

```

# Load required packages
if (!requireNamespace("brms", quietly = TRUE)) {
  install.packages("brms")
}
library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.2.0      v readr      2.1.6
## v forcats    1.0.1      v stringr   1.6.0
## v ggplot2    4.0.2      v tibble    3.3.1
## v lubridate  1.9.5      v tidyr     1.3.2
## v purrr      1.2.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(brms)

## Loading required package: Rcpp
## Loading 'brms' package (version 2.23.0). Useful instructions
## can be found by typing help('brms'). A more detailed introduction
## to the package is available through vignette('brms_overview').
##
## Attaching package: 'brms'
##
## The following object is masked from 'package:stats':
##
##   ar

library(rstan)

## Loading required package: StanHeaders
##
## rstan version 2.32.7 (Stan version 2.32.2)
##
## 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)
##
##
## Attaching package: 'rstan'
##
## The following object is masked from 'package:tidyr':
##
##   extract

```

```
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
library(brms)
library(bayesplot)
```

```
## This is bayesplot version 1.15.0
## - Online documentation and vignettes at mc-stan.org/bayesplot
## - bayesplot theme set to bayesplot::theme_default()
##   * Does _not_ affect other ggplot2 plots
##   * See ?bayesplot_theme_set for details on theme setting
##
## Attaching package: 'bayesplot'
##
## The following object is masked from 'package:brms':
##
##     rhat
```

```
library(cmdstanr)
```

```
## This is cmdstanr version 0.8.0
## - CmdStanR documentation and vignettes: mc-stan.org/cmdstanr
## - Use set_cmdstan_path() to set the path to CmdStan
## - Use install_cmdstan() to install CmdStan
```

```
library(lme4)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Attaching package: 'lme4'
##
## The following object is masked from 'package:brms':
##
##     ngrps
```

```
library(lmerTest)
```

```
##
## Attaching package: 'lmerTest'
##
## The following object is masked from 'package:lme4':
##
##     lmer
##
## The following object is masked from 'package:stats':
##
##     step
```

```
library(rstan)
library(posterior)
```

```
## This is posterior version 1.6.1
##
## Attaching package: 'posterior'
##
## The following object is masked from 'package:bayesplot':
##
##     rhat
##
## The following objects are masked from 'package:rstan':
##
##     ess_bulk, ess_tail
##
## The following objects are masked from 'package:stats':
##
##     mad, sd, var
##
## The following objects are masked from 'package:base':
##
##     %in%, match
```

```
library(performance)
library(sjPlot)
```

```
##
## Attaching package: 'sjPlot'
##
## The following object is masked from 'package:ggplot2':
##
##     set_theme
```

```
df_p1 <- read.csv("/Users/debarpita/Desktop/arjun/trial_wise_dataset_post.csv")
df_p2 <- read.csv("/Users/debarpita/Desktop/arjun/trial_wise_dataset_pre.csv")
library(dplyr)
df <- bind_rows(df_p1, df_p2)
```

```
summary(df$HandlingTime)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  1.968   2.456   2.737   3.721   3.480  30.424
```

Turkey IQR outlier rule - removes extremely high outliers.

```
Q1 <- quantile(df$HandlingTime, 0.25, na.rm = TRUE)
Q3 <- quantile(df$HandlingTime, 0.75, na.rm = TRUE)
IQR <- Q3 - Q1
df <- df %>%
  filter(HandlingTime >= Q1 - 1.5 * IQR,
         HandlingTime <= Q3 + 1.5 * IQR)
```

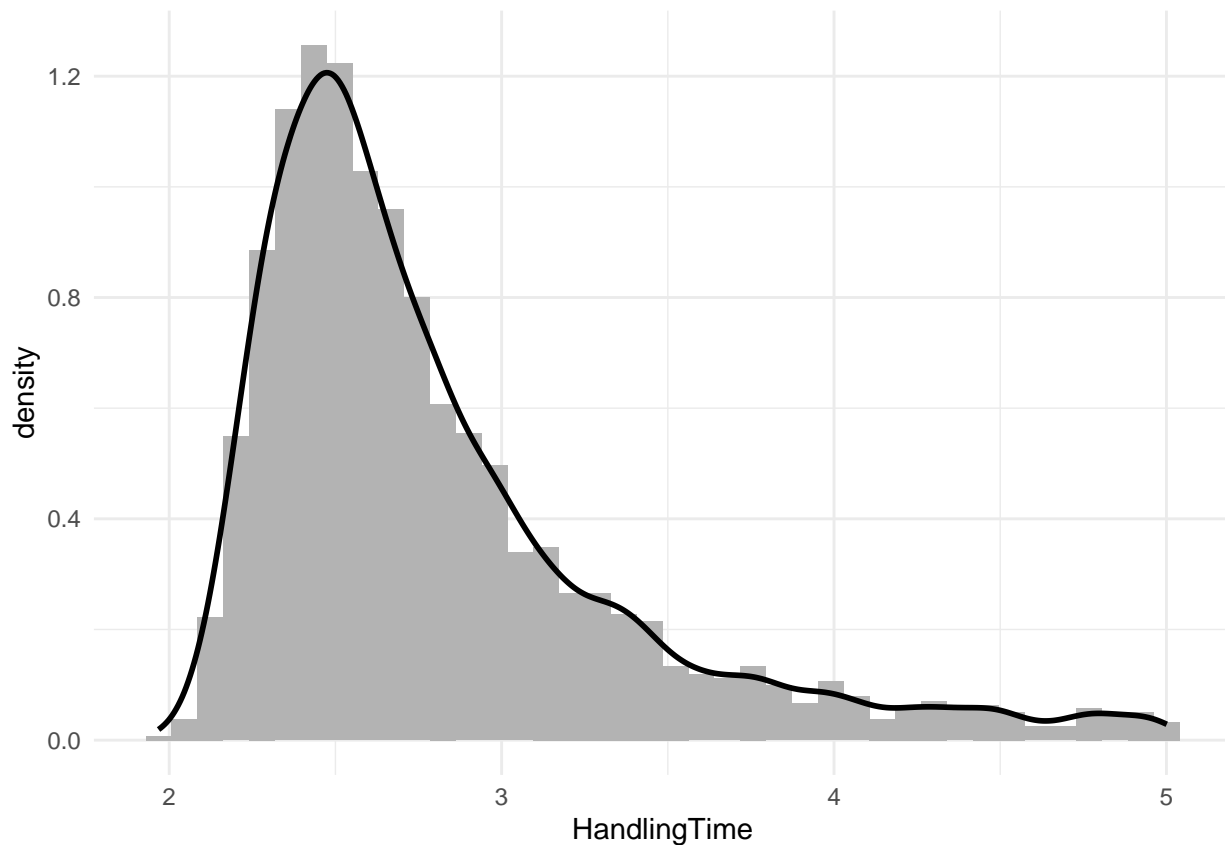
```
df <- df %>%
  mutate(HandlingTime_safe = HandlingTime + 1e-6)
df <- df %>%
  mutate(across(c(Reward, trait_anxiety_score, TotalCumulativeReward),
    list(z = ~scale(.)[,1])))
```

```
library(ggplot2)
```

```
ggplot(df, aes(x = HandlingTime)) +
  geom_histogram(aes(y = ..density..), bins = 40, fill = "grey70") +
  geom_density(size = 1) +
  theme_minimal()
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once per session.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

```
## Warning: The dot-dot notation ('..density..') was deprecated in ggplot2 3.4.0.
## i Please use 'after_stat(density)' instead.
## This warning is displayed once per session.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```



Each participant has a different baseline handling time but the effect of stimulation is assumed to be the same for everyone

```
model_gamma <- brm(  
  HandlingTime_safe ~ stim + (1|Participant_ID),  
  data = df,  
  family = Gamma(link = "log"),  
  chains = 4,  
  iter = 3000,  
  warmup = 1000,  
  cores = 4  
)
```

```
## Compiling Stan program...
```

```
## Start sampling
```

```
model_student <- brm(  
  HandlingTime_safe ~ stim + (1|Participant_ID),  
  data = df,  
  family = student(link = "log"),  
  chains = 4,  
  iter = 3000,  
  warmup = 1000,  
  cores = 4  
)
```

```
## Compiling Stan program...
```

```
## Start sampling
```

```
## 4:              70.353 seconds (Total)  
## Chain 4:
```

```
model_lognorm <- brm(  
  HandlingTime_safe ~ stim + (1|Participant_ID),  
  data = df,  
  family = lognormal(),  
  chains = 4,  
  iter = 3000,  
  warmup = 1000,  
  cores = 4  
)
```

```
## Compiling Stan program...
```

```
## Start sampling
```

```
##              70.353 seconds (Total)  
## Chain 4:
```

For interactions in Bayesian models, z-scoring is strongly recommended as it makes priors behave properly and improves convergence

```
df <- df %>%
  mutate(across(c(Reward, trait_anxiety_score,
                  TotalCumulativeReward),
              list(z = ~scale(.)[,1])))
```

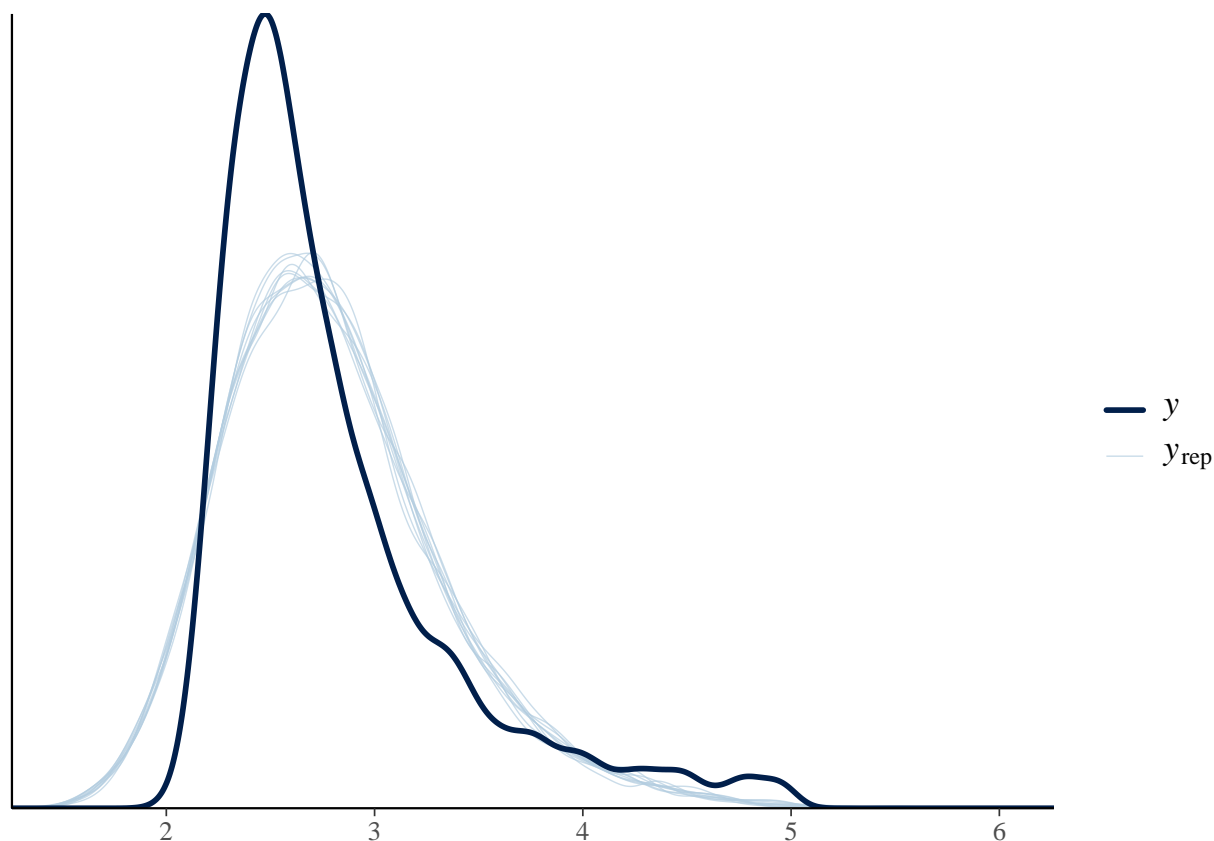
```
summary(model_gamma)
```

```
## Family: gamma
## Links: mu = log
## Formula: HandlingTime_safe ~ stim + (1 | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)    0.13     0.02    0.10    0.19 1.01      680     1119
##
## Regression Coefficients:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept     1.10     0.03    1.04    1.16 1.00      551     1021
## stim         -0.09     0.00   -0.10   -0.09 1.00     3390     3692
##
## Further Distributional Parameters:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## shape    49.36     0.97   47.49   51.29 1.00     2670     3110
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

sd(Intercept) = 0.14 between-person variability in baseline handling time. $\exp(0.14) = 1.15$ this implies that a typical participant is about 15% faster or slower than the average person High variability - people are very different from each other Estimate(Intercept) = 1.10 -> $\exp(1.10) = 3.00$ - The average handling time before stimulation. confidence interval doesn't include 0 which is why it is credible stim = -0.09 -> $\exp(-0.09) = 0.914$ Handling time is reduced by ~8.6% under stimulation. CI narrow which means model is confident about above effect and it is consistent across trials Shape > 5: implies normal Rhat is 1.00 so converged Bulk_ESS -> middle of the distribution Tail_ESS -> For the extreme values ess values greater than 400 so of no concern stim error is within participant difference which is much smaller than intercept which is in between participants

```
pp_check(model_gamma)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



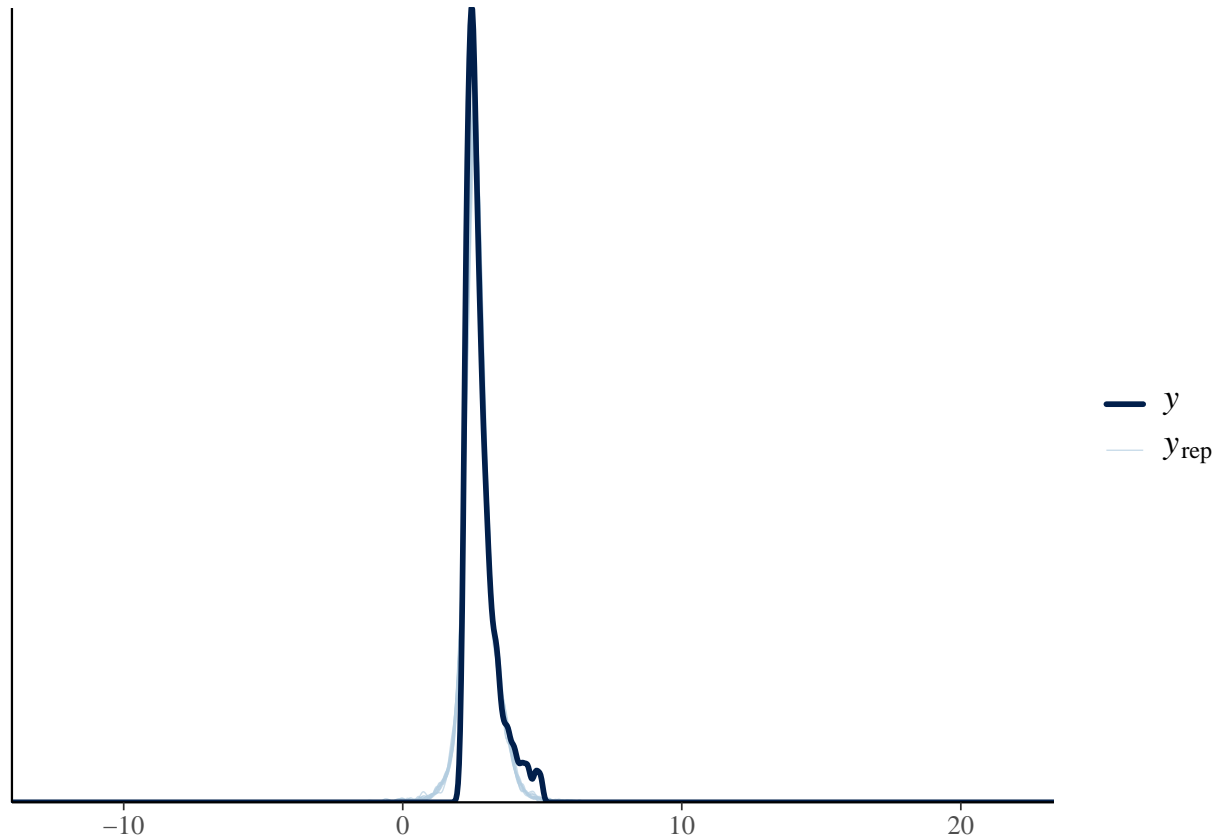
```
summary(model_student)
```

```
## Family: student
## Links: mu = log
## Formula: HandlingTime_safe ~ stim + (1 | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.13 0.02 0.10 0.19 1.00 784 1169
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 1.04 0.03 0.99 1.10 1.01 709 998
## stim -0.07 0.00 -0.07 -0.06 1.00 3805 4048
##
## Further Distributional Parameters:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma 0.23 0.00 0.22 0.24 1.00 2481 3625
## nu 2.15 0.08 2.00 2.32 1.00 2505 3344
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
```

```
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_student)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



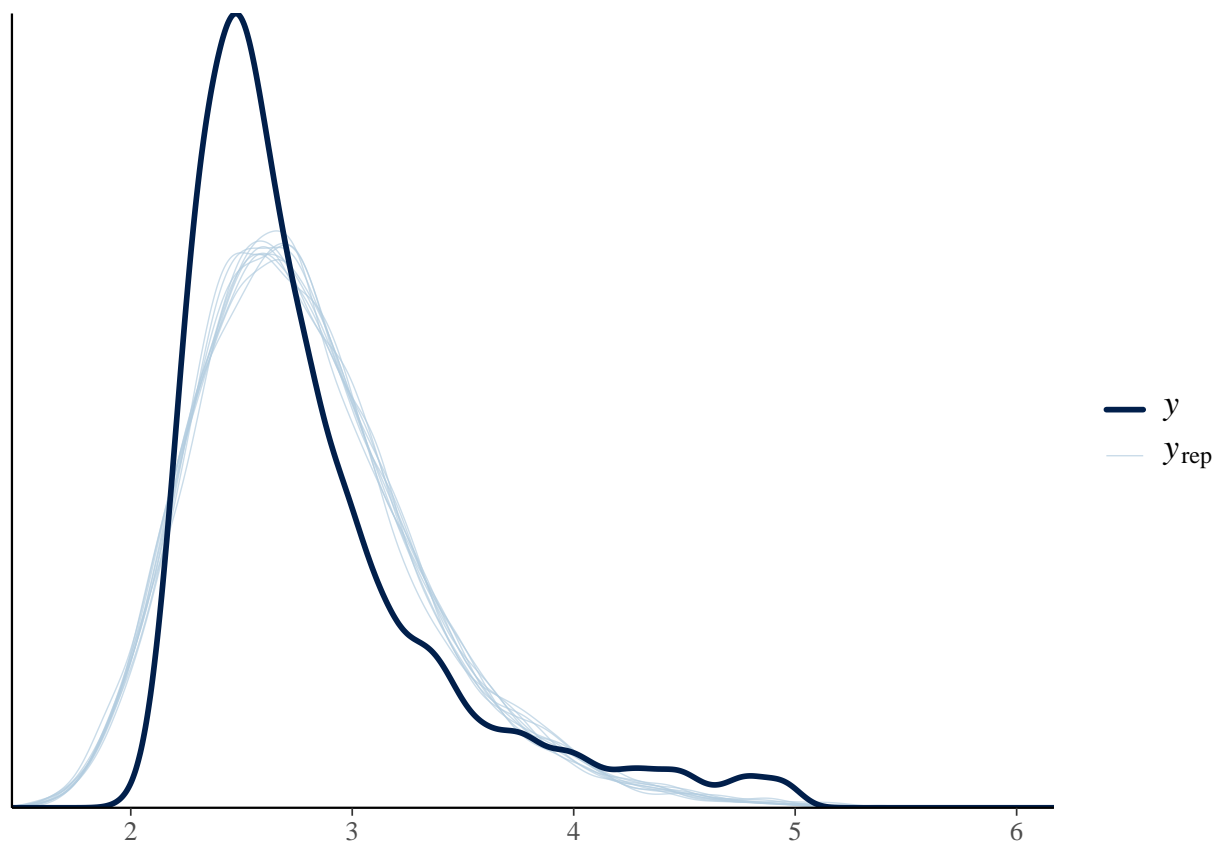
```
summary(model_lognorm)
```

```
## Family: lognormal
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim + (1 | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.13 0.02 0.10 0.18 1.01 636 1210
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 1.09 0.03 1.03 1.14 1.02 551 805
```

```
## stim      -0.09      0.00     -0.10     -0.08 1.00      3230      3545
##
## Further Distributional Parameters:
##      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma    0.14      0.00      0.14      0.14 1.00      2169      2684
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_lognorm)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
library(loo)
```

```
## This is loo version 2.9.0
```

```
## - Online documentation and vignettes at mc-stan.org/loo
```

```
## - As of v2.0.0 loo defaults to 1 core but we recommend using as many as possible. Use the 'cores' arg
```

```
##
```

```
## Attaching package: 'loo'
```

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

```
# Compute LOO for each model
loo_gamma    <- loo(model_gamma)
loo_student  <- loo(model_student)
loo_lognorm  <- loo(model_lognorm)

# Compare models
loo_compare(loo_gamma, loo_student, loo_lognorm)
```

```
##              elpd_diff se_diff
## model_student    0.0      0.0
## model_lognorm  -41.6     39.8
## model_gamma    -201.7    45.9
```

elpd_diff (Expected Log Predictive Density Difference) -> higher is better se_diff (Standard Error of Difference) -> uncertainty in the comparison. If $|\text{elpd_diff}| > 2 \times \text{se_diff} \rightarrow$ strong evidence

Student and lognorm are clearly close competitors with student being the better model.

Each participant has their own baseline and their own stimulation effect. Lines are no longer parallel.

```
model_lognorm_own <- brm(
  HandlingTime_safe ~ stim + (1 + stim | Participant_ID),
  data = df,
  family = lognormal(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## Start sampling
```

```
summary(model_lognorm_own)
```

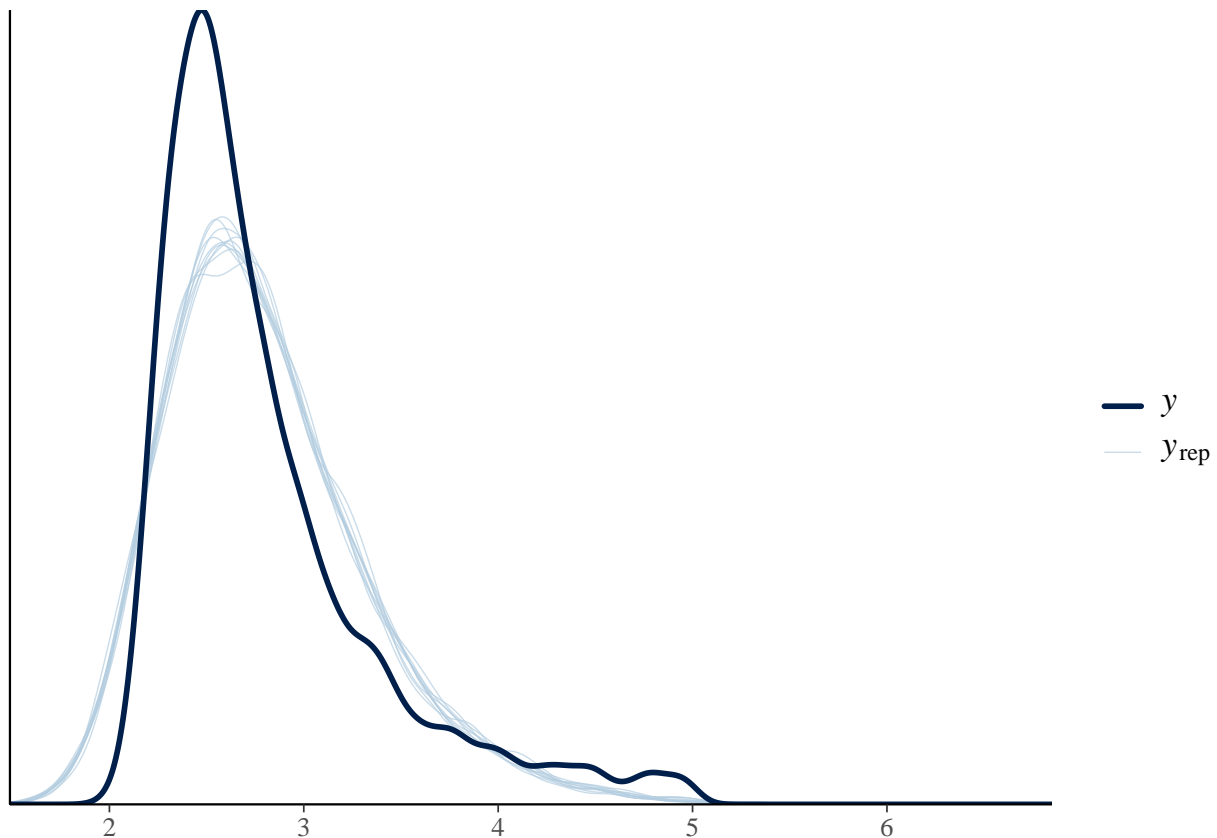
```
## Family: lognormal
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim + (1 + stim | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##
```

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sd(Intercept)	0.14	0.03	0.10	0.20	1.00	1501	2708
sd(stim)	0.07	0.01	0.05	0.10	1.00	1729	3292

```
## cor(Intercept,stim)    -0.23      0.21    -0.61      0.22 1.00      2250      3477
##
## Regression Coefficients:
##      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      1.09      0.03      1.02      1.15 1.00      1022      2006
## stim          -0.09      0.02     -0.12     -0.06 1.00      1924      3364
##
## Further Distributional Parameters:
##      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.14      0.00      0.13      0.14 1.00      9427      5738
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_lognorm_own)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
model_student_own <- brm(
  HandlingTime_safe ~ stim + (1 + stim | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
```

```

warmup = 1000,
cores = 4
)

## Compiling Stan program...

## Start sampling

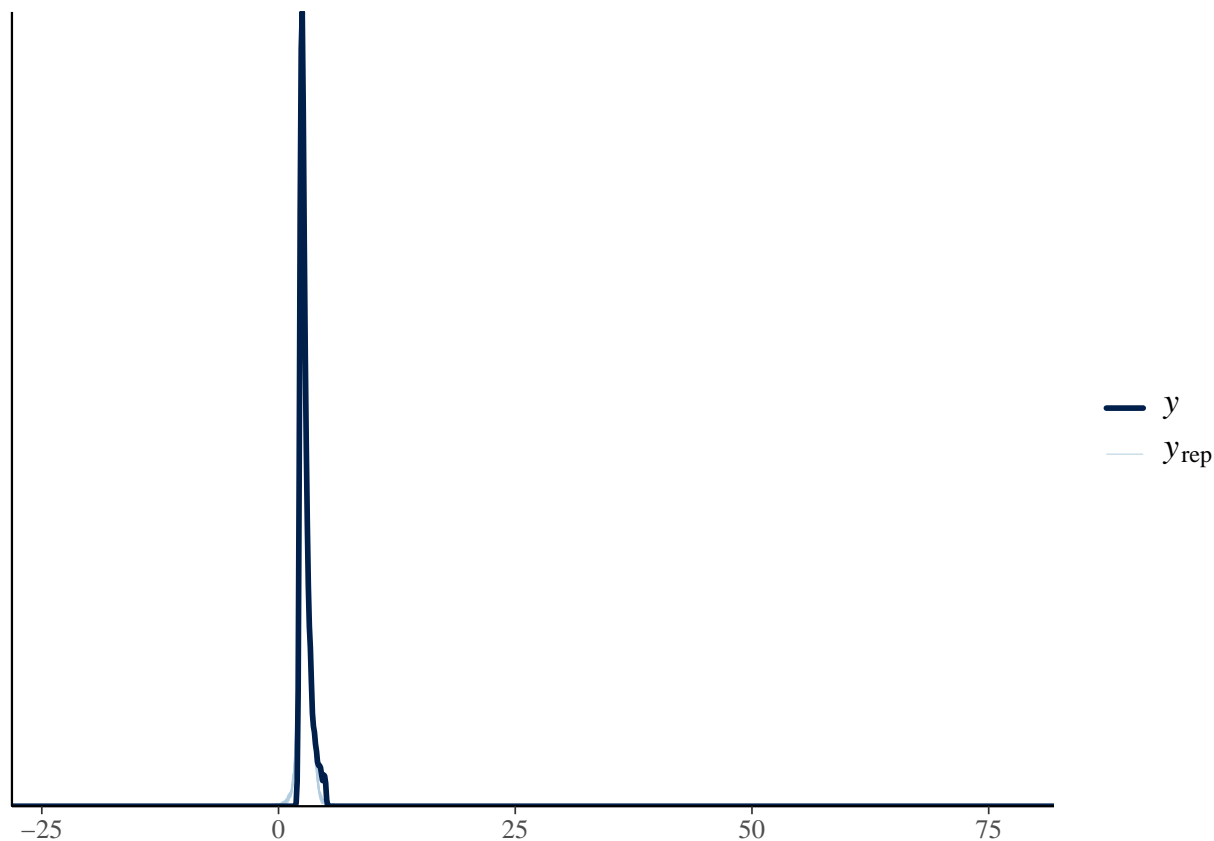
summary(model_student_own)

## Family: student
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim + (1 + stim | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      0.44      0.08      0.32      0.63 1.00      1712      2959
## sd(stim)            0.17      0.03      0.12      0.25 1.00      1996      3609
## cor(Intercept,stim) -0.37      0.20     -0.72      0.07 1.00      2810      3828
##
## Regression Coefficients:
##
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      2.89      0.10      2.70      3.08 1.00      1288      1962
## stim          -0.21      0.04     -0.29     -0.12 1.00      1785      2832
##
## Further Distributional Parameters:
##
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.22      0.00      0.21      0.23 1.00      5843      5248
## nu          2.14      0.08      1.99      2.31 1.00      5525      6082
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

pp_check(model_student_own)

## Using 10 posterior draws for ppc type 'dens_overlay' by default.

```



```
model_reward_log <- brm(
  HandlingTime_safe ~ Reward_z + (1 | Participant_ID),
  data = df,
  family = lognormal(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be biased
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

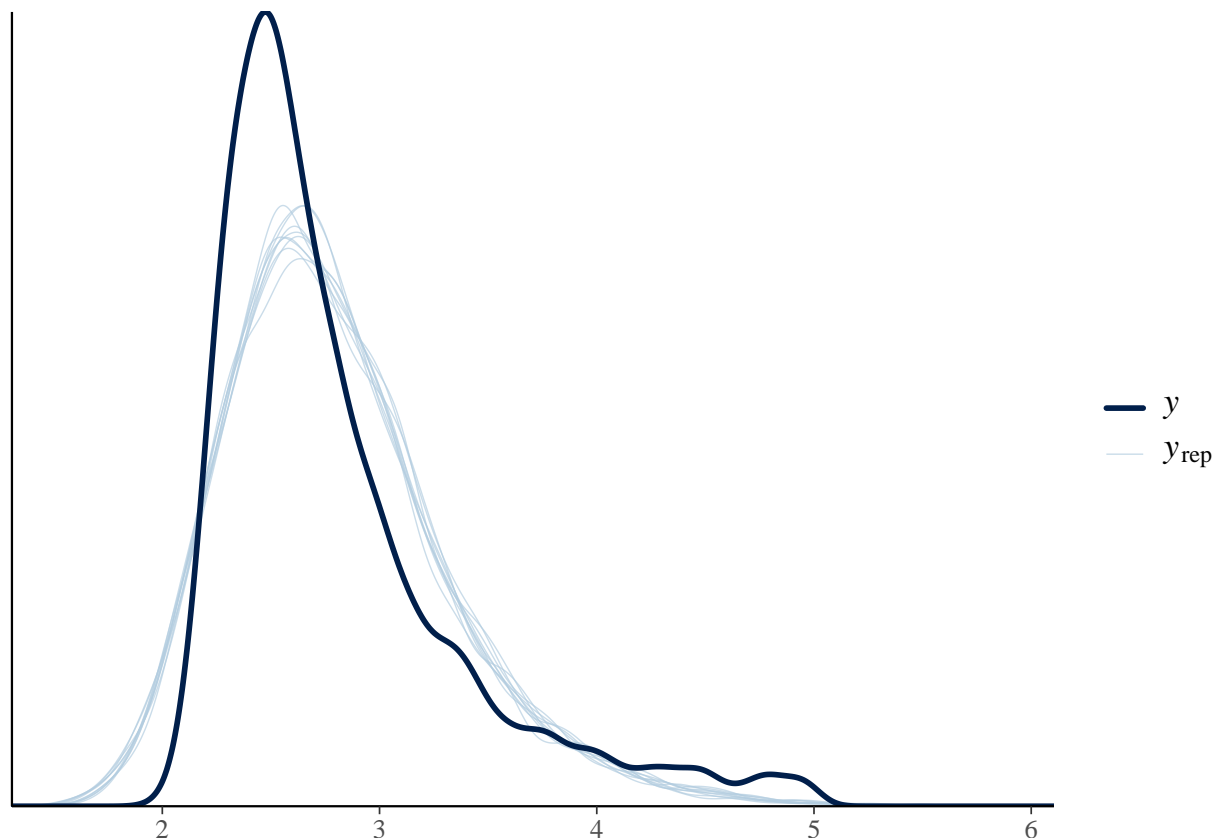
```
summary(model_reward_log)
```

```
## Family: lognormal
## Links: mu = identity
```

```
## Formula: HandlingTime_safe ~ Reward_z + (1 | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.13 0.02 0.10 0.19 1.01 732 1261
##
## Regression Coefficients:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 1.04 0.03 0.98 1.10 1.01 400 659
## Reward_z -0.03 0.00 -0.03 -0.02 1.00 5395 5038
##
## Further Distributional Parameters:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma 0.14 0.00 0.14 0.15 1.00 2604 3232
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_reward_log)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```

model_reward_student <- brm(
  HandlingTime_safe ~ Reward_z + (1 | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)

```

```
## Compiling Stan program...
```

```
## Start sampling
```

```
summary(model_reward_student)
```

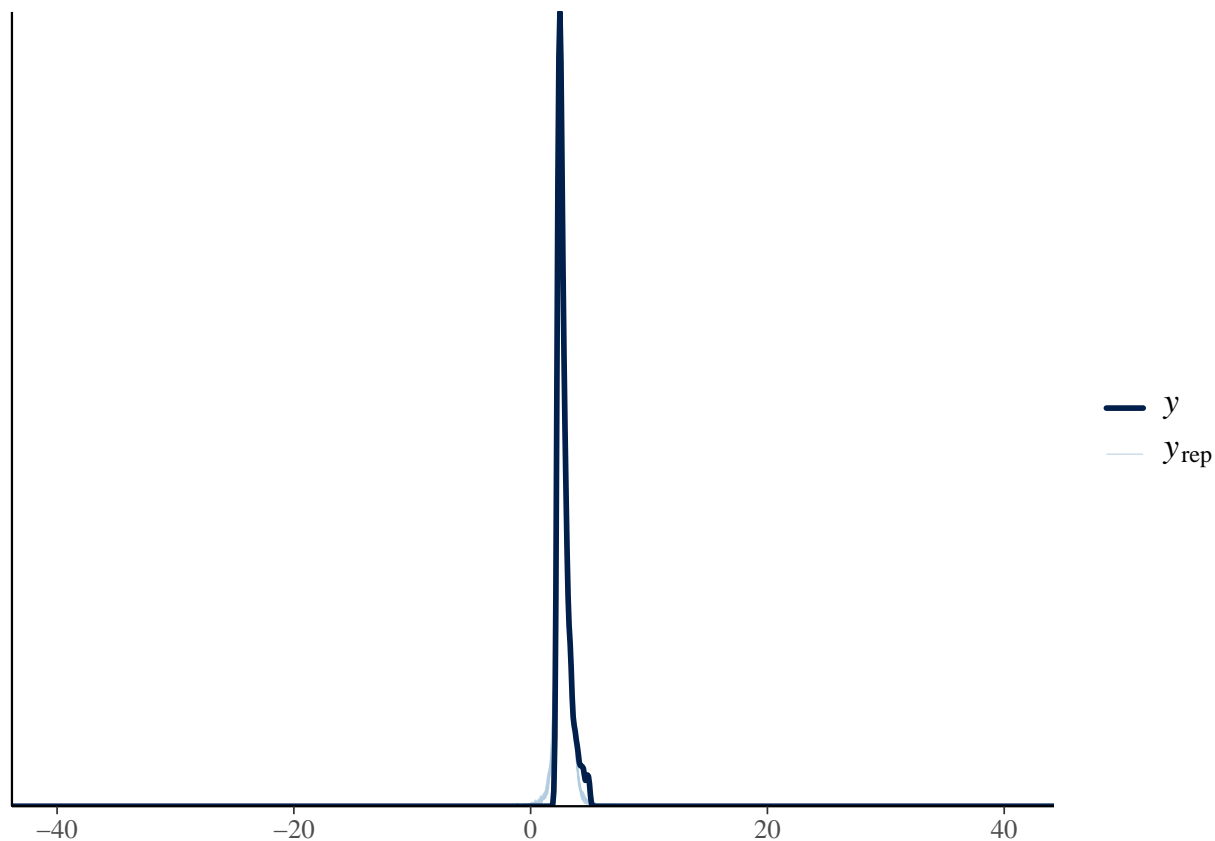
```

## Family: student
## Links: mu = identity
## Formula: HandlingTime_safe ~ Reward_z + (1 | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)    0.39     0.07    0.28    0.56 1.01      869    1578
##
## Regression Coefficients:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      2.76     0.09    2.58    2.93 1.00      677     892
## Reward_z      -0.06     0.00   -0.07   -0.05 1.00     3925    3734
##
## Further Distributional Parameters:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma    0.23     0.00    0.22    0.24 1.00     2570    3233
## nu       2.11     0.08    1.96    2.28 1.00     2512    3206
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```

```
pp_check(model_reward_student)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
model_stim_anx <- brm(
  HandlingTime_safe ~ stim * trait_anxiety_score_z + Reward_z +
    (1 + stim | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

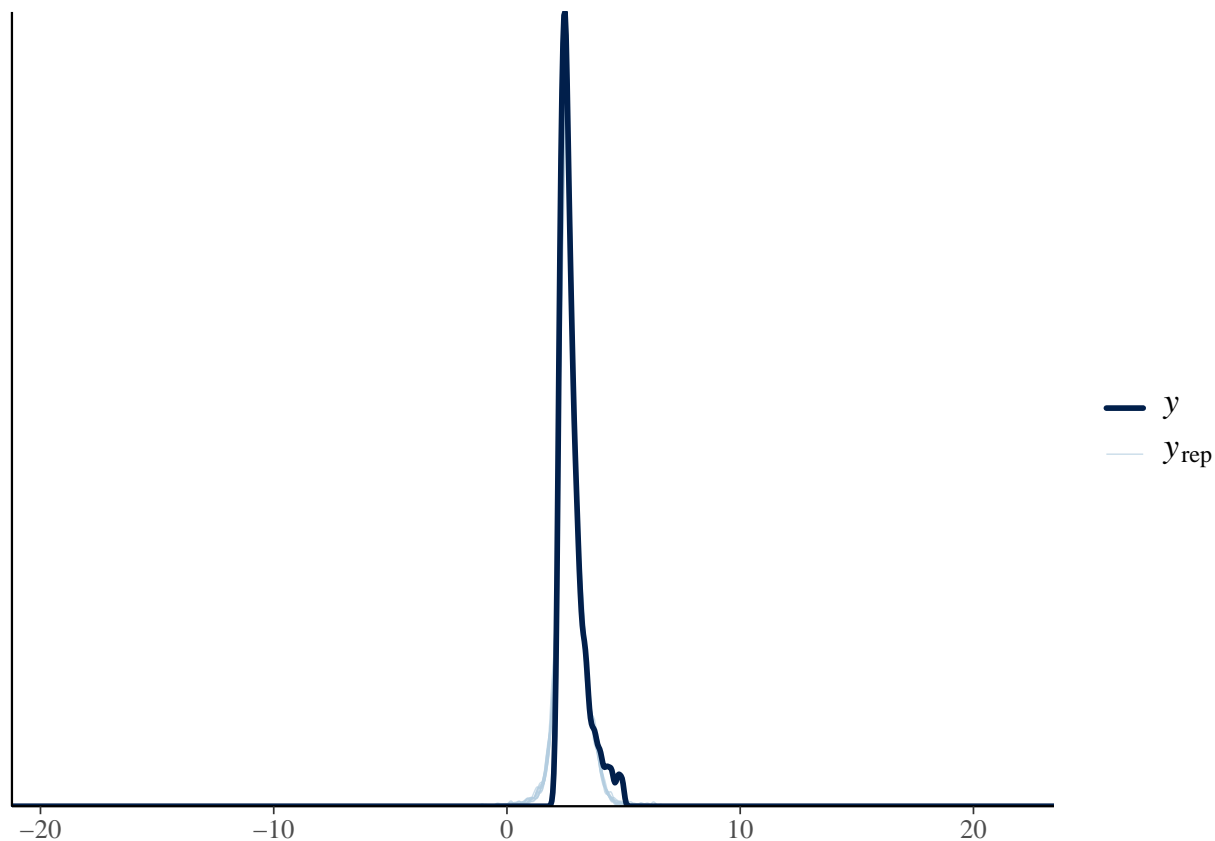
```
summary(model_stim_anx)
```

```
## Family: student
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim * trait_anxiety_score_z + Reward_z + (1 + stim | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
```

```
##          total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##          Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      0.43    0.08    0.31    0.61 1.00    2459    3860
## sd(stim)           0.18    0.03    0.12    0.26 1.00    2706    4548
## cor(Intercept,stim) -0.35    0.21   -0.71    0.10 1.00    4177    4246
##
## Regression Coefficients:
##          Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## Intercept          2.87    0.10    2.68    3.06 1.00    2467
## stim              -0.19    0.04   -0.27   -0.10 1.00    3501
## trait_anxiety_score_z -0.14    0.10   -0.33    0.05 1.00    3221
## Reward_z          -0.02    0.00   -0.03   -0.01 1.00    8408
## stim:trait_anxiety_score_z 0.03    0.04   -0.05    0.11 1.00    4255
##
##          Tail_ESS
## Intercept          3224
## stim              4360
## trait_anxiety_score_z 3724
## Reward_z          5736
## stim:trait_anxiety_score_z 4662
##
## Further Distributional Parameters:
##          Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.22    0.00    0.21    0.23 1.00    7581    5916
## nu         2.13    0.09    1.97    2.30 1.00    7147    5803
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_stim_anx)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
model_stim_anx_log <- brm(
  HandlingTime_safe ~ stim * trait_anxiety_score_z + Reward_z +
    (1 + stim | Participant_ID),
  data = df,
  family = lognormal,
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

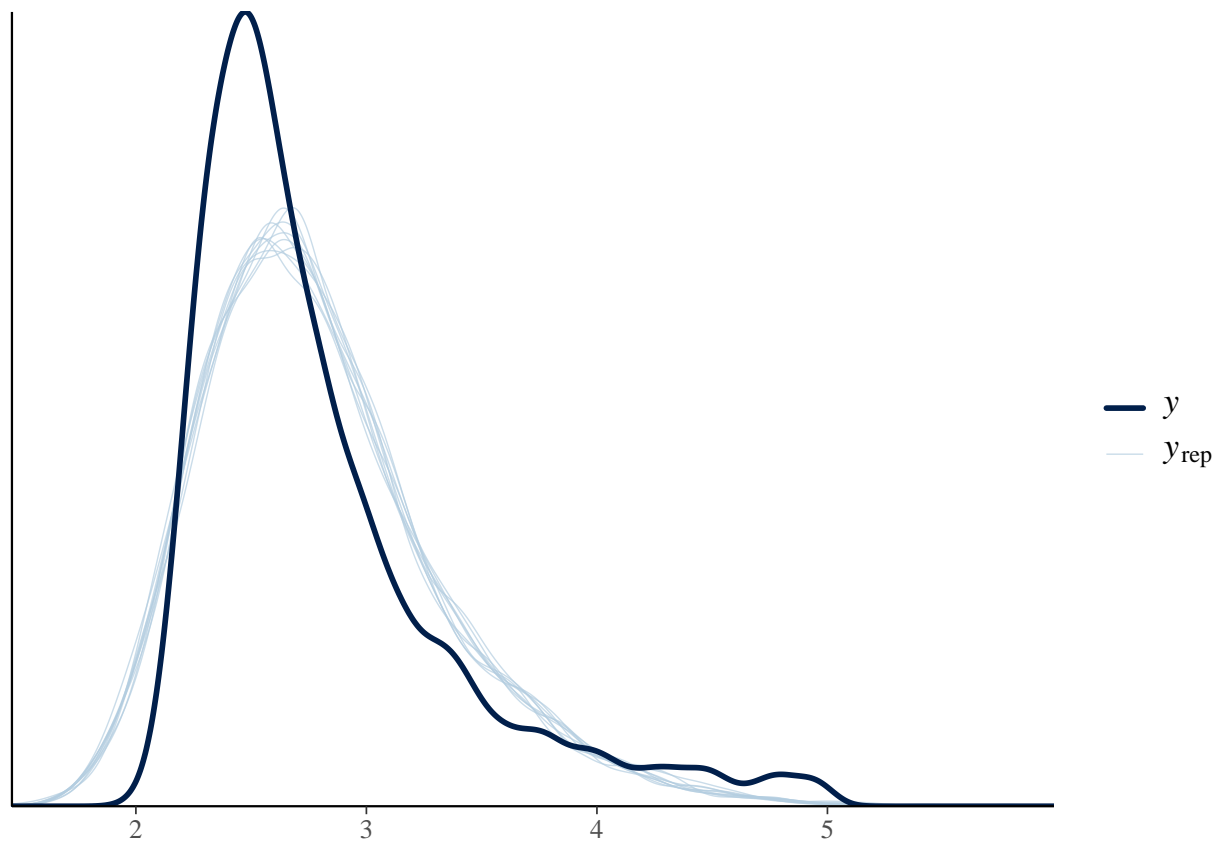
```
summary(model_stim_anx_log)
```

```
## Family: lognormal
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim * trait_anxiety_score_z + Reward_z + (1 + stim | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
```

```
##           total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      0.14      0.02    0.10    0.20 1.00     2376     4071
## sd(stim)            0.07      0.01    0.05    0.10 1.00     2588     3727
## cor(Intercept,stim) -0.21      0.23   -0.60    0.26 1.00     3363     4445
##
## Regression Coefficients:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## Intercept          1.09      0.03    1.02    1.15 1.00     2145
## stim               -0.09      0.02   -0.12   -0.05 1.00     3278
## trait_anxiety_score_z -0.04      0.03   -0.10    0.02 1.00     2781
## Reward_z           -0.00      0.00   -0.01    0.00 1.00    11439
## stim:trait_anxiety_score_z 0.01      0.02   -0.02    0.04 1.00     3407
##           Tail_ESS
## Intercept          2818
## stim              4057
## trait_anxiety_score_z 3584
## Reward_z          5367
## stim:trait_anxiety_score_z 4561
##
## Further Distributional Parameters:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.14      0.00    0.13    0.14 1.00     9724     5326
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_stim_anx_log)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
df <- df %>%
  group_by(Participant_ID, stim_cat, env, patch_id) %>%
  mutate(trial_in_patch = row_number()) %>%
  ungroup()
```

```
model_stim_cum <- brm(
  HandlingTime_safe ~ stim * TotalCumulativeReward_z +
    (1 + stim | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

```
summary(model_stim_cum)
```

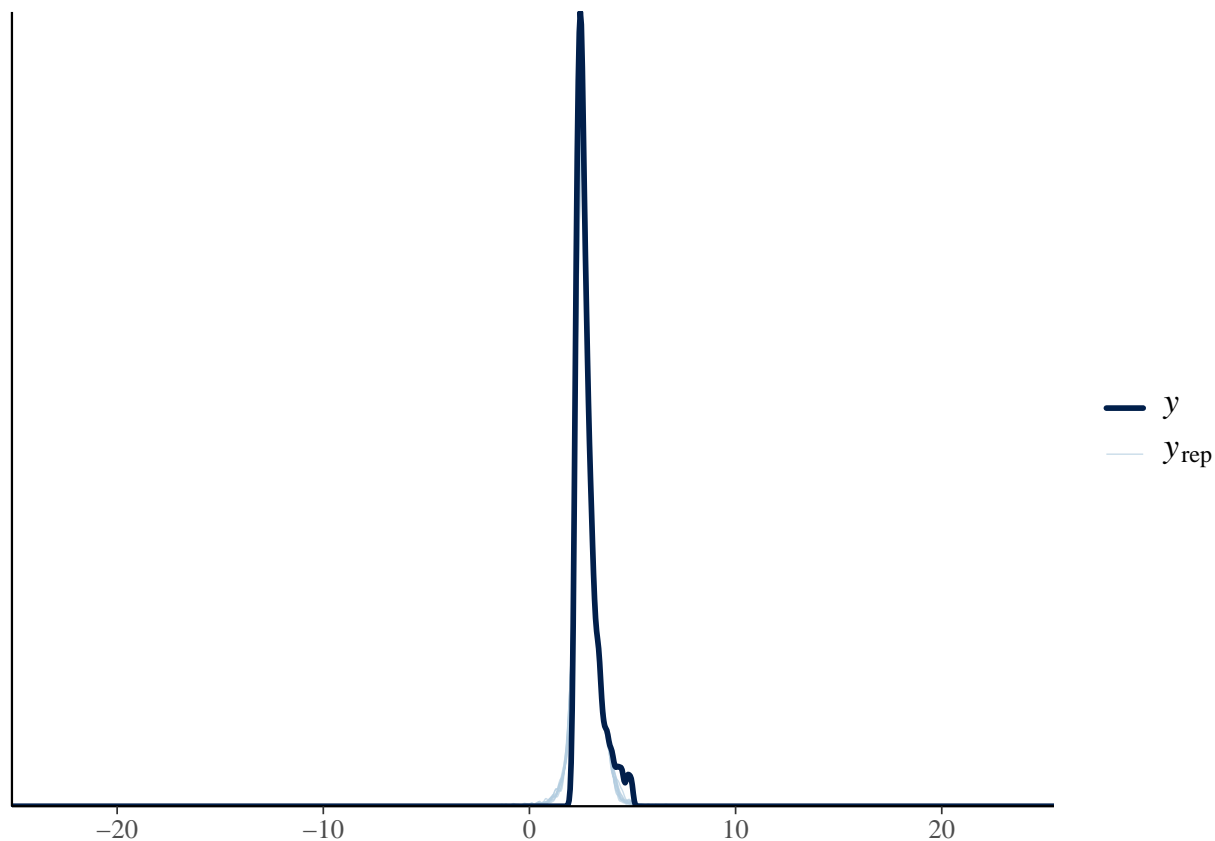
```

## Family: student
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim * TotalCumulativeReward_z + (1 + stim | Participant_ID)
## Data: df (Number of observations: 5079)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      0.43    0.08    0.31    0.61 1.00    1613    2802
## sd(stim)           0.17    0.03    0.12    0.24 1.00    1745    3475
## cor(Intercept,stim) -0.37    0.20   -0.72    0.06 1.00    2954    3867
##
## Regression Coefficients:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS
## Intercept      2.83    0.10    2.65    3.03 1.00     956
## stim           -0.13    0.04   -0.21   -0.06 1.00    2149
## TotalCumulativeReward_z -0.10    0.01   -0.12   -0.07 1.00    5457
## stim:TotalCumulativeReward_z 0.06    0.01    0.03    0.08 1.00    5651
##
##      Tail_ESS
## Intercept    1631
## stim         3175
## TotalCumulativeReward_z 5402
## stim:TotalCumulativeReward_z 5344
##
## Further Distributional Parameters:
##      Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.22    0.00    0.21    0.23 1.00     6152     5548
## nu         2.17    0.09    2.01    2.35 1.00     5814     5064
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```

```
pp_check(model_stim_cum)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
df <- df %>%
  group_by(Participant_ID, stim_cat, env) %>%
  mutate(
    AvgRewardRate_before = lag(AvgRewardRate)
  ) %>%
  ungroup()
```

```
df <- df %>%
  filter(!is.na(AvgRewardRate_before))
```

```
df <- df %>%
  mutate(
    trial_in_patch_z = scale(trial_in_patch)[,1],
    patch_id_z = scale(patch_id)[,1],
    AvgRewardRate_before_z = scale(AvgRewardRate_before)[,1]
  )
```

```
model_L1 <- brm(
  HandlingTime_safe ~ stim_cat * Reward_z +
    trial_in_patch_z +
    patch_id_z +
    (1 + stim_cat | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
```

```

iter = 3000,
warmup = 1000,
cores = 4
)

```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

```
summary(model_L1)
```

```
## Family: student
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim_cat * Reward_z + trial_in_patch_z + patch_id_z + (1 + stim_cat | Participant_ID)
## Data: df (Number of observations: 4995)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##
```

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS
sd(Intercept)	0.39	0.07	0.29	0.55	1.00	1665
sd(stim_catpre)	0.19	0.04	0.13	0.27	1.00	1918
cor(Intercept,stim_catpre)	0.01	0.24	-0.45	0.47	1.00	2257

```
##
```

	Tail_ESS
sd(Intercept)	2889
sd(stim_catpre)	3370
cor(Intercept,stim_catpre)	3490

```
##
```

Regression Coefficients:

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS
Intercept	2.69	0.09	2.52	2.87	1.01	944
stim_catpre	0.17	0.04	0.09	0.26	1.00	1874
Reward_z	-0.02	0.01	-0.04	-0.01	1.00	5277
trial_in_patch_z	-0.02	0.01	-0.03	-0.00	1.00	5449
patch_id_z	-0.07	0.01	-0.08	-0.06	1.00	7265
stim_catpre:Reward_z	-0.04	0.01	-0.07	-0.02	1.00	6836

```
##
```

	Tail_ESS
Intercept	1560
stim_catpre	3101
Reward_z	5419
trial_in_patch_z	5370
patch_id_z	6360
stim_catpre:Reward_z	5321

```
##
```

Further Distributional Parameters:

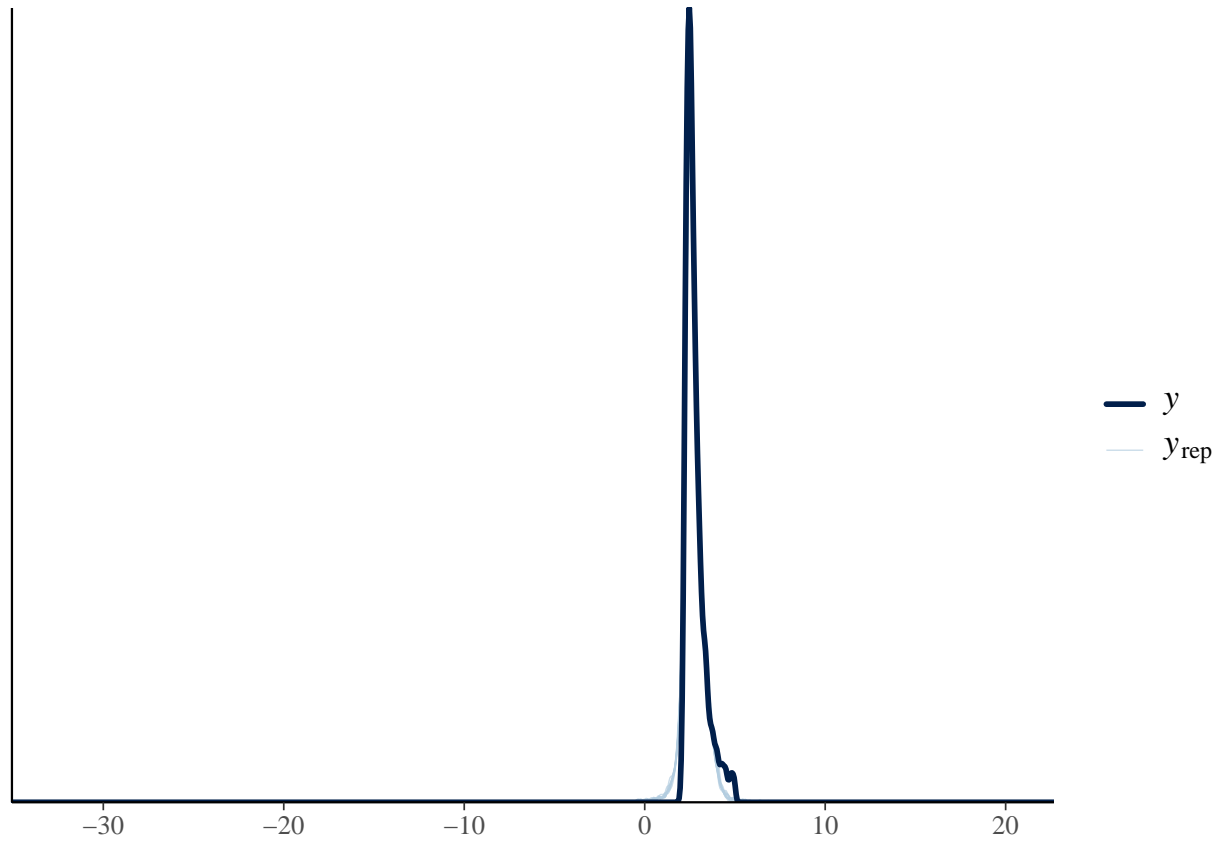
	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sigma	0.21	0.00	0.20	0.22	1.00	5181	5522
nu	2.16	0.09	1.99	2.34	1.00	5306	4874

```
##
```

```
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
pp_check(model_L1)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
model_L2 <- brm(
  HandlingTime_safe ~ stim_cat * Reward_z +
    trial_in_patch_z +
    patch_id_z +
    AvgRewardRate_before_z +
    (1 + stim_cat | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

```
summary(model_L2)
```

```
## Family: student
## Links: mu = identity
## Formula: HandlingTime_safe ~ stim_cat * Reward_z + trial_in_patch_z + patch_id_z + AvgRewardRate_bef
## Data: df (Number of observations: 4995)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##
```

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS
## sd(Intercept)	0.38	0.07	0.28	0.55	1.00	1596
## sd(stim_catpre)	0.18	0.04	0.12	0.26	1.00	1650
## cor(Intercept,stim_catpre)	0.04	0.23	-0.42	0.47	1.00	2152

```
##
```

	Tail_ESS
## sd(Intercept)	2398
## sd(stim_catpre)	3450
## cor(Intercept,stim_catpre)	3795

```
##
```

Regression Coefficients:

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS
## Intercept	2.70	0.09	2.53	2.87	1.00	1070
## stim_catpre	0.14	0.04	0.05	0.22	1.00	2003
## Reward_z	-0.02	0.01	-0.03	-0.01	1.00	5688
## trial_in_patch_z	-0.01	0.01	-0.02	0.00	1.00	5455
## patch_id_z	-0.07	0.01	-0.08	-0.06	1.00	6589
## AvgRewardRate_before_z	-0.03	0.01	-0.04	-0.01	1.00	6223
## stim_catpre:Reward_z	-0.04	0.01	-0.07	-0.02	1.00	7685

```
##
```

	Tail_ESS
## Intercept	1836
## stim_catpre	3371
## Reward_z	5381
## trial_in_patch_z	5600
## patch_id_z	6128
## AvgRewardRate_before_z	5472
## stim_catpre:Reward_z	5178

```
##
```

Further Distributional Parameters:

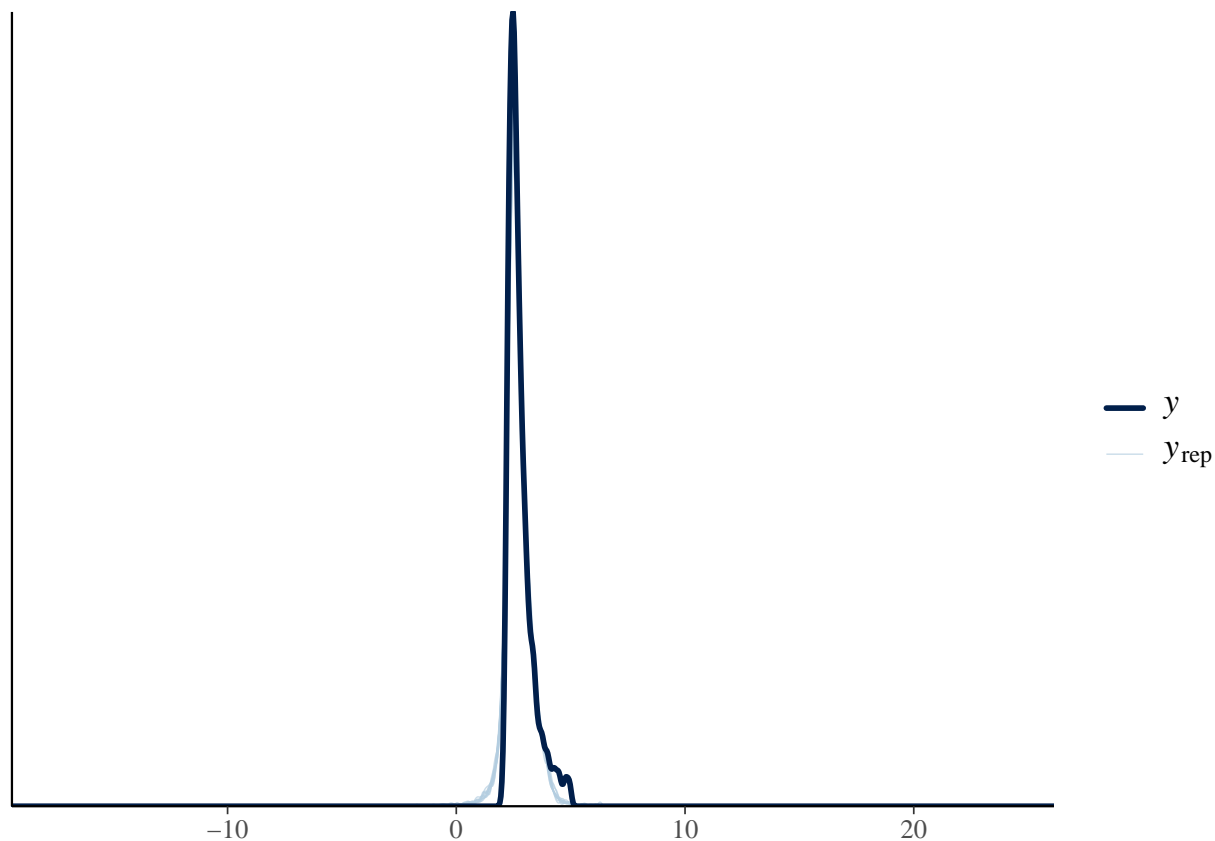
	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
## sigma	0.21	0.00	0.20	0.22	1.00	6409	5716
## nu	2.17	0.09	2.00	2.34	1.00	6283	5527

```
##
```

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).

```
pp_check(model_L2)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
model_L3 <- brm(
  HandlingTime_safe ~ stim_cat * AvgRewardRate_before_z +
    trial_in_patch_z +
    patch_id_z +
    (1 + stim_cat | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

```
summary(model_L3)
```

```
## Family: student
```

```
## Links: mu = identity
```

```
## Formula: HandlingTime_safe ~ stim_cat * AvgRewardRate_before_z + trial_in_patch_z + patch_id_z + (1 + stim_cat | Participant_ID)
```

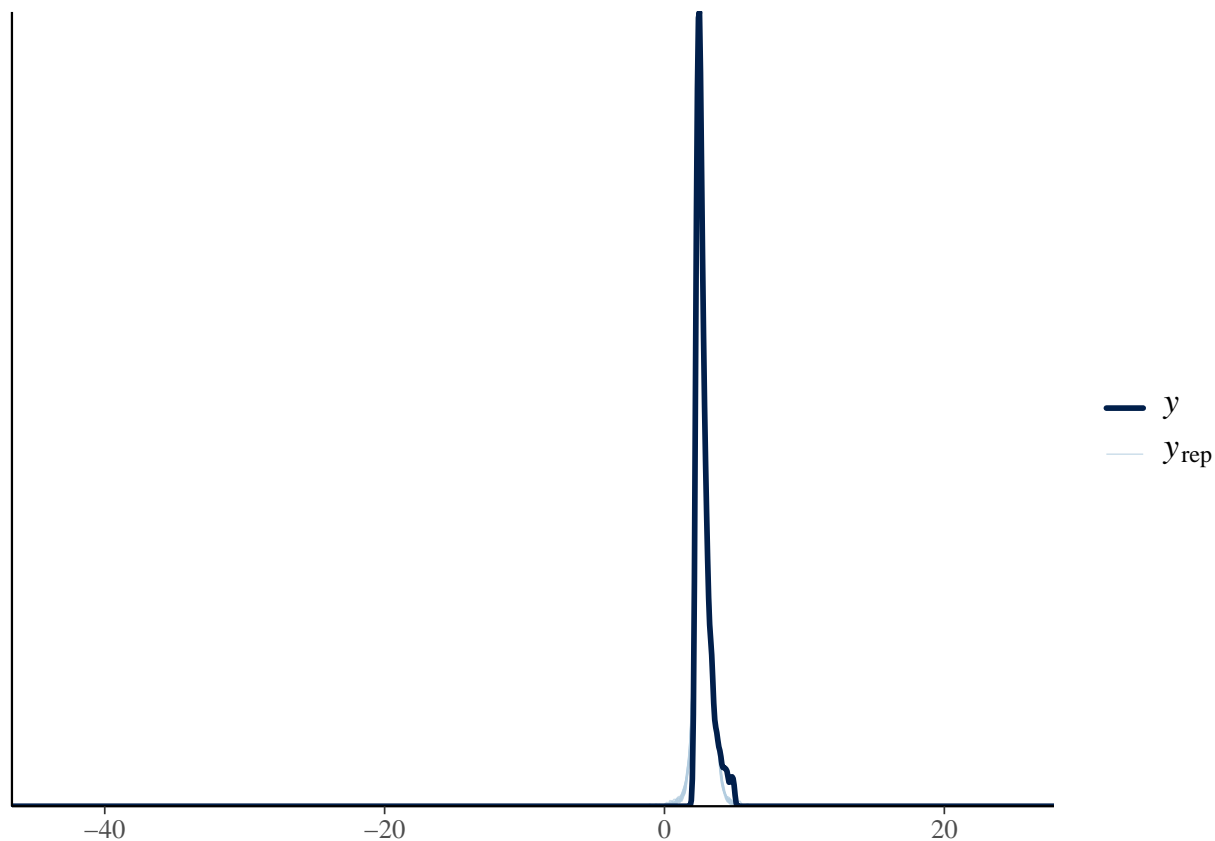
```

## Data: df (Number of observations: 4995)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## sd(Intercept) 0.40 0.07 0.29 0.56 1.00 1695
## sd(stim_catpre) 0.15 0.03 0.10 0.22 1.00 1659
## cor(Intercept,stim_catpre) -0.23 0.23 -0.63 0.25 1.00 2557
##
## Tail_ESS
## sd(Intercept) 2689
## sd(stim_catpre) 3152
## cor(Intercept,stim_catpre) 3809
##
## Regression Coefficients:
##
## Estimate Est.Error l-95% CI u-95% CI Rhat
## Intercept 2.67 0.09 2.49 2.84 1.00
## stim_catpre -0.06 0.05 -0.16 0.04 1.00
## AvgRewardRate_before_z -0.01 0.01 -0.03 0.01 1.00
## trial_in_patch_z 0.01 0.00 0.00 0.02 1.00
## patch_id_z -0.09 0.01 -0.10 -0.08 1.00
## stim_catpre:AvgRewardRate_before_z -0.30 0.04 -0.37 -0.23 1.00
##
## Bulk_ESS Tail_ESS
## Intercept 907 1434
## stim_catpre 2464 3731
## AvgRewardRate_before_z 6562 6074
## trial_in_patch_z 10036 6030
## patch_id_z 6091 5988
## stim_catpre:AvgRewardRate_before_z 4578 4975
##
## Further Distributional Parameters:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma 0.21 0.00 0.21 0.22 1.00 5526 5864
## nu 2.21 0.09 2.04 2.40 1.00 5831 5823
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```

```
pp_check(model_L3)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
model_L4 <- brm(
  HandlingTime_safe ~ stim_cat * AvgRewardRate_before_z +
    trial_in_patch_z +
    patch_id_z +
    trait_anxiety_score_z +
    (1 + stim_cat | Participant_ID),
  data = df,
  family = student(),
  chains = 4,
  iter = 3000,
  warmup = 1000,
  cores = 4
)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Start sampling
```

```
summary(model_L4)
```

```
## Family: student
## Links: mu = identity
```

```

## Formula: HandlingTime_safe ~ stim_cat * AvgRewardRate_before_z + trial_in_patch_z + patch_id_z + tra
## Data: df (Number of observations: 4995)
## Draws: 4 chains, each with iter = 3000; warmup = 1000; thin = 1;
## total post-warmup draws = 8000
##
## Multilevel Hyperparameters:
## ~Participant_ID (Number of levels: 21)
##
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## sd(Intercept) 0.39 0.07 0.28 0.55 1.00 1841
## sd(stim_catpre) 0.15 0.03 0.10 0.23 1.00 1845
## cor(Intercept,stim_catpre) -0.30 0.22 -0.67 0.18 1.00 3324
##
## Tail_ESS
## sd(Intercept) 2825
## sd(stim_catpre) 3503
## cor(Intercept,stim_catpre) 3966
##
## Regression Coefficients:
##
## Estimate Est.Error l-95% CI u-95% CI Rhat
## Intercept 2.67 0.08 2.49 2.83 1.00
## stim_catpre -0.07 0.05 -0.17 0.03 1.00
## AvgRewardRate_before_z -0.01 0.01 -0.03 0.01 1.00
## trial_in_patch_z 0.01 0.00 0.00 0.02 1.00
## patch_id_z -0.09 0.01 -0.10 -0.08 1.00
## trait_anxiety_score_z -0.13 0.08 -0.29 0.03 1.01
## stim_catpre:AvgRewardRate_before_z -0.30 0.04 -0.37 -0.23 1.00
##
## Bulk_ESS Tail_ESS
## Intercept 1771 2376
## stim_catpre 3827 4893
## AvgRewardRate_before_z 6874 5419
## trial_in_patch_z 10080 6121
## patch_id_z 7984 6539
## trait_anxiety_score_z 1841 2890
## stim_catpre:AvgRewardRate_before_z 6270 5570
##
## Further Distributional Parameters:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma 0.21 0.00 0.21 0.22 1.00 7603 5817
## nu 2.21 0.09 2.03 2.39 1.00 7555 5652
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```

```
pp_check(model_L4)
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```

