Changes in smoking and physical activity-related behavior and identity between prescreening and followup-questionnaire

Nele Albers

12 June, 2024

Contents

Introduction **1**

Setup **1**

Data file **1**

Models **2**
Quitter self-identity 2
Smoking frequency 4
Weekly exercise 9

Introduction

This file is meant to reproduce our analyses of the changes in smoking frequency level, weekly exercise level, and quitter self-identity between the prescreening questionnaire and the follow-up questionnaire reported in "Step 5: Training the Model - Study."

Authored by Nele Albers, Mark A. Neerincx, and Willem-Paul Brinkman.

Setup

First, we load the `rethinking` package, which we need to fit and sample from models. We also load `formatR` for formatting.

```
library(formatR)  # For formatting
library(rethinking)  # For Bayesian models
```

Also, we set the number of iterations and number of chains used for fitting the models.

```
NUM_ITERATIONS = 2000  # our value: 2000
NUM_CHAINS = 4  # our value: 4
```

Data file

We load the pre-processed data.

```
df = read.csv(file = "Data/prefu_preprocessed.csv")
```

And we create a datalist to be used for the models.

```
dat_list <- list(
  p_id = df$pid,
  pre_post = df$pre_post,
  we = df$weekly_exercise + 1,  # values need to start at 1
  qsi = df$QSI,
  sf = df$smoking_frequency + 4  # values need to start at 1
)
```

Models

Quitter self-identity

First, we fit a model for the quitter self-identity. Possible values range from 0 to 4.

```
set.seed(18)  # For reproducibility
ml.qsi <- ulam(alist(qsi ~ dnorm(mu, sigma), mu <- a_bar +
    z[p_id] * sigma_a + b * pre_post, a_bar ~ dnorm(2,
    10), sigma_a ~ dexp(1), z[p_id] ~ dnorm(0, 1),
    b ~ dnorm(0, 10), sigma ~ dexp(1)), data = dat_list,
    chains = NUM_CHAINS, log_lik = TRUE, cores = NUM_CHAINS,
    iter = NUM_ITERATIONS)
```

```
## Running MCMC with 4 parallel chains, with 1 thread(s) per chain...
##
## Chain 1 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: normal_lpdf: Scale parameter is 0, but must be positive! (in '/tmp/RtmpPT6pQo/model
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 2 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 1 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 2 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 4 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 3 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 4 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 2 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 3 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 1 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 2 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 4 Iteration:  400 / 2000 [ 20%]  (Warmup)
```

```
## Chain 1 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 2 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 3 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 4 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 2 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 3 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 1 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 3 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 4 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 1 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 2 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 1 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 2 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 3 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 4 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 1 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 2 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 4 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 1 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 3 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 2 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 4 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 1 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 3 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 2 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 4 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1 Iteration: 1800 / 2000 [ 90%]  (Sampling)
```

```
## Chain 3 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 1 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 3 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 4 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 1 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1 finished in 5.4 seconds.
## Chain 3 finished in 5.3 seconds.
## Chain 2 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 4 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2 finished in 5.4 seconds.
## Chain 4 finished in 5.4 seconds.
##
## All 4 chains finished successfully.
## Mean chain execution time: 5.4 seconds.
## Total execution time: 5.6 seconds.
```

```
# Print estimators
precis(ml.qsi, prob = 0.95, depth = 1)
```

```
## 245 vector or matrix parameters hidden. Use depth=2 to show them.
```

```
##              mean         sd        2.5%     97.5%     rhat ess_bulk
## a_bar   2.6264942 0.08060552 2.470476250 2.7868823 1.001535 4593.859
## sigma_a 0.4842601 0.03768004 0.412737800 0.5576370 1.001637 1443.882
## b       0.1027312 0.04746072 0.006661239 0.1939390 1.001084 5878.147
## sigma   0.5333115 0.02401623 0.488367775 0.5823779 1.000812 2026.040
```

Smoking frequency

Next, we fit a model for the smoking frequency. This is an ordered categorical variable with eight levels.

```
set.seed(18)  # For reproducibility
ml.sf <- ulam(alist(sf ~ dordlogit(phi, cutpoints),
    phi <- a_bar + z[p_id] * sigma_a + b * pre_post,
    cutpoints ~ dnorm(0, 1.5), a_bar ~ dnorm(0, 0.5),
    sigma_a ~ dexp(1), z[p_id] ~ dnorm(0, 1), b ~
        dnorm(0, 10)), data = dat_list, chains = NUM_CHAINS,
    log_lik = TRUE, cores = NUM_CHAINS, iter = NUM_ITERATIONS)
```

```
## Running MCMC with 4 parallel chains, with 1 thread(s) per chain...
##
## Chain 1 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
```

```
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -0
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 3 is in
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 7 is 3
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 1
## Chain 2 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 2 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 2
## Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 2 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 2
## Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of the
```

```
## Chain 2 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -5
## Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 2
## Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 2 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 6 is in
## Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 2
## Chain 3 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 3 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 3
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 3 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -1
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 3
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 3 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 3 is -0
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 3
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 3 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 3 is -1
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 3
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 3 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 6 is in
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 3
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of the
```

```
## Chain 3 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 7 is 7
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 3
## Chain 4 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -i
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 4
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -i
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 4
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -3
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 4
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 3 is in
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 4
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 7 is 1
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like covar
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or mi
## Chain 4
## Chain 2 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 3 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 1 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 2 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 3 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 4 Iteration:  200 / 2000 [ 10%]  (Warmup)
```

```
## Chain 3 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 1 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 2 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 2 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 1 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 3 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 4 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 2 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 3 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 1 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 2 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 2 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 3 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 1 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 4 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 2 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 3 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 1 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 4 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 2 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 3 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 1 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 4 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 2 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 3 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 1 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 4 Iteration: 1400 / 2000 [ 70%]  (Sampling)
```

```
## Chain 2 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 1 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 3 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 4 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 2 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 1 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 3 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 4 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 2 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2 finished in 45.4 seconds.
## Chain 3 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3 finished in 45.7 seconds.
## Chain 4 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 1 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1 finished in 46.6 seconds.
## Chain 4 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4 finished in 47.9 seconds.
##
## All 4 chains finished successfully.
## Mean chain execution time: 46.4 seconds.
## Total execution time: 48.1 seconds.
```

```
# Print estimators
precis(ml.sf, prob = 0.95, depth = 1)
```

```
## 252 vector or matrix parameters hidden. Use depth=2 to show them.
```

```
##               mean        sd       2.5%      97.5%     rhat ess_bulk
## a_bar    1.4594300 0.4078764  0.6603914  2.2608755 1.000962 8751.943
## sigma_a  2.1633189 0.2083059  1.7625258  2.5799568 1.001722 1235.147
## b       -0.9614968 0.1705213 -1.3030708 -0.6308314 1.000927 7260.814
```

Weekly exercise

And we also fit a model for the weekly exercise. This is an ordered categorical variable with three levels.

```
set.seed(18)  # For reproducibility
ml.we <- ulam(alist(we ~ dordlogit(phi, cutpoints),
    phi <- a_bar + z[p_id] * sigma_a + b * pre_post,
    cutpoints ~ dnorm(0, 1.5), a_bar ~ dnorm(0, 0.5),
    sigma_a ~ dexp(1), z[p_id] ~ dnorm(0, 1), b ~
        dnorm(0, 10)), data = dat_list, chains = NUM_CHAINS,
    log_lik = TRUE, cores = NUM_CHAINS, iter = NUM_ITERATIONS)
```

```
## Running MCMC with 4 parallel chains, with 1 thread(s) per chain...
##
## Chain 1 Iteration:    1 / 2000 [  0%]  (Warmup)
```

```
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of th
```

```
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of th
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of th
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 1
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of th
## Chain 1 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 1
## Chain 2 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of th
## Chain 3 Exception: ordered_logistic: Final cut-point is inf, but must be finite! (in '/tmp/RtmpPT6pQ
## Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 3
## Chain 4 Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of th
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 4
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of th
## Chain 4 Exception: ordered_logistic: Cut-points is not a valid ordered vector. The element at 2 is -
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or m
```

```
## Chain 4
## Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 4 Exception: ordered_logistic: Final cut-point is inf, but must be finite! (in '/tmp/RtmpPT6pQ
## Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 4
## Chain 1 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 4 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 3 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 2 Iteration:  100 / 2000 [  5%]  (Warmup)
## Chain 1 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 2 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3 Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 2 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 4 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3 Iteration:  300 / 2000 [ 15%]  (Warmup)
## Chain 1 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 2 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 3 Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 1 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 3 Iteration:  500 / 2000 [ 25%]  (Warmup)
## Chain 4 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 1 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3 Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 1 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 4 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 3 Iteration:  700 / 2000 [ 35%]  (Warmup)
## Chain 2 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 3 Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 4 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3 Iteration:  900 / 2000 [ 45%]  (Warmup)
## Chain 2 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2 Iteration: 1100 / 2000 [ 55%]  (Sampling)
```

```
## Chain 4 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 1 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 3 Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3 Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3 Iteration: 1100 / 2000 [ 55%]  (Sampling)
## Chain 4 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 1 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 3 Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 3 Iteration: 1300 / 2000 [ 65%]  (Sampling)
## Chain 2 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 1 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 3 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4 Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 2 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 1 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 3 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4 Iteration: 1500 / 2000 [ 75%]  (Sampling)
## Chain 2 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 1 finished in 35.4 seconds.
## Chain 2 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 4 Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2 finished in 37.5 seconds.
## Chain 3 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 4 Iteration: 1700 / 2000 [ 85%]  (Sampling)
## Chain 3 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3 finished in 39.1 seconds.
## Chain 4 Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4 Iteration: 1900 / 2000 [ 95%]  (Sampling)
## Chain 4 Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4 finished in 41.5 seconds.
##
## All 4 chains finished successfully.
## Mean chain execution time: 38.4 seconds.
## Total execution time: 41.7 seconds.
```

```
# Print estimators
precis(ml.we, prob = 0.95, depth = 1)
```

```
## 247 vector or matrix parameters hidden. Use depth=2 to show them.
```

```
##                mean        sd       2.5%     97.5%     rhat  ess_bulk
## a_bar   -0.004943376 0.4571128 -0.8956368 0.9086421 1.000292 2981.2547
```

```
## sigma_a  3.324518920 0.3714516  2.6484855 4.0963695 1.005711  986.0881
## b        0.185695467 0.2176186 -0.2501075 0.6307969 1.002455 3874.2261
```