SCI 2025: Homework 5

Question 1: 9H3

Sometimes changing a prior for one parameter has unanticipated effects on other parameters. This is because when a parameter is highly correlated with another parameter in the posterior, the prior influences both parameters. Here's an example to work and think through. Go back to the leg length example in Chapter 6 and use the code there to simulate height and leg lengths for 100 imagined individuals. Below is the model you fit before, resulting in a highly correlated posterior for the two beta parameters. This time, fit the model using `ulam`:

```r
library(rethinking)

N <- 100 # number of individuals
set.seed(909)
height <- rnorm(N,10,2) # sim total height of each
leg_prop <- runif(N,0.4,0.5) # leg as proportion of height
leg_left <- leg_prop*height + # sim left leg as proportion + error
    rnorm( N,0,0.02)
leg_right <- leg_prop*height + #sim right leg as proportion + error
    rnorm( N,0,0.02)
# combine into data frame
d <- data.frame(height,leg_left,leg_right)
```

```r
m5.8s <- ulam(
    alist(
        height ~ dnorm(mu, sigma),
        mu <- a + bl*leg_left + br*leg_right,
        a ~ dnorm(10,100),
        bl ~ dnorm(2,10),
        br ~ dnorm(2,10),
        sigma ~ dexp(1)
    ), data=d, chains=4,
    start=list(a=10, bl=0, br=0.1, sigma=1),
    log_lik=TRUE
    )
```

Let's take a look at the coordinates (posterior values) for `bl` and `br` over iterations from our chains.

```r
post <- extract.samples(m5.8s) # extract posterior samples

plot(post$bl[1:30], type = 'l', pch=16, col=rangi2, xlab="iteration",
ylab="parameter value", ylim = c(-8, 10))
points(post$bl[1:30], pch=16, col=rangi2)

lines(post$br[1:30], col="orange")
points(post$br[1:30], pch=16, col="orange")
legend("topright", legend=c("bl", "br"), col=c(rangi2, "orange"), lty=1, cex=0.8)
```

What do you notice about the positions of `br` and `bl` over iterations? Can you explain why this is happening?

Question 2:

One way to improve the sampling of HMC with highly correlated parameters is to set a prior on the total prior contribution, plus a proportional allocation (e.g., 50% of the effect to right leg, 50% to left leg). See example below:

```
m5.8s2 <- ulam(
    alist(
        height ~ dnorm(mu, sigma),
        transpars> bl <- b*theta,
        transpars> br <- b*(1 - theta),
        mu <- a + bl*leg_left + br*leg_right,
        a ~ dnorm(10,100),
        b ~ dnorm(2,10),
        theta ~ dbeta(2, 2),
        sigma ~ dexp(1)
    ), data=d, chains=4,
    start=list(a=10, b=1, theta=0.5, sigma=1),
    log_lik=TRUE
    )
```

`transpars` refers to the "transformed parameters" block in a Stan program, and can be used whenever we have parameters that are derived deterministically from other parameters.

Take a look at the summary of the model above, as well as the first model we fit. Do you notice any differences in the HMC sampling given this new parameterization? What about warnings and sampling speed?

Question 3:

For the two models above, use WAIC or PSIS to compute the effective number of parameters (i.e., the penalty term). You will need to use `log_lik=TRUE` when fitting the model to instruct `ulam` to compute the terms that both WAIC and PSIS need.

Which model has lower effective number of parameters? Can you explain it? Which model has the better expected out-of-sample predictive accuracy?

Question 4

In question 2 I said that the variance decomposition approach I used (sometimes called a 'dirichlet decomposition' when generalied to more than two predictors) is a strategy for handling highly correlated predictors. What assumptions are we making when we use this approach? What potential downsides or limitations occur to you?

Question 5

Re-fit the model in question 2, but this time make the prior for `theta` concentrated around most of the effect going to the right leg (if you're not sure how, try simulating some values from a beta distribution with `rbeta()`). Do you notice and differences in sampling, and effective number of parameters? Can you explain it?