Working with Priors in the `rstanarm` package

Bayesian Data Analysis

Steve Buyske

Coding priors in `rstanarm`

- In `rstanarm`, you can specify arguments for
  - `prior_intercept`, model intercept, *after centering predictors*
  - `prior_aux`, used in linear regression for σ
  - `prior`, used for all the (overall) regression coefficients
  - `prior_covariance`, used for covariance matrices in hierarchical models with varying slopes and intercepts

Defaults in `rstanarm`

- The defaults in `stan_glm()` and `stan_glmer()` are
  - `prior_intercept = normal(`mean of y`, 2.5, autoscale = TRUE)`
  - `prior_aux = exponential(rate = 1, autoscale = TRUE)`
  - `prior = normal(0, 2.5, autoscale = TRUE)`
  - `prior_covariance = decov(regularization = 1, concentration = 1, shape = 1, scale = 1)`
- All but the last are **scaled** (that's what `autoscale = TRUE` does)

To get the unscaled version of the default priors

- For `prior_intercept`, multiply by $\text{sd}(y)$
- For `prior`, multiply by $\text{sd}(y)/\text{sd}(x)$
- For `prior_aux`, *divide* by $\text{sd}(y)$

- You can examine these by using `prior_summary()`:

```
prior_summary(gini_stan)
```

```
## Priors for model 'gini_stan'
## ------
## Intercept (after predictors centered)
##   Specified prior:
##     ~ normal(location = 73, scale = 2.5)
##   Adjusted prior:
##     ~ normal(location = 73, scale = 18)
##
## Coefficients
##   Specified prior:
##     ~ normal(location = 0, scale = 2.5)
##   Adjusted prior:
##     ~ normal(location = 0, scale = 2.3)
##
## Auxiliary (sigma)
##   Specified prior:
```

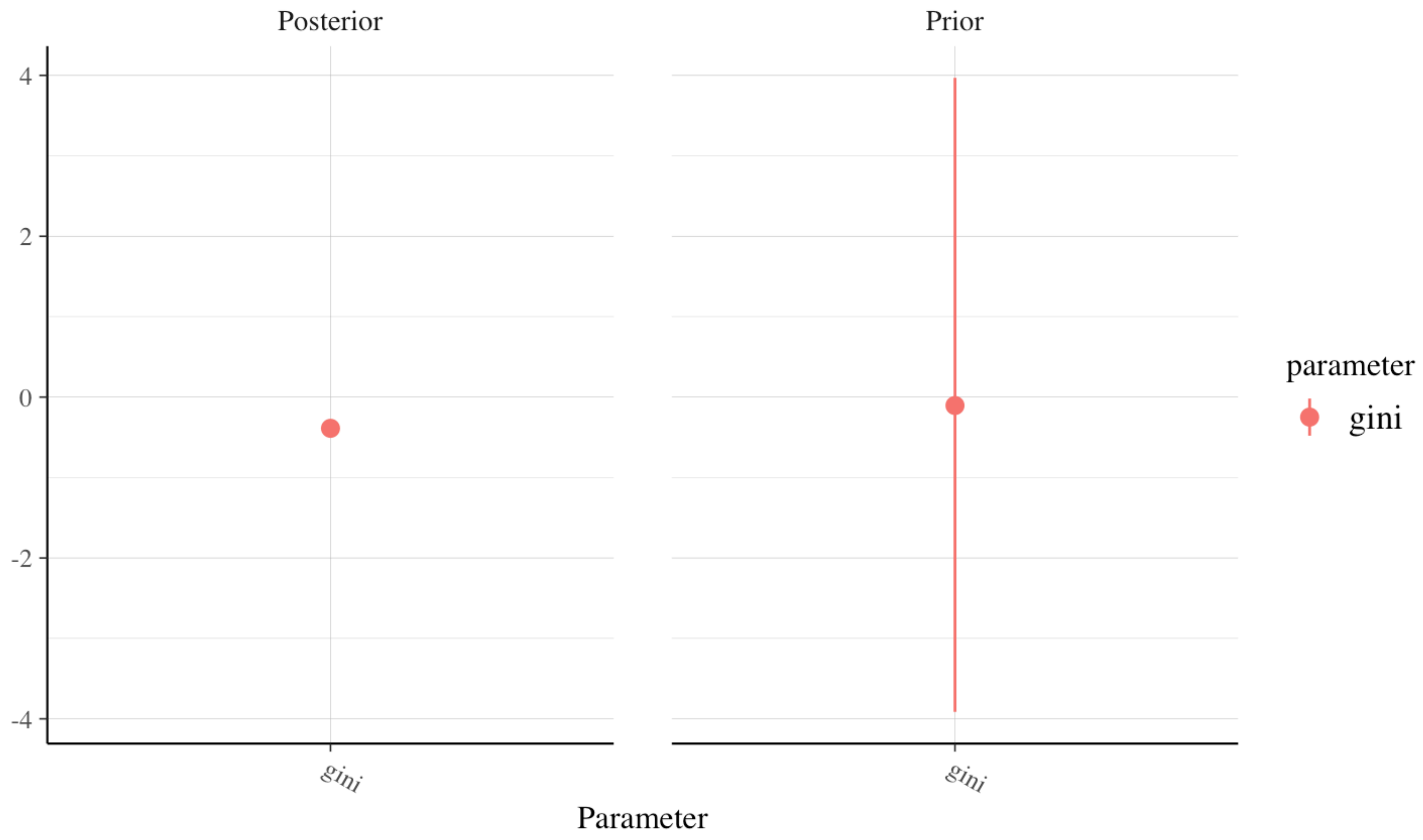
- Or with `describe_prior()`, which does not include the priors for σ or for the group covariance.

```
describe_prior(gini_stan)
```

```
##     Parameter Prior_Distribution Prior_Location Prior_Scale
## 1 (Intercept)             normal       72.92203   17.927098
## 2        gini             normal        0.00000    2.340514
```

- And you can compare the posterior and the prior with `posterior_vs_prior()`

```
posterior_vs_prior(gini_stan, par = "gini")
```



Specifying priors in `rstanarm` functions.

- As a running example, we will use the sleep study example from Week 4.

```
sleep2 <- stan_glmer(Reaction ~ Days + (Days | Subject),  data = sleepstudy, iter = 7500, cores = 4)
```

- If you want a flat prior, just use `NULL` as the value to the argument from a few slides back.

```
sleep2b <- stan_glmer(Reaction ~ Days + (Days | Subject),  data = sleepstudy, iter = 7500, cores = 4, prior = NULL)
```

```
describe_prior(sleep2)
```

```
##     Parameter Prior_Distribution Prior_Location Prior_Scale
## 1 (Intercept)             normal       298.5079   140.82189
## 2        Days             normal         0.0000    48.89151
```

```
describe_prior(sleep2b)
```

```
##     Parameter Prior_Distribution Prior_Location Prior_Scale
## 1 (Intercept)             normal       298.5079    140.8219
## 2        Days            uniform             NA          NA
```

```
describe_posterior(sleep2)
```

```
## # Description of Posterior Distributions
##
## Parameter   |  Median |             89% CI |      pd |        89% ROPE | % in ROPE |  Rhat |      ESS
## -----------------------------------------------------------------------------------------------------
## (Intercept) | 251.480 | [241.168, 262.273] | 100.00% | [-5.633, 5.633] |         0 | 1.000 | 8615.117
## Days        |  10.475 | [  7.812,  13.345] | 100.00% | [-5.633, 5.633] |         0 | 1.000 | 5506.912
```

```
describe_posterior(sleep2b)
```

```
## # Description of Posterior Distributions
##
## Parameter   |  Median |             89% CI |      pd |        89% ROPE | % in ROPE |  Rhat |      ESS
## -----------------------------------------------------------------------------------------------------
## (Intercept) | 251.364 | [240.567, 261.996] | 100.00% | [-5.633, 5.633] |         0 | 1.000 | 8232.685
## Days        |  10.458 | [  7.669,  13.258] | 100.00% | [-5.633, 5.633] |         0 | 1.001 | 4920.429
```

- You can see that in this case the choice of a prior makes virtually no difference.
- One prior is weakly informative while the other is uninformative, but the similiar posteriors suggest that the posterior is not sensitive to the prior.

- Suppose you had wanted to use a strong prior that the regression coefficient for `Days` was close to zero.
- To use a normal prior, the syntax is `normal(location, scale)`; for Student's t prior, the syntax is `student_t(df, location, scale)`
- So you might fit a model with

```
sleep2c <- stan_glmer(Reaction ~ Days + (Days | Subject),
                      data = sleepstudy,
                      iter = 7500,
                      cores = 4,
                      prior = normal(0, 3, autoscale = FALSE)
)
```

```
describe_prior(sleep2)
```

```
##     Parameter Prior_Distribution Prior_Location Prior_Scale
## 1 (Intercept)             normal       298.5079   140.82189
## 2        Days             normal         0.0000    48.89151
```

```
describe_prior(sleep2c)
```

```
##     Parameter Prior_Distribution Prior_Location Prior_Scale
## 1 (Intercept)             normal       298.5079    140.8219
## 2        Days             normal         0.0000      3.0000
```

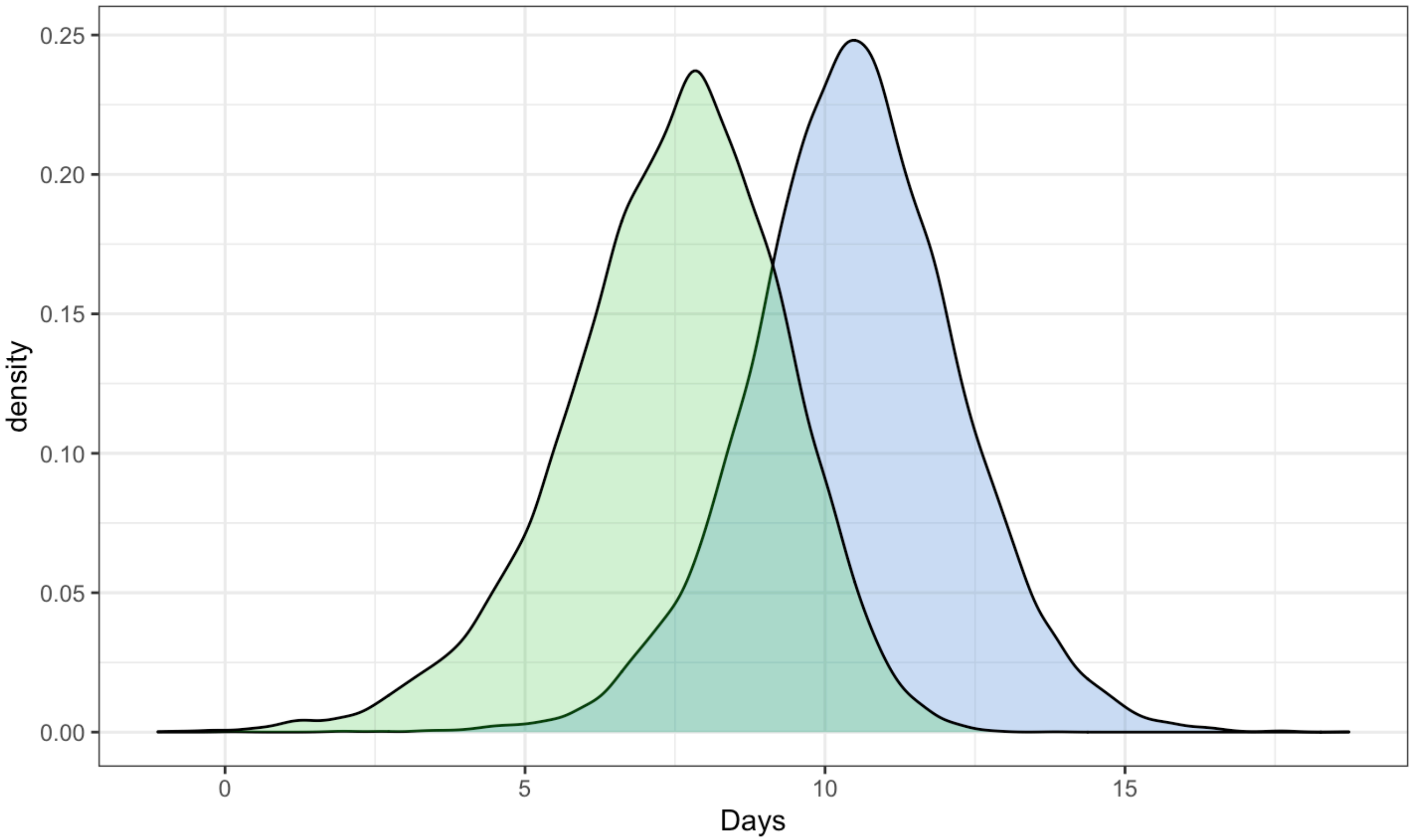
```
describe_posterior(sleep2)
```

```
## # Description of Posterior Distributions
##
## Parameter   |  Median |             89% CI |      pd |        89% ROPE | % in ROPE |  Rhat |      ESS
## -----------------------------------------------------------------------------------------------------
## (Intercept) | 251.480 | [241.168, 262.273] | 100.00% | [-5.633, 5.633] |         0 | 1.000 | 8615.117
## Days        |  10.475 | [  7.812,  13.345] | 100.00% | [-5.633, 5.633] |         0 | 1.000 | 5506.912
```

```
describe_posterior(sleep2c)
```

```
## # Description of Posterior Distributions
##
## Parameter   |  Median |             89% CI |      pd |        89% ROPE | % in ROPE |  Rhat |      ESS
## -----------------------------------------------------------------------------------------------------
## (Intercept) | 252.523 | [241.291, 264.011] | 100.00% | [-5.633, 5.633] |     0.000 | 1.001 | 4924.608
## Days        |   7.627 | [  4.767,  10.466] |  99.94% | [-5.633, 5.633] |     8.224 | 1.001 | 4227.010
```

- You can see that the normal(0, 3) prior had a noticeable effect on the posteror.
- The figure below (not shown in the video) shows how the informative prior, with a much smaller spread, has shifted the posterior to the left. The shapes are otherwise pretty similar.



The `autoscale` argument

- What's going on with the `autoscale` argument?
- The default in `rstanarm` is to rescale the priors, by multiplying the scale for slope coefficients by $\mathrm{sd}(y)/\mathrm{sd}(x)$, for example.
    - This is convenient—once you know the default is `scale = 2.5`, you can change how informative you want the prior to be by changing that without any more calculation.
- If however, you already know the scale that you want, you can use `autoscale = FALSE` to prevent any rescaling.

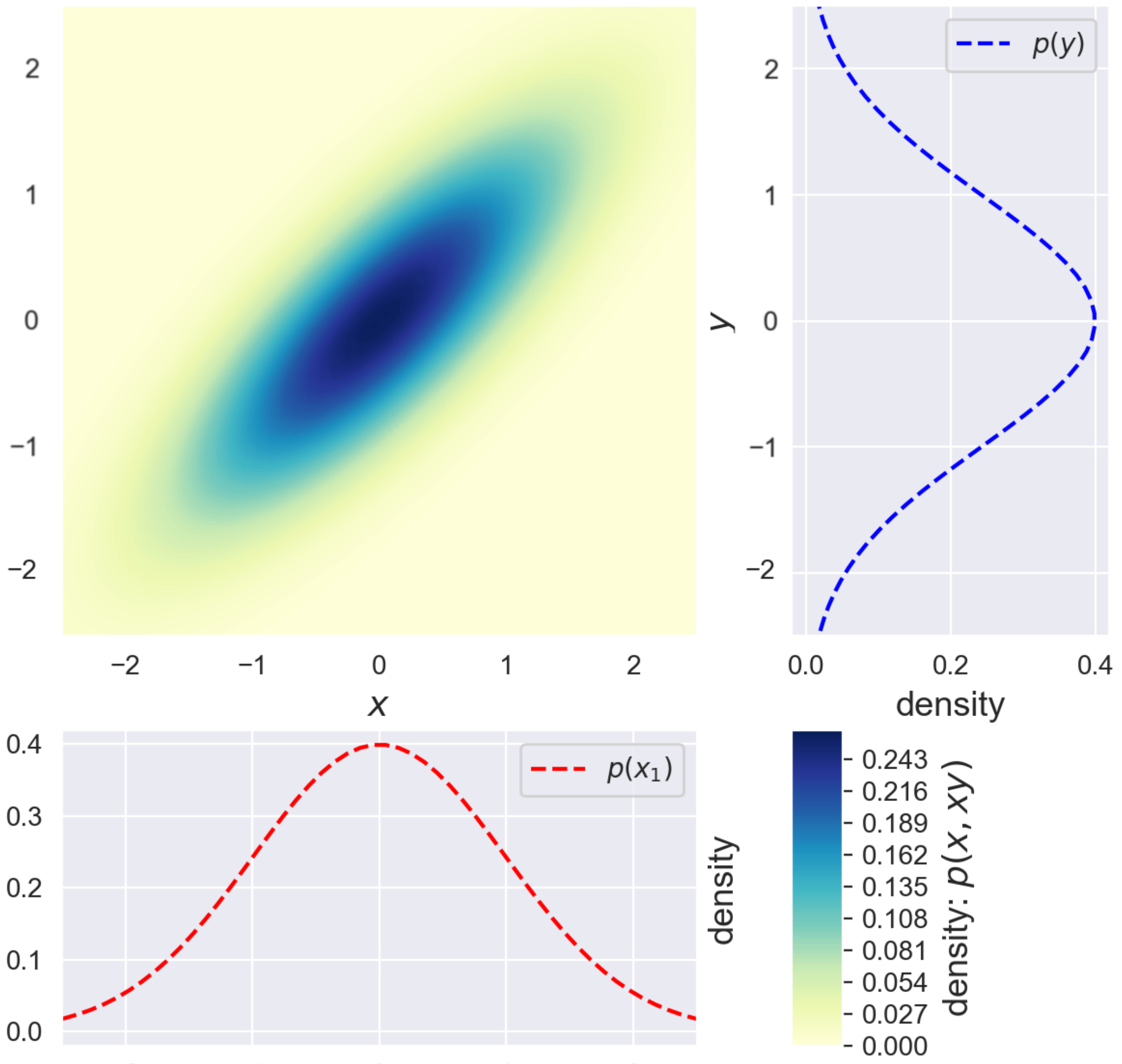
Multiple regression terms

- If you have multiple regression terms and want different priors for each, you can supply a vector for the `location` and `scale` arguments (where I've just made up the AM_PM variable):

```
stan_glmer(Reaction ~ Days + AM_PM + (Days | Subject),
           data = sleepstudy,
           prior = normal(
             location = c(0, 2),
             scale = c(2.5, 5)
           )
)
```

The covariance prior

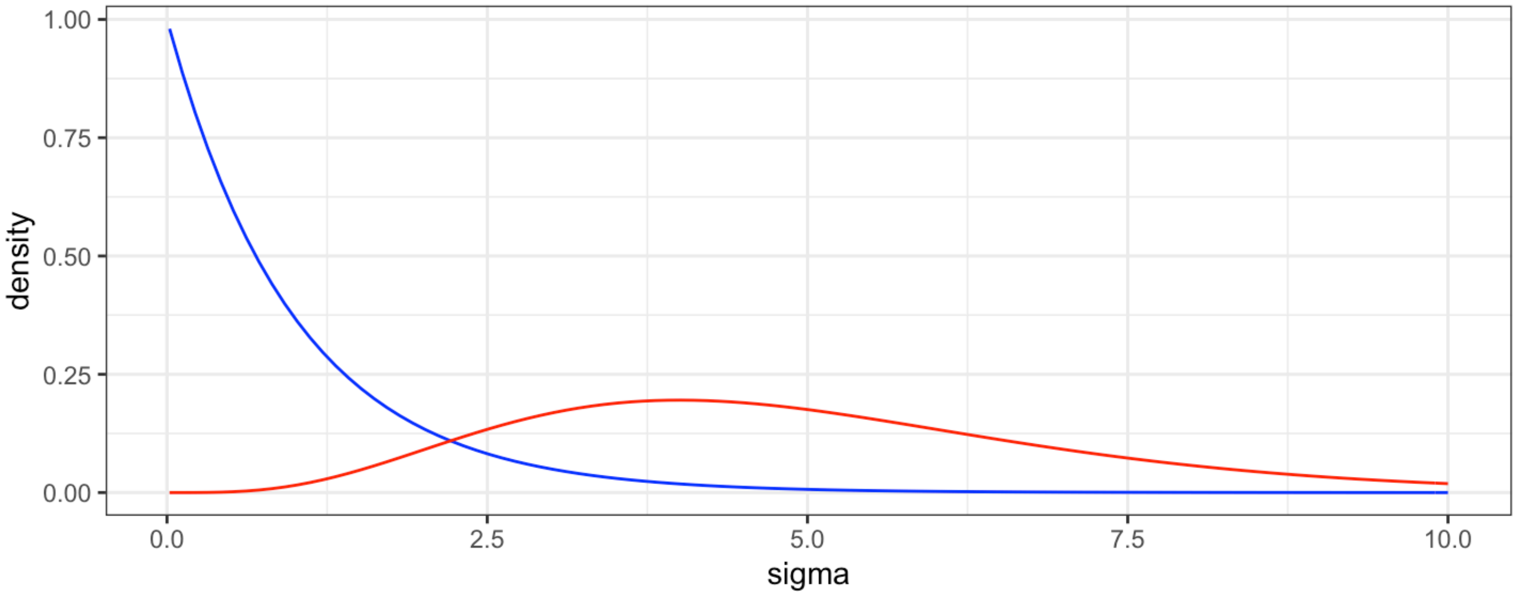
- The prior for related to any group-specific parameters is a bit complicated.
- Think of the varying slopes and intercepts as a random vector.
    - We assume that the distribution of that random vector is multivariate normal
    - with mean 0 (that is, the 0 vector)
    - and unknown covariance matrix that will need to be estimated.


2
1
0
−1
−2
−2
−1
0
1
2
x
p(y)
y
2
1
0
−1
−2
0.0
0.2
0.4
density
0.4
0.3
0.2
0.1
0.0
$p(x_1)$
−2
−1
0
1
2
density
0.243
0.216
0.189
0.162
0.135
0.108
0.081
0.054
0.027
0.000
density: $p(x, xy)$

The covariance prior cont.

- The default prior for that covariance matrix is `prior_covariance = decov(regularization = 1, concentration = 1, shape = 1, scale = 1)`
  - this is a combination of uniform distribution on correlation matrices and an exponential distribution on the the standard deviations.

- Hierarchical models sometimes have trouble converging. I have found that changing the covariance prior often helps: `prior_covariance = decov(shape = 5)`
  - The standard deviation of the group-level parameters might be getting too close to zero. This prior pulls it a bit away from 0.



Concluding remarks

- If you don't have a good sense of what prior to use for a particular analysis, stick with the defaults for `stan_glm()` and `stan_glmer()`
  - The defaults are good choices of weakly informative priors
- If you do have particular priors you want to use, use them. You can leave everything else as the default.
- Changing `pprior_covariance`, even as little as to
  - `prior_covariance = decov(shape = 2)`
  - can help with convergence
- Type `priors` into the help box for lots more choices or read https://cran.r-project.org/web/packages/rstanarm/vignettes/priors.html for more detail.