CH 13

Emilio Horner

2023-02-03

CH 13

1e.

This distribution αtank ~ Normal(0, 1) has more shrinkage.

2e.

yi ~ Binomial(1, pi) logit(pi) = αgroup[i] + βxi αgroup ~ Normal(α_bar, σα) α_bar ~ Normal(0, 1.5) σα ~ Exponential(1) β ~ Normal(0, 0.5)

3e.

yi ~ Normal(μi, σ) μi = α_group[i] + βx_i α_group[i] ~ Normal(α_bar, σα) β ~ Normal(0, 1) σ ~ Exponential(1) α_bar ~ Normal(0, 1) σα ~ Exponential(1)

4e.

yi ~ Poisson(pi) log(pi) = αgroup[i] + βxi αgroup ~ Normal(α_bar, σα) α_bar ~ Normal(0, 1) σα ~ Exponential(1) β ~ Normal(0, 1)

5e.

yi ~ Poisson(pi) log(pi) = αgroup[i] + αgroupb[i] + βxi αgroup ~ Normal(α_bar, σgroup) αgroupb ~ Normal(0, σgroupb) α_bar ~ Normal(0, 1) σgroup ~ Exponential(1) σgroupb ~ Exponential(1) β ~ Normal(0, 1)

1m. Revisit the Reed frog survival data, data(reedfrogs), and add the predation and size treatment variables to the varying intercepts model. Consider models with either main effect alone,both main effects, as well as a model including both and their interaction. Instead of focusing on inferences about these two predictor variables, focus on the inferred variation across tanks. Explain why it changes as it does across models.

```
library(rethinking)
```

```
## Loading required package: rstan
```

```
## Loading required package: StanHeaders
```

```
##
## rstan version 2.26.13 (Stan version 2.26.1)
```

```
## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)
## For within-chain threading using `reduce_sum()` or `map_rect()` Stan functions,
## change `threads_per_chain` option:
## rstan_options(threads_per_chain = 1)
```

```
## Do not specify '-march=native' in 'LOCAL_CPPFLAGS' or a Makevars file
```

```
## Loading required package: parallel
```

```
## Loading required package: dagitty
```

```
## rethinking (Version 2.01)
```

```
##
## Attaching package: 'rethinking'
```

```
## The following object is masked from 'package:stats':
##
##     rstudent
```

```
library(tidyr)
```

```
##
## Attaching package: 'tidyr'
```

```
## The following object is masked from 'package:rstan':
##
##     extract
```

```
library(rstan)
data(reedfrogs)
d <- reedfrogs
str(d)
```

```
## 'data.frame':    48 obs. of  5 variables:
##  $ density : int  10 10 10 10 10 10 10 10 10 10 ...
##  $ pred    : Factor w/ 2 levels "no","pred": 1 1 1 1 1 1 1 1 2 2 ...
##  $ size    : Factor w/ 2 levels "big","small": 1 1 1 1 2 2 2 2 2 1 1 ...
##  $ surv    : int  9 10 7 10 9 9 10 9 4 9 ...
##  $ propsurv: num  0.9 1 0.7 1 0.9 0.9 1 0.9 0.4 0.9 ...
```

$dtank < -1 : nrow(d)dat < -list(S = d$surv, P = d$pred, $Z = d$size, tank = d$tank)

approximate posterior

m13.1 <- ulam(alist(S ~ dbinom(Z, p) , S ~ dbinom(P, p), logit(p) <- a[tank] , a[tank] ~ dnorm(0 , 1.5)), data=dat , chains=4 , log_lik=TRUE)

$dtank < -1 : nrow(d)dat < -list(S = d$surv, P = d$pred, $Z = d$size, tank = d$tank)

m13.2 <- ulam(alist(S ~ dbinom(Z , p) , S ~ dbinom(P , p) logit(p) <- a[tank] , a[tank] ~ dnorm(a_bar , sigma) , a_bar ~ dnorm(0 , 1.5) sigma ~ dexp(1)), data=dat , chains=4 , log_lik=TRUE)

I think this is what the code would look like, but I can't get it to run without an error message.

2m.

I would just do:

WAIC(m13.1)

WAIC(m13.2)

3m. Cauchy distribution in place of the Gaussian distribution. Cauchy has a higher peak.

I can't figure this out, and have run out of time.