiple linear regression relies on an assumption of normality. Specifically, it assumes that the *residuals* are normally distributed. It's actually okay if the predictors X and the outcome Y are non-normal, so long as the residuals ϵ are normal. See Section **??**.
- *Linearity*. A pretty fundamental assumption of the linear regression model is that the relationship between X and Y actually is linear! Regardless of whether it's a simple regression or a multiple regression, we assume that the relationships involved are linear.
- *Homogeneity of variance*. Strictly speaking, the regression model assumes that each residual ϵ_i is generated from a normal distribution with mean 0, and (more importantly for the current purposes) with a standard deviation σ that is the same for every single residual. In practice, it's impossible to test the assumption that every residual is identically distributed. Instead, what we care about is that the standard deviation of the residual is the same for all values of $\hat{Y}$, and (if we're being especially paranoid) all values of every predictor X in the model.
- *Uncorrelated predictors*. The idea here is that, in a multiple regression model, you don't want your predictors to be too strongly correlated with each other. This isn't "technically" an assumption of the regression model, but in practice it's required. Predictors that are too strongly correlated with each other (referred to as "collinearity") can cause problems when evaluating the model.

- *Residuals are independent of each other*. This is really just a "catch all" assumption, to the effect that "there's nothing else funny going on in the residuals". If there is something weird (e.g., the residuals all depend heavily on some other unmeasured variable) going on, it might screw things up.
- *No "bad" outliers*. Again, not actually a technical assumption of the model (or rather, it's sort of implied by all the others), but there is an implicit assumption that your regression model isn't being too strongly influenced by one or two anomalous data points because this raises questions about the adequacy of the model and the trustworthiness of the data in some cases. See Section **??**.

1.10 Model checking

The main focus of this section is **regression diagnostics**, a term that refers to the art of checking that the assumptions of your regression model have been met, figuring out how to fix the model if the assumptions are violated, and generally to check that nothing "funny" is going on. I refer to this as the "art" of model checking with good reason. It's not easy, and while there are a lot of fairly standardised tools that you can use to diagnose and maybe even cure the problems that ail your model (if there are any, that is!), you really do need to exercise a certain amount of judgement when doing this. It's easy to get lost in all the details of checking this thing or that thing, and it's quite exhausting to try to remember what all the different things are. This has the very nasty side effect that a lot of people get frustrated when trying to learn *all* the tools, so instead they decide not to do *any* model checking. This is a bit of a worry!

In this section I describe several different things you can do to check that your regression model is doing what it's supposed to. It doesn't cover the full space of things you could do, but it's still much more detailed than what I see a lot of people doing in practice, and even I don't usually cover all of this in my intro stats class either. However, I do think it's important that you get a sense of what tools are at your disposal, so I'll try to introduce a bunch of them here. Finally, I should note that this section draws quite heavily from the **Fox2011** text, the book associated with the `car` package that is used to conduct regression analysis in R. The `car` package is notable for providing some excellent tools for regression diagnostics, and the book itself talks about them in an admirably clear fashion. I don't want to sound too gushy about it, but I do think that **Fox2011** is well worth reading, even if some of the advanced diagnostic techniques are only available in R and not JASP.

1.10.1 Three kinds of residuals

The majority of regression diagnostics revolve around looking at the residuals, and by now you've probably formed a sufficiently pessimistic theory of statistics to be able to guess that, precisely *because* of the fact that we care a lot about the residuals, there are several different kinds of residual that we might consider. In particular, the following three kinds of residuals are referred to in this section: "ordinary residuals", "standardised residuals", and "Studentised residuals". There is a fourth kind that you'll see referred to in some of the Figures, and that's the "Pearson residual". However, for the models that we're talking about in this chapter the Pearson residual is identical to the ordinary residual.

The first and simplest kind of residuals that we care about are **ordinary residuals**. These are the actual raw residuals that I've been talking about throughout this chapter so far. The ordinary residual is just the difference between the fitted value $\hat{Y}_i$ and the observed value Y_i. I've been using the notation ϵ_i to refer to the i-th ordinary residual, and darn it, I'm going to stick to it. With this in mind, we have the very simple equation

$$\epsilon_i = Y_i - \hat{Y}_i$$

This is of course what we saw earlier, and unless I specifically refer to some other kind of residual, this is the one I'm talking about. So there's nothing new here. I just wanted to repeat myself. One drawback to using ordinary residuals is that they're always on a different scale, depending on what the outcome variable is and how good the regression model is. That is, unless you've decided to run a regression model without an intercept term, the ordinary residuals will have mean 0 but the variance is different for every regression. In a lot of contexts, especially where you're only interested in the *pattern* of the residuals and not their actual values, it's convenient to estimate the **standardised residuals**, which are normalised in such a way as to have standard deviation 1.

The way we calculate these is to divide the ordinary residual by an estimate of the (population) standard deviation of these residuals. For technical reasons, mumble mumble, the formula for this is

$$\epsilon'_i = \frac{\epsilon_i}{\hat{\sigma}\sqrt{1-h_i}}$$

where $\hat{\sigma}$ in this context is the estimated population standard deviation of the ordinary residuals, and h_i is the "hat value" of the ith observation. I haven't explained hat values to you yet (but have no fear,[a] it's coming shortly), so this won't make a lot of sense. For now, it's enough to interpret the standardised residuals as if we'd converted the ordinary residuals to z-scores. In fact,

that is more or less the truth, it's just that we're being a bit fancier.

[a] Or have no hope, as the case may be.

The third kind of residuals are **Studentised residuals** (also called "jackknifed residuals") and they're even fancier than standardised residuals. Again, the idea is to take the ordinary residual and divide it by some quantity in order to estimate some standardised notion of the residual.

The formula for doing the calculations this time is subtly different

$$\varepsilon_i^* = \frac{\varepsilon_i}{\hat{\sigma}_{(-i)}\sqrt{1-h_i}}$$

Notice that our estimate of the standard deviation here is written $\hat{\sigma}_{(-i)}$. What this corresponds to is the estimate of the residual standard deviation that you *would have obtained* if you just deleted the ith observation from the data set. This sounds like the sort of thing that would be a nightmare to calculate, since it seems to be saying that you have to run N new regression models (even a modern computer might grumble a bit at that, especially if you've got a large data set). Fortunately, some terribly clever person has shown that this standard deviation estimate is actually given by the following equation:

$$\hat{\sigma}_{(-i)} = \hat{\sigma}\sqrt{\frac{N-K-1-\varepsilon_i'^2}{N-K-2}}$$

Isn't that a pip?

Before moving on, I should point out that you don't often need to obtain these residuals yourself, even though they are at the heart of almost all regression diagnostics. Most of the time the various options that provide the diagnostics, or assumption checks, will take care of these calculations for you. Even so, it's always nice to know how to actually get hold of these things yourself in case you ever need to do something non-standard.

1.10.2 Three kinds of anomalous data

One danger that you can run into with linear regression models is that your analysis might be disproportionately sensitive to a smallish number of "unusual" or "anomalous" observations. I discussed this idea previously in Section **??** in the context of discussing the outliers that get automatically identified by the boxplot option under 'Exploration' - 'Descriptives', but this time we need to be much more precise. In the context of linear regression, there are three conceptually distinct ways in

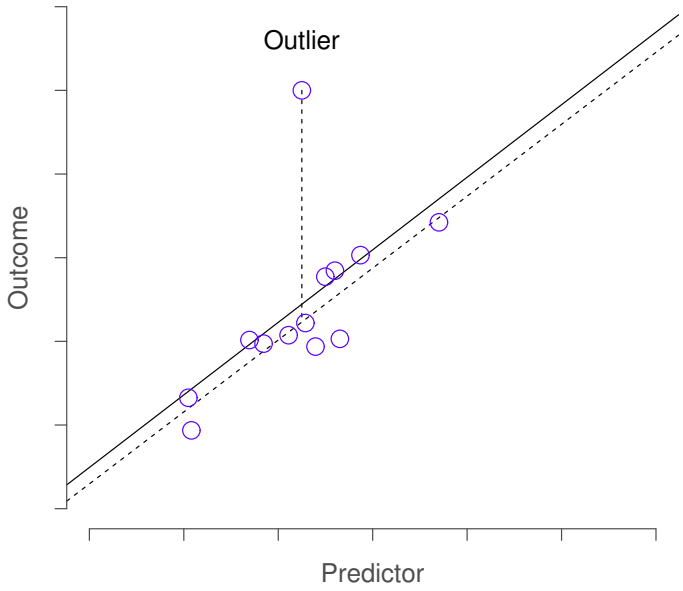


Figure1.15 An illustration of outliers. The dotted lines plot the regression line that would have been estimated without the anomalous observation included, and the corresponding residual (i.e., the Studentised residual). The solid line shows the regression line with the anomalous observation included. The outlier has an unusual value on the outcome (y axis location) but not the predictor (x axis location), and lies a long way from the regression line.

which an observation might be called "anomalous". All three are interesting, but they have rather different implications for your analysis.

The first kind of unusual observation is an **outlier**. The definition of an outlier (in this context) is an observation that is very different from what the regression model predicts. An example is shown in Figure **??**. In practice, we operationalise this concept by saying that an outlier is an observation that has a very large Studentised residual, ε_i^*. Outliers are interesting: a big outlier *might* correspond to junk data, e.g., the variables might have been recorded incorrectly in the data set, or some other defect may be detectable. Note that you shouldn't throw an observation away just because it's an outlier. But the fact that it's an outlier is often a cue to look more closely at that case and try to find out why it's so different.

The second way in which an observation can be unusual is if it has high **leverage**, which happens

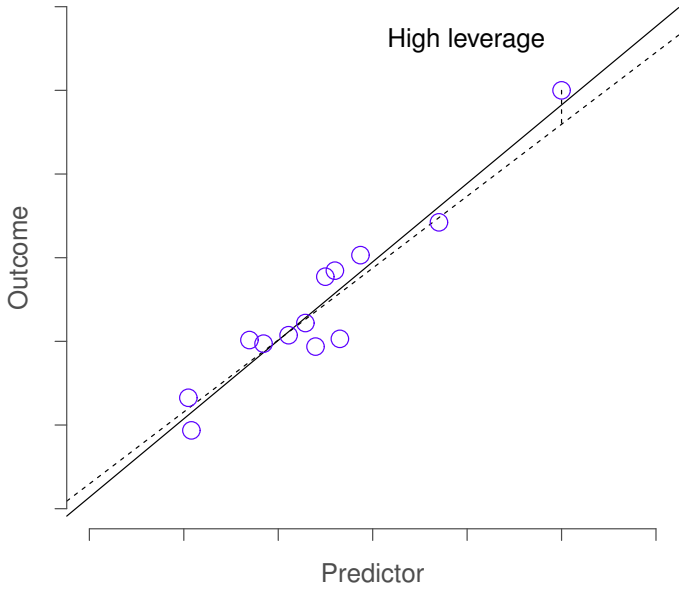


Figure1.16 An illustration of high leverage points. The anomalous observation in this case is unusual both in terms of the predictor (x axis) and the outcome (y axis), but this unusualness is highly consistent with the pattern of correlations that exists among the other observations. The observation falls very close to the regression line and does not distort it.

when the observation is very different from all the other observations. This doesn't necessarily have to correspond to a large residual. If the observation happens to be unusual on all variables in precisely the same way, it can actually lie very close to the regression line. An example of this is shown in Figure **??**. The leverage of an observation is operationalised in terms of its *hat value*, usually written h_i. The formula for the hat value is rather complicated[*9] but its interpretation is not: h_i is a measure of the extent to which the i-th observation is "in control" of where the regression line ends up going.

In general, if an observation lies far away from the other ones in terms of the predictor variables,

[*9]Again, for the linear algebra fanatics: the "hat matrix" is defined to be that matrix H that converts the vector of observed values $\boldsymbol{y}$ into a vector of fitted values $\hat{\boldsymbol{y}}$, such that $\hat{\boldsymbol{y}} = \mathrm{H}\boldsymbol{y}$. The name comes from the fact that this is the matrix that "puts a hat on $\boldsymbol{y}$". The hat *value* of the i-th observation is the i-th diagonal element of this matrix (so technically I should be writing it as h_{ii} rather than h_i). Oh, and in case you care, here's how it's calculated: $\mathrm{H} = \mathrm{X}(\mathrm{X}'\mathrm{X})^{-1}\mathrm{X}'$. Pretty, isn't it?

it will have a large hat value (as a rough guide, high leverage is when the hat value is more than 2-3 times the average; and note that the sum of the hat values is constrained to be equal to $K + 1$). High leverage points are also worth looking at in more detail, but they're much less likely to be a cause for concern unless they are also outliers.

This brings us to our third measure of unusualness, the **influence** of an observation. A high influence observation is an outlier that has high leverage. That is, it is an observation that is very different to all the other ones in some respect, and also lies a long way from the regression line. This is illustrated in Figure **??**. Notice the contrast to the previous two figures. Outliers don't move the regression line much and neither do high leverage points. But something that is both an outlier and has high leverage, well that has a big effect on the regression line. That's why we call these points high influence, and it's why they're the biggest worry. We operationalise influence in terms of a measure known as **Cook's distance**.

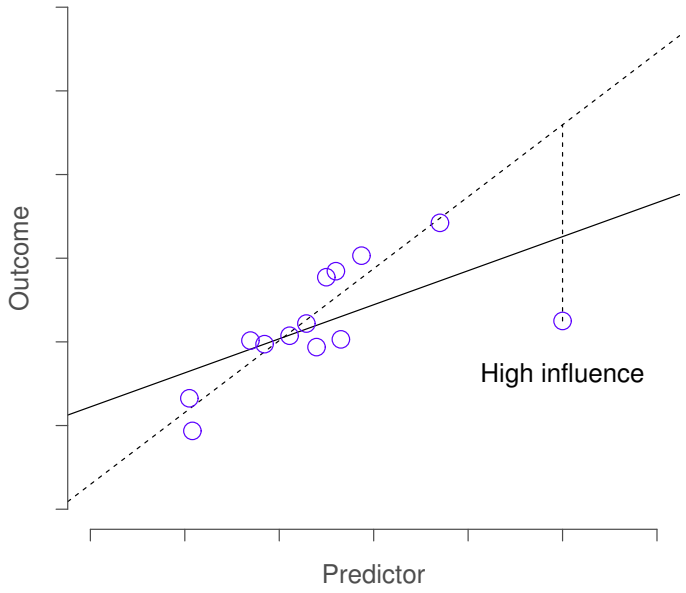


Figure1.17 An illustration of high influence points. In this case, the anomalous observation is highly unusual on the predictor variable (x axis), and falls a long way from the regression line. As a consequence, the regression line is highly distorted, even though (in this case) the anomalous observation is entirely typical in terms of the outcome variable (y axis).

$$D_i = \frac{\varepsilon_i^{*2}}{K+1} \times \frac{h_i}{1-h_i}$$

Notice that this is a multiplication of something that measures the outlier-ness of the observation (the bit on the left), and something that measures the leverage of the observation (the bit on the right).

In order to have a large Cook's distance an observation must be a fairly substantial outlier *and* have high leverage. As a rough guide, Cook's distance greater than 1 is often considered large (that's what I typically use as a quick and dirty rule).

In JASP, information about Cook's distance can be calculated by clicking on 'Casewise diagnostics' under the 'Statistics' menu. There are two ways to visualize these data. First, you can select 'All' to see Cook's distance for each case (i.e., row of data); see Figure **??**. Alternatively, you can opt to display *only* those cases for which Cook's distance is greater than some threshold; the default in JASP is 1. In either case, you can see that we have no data cases that are beyond this threshold.

Casewise Diagnostics

Case Number	Std. Residual	dan.grump	Predicted Value	Residual	Cook's Distance
1	−0.497	56.000	58.140	−2.140	0.002
2	1.104	60.000	55.292	4.708	0.017
3	0.464	82.000	80.045	1.955	0.005
4	−0.477	55.000	57.060	−2.060	0.001
5	0.168	67.000	66.281	0.719	0.000
6	−0.095	72.000	72.407	−0.407	0.000
7	0.053	53.000	52.773	0.227	0.000
8	−0.393	60.000	61.700	−1.700	0.001
9	0.047	60.000	59.797	0.203	0.000
10	0.890	71.000	67.148	3.852	0.003
11	0.959	72.000	68.001	3.999	0.027
12	−1.139	65.000	69.912	−4.912	0.008

Figure1.18 JASP output showing Cook's distance for each case/row of data

An obvious question to ask next is, if you do have large values of Cook's distance what should you do? As always, there's no hard and fast rule. Probably the first thing to do is to try running the regression with the outlier with the greatest Cook's distance[*10] excluded and see what happens to the model performance and to the regression coefficients. If they really are substantially different, it's time to start digging into your data set and your notes that you no doubt were scribbling as your ran your study. Try to figure out *why* the point is so different. If you start to become convinced that this one data point is badly distorting your results then you might consider excluding it, but that's less than ideal unless you have a solid explanation for why this particular case is qualitatively

[*10] although currently there isn't a very easy way to do this in JASP, so a more powerful regression program such as the `car` package in R would be better for this more advanced analysis

different from the others and therefore deserves to be handled separately.

1.10.3 Checking the normality of the residuals

Like many of the statistical tools we've discussed in this book, regression models rely on a normality assumption. In this case, we assume that the residuals are normally distributed. The first thing we can do is draw a QQ-plot in JASP via the 'Plots' - 'Q-Q plot standardized residuals' option. The output is shown in Figure **??**, showing the standardised residuals plotted as a function of their theoretical quantiles according to the regression model.

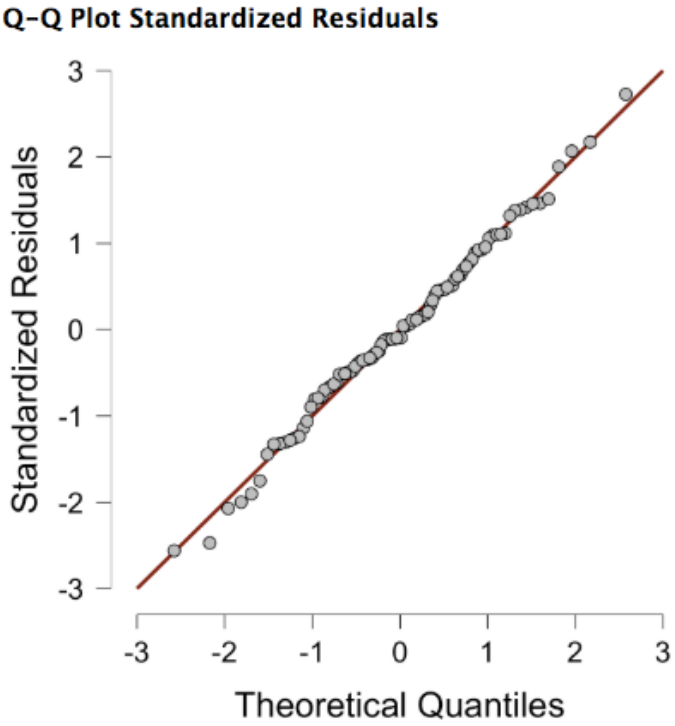


Figure1.19 Plot of the theoretical quantiles according to the model, against the quantiles of the standardised residuals, produced in JASP.

Another thing we should check is the relationship between the fitted values and the residuals themselves. We can get JASP to do this using the various 'Residuals Plots' choices, each of which provides a scatterplot for the predictor variables, the outcome variable, and the fitted values against residuals; see Figure **??**. In these plots we are looking for a fairly uniform distribution of 'dots', with no clear bunching or patterning of the 'dots'. Looking at these plots, there is nothing particularly

worrying as the dots are fairly evenly spread across the whole plot. There may be a little bit of non-uniformity in plot (b), but it is not a strong deviation and probably not worth worrying about.

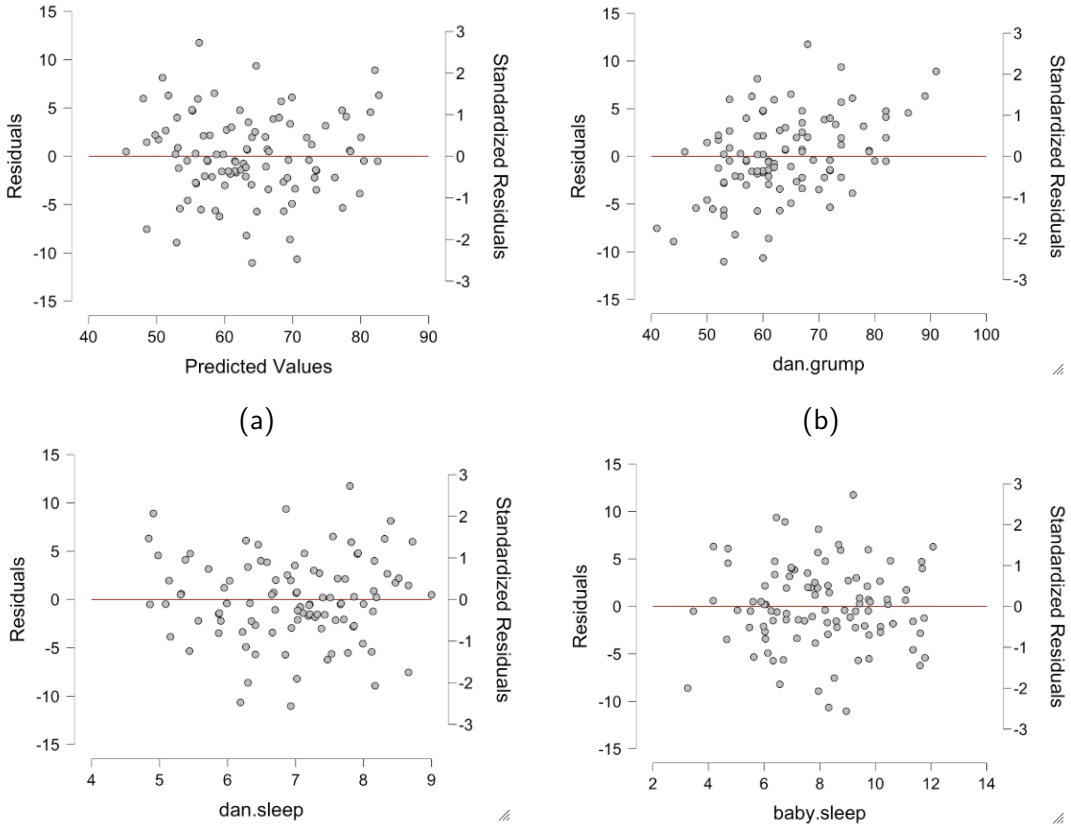


Figure1.20 Residuals plots produced in JASP

If we were worried, then in a lot of cases the solution to this problem (and many others) is to transform one or more of the variables. Transformations are beyond the scope of this text.

1.11 Model selection

One fairly major problem that remains is the problem of "model selection". That is, if we have a data set that contains several variables, which ones should we include as predictors, and which ones should we not include? In other words, we have a problem of **variable selection**. In general,

model selection is a complex business but it's made somewhat simpler if we restrict ourselves to the problem of choosing a subset of the variables that ought to be included in the model. Nevertheless, I'm not going to try covering even this reduced topic in a lot of detail. Instead, I'll talk about two broad principles that you need to think about, and then discuss one concrete tool to help you select a subset of variables to include in your model. First, the two principles:

- It's nice to have an actual substantive basis for your choices. That is, in a lot of situations you the researcher have good reasons to pick out a smallish number of possible regression models that are of theoretical interest. These models will have a sensible interpretation in the context of your field. Never discount the importance of this. Statistics serves the scientific process, not the other way around.
- To the extent that your choices rely on statistical inference, there is a trade off between simplicity and goodness of fit. As you add more predictors to the model you make it more complex. Each predictor adds a new free parameter (i.e., a new regression coefficient), and each new parameter increases the model's capacity to "absorb" random variations. So the goodness of fit (e.g., R^2) continues to rise, sometimes trivially or by chance, as you add more predictors no matter what. If you want your model to be able to generalise well to new observations you need to avoid throwing in too many variables.

This latter principle is often referred to as **Ockham's razor** and is often summarised in terms of the following pithy saying: *do not multiply entities beyond necessity*. In this context, it means don't chuck in a bunch of largely irrelevant predictors just to boost your R^2. Hmm. Yeah, the original was better.

In any case, what we need is an actual mathematical criterion that will implement the qualitative principle behind Ockham's razor in the context of selecting a regression model. As it turns out there are several possibilities. The one that I'll talk about is the **Akaike information criterion** (**Akaike1974**); currently, it is not part of JASP's standard output, but it is quite easy to compute with the model fit data that JASP does produce.

In the context of a linear regression model, the AIC for a model that has n observations and K predictor variables (not including the intercept) can be computed[a] as

$$AIC = n \ln(SS_{res}) + 2K$$

[a]Strictly speaking, this formula is not completely correct. Akaike's original definition was in terms of something called a *maximum likelihood estimate* for the model, and as such, there are some other terms that appear in the computation. However, many of them don't depend on the model, and given that the purpose of the AIC is to *compare models*, these terms will be present in all models and will mathematically "wash out". Thus, I am presenting a 'bare bones' version of the formula that is sufficient for our purposes.

Here's the basic principle behind using AIC for model comparison: the smaller the AIC value, the better the model performance. If we ignore the low level details it's fairly obvious what the AIC does. On the left we have a term that decreases as the model predictions get better; on the right we have a term that increases as the model complexity increases. The best model is the one that both fits the data well (small SS_{res}, left hand side) and uses as few predictors as possible (small K, right hand side). In short, this is a simple mathematical implementation of Ockham's razor.

Let's demonstrate how AIC can be used to compare the two regression models we have computed in this chapter. Consider first the regression model with only one predictor: `dan.sleep` (see Figure **??**). In this model, we have $n = 100$ observations, $K = 1$ predictor, and $SS_{res} = 1838.714$. Thus,

$$\begin{aligned} AIC_1 &= n \ln(SS_{res}) + 2K \\ &= 100 \ln(1838.714) + 2(1) \\ &= 753.68 \end{aligned}$$

Now consider the second model using two predictors: `dan.sleep` and `baby.sleep` (see Figure **??**). In this model, we have $n = 100$ observations, $K = 2$ predictors, and $SS_{res} = 1838.685$. This gives us

$$\begin{aligned} AIC_2 &= n \ln(SS_{res}) + 2K \\ &= 100 \ln(1838.685) + 2(2) \\ &= 755.68. \end{aligned}$$

Since $AIC_1 < AIC_2$, this tells us that Model 1 is the better fit, which confirms our intuitions. Adding `baby.sleep` doesn't add much to the model fit, but it increases model complexity. AIC balances these two requirements; the penalty for adding an additional parameter is not outweighed by the meager improvement in model fit.

1.12

Summary

- Want to know how strong the relationship is between two variables? Calculate a correlation (Section **??**).
- Drawing scatterplots (Section **??**).
- Basic ideas in linear regression and how regression models are estimated (Sections **??** and **??**).
- Multiple linear regression (Section **??**).
- Measuring the overall performance of a regression model using R^2 (Section **??**).
- Hypothesis tests for regression models (Section **??**)
- Calculating confidence intervals for regression coefficients and standardised coefficients (Section **??**).
- The assumptions of regression (Section **??**) and how to check them (Section **??**).
- Selecting a regression model (Section **??**).