8 Conditional Manatees

The manatee (*Trichechus manatus*) is a slow-moving, aquatic mammal that lives in warm, shallow water. Manatees have no natural predators, but they do share their waters with motor boats. And motor boats have propellers. While manatees are related to elephants and have very thick skins, propeller blades can and do kill them. A majority of adult manatees bear some kind of scar earned in a collision with a boat (FIGURE 8.1, top).[128]

The Armstrong Whitworth A.W.38 Whitley was a frontline Royal Air Force bomber. During the second World War, the A.W.38 carried bombs and pamphlets into German territory. Unlike the manatee, the A.W.38 has fierce natural enemies: artillery and interceptor fire. Many planes never returned from their missions. And those that survived had the scars to prove it (FIGURE 8.1, bottom).

How is a manatee like an A.W.38 bomber? In both cases—manatee propeller scars and bomber bullet holes—we'd like to do something to improve the odds, to help manatees and bombers survive. Most observers intuit that helping manatees or bombers means reducing the kind of damage we see on them. For manatees, this might mean requiring propeller guards (on the boats, not the manatees). For bombers, it'd mean adding armor to the parts of the plane that show the most damage.

But in both cases, the evidence misleads us. Propellers do not cause most of the injury and death caused to manatees. Rather autopsies confirm that collisions with blunt parts of the boat, like the keel, do far more damage. Similarly, up-armoring the damaged portions of returning bombers did little good. Instead, improving the A.W.38 bomber meant armoring the *undamaged* sections.[129] The evidence from surviving manatees and bombers is misleading, because it is *conditional* on survival. Manatees and bombers that perished look different. A manatee struck by a keel is less likely to live than another grazed by a propeller. So among the survivors, propeller scars are common. Similarly, bombers that returned home conspicuously lacked damage to the cockpit and engines. They got lucky. Bombers that never returned home were less so. To get the right answer, in either context, we have to realize that the kind of damage seen is conditional on survival.

CONDITIONING is one of the most important principles of statistical inference. Data, like the manatee scars and bomber damage, are conditional on how they get into our sample. Posterior distributions are conditional on the data. All model-based inference is conditional on the model. Every inference is conditional on something, whether we notice it or not.

And a large part of the power of statistical modeling comes from creating devices that allow probability to be conditional of aspects of each case. The linear models you've grown to love are just crude devices that allow each outcome y_i to be conditional on a set of predictors for each case i. Like the epicycles of the Ptolemaic and Kopernikan models (Chapters 4 and 7), linear models give us a way to describe conditionality.

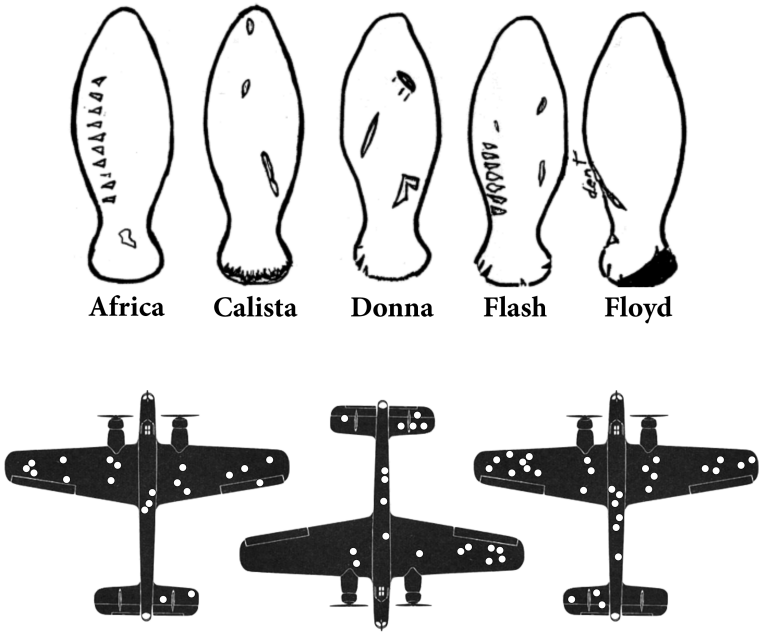


FIGURE 8.1. TOP: Dorsal scars for 5 adult Florida manatees. Rows of short scars, for example on the individuals Africa and Flash, are indicative of propeller laceration. BOTTOM: Three exemplars of damage on A.W.38 bombers returning from missions.

Simple linear models frequently fail to provide enough conditioning, however. Every model so far in this book has assumed that each predictor has an independent association with the mean of the outcome. What if we want to allow the association to be conditional? For example, in the primate milk data from the previous chapters, suppose the relationship between milk energy and brain size varies by taxonomic group (ape, monkey, prosimian). This is the same as suggesting that the influence of brain size on milk energy is conditional on taxonomic group. The linear models of previous chapters cannot address this question.

To model deeper conditionality—where the importance of one predictor depends upon another predictor—we need **INTERACTION** (also known as **MODERATION**). Interaction is a kind of conditioning, a way of allowing parameters (really their posterior distributions) to be conditional on further aspects of the data. The simplest kind of interaction, a linear interaction, is built by extending the linear modeling strategy to parameters within the linear model. So it is akin to placing epicycles on epicycles in the Ptolemaic and Kopernikan models. It is descriptive, but very powerful.

More generally, interactions are central to most statistical models beyond the cozy world of Gaussian outcomes and linear models of the mean. In generalized linear models (GLMs, Chapter 10 and onwards), even when one does not explicitly define variables as interacting, they will always interact to some degree. Multilevel models induce similar effects. Common sorts of multilevel models are essentially massive interaction models, in which estimates (intercepts and slopes) are conditional on clusters (person, genus, village, city, galaxy) in the data. Multilevel interaction effects are complex. They're not just allowing the impact of a

predictor variable to change depending upon some other variable, but they are also estimating aspects of the *distribution* of those changes. This may sound like genius, or madness, or both. Regardless, you can't have the power of multilevel modeling without it.

Models that allow for complex interactions are easy to fit to data. But they can be considerably harder to understand. And so I spend this chapter reviewing simple interaction effects: how to specify them, how to interpret them, and how to plot them. The chapter starts with a case of an interaction between a single categorical (indicator) variable and a single continuous variable. In this context, it is easy to appreciate the sort of hypothesis that an interaction allows for. Then the chapter moves on to more complex interactions between multiple continuous predictor variables. These are harder. In every section of this chapter, the model predictions are visualized, averaging over uncertainty in parameters.

Interactions are common, but they are not easy. My hope is that this chapter lays a solid foundation for interpreting generalized linear and multilevel models in later chapters.

Rethinking: Statistics all-star, Abraham Wald. The World War II bombers story is the work of Abraham Wald (1902–1950). Wald was born in what is now Romania, but immigrated to the United States after the Nazi invasion of Austria. Wald made many contributions over his short life. Perhaps most germane to the current material, Wald proved that for many types of rules for making statistical decisions, there will exist a Bayesian rule that is at least as good as any non-Bayesian one. Wald proved this, remarkably, beginning with non-Bayesian premises, and so anti-Bayesians could not ignore it. This work was summarized in Wald's 1950 book, published just before his death.[130] Wald died much too young, from a plane crash while touring India.

8.1. Building an interaction

Africa is special. The second largest continent, it is the most culturally and genetically diverse. Africa has about 3 billion fewer people than Asia, but it has just as many living languages. Africa is so genetically diverse that most of the genetic variation outside of Africa is just a subset of the variation within Africa. Africa is also geographically special, in a puzzling way: Bad geography tends to be related to bad economies outside of Africa, but African economies may actually benefit from bad geography.

To appreciate the puzzle, look at regressions of terrain ruggedness—a particular kind of bad geography—against economic performance (log GDP[131] per capita in the year 2000), both inside and outside of Africa (FIGURE 8.2). The variable `rugged` is a Terrain Ruggedness Index[132] that quantifies the topographic heterogeneity of a landscape. The outcome variable here is the logarithm of real gross domestic product per capita, from the year 2000, `rgdppc_2000`. We use the logarithm of it, because the logarithm of GDP is the *magnitude* of GDP. Since wealth generates wealth, it tends to be exponentially related to anything that increases it. This is like saying that the absolute distances in wealth grow increasingly large, as nations become wealthier. So when we work with logarithms instead, we can work on a more evenly spaced scale of magnitudes. Regardless, keep in mind that a log transform loses no information. It just changes what we the model assumes about the shape of the association between variables. In this case, raw GDP is not linearly associated with anything, because of its exponential pattern. But log GDP is linearly associated with lots of things.

What is going on in this figure? It makes sense that ruggedness is associated with poorer countries, in most of the world. Rugged terrain means transport is difficult. Which means market access is hampered. Which means reduced gross domestic product. So the reversed

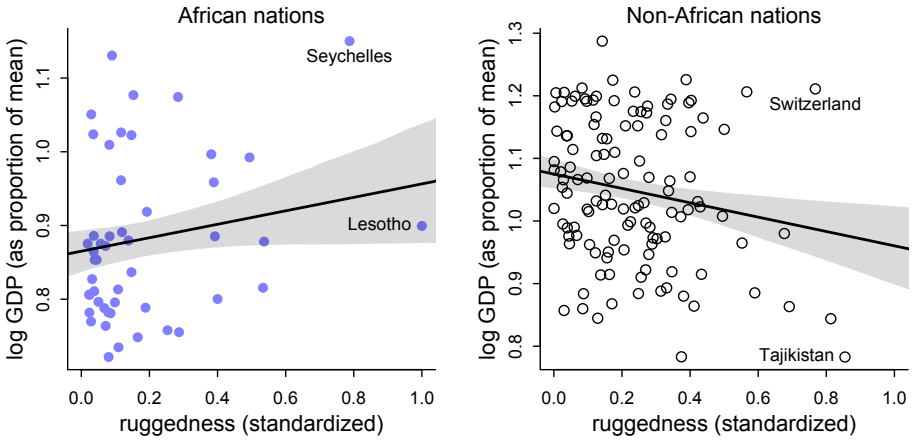
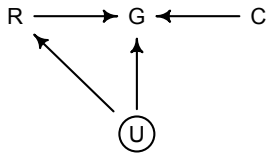


FIGURE 8.2. Separate linear regressions inside and outside of Africa, for log-GDP against terrain ruggedness. The slope is positive inside Africa, but negative outside. How can we recover this reversal of the slope, using the combined data?

relationship within Africa is puzzling. Why should difficult terrain be associated with higher GDP per capita?

If this relationship is at all causal, it may be because rugged regions of Africa were protected against the Atlantic and Indian Ocean slave trades. Slavers preferred to raid easily accessed settlements, with easy routes to the sea. Those regions that suffered under the slave trade understandably continue to suffer economically, long after the decline of slave-trading markets. However, an outcome like GDP has many influences, and is furthermore a strange measure of economic activity. And ruggedness is correlated with other geographic features, like coastlines, that also influence the economy. So it is hard to be sure what's going on here.

The causal hypothesis, in DAG form, might be (but see the Overthinking box at the end of this section):



where R is terrain ruggedness, G is GDP, C is continent, and U is some set of unobserved confounds (like distance to coast). Let's ignore U for now. You'll consider some confounds in the practice problems at the end. Focus instead on the implication that R and C both influence G. This could mean that they are independent influences or rather that they interact (one moderates the influence of the other). The DAG does not display an interaction. That's because DAGs do not specify how variables combine to influence other variables. The DAG above implies only that there is some function that uses R and C to generate G. In typical notation, $G = f(R, C)$.

So we need a statistical approach to judge different propositions for $f(R, C)$. How do we make a model that produces the conditionality in FIGURE 8.2? We could cheat by splitting

the data into two data frames, one for Africa and one for all the other continents. But it's not a good idea to split the data in this way. Here are four reasons.

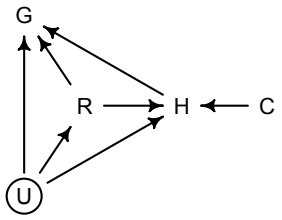
First, there are usually some parameters, such as σ, that the model says do not depend in any way upon continent. By splitting the data table, you are hurting the accuracy of the estimates for these parameters, because you are essentially making two less-accurate estimates instead of pooling all of the evidence into one estimate. In effect, you have accidentally assumed that variance differs between African and non-African nations. Now, there's nothing wrong with that sort of assumption. But you want to avoid accidental assumptions.

Second, in order to acquire probability statements about the variable you used to split the data, `cont_africa` in this case, you need to include it in the model. Otherwise, you have only the weakest sort of statistical argument. Isn't there uncertainty about the predictive value of distinguishing between African and non-African nations? Of course there is. Unless you analyze all of the data in a single model, you can't easily quantify that uncertainty. If you just let the posterior distribution do the work for you, you'll have a useful measure of that uncertainty.

Third, we may want to use information criteria or another method to compare models. In order to compare a model that treats all continents the same way to a model that allows different slopes in different continents, we need models that use all of the same data (as explained in Chapter 7). This means we can't split the data, but have to make the model split the data.

Fourth, once you begin using multilevel models (Chapter 13), you'll see that there are advantages to borrowing information across categories like "Africa" and "not Africa." This is especially true when sample sizes vary across categories, such that overfitting risk is higher within some categories. In other words, what we learn about ruggedness outside of Africa should have some effect on our estimate within Africa, and visa versa. Multilevel models (Chapter 13) borrow information in this way, in order to improve estimates in all categories. When we split the data, this borrowing is impossible.

Overthinking: Not so simple causation. The terrain ruggedness DAG in the preceding section is simple. But the truth isn't so simple. Continent isn't really the cause of interest. Rather there are hypothetical historical exposures to colonialism and the slave trade that have persistent influences on economic performance. Terrain features, like ruggedness, that causally reduced those historical factors may indirectly influence economy. Like this:



H stands for historical factors like exposure to slave trade. The total causal influence of R contains both a direct path $R \rightarrow G$ (this is presumably always negative) and an indirect path $R \rightarrow H \rightarrow G$. The second path is the one that covaries with continent C, because H is strongly associated with C. Note that the confounds U could influence any of these variables (except for C). If for example distance to coast is really what influenced H in the past, not terrain ruggedness, then the association of terrain ruggedness with GDP is non-causal. The data contain a large number of potential confounds that

you might consider.

8.1.1. Making a rugged model. Let's see how to recover the reversal of slope, within a single model. We'll begin by fitting a single model to all the data, ignoring continent. This will let us think through the model structure and priors before facing the devil of interaction. To get started, load the data and preform some pre-processing:

R code 8.1

```
library(rethinking)
data(rugged)
d <- rugged

# make log version of outcome
d$log_gdp <- log( d$rgdppc_2000 )

# extract countries with GDP data
dd <- d[ complete.cases(d$rgdppc_2000) , ]

# rescale variables
dd$log_gdp_std <- dd$log_gdp / mean(dd$log_gdp)
dd$rugged_std <- dd$rugged / max(dd$rugged)
```

Each row in these data is a country, and the various columns are economic, geographic, and historical features.[133] Raw magnitudes of GDP and terrain ruggedness aren't meaningful to humans. So I've scaled the variables to make the units easier to work with. The usual standardization is to subtract the mean and divide by the standard deviation. This makes a variable into z-scores. We don't want to do that here, because zero ruggedness is meaningful. So instead terrain ruggedness is divided by the maximum value observed. This means it ends up scaled from totally flat (zero) to the maximum in the sample at 1 (Lesotho, a very rugged and beautiful place). Similarly, log GDP is divided by the average value. So it is rescaled as a proportion of the international average. 1 means average, 0.8 means 80% of the average, and 1.1 means 10% more than average.

To build a Bayesian model for this relationship, we'll again use our geocentric skeleton:

$$\log(y_i) \sim \text{Normal}(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta(r_i - \bar{r})$$

where y_i is GDP for nation i, r_i is terrain ruggedness for nation i, and $\bar{r}$ is the average ruggedness in the whole sample. It's value is 0.215—most nations aren't that rugged. Remember that using $\bar{r}$ just makes it easier to assign a prior to the intercept α.

The hard thinking here comes when we specify priors. If you are like me, you don't have much scientific information about plausible associations between log GDP and terrain ruggedness. But even when we don't know much about the context, the measurements themselves constrain the priors in useful ways. The scaled outcome and predictor will make this easier. Consider first the intercept, α, defined as the log GDP when ruggedness is at the sample mean. So it must be close to 1, because we scaled the outcome so that the mean is 1. Let's start with a guess at:

$$\alpha \sim \text{Normal}(1, 1)$$

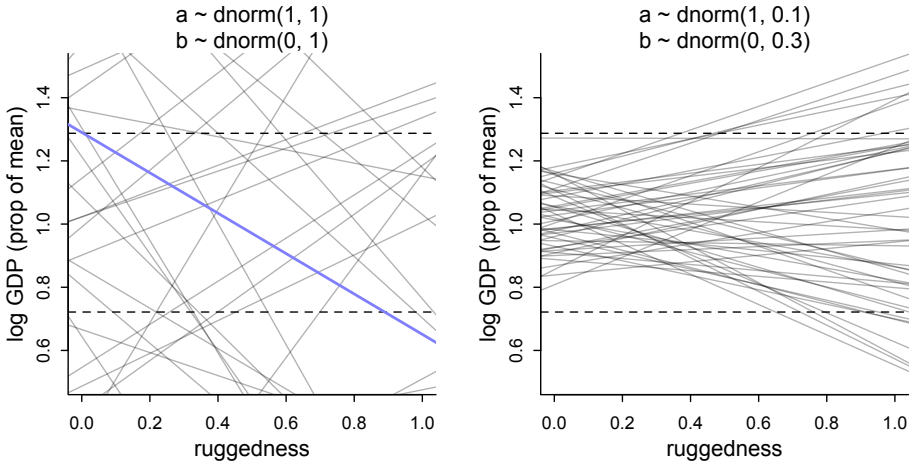


FIGURE 8.3. Simulating in search of reasonable priors for the terrain ruggedness example. The dashed horizontal lines indicate the minimum and maximum observed GDP values. Left: The first guess with very vague priors. Right: The improved model with much more plausible priors.

Now for β, the slope. If we center it on zero, that indicates no bias for positive or negative, which makes sense. But what about the standard deviation? Let's start with a guess at 1:

$$\beta \sim \text{Normal}(0, 1)$$

We'll evaluate this guess by simulating prior predictive distributions. The last thing we need is a prior for σ. Let's assign something very broad, $\sigma \sim \text{Exponential}(1)$. In the problems at the end of the chapter, I'll ask you to confront this prior as well. But we'll ignore it for the rest of this example.

All together, we have our first candidate model for fitting a line to the terrain ruggedness data:

R code 8.2

```
m8.1 <- quap(
    alist(
        log_gdp_std ~ dnorm( mu , sigma ) ,
        mu <- a + b*( rugged_std - 0.215 ) ,
        a ~ dnorm( 1 , 1 ) ,
        b ~ dnorm( 0 , 1 ) ,
        sigma ~ dexp( 1 )
    ) , data=dd )
```

We're not going to look at the posterior predictions yet, but rather at the prior predictions. Let's extract the prior and plot the implied lines. We'll do this using `link`, as in earlier chapters.

R code 8.3

```
set.seed(7)
prior <- extract.prior( m8.1 )
```

```
# set up the plot dimensions
plot( NULL , xlim=c(0,1) , ylim=c(0.5,1.5) ,
    xlab="ruggedness" , ylab="log GDP" )
abline( h=min(dd$log_gdp_std) , lty=2 )
abline( h=max(dd$log_gdp_std) , lty=2 )

# draw 50 lines from the prior
rugged_seq <- seq( from=-0.1 , to=1.1 , length.out=30 )
mu <- link( m8.1 , post=prior , data=data.frame(rugged_std=rugged_seq) )
for ( i in 1:50 ) lines( rugged_seq , mu[i,] , col=col.alpha("black",0.3) )
```

The result is displayed on the left side of FIGURE 8.3. The horizontal dashed lines show the maximum and minimum observed log GDP values. The regression lines trend both positive and negative, as they should, but many of these lines are in impossible territory. Considering only the measurement scales, the lines have to pass closer to the point where ruggedness is average (0.215 on the horizontal axis) and proportional log GDP is 1. Instead there are lots of lines that expect average GDP outside observed ranges. So we need a tighter standard deviation on the α prior. Something like $\alpha \sim \text{Normal}(0, 0.1)$ will put most of the plausibility within the observed GDP values. Remember: 95% of the Gaussian mass is within 2 standard deviations. So a $\text{Normal}(0, 0.1)$ prior assigns 95% of the plausibility between 0.8 and 1.2. That is still very vague, but at least it isn't ridiculous.

At the same time, the slopes are too variable. It is not plausible that terrain ruggedness explains most of the observed variation in log GDP. An implausibly strong association would be, for example, a line that goes from minimum ruggedness and extreme GDP on one end to maximum ruggedness and the opposite extreme of GDP on the other end. I've highlighted such a line in blue. The slope of such a line must be about $1.3 - 0.7 = 0.6$, the difference between the maximum and minimum observed proportional log GDP. But very many lines in the prior have much more extreme slopes than this. Under the $\beta \sim \text{Normal}(0, 1)$ prior, more than half of all slopes will have absolute value greater than 0.6.

R code 8.4

```
sum( abs(prior$b) > 0.6 ) / length(prior$b)
```

```
[1] 0.545
```

Let's try instead $\beta \sim \text{Normal}(0, 0.3)$. This prior makes a slope of 0.6 two standard deviations out. That is still a bit too plausible for reality, but it's a lot better than before.

With these two changes, now the model is:

R code 8.5

```
m8.1 <- quap(
    alist(
        log_gdp_std ~ dnorm( mu , sigma ) ,
        mu <- a + b*( rugged_std - 0.215 ) ,
        a ~ dnorm( 1 , 0.1 ) ,
        b ~ dnorm( 0 , 0.3 ) ,
        sigma ~ dexp(1)
    ) , data=dd )
```

You can extract the prior and plot the implied lines using the same code as before. The result is shown on the right side of FIGURE 8.3. Some of these slopes are still implausibly strong. But in the main, this is a much better set of priors.

Let's look at the posterior distribution now:

R code 8.6

```
precis( m8.1 )
```

```
      mean   sd  5.5% 94.5%
a     1.00 0.01  0.98  1.02
b     0.00 0.05 -0.09  0.09
sigma 0.14 0.01  0.12  0.15
```

Really no overall association between terrain ruggedness and log GDP. Next we'll see how to split apart the continents.

Rethinking: Practicing for when it matters. The exercise in FIGURE 8.3 is really not necessary in this example, because there is enough data, and the model is simple enough, that even awful priors get washed out. You could even use completely flat priors (don't!), and it would all be fine. But we practice doing things right not because it always matters. Rather, we practice doing things right so that we are ready when it matters.

8.1.2. Adding an indicator variable isn't enough. The first thing to realize is that just including an indicator variable for African nations, `cont_africa` here, won't reveal the reversed slope. It's worth fitting this model to prove it to yourself, though. I'm going to walk through this as a simple model comparison exercise, just so you begin to get some applied examples of concepts you've accumulated from earlier chapters. Note that model comparison here is not about selecting a model. Scientific considerations already select the relevant model. Instead it is about measuring the impact of model differences while accounting for overfitting risk.

To build a model that allows nations inside and outside Africa to have different intercepts, we need to modify the model for μ_i so that the mean is conditional on continent. The conventional way to do this would be to just add another term to the linear model:

$$\mu_i = \alpha + \beta(r_i - \bar{r}) + \gamma A_i$$

where A_i is `cont_africa`, a 0/1 indicator variable. But let's not follow this convention. In fact, this convention is often a bad idea. It took me years to figure this out, and I'm trying to save you from the horrors I've seen. The problem here, and in general, is that we need a prior for γ. Okay, we can do priors. But what that prior will necessarily do is tell the model that μ_i for a nation in Africa is more uncertain, before seeing the data, than μ_i outside Africa. And that makes no sense. This is the same issue we confronted back in Chapter 4, when I introduced categorical variables.

There is a simple solution: Nations in Africa will get one intercept and those outside Africa another. This is what μ_i looks like now:

$$\mu_i = \alpha_{\text{CID}[i]} + \beta(r_i - \bar{r})$$

where CID is an index variable, continent ID. It takes the value 1 for African nations and 2 for all other nations. This means there are two parameters, α_1 and α_2, one for each unique index value. The notation $\text{CID}[i]$ just means the value of CID on row i. I use the bracket notation

with index variables, because it is easier to read than adding a second level of subscript, α_{CID_i}. We can build this index ourselves:

R code 8.7
```
# make variable to index Africa (1) or not (2)
dd$cid <- ifelse( dd$cont_africa==1 , 1 , 2 )
```

Using this approach, instead of the conventional approach of adding another term with the 0/1 indicator variable, doesn't force us to say that the mean for Africa is inherently less certain than the mean for all other continents. We can just reuse the same prior as before. After all, whatever Africa's average log GDP, it is surely within plus-or-minus 0.2 of 1. But keep in mind that this is structurally the same model you'd get in the conventional approach. It is just much easier this way to assign sensible priors.

To define the model in `quap`, we need to add brackets in the linear model and the prior:

R code 8.8
```
m8.2 <- quap(
    alist(
        log_gdp_std ~ dnorm( mu , sigma ) ,
        mu <- a[cid] + b*( rugged_std - 0.215 ) ,
        a[cid] ~ dnorm( 1 , 0.1 ) ,
        b ~ dnorm( 0 , 0.3 ) ,
        sigma ~ dexp( 1 )
    ) , data=dd )
```

Now to compare these models, using WAIC:

R code 8.9
```
compare( m8.1 , m8.2 )
```

```
       WAIC pWAIC dWAIC weight   SE   dSE
m8.2 -252.3   4.3   0.0      1 15.3    NA
m8.1 -188.7   2.7  63.5      0 13.3 15.15
```

`m8.2` gets all the model weight. And while the standard error of the difference in WAIC is 15, the difference itself is 64. So the continent variable seems to be picking up some important association in the sample. The `precis` output gives a good hint. Note that we need to use `depth=2` to display the vector parameter `a`. With only two parameters in `a`, it wouldn't be bad to display it by default. But often a vector like this has hundreds of values, and you don't want to see each one in a table.

R code 8.10
```
precis( m8.2 , depth=2 )
```

```
       mean   sd  5.5% 94.5%
a[1]   0.88 0.02  0.85  0.91
a[2]   1.05 0.01  1.03  1.07
b     -0.05 0.05 -0.12  0.03
sigma  0.11 0.01  0.10  0.12
```

The parameter `a[1]` is the intercept for African nations. It seems reliably lower than `a[2]`. The posterior contrast between the two intercepts is:

R code 8.11

```
post <- extract.samples(m8.2)
diff_a1_a2 <- post$a[,1] - post$a[,2]
PI( diff_a1_a2 )
```

```
        5%        94%
-0.1990056 -0.1378378
```

The difference is reliably below zero.

Let's plot the posterior predictions for `m8.2`, so you can see how, despite it's predictive superiority to `m8.1`, it still doesn't manage different slopes inside and outside of Africa. To sample from the posterior and compute the predicted means and intervals for both African and non-African nations:

R code 8.12

```
rugged.seq <- seq( from=-0.1 , to=1.1 , length.out=30 )

# compute mu over samples, fixing cid=2
mu.NotAfrica <- link( m8.2 ,
    data=data.frame( cid=2 , rugged_std=rugged.seq ) )

# compute mu over samples, fixing cid=1
mu.Africa <- link( m8.2 ,
    data=data.frame( cid=1 , rugged_std=rugged.seq ) )

# summarize to means and intervals
mu.NotAfrica_mu <- apply( mu.NotAfrica , 2 , mean )
mu.NotAfrica_ci <- apply( mu.NotAfrica , 2 , PI , prob=0.97 )
mu.Africa_mu <- apply( mu.Africa , 2 , mean )
mu.Africa_ci <- apply( mu.Africa , 2 , PI , prob=0.97 )
```

I show these posterior predictions (retrodictions) in FIGURE 8.4. African nations are shown in blue, while nations outside Africa are shown in gray. What you've ended up with here is a rather weak negative relationship between economic development and ruggedness. The African nations do have lower overall economic development, and so the blue regression line is below, but parallel to, the black line. All including a dummy variable for African nations has done is allow the model to predict a lower mean for African nations. It can't do anything to the slope of the line. The fact that WAIC tells you that the model with the dummy variable is hugely better only indicates that African nations on average do have lower GDP. It's still a bad model.

Rethinking: Why 97%? In the code block just above, and therefore also in FIGURE 8.4, I used 97% intervals of the expected mean. This is a rather non-standard percentile interval. So why use 97%? In this book, I use non-standard percents to constantly remind the reader that conventions like 95% and 5% are arbitrary. Furthermore, boundaries are meaningless. There is continuous change in probability as we move away from the expected value. So one side of the boundary is almost equally probable as the other side. Also, 97 is a prime number. That doesn't mean it is a better choice than any other number here, but it's no less silly than using a multiple of 5, just because we have five digits on each hand. Resist the tyranny of the Tetrapoda.

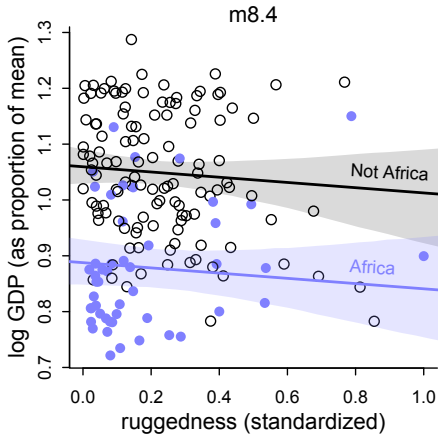


FIGURE 8.4. Including an indicator for African nations has no effect on the slope. African nations are shown in blue. Non-African nations are shown in black. Regression means for each subset of nations are shown in corresponding colors, along with 97% intervals shown by shading.

8.1.3. Adding an interaction does work. How can you recover the change in slope you saw at the start of this section? You need a proper interaction effect. This just means we also make the slope conditional on continent. The definition of μ_i in the model you just plotted, in math form, is:

$$\mu_i = \alpha_{\text{CID}[i]} + \beta(r_i - \bar{r})$$

And now we'll double-down on our indexing to make the slope conditional as well:

$$\mu_i = \alpha_{\text{CID}[i]} + \beta_{\text{CID}[i]}(r_i - \bar{r})$$

And again, there is a conventional approach to specifying an interaction that uses an indicator variable and a new interaction parameter. It would look like this:

$$\mu_i = \alpha_{\text{CID}[i]} + (\beta + \gamma A_i)(r_i - \bar{r})$$

where A_i is a 0/1 indicator for African nations. This is equivalent to our index approach, but it is much harder to state sensible priors. Any prior we put on γ makes the slope inside Africa more uncertain than the slope outside Africa. And again that makes no sense. But in the indexing approach, we can easily assign the same prior to the slope, no matter which continent.

To approximate the posterior of this new model, you can just use `quap` as before. Here's the code that includes an interaction between ruggedness and being in Africa:

R code 8.13

```
m8.3 <- quap(
    alist(
        log_gdp_std ~ dnorm( mu , sigma ) ,
        mu <- a[cid] + b[cid]*( rugged_std - 0.215 ) ,
        a[cid] ~ dnorm( 1 , 0.1 ) ,
        b[cid] ~ dnorm( 0 , 0.3 ) ,
        sigma ~ dexp( 1 )
    ) , data=dd )
```

Let's inspect the marginal posterior distributions:

R code 8.14
```
precis( m8.5 , depth=2 )
```

```
       mean   sd  5.5% 94.5%
a[1]   0.89 0.02  0.86  0.91
a[2]   1.05 0.01  1.03  1.07
b[1]   0.13 0.07  0.01  0.25
b[2]  -0.14 0.05 -0.23 -0.06
sigma  0.11 0.01  0.10  0.12
```

The slope is essentially reversed inside Africa, 0.13 instead of -0.14.

How much does allowing the slope to vary improve expected prediction? Let's use PSIS to compare this new model to the previous two. You could use WAIC here as well. It'll give almost identical results. But it won't give us a sweet Pareto k warning.

R code 8.15
```
compare( m8.1 , m8.2 , m8.3 , func=PSIS )
```

```
Some Pareto k values are high (>0.5).
       PSIS pPSIS dPSIS weight    SE   dSE
m8.3 -258.7   5.4   0.0   0.97 15.40    NA
m8.2 -251.5   4.6   7.2   0.03 15.47  6.84
m8.1 -188.6   2.8  70.2   0.00 13.33 15.56
```

Model family `m8.3` has more than 95% of the model weight. That's very strong support for including the interaction effect, if prediction is our goal. But the modicum of weight given to `m8.2` suggests that the posterior means for the slopes in `m8.3` are a little overfit. And the standard error of the difference in PSIS between the top two models is almost the same as the difference itself. If you plot PSIS Pareto k values for `m8.3`, you'll notice some influential countries.

R code 8.16
```
plot( PSIS( m8.3 , pointwise=TRUE )$k )
```

You'll explore this in the practice problems at the end of the chapter. This is possibly a good context for robust regression, like the Student-t regression we did in Chapter 7.

Remember that these comparisons are not reliable guides to causal inference. They just tell us how important features are for prediction. Real causal effects may not be important for overall prediction in any given sample. Prediction and inference are just different questions. Still, overfitting always happens. So anticipating and measuring it matters for inference as well.

8.1.4. Plotting the interaction. Plotting this model doesn't really require any new tricks. The goal is to make two plots. In the first, we'll display nations in Africa and overlay the posterior mean regression line and the 97% interval of that line. In the second, we'll display nations outside of Africa instead.

R code 8.17
```
# plot Africa - cid=1
d.A1 <- dd[ dd$cid==1 , ]
plot( d.A1$rugged_std , d.A1$log_gdp_std , pch=16 , col=rangi2 ,
    xlab="ruggedness (standardized)" , ylab="log GDP (as proportion of mean)" ,
    xlim=c(0,1) )
```

```
mu <- link( m8.3 , data=data.frame( cid=1 , rugged_std=rugged_seq ) )
mu_mean <- apply( mu , 2 , mean )
mu_ci <- apply( mu , 2 , PI , prob=0.97 )
lines( rugged_seq , mu_mean , lwd=2 )
shade( mu_ci , rugged_seq , col=col.alpha(rangi2,0.3) )
mtext("African nations")

# plot non-Africa - cid=2
d.A0 <- dd[ dd$cid==2 , ]
plot( d.A0$rugged_std , d.A0$log_gdp_std , pch=1 , col="black" ,
    xlab="ruggedness (standardized)" , ylab="log GDP (as proportion of mean)" ,
    xlim=c(0,1) )
mu <- link( m8.3 , data=data.frame( cid=2 , rugged_std=rugged_seq ) )
mu_mean <- apply( mu , 2 , mean )
mu_ci <- apply( mu , 2 , PI , prob=0.97 )
lines( rugged_seq , mu_mean , lwd=2 )
shade( mu_ci , rugged_seq )
mtext("Non-African nations")
```

And the result is shown in FIGURE 8.5. Finally, the slope reverses direction inside and outside of Africa. And because we achieved this inside a single model, we could statistically evaluate the value of this reversal.

8.2. Symmetry of interactions

Buridan's ass is a toy philosophical problem in which an ass who always moves towards the closest pile of food will starve to death when he finds himself equidistant between two identical piles. The basic problem is one of symmetry: How can the ass decide between two identical options? Like many toy problems, you can't take this one too seriously. Of course the ass will not starve. But thinking about how the symmetry is broken can be productive.

Interactions are like Buridan's ass. Like the two piles of identical food, a simple interaction model contains two symmetrical interpretations. Absent some other information, outside the model, there's no logical basis for preferring one over the other. Consider for

Rethinking: All Greek to me. We use these Greek symbols α and β because it is conventional. They don't have special meanings. If you prefer some other Greek symbol like ω—why should α get all the attention?—feel free to use that instead. It is conventional to use Greek letters for unobserved variables (parameters) and Roman letters for observed variables (data). That convention does have some value, because it helps others read your models. But breaking the convention is not an error, and sometimes it is better to use a familiar Roman symbol than an unfamiliar Greek one like ξ or ζ. If your readers cannot say the symbol's name, it could make understanding the model harder.

A core problem with the convention of using Greek for unobserved and Roman for observed variables is that in many models the same variable can be both observed and unobserved. This happens, for example, when data are missing for some cases. It also happens in "occupancy" detection models, where specific values of the outcome (usually zero) cannot be trusted. We will deal with these issues explicitly in Chapter 15.

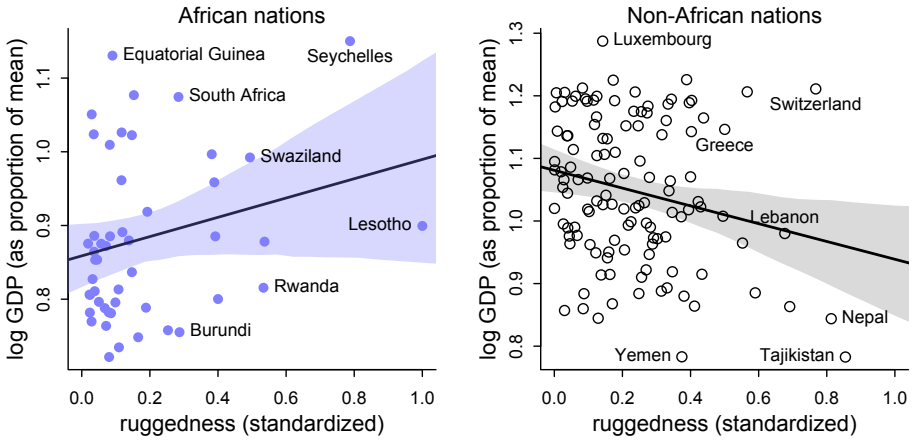


FIGURE 8.5. Posterior predictions for the terrain ruggedness model, including the interaction between Africa and ruggedness. Shaded regions are 97% posterior intervals of the mean.

example the GDP and terrain ruggedness problem. The interaction there has two equally valid phrasings.

(1) How much does the association between ruggedness and log GDP depend upon whether the nation is in Africa?
(2) How much does the association of Africa with log GDP depend upon ruggedness?

While these two possibilities sound different to most humans, your golem thinks they are identical.

In this section, we'll examine this fact, first mathematically. Then we'll plot the ruggedness and GDP example again, but with the reverse phrasing—the association between Africa and GDP depends upon ruggedness.

Consider yet again the model for μ_i:

$$\mu_i = \alpha_{\text{CID}[i]} + \beta_{\text{CID}[i]}(r_i - \bar{r})$$

The interpretation previously has been that the slope is conditional on continent. But it's also fine to say that the intercept is conditional on ruggedness. It's easier to see this if we write the above expression another way:

$$\mu_i = \underbrace{(2 - \text{CID}_i)(\alpha_1 + \beta_1(r_i - \bar{r}))}_{\text{CID}[i]=1} + \underbrace{(\text{CID}_i - 1)(\alpha_2 + \beta_2(r_i - \bar{r}))}_{\text{CID}[i]=2}$$

This looks weird, but it's the same model. When $\text{CID}_i = 1$, only the first term, the Africa parameters, remains. The second term vanishes to zero. When instead $\text{CID}_i = 2$, the first term vanishes to zero and only the second term remains. Now if we imagine switching a nation to Africa, in order to know what this does for the prediction, we have to know the ruggedness (unless we are exactly at the average ruggedness, $\bar{r}$).

It'll be helpful to plot the reverse interpretation: *The association of being in Africa with log GDP depends upon terrain ruggedness*. What we'll do is compute the difference between

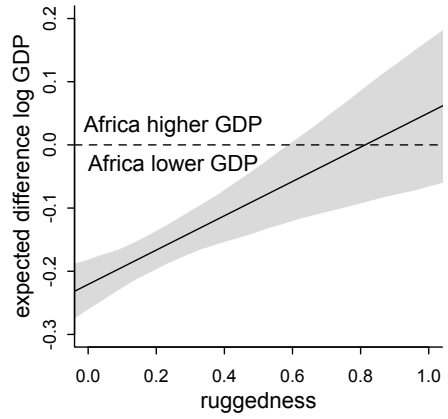


FIGURE 8.6. The other side of the interaction between ruggedness and continent. The vertical axis is the difference in expected proportional log GDP for a nation in Africa and one outside Africa. At low ruggedness, we expect "moving" a nation to Africa to hurt its economy. But at high ruggedness, the opposite is true. The association between continent and economy depends upon ruggedness, just as much as the association between ruggedness and economy depends upon continent.

a nation in Africa and outside Africa, holding its ruggedness constant. To do this, you can just run `link` twice and then subtract the second result from the first:

R code 8.18

```
rugged_seq <- seq(from=-0.2,to=1.2,length.out=30)
muA <- link( m8.3 , data=data.frame(cid=1,rugged_std=rugged_seq) )
muN <- link( m8.3 , data=data.frame(cid=2,rugged_std=rugged_seq) )
delta <- muA - muN
```

Then you can summarize and plot the difference in expected log GDP contained in `delta`.

The result is shown in FIGURE 8.6. This plot is *counter-factual*. There is no raw data here. Instead we are seeing through the model's eyes and imagining comparisons between identical nations inside and outside Africa, as if we could independently manipulate continent and also terrain ruggedness. Below the horizontal dashed line, African nations have lower expected GDP. This is the case for most terrain ruggedness values. But at the highest ruggedness values, a nation is possibly better off inside Africa than outside it. Really it is hard to find any reliable difference inside and outside Africa, at high ruggedness values. It is only in smooth nations that being in Africa is a liability for the economy.

This perspective on the GDP and terrain ruggedness is completely consistent with the previous perspective. It's simultaneously true in these data (and with this model) that (1) the influence of ruggedness depends upon continent and (2) the influence of continent depends upon ruggedness. Indeed, something is gained by looking at the data in this symmetrical perspective. Just inspecting the first view of the interaction, back on page 255, it's not obvious that African nations are on average nearly always worse off. It's just at very high values of `rugged` that nations inside and outside of Africa have the same expected log GDP. This second way of plotting the interaction makes this clearer.

Simple interactions are symmetric, just like the choice facing Buridan's ass. Within the model, there's no basis to prefer one interpretation over the other, because in fact they are the same interpretation. But when we reason causally about models, our minds tend to prefer one interpretation over the other, because it's usually easier to imagine manipulating one of the predictor variables instead of the other. In this case, it's hard to imagine manipulating which continent a nation is on. But it's easy to imagine manipulating terrain ruggedness, by flattening hills or blasting tunnels through mountains.[134] If in fact the explanation for

Africa's unusually positive relationship with terrain ruggedness is due to historical causes, not contemporary terrain, then tunnels might improve economies in the present. At the same time, continent is not really a cause of economic activity. Rather there are historical and political factors associated with continents, and we use the continent variable as a proxy for those factors. It is manipulation of those other factors that would matter.

8.3. Continuous interactions

I want to convince the reader that interaction effects are difficult to interpret. They are nearly impossible to interpret, using only posterior means and standard deviations. Once interactions exist, multiple parameters are always in play at the same time. It is hard enough with the simple, categorical interactions from the terrain ruggedness example. Once we start modeling interactions among more than one continuous variables, it gets much harder. It's one thing to make a slope conditional upon a *category*. In such a context, the model reduces to estimating a different slope for each category. But it's quite a lot harder to understand that a slope varies in a continuous fashion with a continuous variable. Interpretation is much harder in this case, even though the mathematics of the model are essentially the same as in the categorical case.

In pursuit of clarifying the construction and interpretation of CONTINUOUS INTERACTIONS among two or more continuous predictor variables, in this section I develop a simple regression example and show you a way to plot the two-way interaction between two continuous variables. The method I present for plotting this interaction is a *triptych* plot, a panel of three complementary figures that comprise a whole picture of the regression results. There's nothing magic about having three figures—in other cases you might want more or less. Instead, the utility lies in making multiple figures that allow one to see how the interaction alters a slope, across changes in a chosen variable.

8.3.1. A winter flower. The data in this example are sizes of blooms from beds of tulips grown in greenhouses, under different soil and light conditions.[135] Load the data with:

R code 8.19

```
library(rethinking)
data(tulips)
d <- tulips
str(d)
```

```
'data.frame': 27 obs. of  4 variables:
 $ bed   : Factor w/ 3 levels "a","b","c": 1 1 1 1 1 1 1 1 1 2 ...
 $ water : int  1 1 1 2 2 2 3 3 3 1 ...
 $ shade : int  1 2 3 1 2 3 1 2 3 1 ...
 $ blooms: num  0 0 111 183.5 59.2 ...
```

The `blooms` column will be our outcome—what we wish to predict. The `water` and `shade` columns will be our predictor variables. `water` indicates one of three ordered levels of soil moisture, from low (1) to high (3). `shade` indicates one of three ordered levels of light exposure, from high (1) to low (3). The last column, `bed`, indicates a cluster of plants from the same section of the greenhouse.

Since both light and water help plants grow and produce blooms, it stands to reason that the independent effect of each will be to produce bigger blooms. But we'll also be interested in the interaction between these two variables. In the absence of light, for example, it's hard

to see how water will help a plant—photosynthesis depends upon both light and water. Likewise, in the absence of water, sunlight does a plant little good. One way to model such an interdependency is to use an interaction effect. In the absence of a good mechanistic model of the interaction, one that uses a theory about the plant's physiology to hypothesize the functional relationship between light and water, then a simple linear two-way interaction is a good start. But ultimately it's not close to the best that we could do.

8.3.2. The models. I'm going to focus on just two models: (1) the model with both `water` and `shade` but no interaction and (2) the model that also contains the interaction of `water` with `shade`. You could also inspect models that contain only one of these variables, `water` or `shade`, and I encourage the reader to try that at the end and make sure you understand the full ensemble of models.

The causal scenario is simply that water (W) and shade (S) both influence blooms (B): $W \rightarrow B \leftarrow S$. As before, this DAG doesn't tell us the function through which W and S jointly influence B, $B = f(W, S)$. In principle, every unique combination of W and S could have a different mean S. The convention is to do something much simpler. We'll start simple.

The first model, containing no interaction at all (only "main effects"), begins this way:

$$B_i \sim \text{Normal}(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta_W(W_i - \bar{W}) + \beta_S(S_i - \bar{S})$$

where B_i is the value of `blooms` on row i, W_i is the value of `water`, and S_i is the value of `shade`. The symbols $\bar{W}$ and $\bar{S}$ are the means of water and shade, respectively. All together, this is just a linear regression with two predictors, each centered by subtracting its mean.

To make estimation easier, let's center W and S and scale B by its maximum:

R code 8.20

```
d$blooms_std <- d$blooms / max(d$blooms)
d$water_cent <- d$water - mean(d$water)
d$shade_cent <- d$shade - mean(d$shade)
```

Now `blooms_std` ranges from 0 to 1, and both `water_cent` and `shade_cent` range from -1 to 1. I've scaled `blooms` by its maximum observed value, for three reasons. First, the large values on the raw scale will make optimization difficult. Second, it will be easier to assign a reasonable prior this way. Third, we don't want to standardize `blooms`, because zero is a meaningful boundary we want to preserve.

As always in rescaling variables, the goals are to create focal points that you might have prior information about, prior to seeing the actual data values. That way we can assign priors that are not obviously crazy, and in thinking about those priors, we might realize that the model makes no sense. But this is only possible if we think about the relationship between measurements and parameters, and the exercise of rescaling and assigning sensible priors helps us along that path. Even when there are enough data that choice of priors is not crucial, this thought exercise is useful.

There are three parameters (aside from σ) in this model, so we need three priors. As a first, vague guess:

$$\alpha \sim \text{Normal}(0.5, 1)$$
$$\beta_W \sim \text{Normal}(0, 1)$$
$$\beta_S \sim \text{Normal}(0, 1)$$

Centering the prior for α at 0.5 implies that, when both water and shade are at their mean values, the model expects blooms to be halfway to the observed maximum. The two slopes are centered on zero, implying no prior information about direction. This is obviously less information than we have—basic botany informs us that water should have a positive slope and shade a negative slope. But these priors allow us to see which trend the sample shows, while still bounding the slopes to reasonable values. In the problems at the end of the chapter, I'll ask you to use your botany instead.

The prior bounds on the parameters come from the prior standard deviations, all set to 1 here. These are surely too broad. The intercept α must be greater than zero and less than one, for example. But this prior assigns most of the probability outside that range:

R code 8.21

```
a <- rnorm( 1e4 , 0.5 , 1 )
sum( a < 0 | a > 1 ) / length( a )
```

```
[1] 0.6126
```

If it's 0.5 units from the mean to zero, then a standard deviation of 0.25 should put only 5% of the mass outside the valid internal. Let's see:

R code 8.22

```
a <- rnorm( 1e4 , 0.5 , 0.25 )
sum( a < 0 | a > 1 ) / length( a )
```

```
[1] 0.0486
```

Much better.

What about those slopes? What would a very strong effect of water and shade look like? How big could those slopes be in theory? The range of both water and shade is 2—from -1 to 1 is 2 units. To take us from the theoretical minimum of zero blooms on one end to the observed maximum of 1—a range of 1 unit—on the other would require a slope of 0.5 from either variable—$0.5 \times 2 = 1$. So if we assign a standard deviation of 0.25 to each, then 95% of the prior slopes are from -0.5 to 0.5, so either variable could in principle account for the entire range, but it would be unlikely. Remember, the goals here are to assign weakly informative priors to discourage overfitting—impossibly large effects should be assigned low prior probability—and also to force ourselves to think about what the model means.

All together now, in code form:

R code 8.23

```
m8.4 <- quap(
    alist(
        blooms_std ~ dnorm( mu , sigma ) ,
        mu <- a + bw*water_cent + bs*shade_cent ,
        a ~ dnorm( 0.5 , 0.25 ) ,
        bw ~ dnorm( 0 , 0.25 ) ,
        bs ~ dnorm( 0 , 0.25 ) ,
        sigma ~ dexp( 1 )
    ) , data=d )
```

It's a good idea at this point to simulate lines from the prior. But before doing that, let's define the interaction model as well. Then we can talk about how to plot predictions from interactions and see both prior and posterior predictions together.

To build in an interaction between water and shade, we need to construct μ so that the impact of changing either water or shade depends upon the value of the other variable. For example, if water is low, then decreasing the shade (increase light) can't help as much as when water is high. So we want the slope of water, β_W, to be conditional on shade. Likewise for shade being conditional on water (remember Buridan's interaction, 254). How can we do this?

In the previous example, terrain ruggedness, we made a slope conditional on the value of a category. When there are, in principle, an infinite number of categories, then it's harder. In this case, the "categories" of shade and water are, in principle, infinite and ordered. We only observed three levels of water, but the model should be able to make a prediction with a water level intermediate between any two of the observed ones. With continuous interactions, the problem isn't so much the infinite part but rather the ordered part. Even if we only cared about the three observed values, we'd still need to preserve the ordering, which is bigger than which. So what to do?

The conventional answer is to reapply the original geocentrism that justifies a linear regression. When we have two variable, an outcome and a predictor, and we wish to model the mean of the outcome such that it is conditional on the value of a continuous predictor x, we can use a linear model: $\mu_i = \alpha + \beta x_i$. Now in order to make the slope β conditional on yet another variable, we can just recursively apply the same trick.

For brevity, let W_i and S_i be the centered variables. Then if we define the slope β_W with its own linear model γ_W:

$$\begin{aligned} \mu_i &= \alpha + \gamma_{W,i} W_i + \beta_S S_i \\ \gamma_{W,i} &= \beta_W + \beta_{WS} S_i \end{aligned}$$

Now $\gamma_{W,i}$ is the slope defining how quickly blooms change with water level. The parameter β_W is the rate of change, when shade is at its mean value. And β_{WS} is the rate change in $\gamma_{W,i}$ as shade changes—the slope for shade on the slope of water. Remember, it's turtles all the way down. Note the i in $\gamma_{W,i}$—it depends upon the row i, because it has S_i in it.

We also want to allow the association with shade, β_S, to depend upon water. Luckily, because of the symmetry of simple interactions, we get this for free. There is just no way to specify a simple, linear interaction in which you can say the effect of some variable x depends upon z but the effect of z does not depend upon x. I explain this in more detail in the Overthinking box at the end of this section. The impact of this is that it is conventional to substitute $\gamma_{W,i}$ into the equation for μ_i and just state:

$$\mu_i = \alpha + \underbrace{(\beta_W + \beta_{WS} S_i)}_{\gamma_{W,i}} W_i + \beta_S S_i = \alpha + \beta_W W_i + \beta_S S_i + \beta_{WS} S_i W_i$$

And that's the conventional form of a continuous interaction, with the extra term on the far right end holding the product of the two variables.

Let's put this to work on the tulips. The interaction model is:

$$\begin{aligned} B_i &\sim \text{Normal}(\mu_i, \sigma) \\ \mu_i &= \alpha + \beta_W W_i + \beta_S S_i + \beta_{WS} W_i S_i \end{aligned}$$

The last thing we need is a prior for this new interaction parameter, β_{WS}. This is hard, because these epicycle parameters don't have clear natural meaning. Still, implied predictions help. Suppose the strongest plausible interaction is one in which high enough shade makes water

have zero effect. That implies:

$$\gamma_{W,i} = \beta_w + \beta_{WS} S_i = 0$$

If we set $S_i = 1$ (the maximum in the sample), then this means the interaction needs to be the same magnitude as the main effect, but reversed: $\beta_{WS} = -\beta_W$. That is largest conceivable interaction. So if we set the prior for β_{WS} to have the same standard deviation as β_W, maybe that isn't ridiculous. All together now, in code form:

R code 8.24

```
m8.5 <- quap(
    alist(
        blooms_std ~ dnorm( mu , sigma ) ,
        mu <- a + bw*water_cent + bs*shade_cent + bws*water_cent*shade_cent ,
        a ~ dnorm( 0.5 , 0.25 ) ,
        bw ~ dnorm( 0 , 0.25 ) ,
        bs ~ dnorm( 0 , 0.25 ) ,
        bws ~ dnorm( 0 , 0.25 ) ,
        sigma ~ dexp( 1 )
    ) , data=d )
```

And that's the structure of a simple, continuous interaction. Next, let's figure out how to plot these creatures.

Overthinking: How is interaction formed? As in the main text, if you substitute $\gamma_{W,i}$ into μ_i above and expand:

$$\mu_i = \alpha + (\beta_W + \beta_{WS} S_i) W_i + \beta_S S_i = \alpha + \beta_W W_i + \beta_S S_i + \beta_{WS} S_i W_i$$

Now it's possible to refactor this to construct a $\gamma_{S,i}$ that makes the association of shade with blooms depend upon water:

$$\begin{aligned} \mu_i &= \alpha + \beta_W W_i + \gamma_{S,i} S_i \\ \gamma_{S,i} &= \beta_S + \beta_{SW} W_i \end{aligned}$$

So both interpretations are simultaneously true. You could even put both γ definitions into μ at the same time:

$$\begin{aligned} \mu_i &= \alpha + \gamma_{W,i} W_i + \gamma_{S,i} S_i \\ \gamma_{W,i} &= \beta_W + \beta_{WS} S_i \\ \gamma_{S,i} &= \beta_S + \beta_{SW} W_i \end{aligned}$$

Note that I defined two different interaction parameters: β_{WS} and β_{SW}. Now let's substitute the γ definitions into μ and start factoring:

$$\begin{aligned} \mu_i &= \alpha + (\beta_W + \beta_{WS} S_i) W_i + (\beta_S + \beta_{SW} W_i) S_i \\ &= \alpha + \beta_W W_i + \beta_S S_i + (\beta_{WS} + \beta_{SW}) W_i S_i \end{aligned}$$

The only thing we can identify in such a model is the sum $\beta_{WS} + \beta_{SW}$, so really the sum is a single parameter (dimension in the posterior). It's the same interaction model all over again. We just cannot tell the difference between water depending upon shade and shade depending upon water.

8.3.3. Plotting posterior predictions. Golems (models) have awesome powers of reason, but terrible people skills. The golem provides a posterior distribution of plausibility for combinations of parameter values. But for us humans to understand its implications, we need to

decode the posterior into something else. Centered predictors or not, plotting posterior predictions always tells you what the golem is thinking, on the scale of the outcome. That's why we've emphasized plotting so much. But in previous chapters, there were no interactions. As a result, when plotting model predictions as a function of any one predictor, you could hold the other predictors constant at any value you liked. So the choice of which values to set the un-viewed predictor variables to hardly mattered.

Now that'll be different. Once there are interactions in a model, the effect of changing a predictor depends upon the values of the other predictors. Maybe the simplest way to go about plotting such interdependency is to make a frame of multiple bivariate plots. In each plot, you choose different values for the un-viewed variables. Then by comparing the plots to one another, you can see how big of a difference the changes make.

That's what we did for the terrain ruggedness example. But there we needed only two plots, one for Africa and one for everyplace else. Now we'll need more. Here's how you might accomplish this visualization, for the tulip data. I'm going to make three plots in a single panel. Such a panel of three plots that are meant to be viewed together is a **TRIPTYCH**, and triptych plots are very handy for understanding the impact of interactions. Here's the strategy. We want each plot to show the bivariate relationship between water and blooms, as predicted by the model. Each plot will plot predictions for a different value of shade. For this example, it is easy to pick which three values of shade to use, because there are only three values: -1, 0, and 1. But more generally, you might use a representative low value, the median, and a representative high value.

Here's the code to draw posterior predictions for `m8.4`, the non-interaction model. This will loop over three values for shade, compute posterior predictions, then draw 20 lines from the posterior.

R code 8.25

```
par(mfrow=c(1,3)) # 3 plots in 1 row
for ( s in -1:1 ) {
    idx <- which( d$shade_cent==s )
    plot( d$water_cent[idx] , d$blooms_std[idx] , xlim=c(-1,1) , ylim=c(0,1) ,
        xlab="water" , ylab="blooms" , pch=16 , col=rangi2 )
    mu <- link( m8.4 , data=data.frame( shade_cent=s , water_cent=-1:1 ) )
    for ( i in 1:20 ) lines( -1:1 , mu[i,] , col=col.alpha("black",0.3) )
}
```

The result is shown in FIGURE 8.7, along with the same type of plot for the interaction model, `m8.5`. Notice that the top model believes that water helps—there is a positive slope in each plot—and that shade hurts—the lines sink lower moving from left to right. But the slope with water doesn't vary across shade levels. Without the interaction, it cannot vary. In the bottom row, the interaction is turned on. Now the model believes that the effect of water decreases as shade increases. The lines get flat.

What is going on here? The likely explanation for these results is that tulips need both water and light to produce blooms. At low light levels, water can't have much of an effect, because the tulips don't have enough light to produce blooms. At higher light levels, water can matter more, because the tulips have enough light to produce blooms. At very high light levels, light is no longer limiting the blooms, and so water can have a much more dramatic impact on the outcome. The same explanation works symmetrically for shade. If there isn't enough light, then more water hardly helps. You could remake FIGURE 8.7 with shade on the

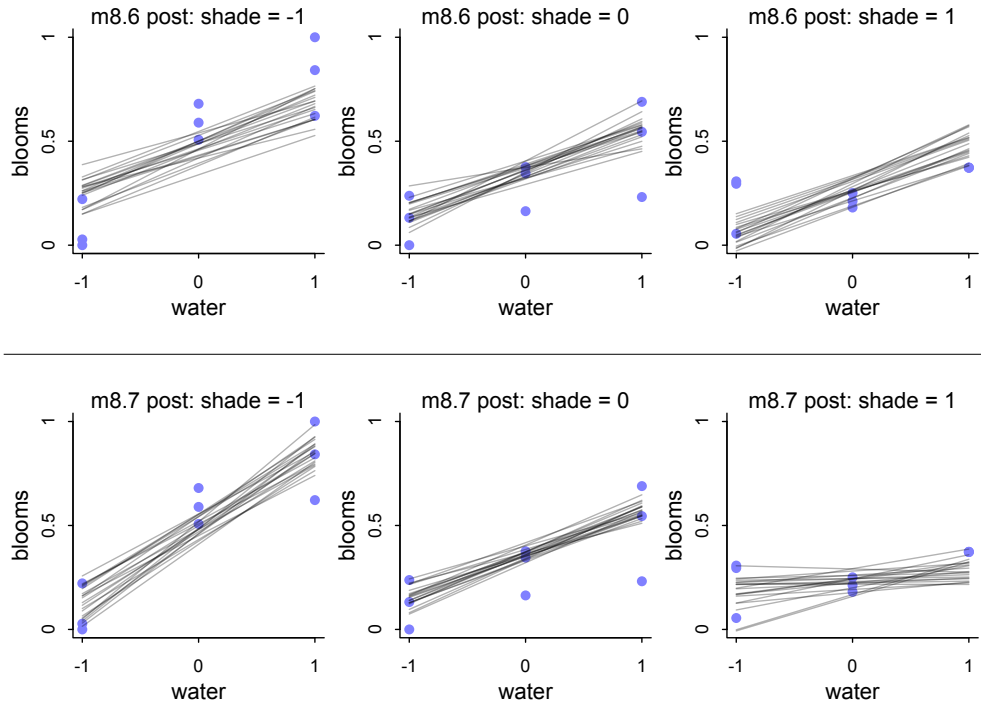


FIGURE 8.7. Triptych plots of posterior predicted blooms across water and shade treatments. Top row: Without an interaction between water and shade. Bottom row: With an interaction between water and shade. Each plot shows 20 posterior lines for each level of shade.

horizontal axes and water level varied from left to right, if you'd like to visualize the model predictions that way.

8.3.4. Plotting prior predictions. And we can use the same technique to finally plot prior predictive simulations as well. This will let us evaluate my guesses from earlier. To produce the prior predictions, all that's need is to extract the prior:

R code 8.26

```
set.seed(7)
prior <- extract.prior(m8.5)
```

And then add `post=prior` as an argument to the `link` call in the previous code. I've also adjusted the vertical range of the prior plots, so we can see more easily the lines that fall outside the valid outcome range.

The result is displayed as FIGURE 8.8. Since the lines are so scattered in the prior—the prior not very informative—it is hard to see that the lines from the same set of samples actually go together in meaningful ways. So I've bolded three lines in the top and in the bottom rows. The three bolded lines in the top row come from the same parameter values. Notice

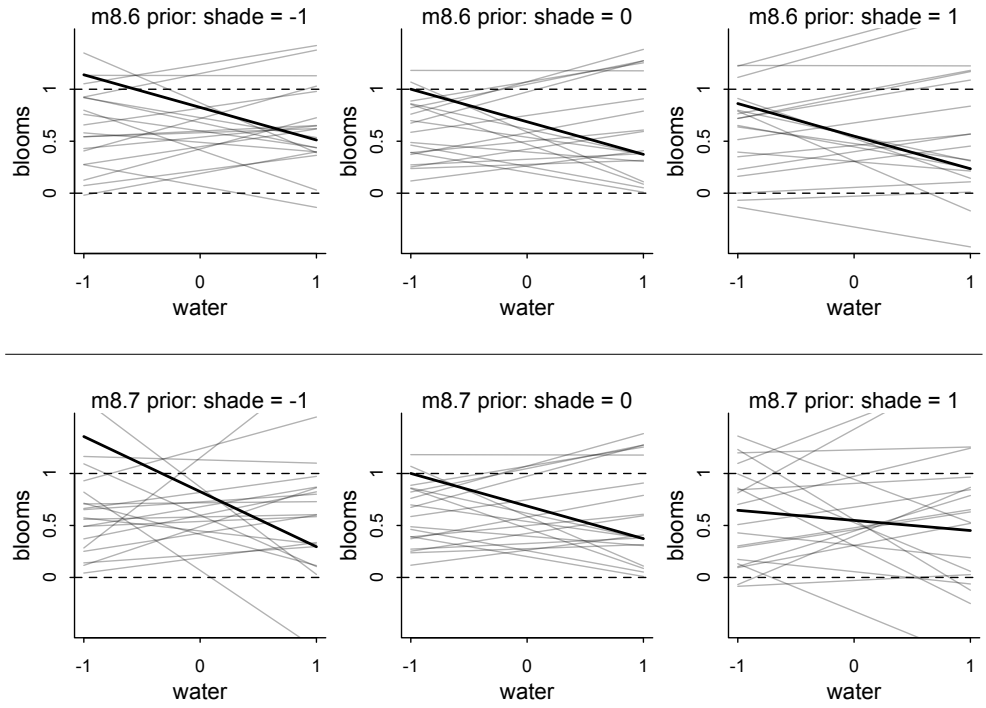


FIGURE 8.8. Triptych plots of prior predicted blooms across water and shade treatments. Top row: Without an interaction between water and shade. Bottom row: With an interaction between water and shade. Each plot shows 20 prior lines for each level of shade.

that all three have the same slope. This is what we expect from a model without an interaction. So while the lines in the prior have lots of different slopes, the slopes for water don't depend upon shade. In the bottom row, the three bolded lines again come from a single prior sample. But now the interaction makes the slope systematically change as shade changes.

What can we say about these priors, overall? They are harmless, but only weakly realistic. Most of the lines stay within the valid outcome space. But silly trends are not rare. We could do better. We could also do a lot worse, such as flat priors which would consider plausible that even a tiny increase in shade would kill all the tulips. If you displayed these priors to your colleagues, a reasonable summary might be, "These priors contain no bias towards positive or negative effects, and at the same time they very weakly bound the effects to realistic ranges."

8.4. Summary

This chapter introduced *interactions*, which allow for the association between a predictor and an outcome to depend upon the value of another predictor. While you can't see them in a DAG, interactions can be important for making accurate inferences. Interactions can be difficult to interpret, and so the chapter also introduced *triptych* plots that help in visualizing the effect of an interaction. No new coding skills were introduced, but the statistical models

considered were among the most complicated so far in the book. To go any further, we're going to need a more capable conditioning engine to fit our models to data. That's the topic of the next chapter.

8.5. Practice

Easy.

8E1. For each of the causal relationships below, name a hypothetical third variable that would lead to an interaction effect.

(1) Bread dough rises because of yeast.
(2) Education leads to higher income.
(3) Gasoline makes a car go.

8E2. Which of the following explanations invokes an interaction?

(1) Caramelizing onions requires cooking over low heat and making sure the onions do not dry out.
(2) A car will go faster when it has more cylinders or when it has a better fuel injector.
(3) Most people acquire their political beliefs from their parents, unless they get them instead from their friends.
(4) Intelligent animal species tend to be either highly social or have manipulative appendages (hands, tentacles, etc.).

8E3. For each of the explanations in **8E2**, write a linear model that expresses the stated relationship.

Medium.

8M1. Recall the tulips example from the chapter. Suppose another set of treatments adjusted the temperature in the greenhouse over two levels: cold and hot. The data in the chapter were collected at the cold temperature. You find none of the plants grown under the hot temperature developed any blooms at all, regardless of the water and shade levels. Can you explain this result in terms of interactions between water, shade, and temperature?

8M2. Can you invent a regression equation that would make the bloom size zero, whenever the temperature is hot?

8M3. In parts of North America, ravens depend upon wolves for their food. This is because ravens are carnivorous but cannot usually kill or open carcasses of prey. Wolves however can and do kill and tear open animals, and they tolerate ravens co-feeding at their kills. This species relationship is generally described as a "species interaction." Can you invent a hypothetical set of data on raven population size in which this relationship would manifest as a statistical interaction? Do you think the biological interaction could be linear? Why or why not?

Hard.

8H1. Return to the `data(tulips)` example in the chapter. Now include the `bed` variable as a predictor in the interaction model. Don't interact `bed` with the other predictors; just include it as a main effect. Note that `bed` is categorical. So to use it properly, you will need to either construct dummy variables or rather an index variable, as explained in Chapter 6.

8H2. Use WAIC to compare the model from **8H1** to a model that omits `bed`. What do you infer from this comparison? Can you reconcile the WAIC results with the posterior distribution of the `bed` coefficients?

8H3. Consider again the `data(rugged)` data on economic development and terrain ruggedness, examined in this chapter. One of the African countries in that example, Seychelles, is far outside the cloud of other nations, being a rare country with both relatively high GDP and high ruggedness. Seychelles is also unusual, in that it is a group of islands far from the coast of mainland Africa, and its main economic activity is tourism.

(a) Focus on model `m8.5` from the chapter. Use WAIC pointwise penalties and PSIS Pareto k values to measure relative influence of each country. By these criteria, is Seychelles influencing the results? Are there other nations that are relatively influential? If so, can you explain why?

(b) Now use robust regression, as described in the previous chapter. Modify `m8.5` to use a Student-t distribution with $\nu = 2$. Does this change the results in a substantial way?

8H4. The values in `data(nettle)` are data on language diversity in 74 nations.[136] The meaning of each column is given below.

(1) `country`: Name of the country
(2) `num.lang`: Number of recognized languages spoken
(3) `area`: Area in square kilometers
(4) `k.pop`: Population, in thousands
(5) `num.stations`: Number of weather stations that provided data for the next two columns
(6) `mean.growing.season`: Average length of growing season, in months
(7) `sd.growing.season`: Standard deviation of length of growing season, in months

Use these data to evaluate the hypothesis that language diversity is partly a product of food security. The notion is that, in productive ecologies, people don't need large social networks to buffer them against risk of food shortfalls. This means ethnic groups can be smaller and more self-sufficient, leading to more languages per capita. In contrast, in a poor ecology, there is more subsistence risk, and so human societies have adapted by building larger networks of mutual obligation to provide food insurance. This in turn creates social forces that help prevent languages from diversifying.

Specifically, you will try to model the number of languages per capita as the outcome variable:

R code 8.27

```
d$lang.per.cap <- d$num.lang / d$k.pop
```

Use the logarithm of this new variable as your regression outcome. (A count model would be better here, but you'll learn those later, in Chapter 11.)

This problem is open ended, allowing you to decide how you address the hypotheses and the uncertain advice the modeling provides. If you think you need to use WAIC anyplace, please do. If you think you need certain priors, argue for them. If you think you need to plot predictions in a certain way, please do. Just try to honestly evaluate the main effects of both `mean.growing.season` and `sd.growing.season`, as well as their two-way interaction, as outlined in parts (a), (b), and (c) below. If you are not sure which approach to use, try several.

(a) Evaluate the hypothesis that language diversity, as measured by `log(lang.per.cap)`, is positively associated with the average length of the growing season, `mean.growing.season`. Consider `log(area)` in your regression(s) as a covariate (not an interaction). Interpret your results.

(b) Now evaluate the hypothesis that language diversity is negatively associated with the standard deviation of length of growing season, `sd.growing.season`. This hypothesis follows from uncertainty in harvest favoring social insurance through larger social networks and therefore fewer languages. Again, consider `log(area)` as a covariate (not an interaction). Interpret your results.

(c) Finally, evaluate the hypothesis that `mean.growing.season` and `sd.growing.season` interact to synergistically reduce language diversity. The idea is that, in nations with longer average growing seasons, high variance makes storage and redistribution even more important than it would