Modeling workflow supplement: An evaluation of M2+ interference correction approaches associated with As and Se in ICP-MS using a multi-day dataset along with ICP-MS/MS/HR-ICP-MS based analysis and hierarchical modeling as a means of assessing bias in fortified drinking waters and single component matrices

January 12, 2022

Contents

Background

In this document, the results of model fitting, checking, and inferences are provided along with the code and workflow. The objectives of the modeling effort were to (1) estimate the effects of the experimental factors (i.e., matrix, day, and tune) on the accuracy and precision of using different internal standards (i.e., in-sample, alt.isotope, etc.) to predict the shifts in the M2+/M1+ factor. The accuracy of these predictions for ^{75}As ($^{150}Nd^{2+}$ and $^{150}Sm^{2+}$) and Se ($^{156}Gd^{2+}$) were determined using ICP-MS/MS/HR-ICP-MS (i.e., "True Value" or, henceforth, "TV") ; and (2) to predict out-of-sample observations in order to quantify expectations about the future performance of these internal standard methods.

This document begins with a section describing the preparation of the experimental data for use in the subsequent modeling. Then there is a section explaining the model's structure. That section is followed by one describing the priors used for each parameter in the model and the implied prior predictive distribution. The next sections then describe model fits, checks, and inferences.

Import and mung data

The data is imported from the local directory and stored as an object in R. All code is not shown in the html notebook, but is available in the R markdown document (i.e., .rmd file).

Print dataframes for both As and Se.

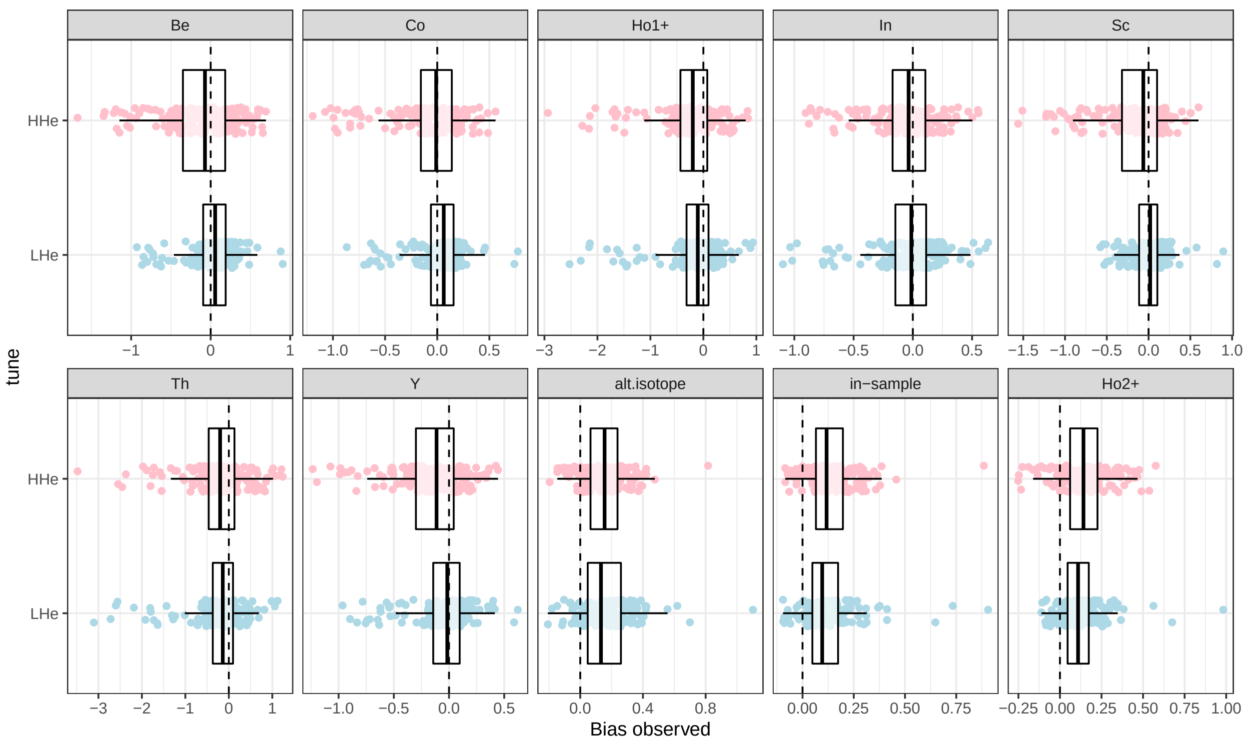
```
## # A tibble: 352 x 14
##    ider    matrix day_expt tune      Alt     Ho2      In     Std      Sc       Y
##    <chr>   <fct>  <fct>    <fct>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 water_~ water~ 316      LHe   -0.0553 -0.0113 -0.204  -0.0593 -0.262  -0.240
##  2 water_~ water~ 316      LHe   -0.047  -0.0199 -0.112  -0.0881 -0.169  -0.122
##  3 water_~ water~ 316      LHe   -0.104  -0.0816 -0.274  -0.0951 -0.367  -0.322
##  4 neat_1~ neat_1 316      LHe    0.121   0.213   0.0695  0.113   0.0889  0.0918
##  5 water_~ water~ 316      LHe    0.0596  0.151  -0.008   0.118  -0.0866 -0.0454
##  6 water_~ water~ 316      LHe    0.119   0.119   0.0491  0.0742  0.0207  0.0362
##  7 water_~ water~ 316      LHe    0.013   0.0321 -0.0858  0.0334 -0.0987 -0.102
##  8 water_~ water~ 316      LHe    0.0515  0.123  -0.0012  0.0909  0.0251  0.0012
##  9 water_~ water~ 316      LHe    0.0621  0.0341 -0.132   0.0738 -0.159  -0.173
## 10 neat_2~ neat_2 316      LHe    0.0723  0.146  -0.0164  0.116   0.058   0.0127
## # ... with 342 more rows, and 4 more variables: Be <dbl>, Co <dbl>, Th <dbl>,
## #   Ho1 <dbl>
```

```
## # A tibble: 352 x 14
##    ider         matrix day_expt tune     Alt    Ho2     In    Std      Sc      Y
##    <chr>        <fct>  <fct>    <fct>  <dbl>  <dbl>  <dbl>  <dbl>   <dbl>  <dbl>
##  1 water_1316L~ water~ 316      LHe   -1.68  -0.531 -3.04   0.387 -3.81   -3.51
##  2 water_3316L~ water~ 316      LHe   -1.09  -2.01  -3.19  -2.09  -3.93   -3.33
##  3 water_4316L~ water~ 316      LHe   -0.694  0.744 -1.75  -0.271 -2.95   -2.38
##  4 neat_1316LHe neat_1 316      LHe   -1.54  -0.408 -2.24  -1.90  -2.00   -1.96
##  5 water_5316L~ water~ 316      LHe   -0.940 -0.161 -2.19  -1.13  -3.19   -2.67
##  6 water_6316L~ water~ 316      LHe   -1.38  -0.707 -1.59  -1.00  -1.96   -1.76
##  7 water_7316L~ water~ 316      LHe   -1.01  -0.119 -1.62  -0.184 -1.78   -1.83
##  8 water_8316L~ water~ 316      LHe   -1.41  -0.132 -1.71   0.655 -1.38   -1.68
##  9 water_9316L~ water~ 316      LHe   -1.26  -0.178 -2.31   0.610 -2.65   -2.84
## 10 neat_2316LHe neat_2 316      LHe   -0.964  1.12  -0.927  0.339  0.0133 -0.559
## # ... with 342 more rows, and 4 more variables: Be <dbl>, Co <dbl>, Th <dbl>,
## #   Ho1 <dbl>
```

For each of As and Se, there were 352 observations of the difference between the IS (internal standards) corrected estimate and the "True Value" ("diff_std") for each of the 10 IS methods over 8 days, 20 matrices, and 2 tune settings. For clarification, the "day_expt" column is a concatenation of month and day corresponding to each of the 8 days of the experiment. The "tune" variable describes the tune setting (high helium = HHe or low helium = LHe) used for each observation. The "matrix" column is self-explanatory. The last 10 columns contain the results for each of the IS methods.

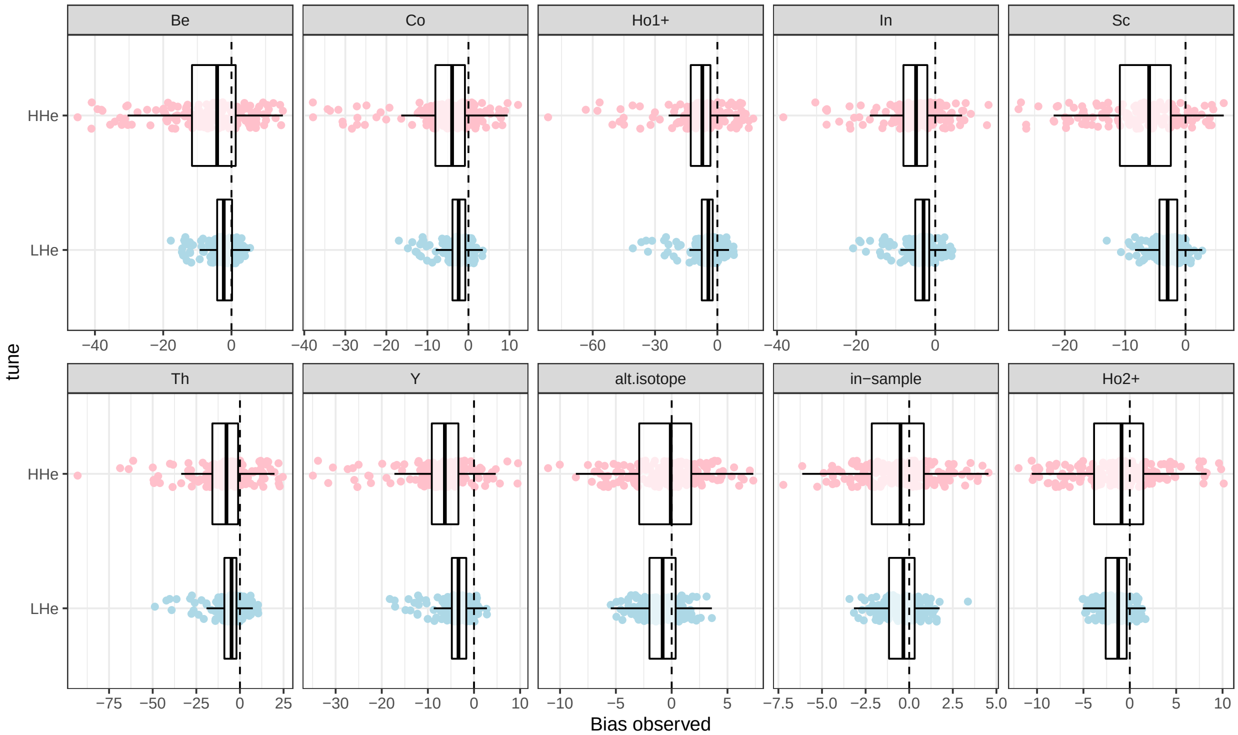
Visualize data

Below is a plot of observed bias (relative to "TV") by tune setting for each IS method for Arsenic.



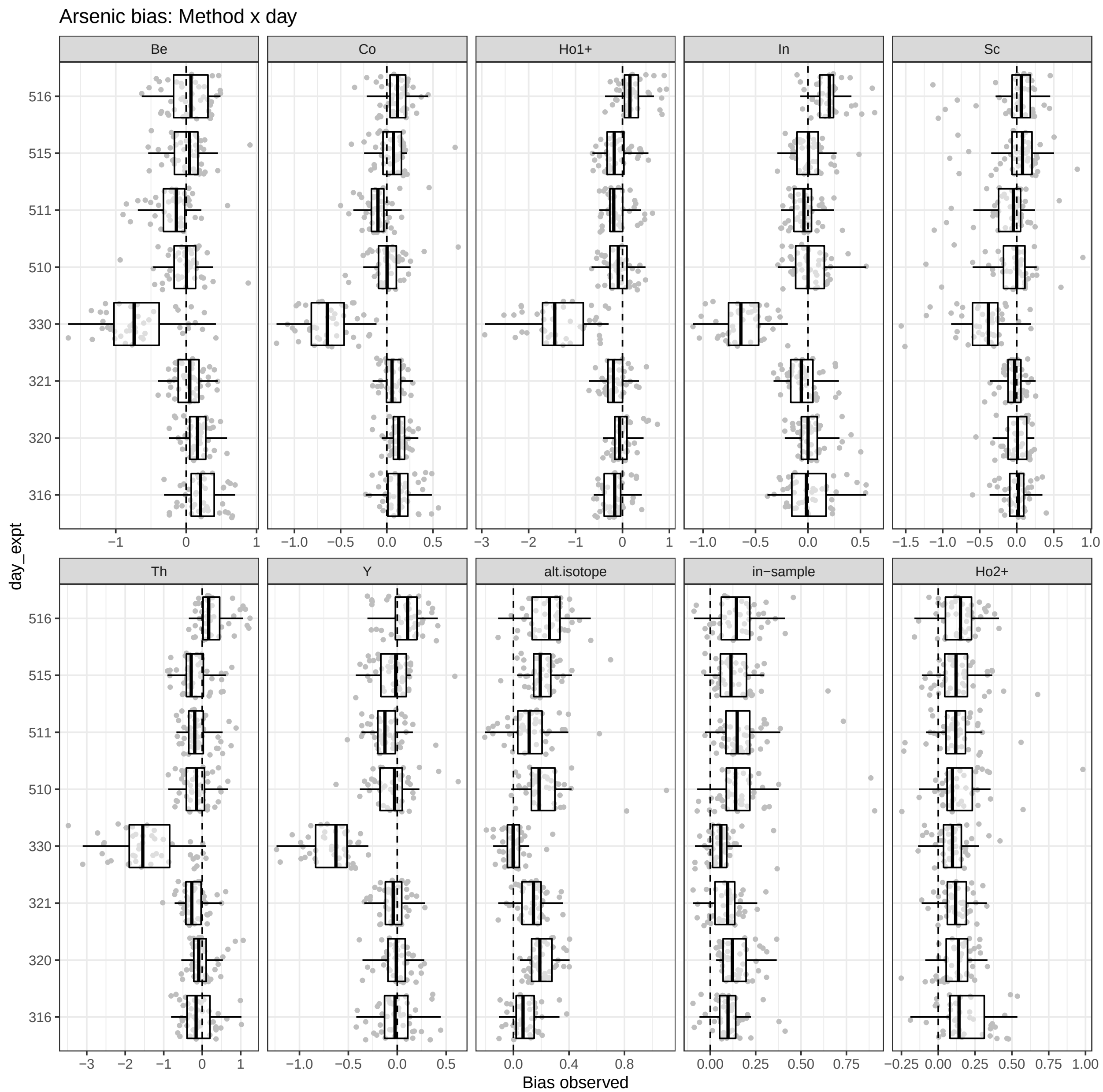
The above plot suggests slight under-corrections, on average, for all of the +2 internal standard methods. Most of the +1 methods look to have over-corrected or resulted in minimal bias, on average. Within method, there are no clear differences in this figure due to tune setting for any method.

Next, the same plot for the selenium observations.



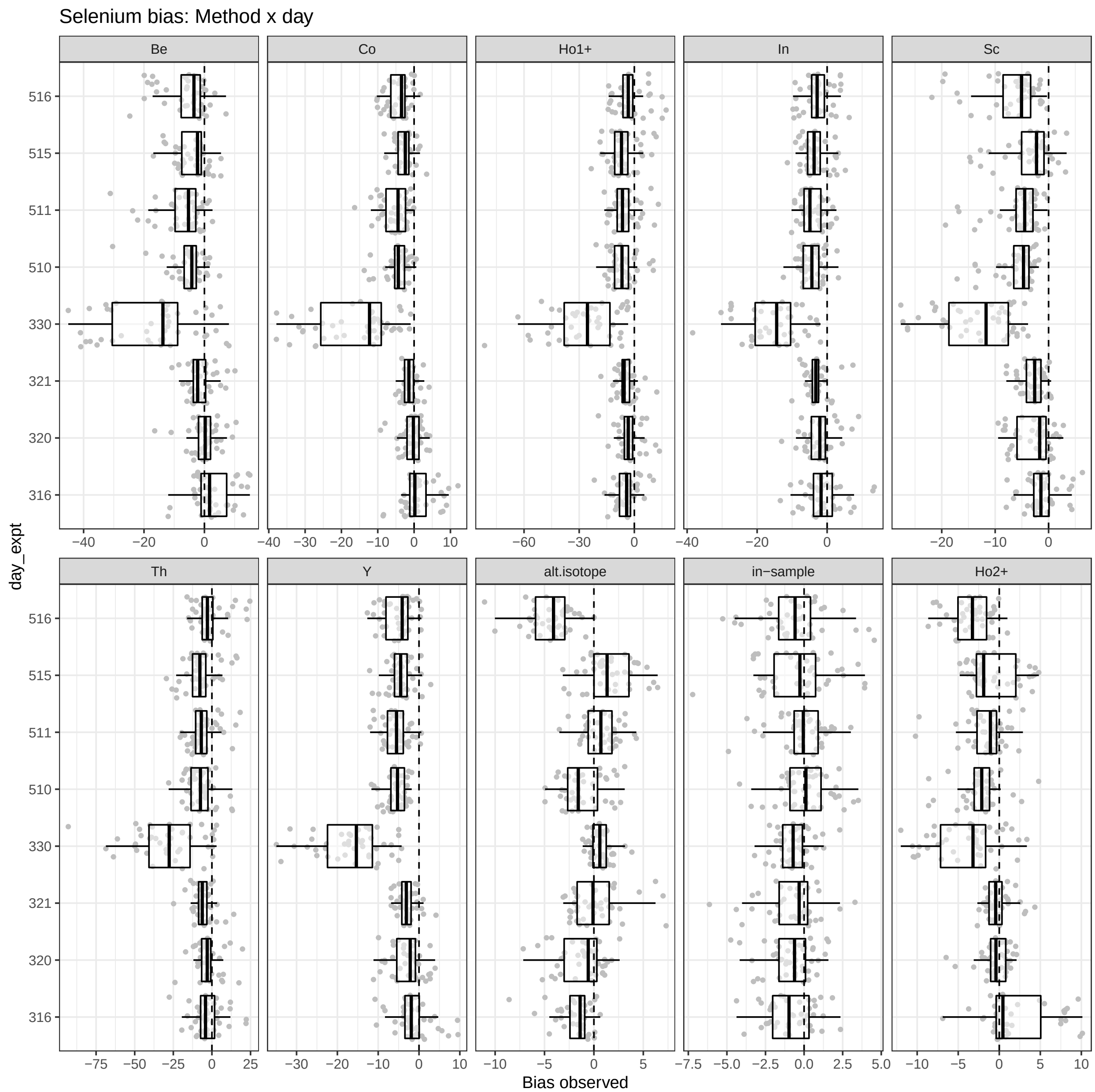
For selenium, above, the +2 methods look to be unbiased to slightly over-corrected, on average. The +1 methods all tended to over-correct fairly clearly. Within method, there looks to be clearer indication of differences due to tune setting for some of the +1 methods: Y and Sc in particular. Otherwise, tune effects are not clear.

Next, a plot of observations by method and day of the experiment for arsenic.



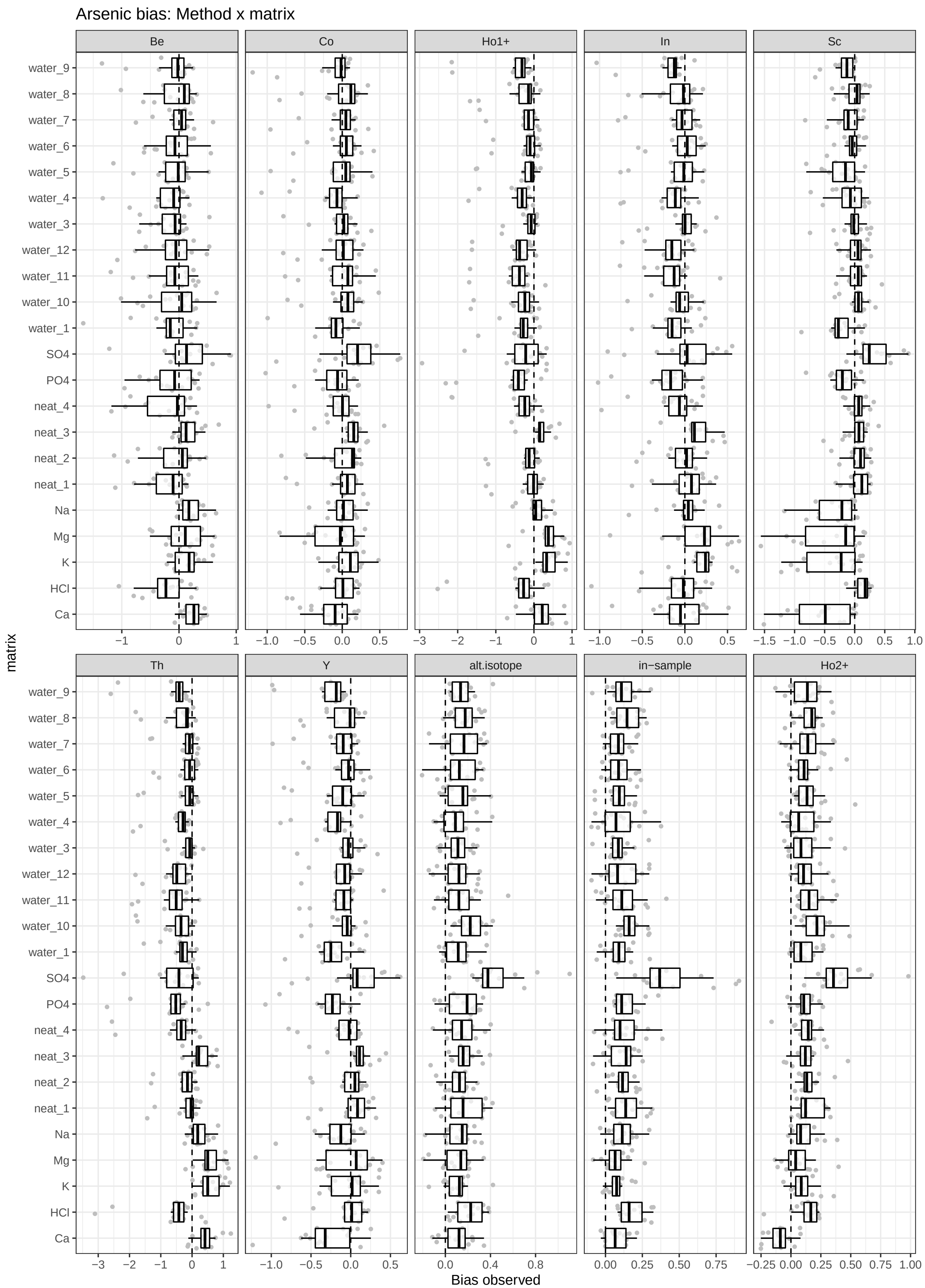
Above, the bias looks to have varied by day, most clearly for the +1 methods. In particular, the day of the cone change (3/30) sticks out as having greater tendency for over-corrections. The cone change looks to have been less important for the +2 methods; though day to day variability was otherwise similar.

And the same plot for selenium.



The pattern for selenium is similar as with arsenic above. However, the day to day variability for the +2 methods looks perhaps more pronounced compared to the +2 methods with arsenic.

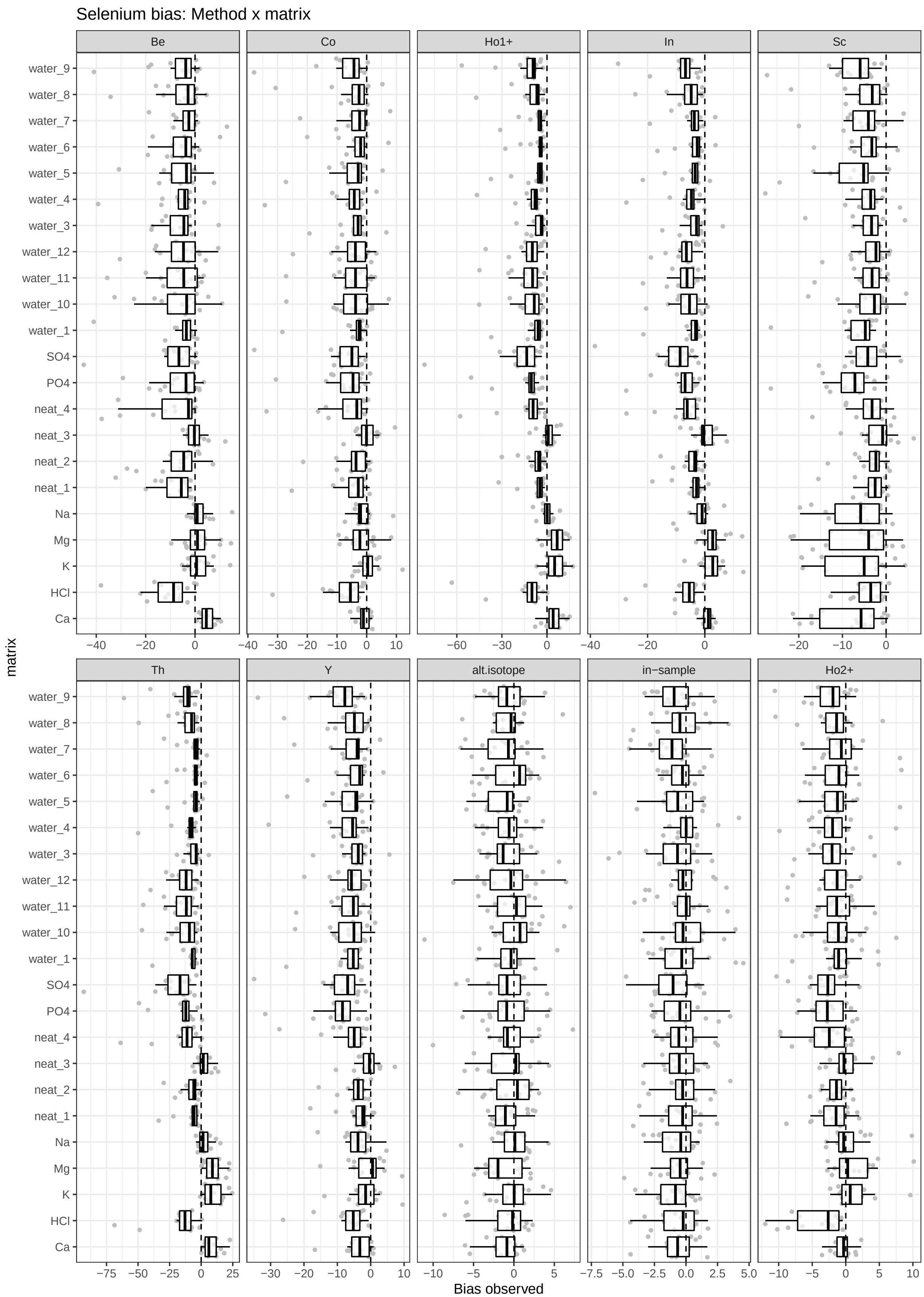
Next, a similar plot, but dividing the panels by matrix within method for arsenic.



Arsenic bias: Method x matrix
Be
Co
Ho1+
In
Sc
Th
Y
alt.isotope
in−sample
Ho2+
matrix
water_9
water_8
water_7
water_6
water_5
water_4
water_3
water_12
water_11
water_10
water_1
SO4
PO4
neat_4
neat_3
neat_2
neat_1
Na
Mg
K
HCl
Ca
Bias observed

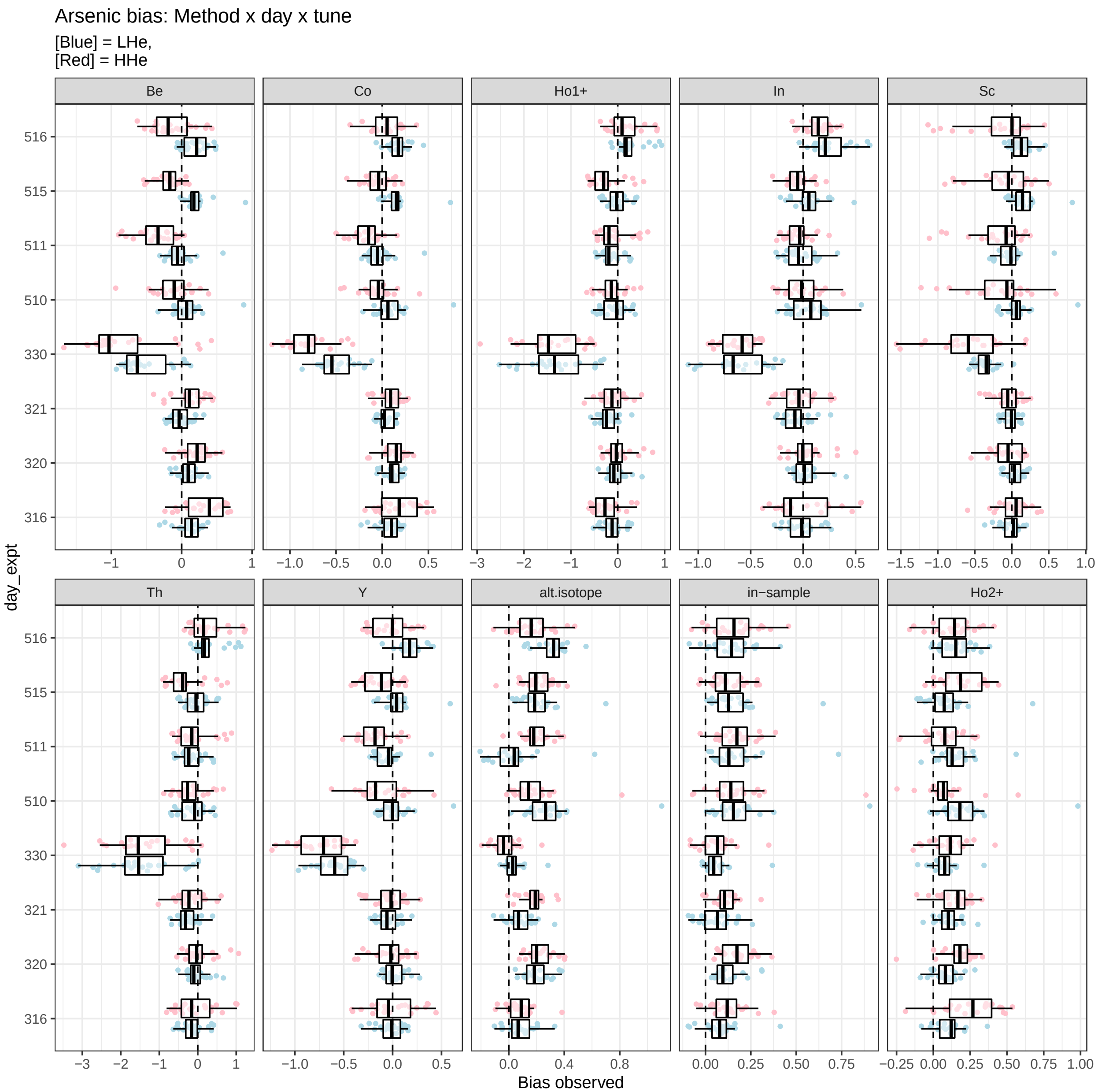
Matrix to matrix variation in bias looks less severe compared to variation by day above. Inferences concerning differences are frustrated more by noise here due to only 16 observations per box. However, looking at Sc, for example, reduced precision and larger over-corrections in the Na, Mg, K, and Ca matrices relative to other matrices stands out. Similarly, the SO_4 matrix in the +2 methods stands out as potentially inducing larger under-corrections relative to all other matrices for those methods.

And the same plot for selenium.


Selenium bias: Method x matrix
Be
Co
Ho1+
In
Sc
Th
Y
alt.isotope
in−sample
Ho2+
water_9
water_8
water_7
water_6
water_5
water_4
water_3
water_12
water_11
water_10
water_1
SO4
PO4
neat_4
neat_3
neat_2
neat_1
Na
Mg
K
HCl
Ca
matrix
Bias observed

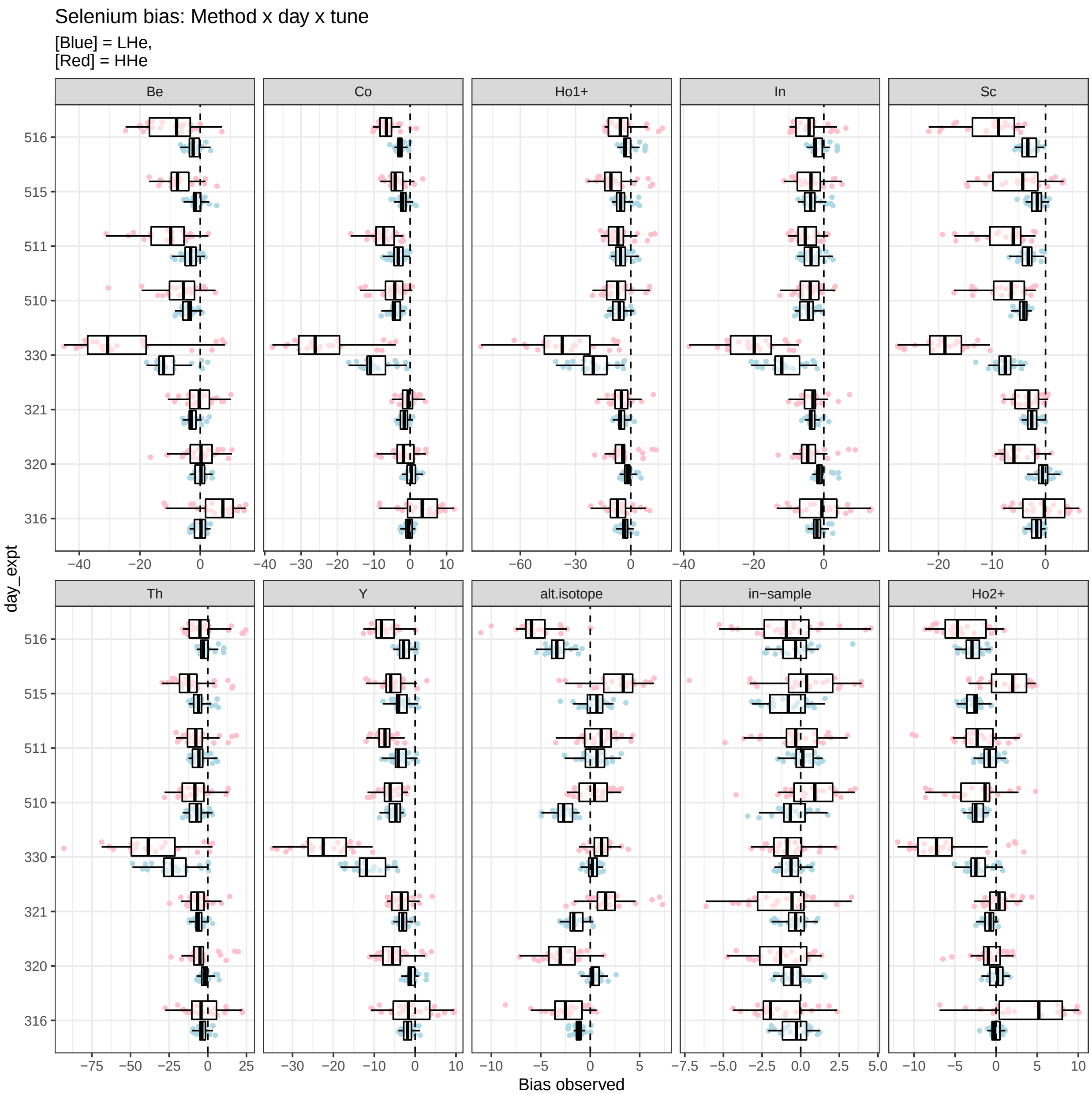
In terms of overall variability due to matrix, this one is very similar to the one for arsenic preceding. Interestingly, the SO_4 matrix doesn't stick out for the +2 methods here, though the pattern for Sc and the 250ppm matrices is similar to the arsenic plot above.

Next is another plot of the observed bias by day (e.g., 516 = 5/16) for arsenic measured for each IS method, but also divided according to tune (red = LHe; blue = HHe).



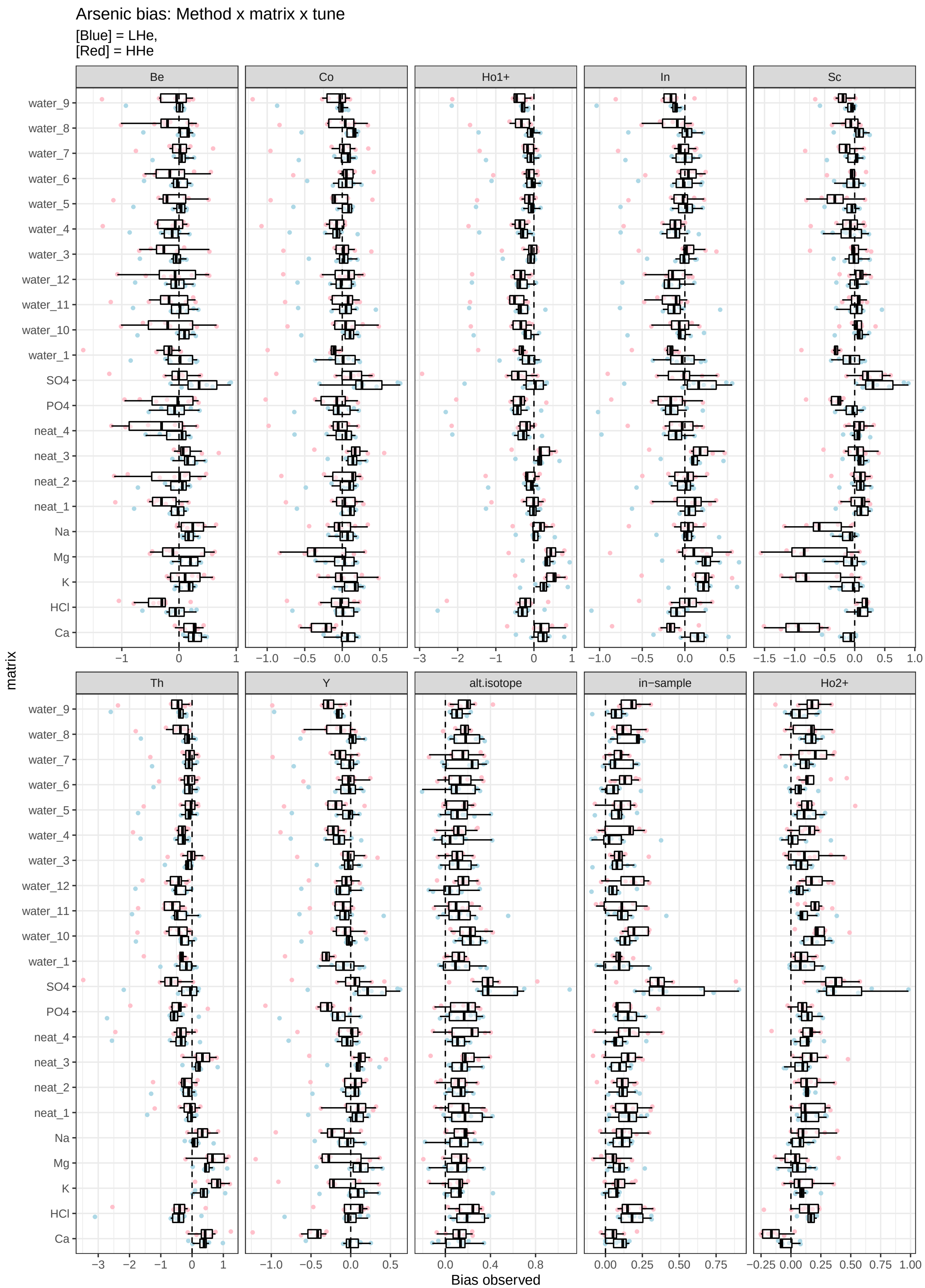
The general patterns by day are still apparent. There are some distinctions within days indicating that the effect of tune may depend on the day (e.g., alt.isotope on 3/21 and 5/11; Co on 3/30 and 5/15; etc.).

And the same plot for selenium is below.



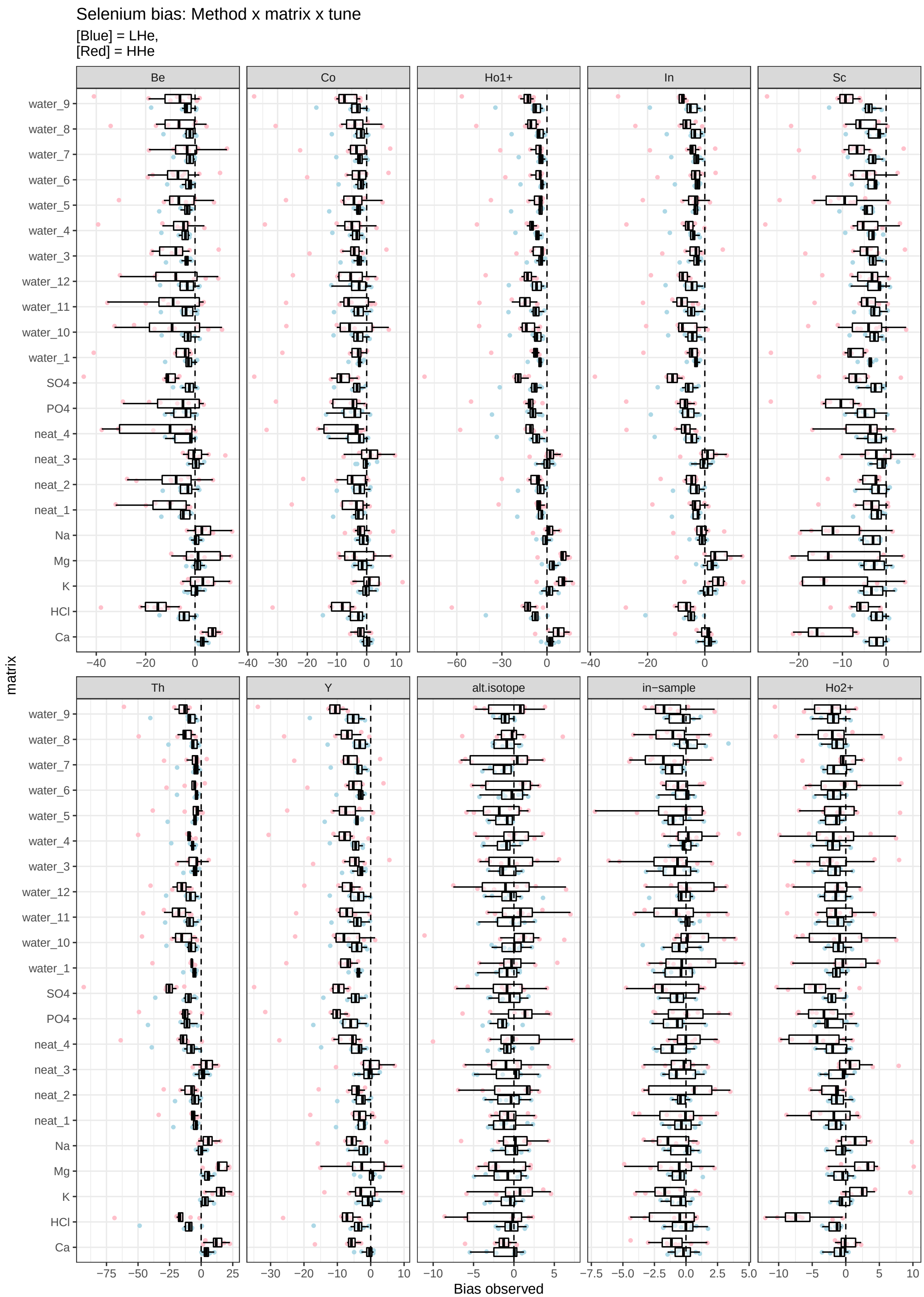
This one above is similar to the arsenic plot preceding, though there look to be more cases where the tune effect potentially differs by day for some methods. That is, the tune effect looks potentially clearer in some cases.

Next, a similar plot, but dividing the panels by matrix and tune within method for arsenic.


Arsenic bias: Method x matrix x tune
[Blue] = LHe,
[Red] = HHe
Be
Co
Ho1+
In
Sc
Th
Y
alt.isotope
in−sample
Ho2+
water_9
water_8
water_7
water_6
water_5
water_4
water_3
water_12
water_11
water_10
water_1
SO4
PO4
neat_4
neat_3
neat_2
neat_1
Na
Mg
K
HCl
Ca
matrix
Bias observed

There are only 8 observations per box above, but matrix to matrix variability in bias again seems smaller compared to day to day above; and the patterns in the 250ppm matrices for the +2 methods are apparent. Within matrix, there are a few potential differences here and there according to tune.

And finally the same plot for selenium.



Selenium bias: Method x matrix x tune
[Blue] = LHe,
[Red] = HHe
Be
Co
Ho1+
In
Sc
Th
Y
alt.isotope
in−sample
Ho2+
water_9
water_8
water_7
water_6
water_5
water_4
water_3
water_12
water_11
water_10
water_1
SO4
PO4
neat_4
neat_3
neat_2
neat_1
Na
Mg
K
HCl
Ca
matrix
Bias observed

Again, the overall patterns are similar to the plot above, but within matrix differences due to tune setting are more pronounced in some cases.

Model structure

Because all 10 IS were included in each physical sample run from a given matrix on a given day, the data were treated as a multivariate response (i.e., 352 observations on 10 IS methods), allowing for observation-level correlation between the IS methods. A multilevel Bayesian approach to modeling was taken, with each model conditioned in some manner on all of the experimental design factors (matrix, day, and tune). However, not all potential interactions among the experimental factors were able to be estimated because the number of observations available to estimate them was limited. For example, there was no replication of observations within a matrix and day of the experiment using the same tune setting for any IS method. For each IS in a specific matrix and specific day, only a single measurement was taken at each tune setting (i.e., n=1 for estimating a method x matrix x day x tune effect). Likewise, matrix x day effects were not estimated due to limited information (n = 2 per IS method, corresponding to 2 tune settings).

The models to follow assumed that the observed bias (difference to the nearest 0.001 ppb between the "True Value" and IS estimated correction value) was generated from a multivariate normal (MVN) distribution, where:

$$Y \sim MVN(\mu, \Sigma)$$

where Y is a multivariate response matrix of size $J = 10$ columns and $N = 352$ rows; μ is the location parameter of the multivariate normal data generating process; and Σ is the covariance matrix containing the scale parameters for each of the $J = 10$ methods on the diagonal, and the between-method correlation parameters on the off-diagonals.

Linear model for μ

For the first model fit below, the location parameter was parameterized such that:

$$\mu_j = \alpha_j + \beta_j * X_{tune_j} + \gamma_{K_j} + \gamma_{L_j}$$

$$\gamma_{K_j} \sim N(0, \sigma_{K_j})$$

$$\gamma_{L_j} \sim N(0, \sigma_{L_j})$$

where α_j references a fixed global intercept for each IS method, $j \in 1, .., J = 10$. This intercept parameter represents the global mean of the low helium tune (LHe) observations for each method, j. The individual observations are not indexed in the notation above, but each of $i \in i = 1, .., N = 352$ observations per method is fit to the same $alpha_j$. The β_j parameter captures the additive effect of the HHe tune on the global mean for each IS method. X_{tune_j} represents a vector holding the tune setting indicator (0 or 1) for each observation, entered as data. The γ_{K_j} term references varying effects, or intercepts, for each of $k \in 1, ..., K = 22$ matrix conditions for each method; and γ_{L_j} references varying effects for each of the $l \in 1, ..., L = 8$ days of the experiment. These varying effects are centered on zero and can vary around the global intercept parameter, the degree to which is determined by a hierarchical scale parameters, σ_{K_j} and σ_{L_j}, estimated from the data.

In the formula syntax of the **R** package **lme4** commonly used to fit mixed effects models, the formula for the linear predictor for μ above, would be:

$$1 + tune + (1|matrix) + (1|day)$$

This is the equivalent syntax employed in the **brms** package (Bürkner 2017), which is used to fit the models below.

Fitting a model for arsenic

The **brms** package (Bürkner 2017) makes it convenient to fit and compare multilevel generalized linear models in a fully Bayesian framework. Below, the model described above is fit to the observational data. For each parameter described above, priors were provided as indicated in the code below. Specifically, $N(0, 1)$ priors were placed over all intercept parameters for each response. This prior is considered weakly informative on the the scale of the observations. This $N(0, 1)$ prior was also placed on the scale parameters of the varying effects, which was also considered weakly informative. Covariances in **Stan** (Stan Development Team 2018c, 2018a, 2018b) are parameterized efficiently by placing priors separately on the standard deviations and the correlation matrix. The covariance term for the multivariate model is parameterized such that:

$$\Sigma = diag(\tau)\Omega diag(\tau)$$

so priors are placed separately on τ, the vector of standard deviations for each IS method, and Ω the correlation matrix. For each standard deviation in the model below, the $N(0, 1)$ prior was again applied and assumed weakly informative. For the correlation matrix, an $LKJ(\eta = 1)$ prior was used, which is uniform over permissible correlation matrices. For more information on prior choice recommendations, see: https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations

```
load("full-analysis-files/df_mv_as.rda")

bf_Std <- bf(Std ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Alt <- bf(Alt ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho2 <- bf(Ho2 ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_In <- bf(In ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Sc <- bf(Sc ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Y <- bf(Y ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Be <- bf(Be ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Co <- bf(Co ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Th <- bf(Th ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho1 <- bf(Ho1 ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

mod1 <- brm(bf_Std +
              bf_Alt +
              bf_Ho2 +
```

```
             bf_In +
             bf_Sc +
             bf_Y +
             bf_Be +
             bf_Co +
             bf_Th +
             bf_Ho1 +
             set_rescor(TRUE),
  data = df_mv_as,
  prior = c(prior(normal(0, 1), class = "Intercept", resp = "Std"),
            prior(normal(0, 1), class = "Intercept", resp = "Alt"),
            prior(normal(0, 1), class = "Intercept", resp = "Ho2"),
            prior(normal(0, 1), class = "Intercept", resp = "In"),
            prior(normal(0, 1), class = "Intercept", resp = "Sc"),
            prior(normal(0, 1), class = "Intercept", resp = "Y"),
            prior(normal(0, 1), class = "Intercept", resp = "Be"),
            prior(normal(0, 1), class = "Intercept", resp = "Co"),
            prior(normal(0, 1), class = "Intercept", resp = "Th"),
            prior(normal(0, 1), class = "Intercept", resp = "Ho1"),

            prior(normal(0, 1), class = "b", resp = "Std"),
            prior(normal(0, 1), class = "b", resp = "Alt"),
            prior(normal(0, 1), class = "b", resp = "Ho2"),
            prior(normal(0, 1), class = "b", resp = "In"),
            prior(normal(0, 1), class = "b", resp = "Sc"),
            prior(normal(0, 1), class = "b", resp = "Y"),
            prior(normal(0, 1), class = "b", resp = "Be"),
            prior(normal(0, 1), class = "b", resp = "Co"),
            prior(normal(0, 1), class = "b", resp = "Th"),
            prior(normal(0, 1), class = "b", resp = "Ho1"),

            prior(normal(0, 1), class = "sd", resp = "Std"),
            prior(normal(0, 1), class = "sd", resp = "Alt"),
            prior(normal(0, 1), class = "sd", resp = "Ho2"),
            prior(normal(0, 1), class = "sd", resp = "In"),
            prior(normal(0, 1), class = "sd", resp = "Sc"),
            prior(normal(0, 1), class = "sd", resp = "Y"),
            prior(normal(0, 1), class = "sd", resp = "Be"),
            prior(normal(0, 1), class = "sd", resp = "Co"),
            prior(normal(0, 1), class = "sd", resp = "Th"),
            prior(normal(0, 1), class = "sd", resp = "Ho1"),

            prior(normal(0, 1), class = "sigma", resp = "Std"),
            prior(normal(0, 1), class = "sigma", resp = "Alt"),
            prior(normal(0, 1), class = "sigma", resp = "Ho2"),
            prior(normal(0, 1), class = "sigma", resp = "In"),
            prior(normal(0, 1), class = "sigma", resp = "Sc"),
            prior(normal(0, 1), class = "sigma", resp = "Y"),
            prior(normal(0, 1), class = "sigma", resp = "Be"),
            prior(normal(0, 1), class = "sigma", resp = "Co"),
            prior(normal(0, 1), class = "sigma", resp = "Th"),
            prior(normal(0, 1), class = "sigma", resp = "Ho1"),
```

```r
            prior(lkj(1), class = "rescor")
            ),
  control = list(adapt_delta = 0.90, max_treedepth = 12),
  init_r = 0.05,
  save_pars = save_pars(all = TRUE),
  seed = 6518,
  chains=4,
  iter=2000,
  cores=4 )

save(mod1, file = "full-analysis-files/mod1_As_mv.rda")
```

Tabular parameter estimates

Next, a summary of the posterior estimates.

```
##  Family: MV(gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian
##   Links: mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
##          mu = identity; sigma = identity
## Formula: Std ~ tune + (1 | matrix) + (1 | day_expt)
##          Alt ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho2 ~ tune + (1 | matrix) + (1 | day_expt)
##          In ~ tune + (1 | matrix) + (1 | day_expt)
##          Sc ~ tune + (1 | matrix) + (1 | day_expt)
##          Y ~ tune + (1 | matrix) + (1 | day_expt)
##          Be ~ tune + (1 | matrix) + (1 | day_expt)
##          Co ~ tune + (1 | matrix) + (1 | day_expt)
##          Th ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho1 ~ tune + (1 | matrix) + (1 | day_expt)
##    Data: df_mv_as (Number of observations: 352)
##   Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup draws = 4000
##
## Priors:
## b_Alt ~ normal(0, 1)
## b_Be ~ normal(0, 1)
## b_Co ~ normal(0, 1)
## b_Ho1 ~ normal(0, 1)
## b_Ho2 ~ normal(0, 1)
## b_In ~ normal(0, 1)
## b_Sc ~ normal(0, 1)
## b_Std ~ normal(0, 1)
## b_Th ~ normal(0, 1)
## b_Y ~ normal(0, 1)
```

```
## Intercept_Alt ~ normal(0, 1)
## Intercept_Be ~ normal(0, 1)
## Intercept_Co ~ normal(0, 1)
## Intercept_Ho1 ~ normal(0, 1)
## Intercept_Ho2 ~ normal(0, 1)
## Intercept_In ~ normal(0, 1)
## Intercept_Sc ~ normal(0, 1)
## Intercept_Std ~ normal(0, 1)
## Intercept_Th ~ normal(0, 1)
## Intercept_Y ~ normal(0, 1)
## Lrescor ~ lkj_corr_cholesky(1)
## sd_Alt ~ normal(0, 1)
## sd_Be ~ normal(0, 1)
## sd_Co ~ normal(0, 1)
## sd_Ho1 ~ normal(0, 1)
## sd_Ho2 ~ normal(0, 1)
## sd_In ~ normal(0, 1)
## sd_Sc ~ normal(0, 1)
## sd_Std ~ normal(0, 1)
## sd_Th ~ normal(0, 1)
## sd_Y ~ normal(0, 1)
## sigma_Alt ~ normal(0, 1)
## sigma_Be ~ normal(0, 1)
## sigma_Co ~ normal(0, 1)
## sigma_Ho1 ~ normal(0, 1)
## sigma_Ho2 ~ normal(0, 1)
## sigma_In ~ normal(0, 1)
## sigma_Sc ~ normal(0, 1)
## sigma_Std ~ normal(0, 1)
## sigma_Th ~ normal(0, 1)
## sigma_Y ~ normal(0, 1)
##
## Group-Level Effects:
## ~day_expt (Number of levels: 8)
##                   Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Std_Intercept)     0.04      0.02     0.02     0.09 1.00     3085     3089
## sd(Alt_Intercept)     0.10      0.04     0.05     0.19 1.00     2751     2747
## sd(Ho2_Intercept)     0.02      0.01     0.00     0.05 1.00     1993     2426
## sd(In_Intercept)      0.28      0.09     0.16     0.52 1.00     4727     2584
## sd(Sc_Intercept)      0.18      0.07     0.10     0.34 1.00     3073     2982
## sd(Y_Intercept)       0.27      0.09     0.15     0.50 1.00     4822     2657
## sd(Be_Intercept)      0.31      0.10     0.18     0.57 1.00     3370     2977
## sd(Co_Intercept)      0.29      0.10     0.17     0.54 1.00     3525     2850
## sd(Th_Intercept)      0.57      0.17     0.34     0.99 1.00     4401     2902
## sd(Ho1_Intercept)     0.56      0.18     0.32     0.99 1.00     5020     2729
##
## ~matrix (Number of levels: 22)
##                   Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Std_Intercept)     0.06      0.01     0.04     0.08 1.00     1649     2770
## sd(Alt_Intercept)     0.05      0.01     0.03     0.08 1.00     1504     2775
## sd(Ho2_Intercept)     0.06      0.01     0.04     0.09 1.00     1606     2287
## sd(In_Intercept)      0.10      0.02     0.07     0.14 1.00     2511     3094
## sd(Sc_Intercept)      0.17      0.03     0.12     0.24 1.00     2571     3079
## sd(Y_Intercept)       0.08      0.02     0.06     0.12 1.00     2853     2997
```

```
## sd(Be_Intercept)      0.13      0.03     0.09     0.20 1.00     2680     2724
## sd(Co_Intercept)      0.05      0.01     0.03     0.08 1.00     2171     2897
## sd(Th_Intercept)      0.38      0.07     0.28     0.53 1.00     2082     2767
## sd(Ho1_Intercept)     0.26      0.05     0.19     0.38 1.00     2010     2523
##
## Population-Level Effects:
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Std_Intercept     0.12      0.02     0.08     0.16 1.00     2900     2697
## Alt_Intercept     0.16      0.04     0.09     0.23 1.00     2720     2988
## Ho2_Intercept     0.12      0.02     0.09     0.16 1.00     3598     3242
## In_Intercept     -0.04      0.11    -0.25     0.19 1.00     2189     2595
## Sc_Intercept     -0.00      0.08    -0.16     0.15 1.00     2227     2478
## Y_Intercept      -0.06      0.10    -0.25     0.14 1.00     2198     2616
## Be_Intercept      0.02      0.12    -0.21     0.26 1.00     2030     2895
## Co_Intercept      0.01      0.10    -0.19     0.22 1.00     2135     2797
## Th_Intercept     -0.23      0.22    -0.66     0.20 1.00     2177     2582
## Ho1_Intercept    -0.19      0.21    -0.62     0.23 1.00     2022     2482
## Std_tuneHHe       0.02      0.01    -0.01     0.04 1.00     4288     3272
## Alt_tuneHHe      -0.00      0.01    -0.03     0.02 1.00     4111     3542
## Ho2_tuneHHe       0.02      0.01    -0.00     0.05 1.00     3977     3479
## In_tuneHHe       -0.03      0.02    -0.06     0.00 1.00     3283     3249
## Sc_tuneHHe       -0.17      0.02    -0.21    -0.13 1.00     4333     3119
## Y_tuneHHe        -0.10      0.02    -0.13    -0.07 1.00     3387     3090
## Be_tuneHHe       -0.14      0.03    -0.19    -0.08 1.00     4509     3154
## Co_tuneHHe       -0.09      0.02    -0.12    -0.06 1.00     3242     3296
## Th_tuneHHe        0.00      0.03    -0.06     0.06 1.00     5398     3520
## Ho1_tuneHHe      -0.06      0.02    -0.10    -0.01 1.00     5037     3492
##
## Family Specific Parameters:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma_Std     0.10      0.00     0.09     0.11 1.00     3013     3433
## sigma_Alt     0.12      0.00     0.11     0.13 1.00     3052     3453
## sigma_Ho2     0.12      0.00     0.11     0.13 1.00     2919     3491
## sigma_In      0.14      0.01     0.13     0.15 1.00     3046     3377
## sigma_Sc      0.21      0.01     0.19     0.23 1.00     4066     3239
## sigma_Y       0.15      0.01     0.14     0.16 1.00     3034     3078
## sigma_Be      0.26      0.01     0.24     0.28 1.00     4449     3559
## sigma_Co      0.15      0.01     0.14     0.16 1.00     3020     3219
## sigma_Th      0.28      0.01     0.26     0.31 1.00     3681     3135
## sigma_Ho1     0.22      0.01     0.20     0.24 1.00     4003     3139
##
## Residual Correlations:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## rescor(Std,Alt)     0.67      0.03     0.60     0.73 1.00     3403     3434
## rescor(Std,Ho2)     0.59      0.04     0.52     0.66 1.00     3041     3441
## rescor(Alt,Ho2)     0.55      0.04     0.47     0.63 1.00     3628     3353
## rescor(Std,In)      0.52      0.04     0.43     0.59 1.00     3069     2760
## rescor(Alt,In)      0.49      0.04     0.40     0.57 1.00     3159     3190
## rescor(Ho2,In)      0.59      0.04     0.51     0.66 1.00     3582     3074
## rescor(Std,Sc)      0.42      0.05     0.33     0.51 1.00     3373     3310
## rescor(Alt,Sc)      0.40      0.05     0.31     0.49 1.00     3809     3337
## rescor(Ho2,Sc)      0.54      0.04     0.46     0.61 1.00     4236     2825
## rescor(In,Sc)       0.67      0.03     0.60     0.73 1.00     4338     3011
## rescor(Std,Y)       0.51      0.04     0.43     0.59 1.00     2940     3126
```

```
## rescor(Alt,Y)        0.48      0.04      0.40      0.56 1.00     3034     3171
## rescor(Ho2,Y)        0.62      0.03      0.55      0.68 1.00     3767     2785
## rescor(In,Y)         0.93      0.01      0.92      0.95 1.00     4334     3322
## rescor(Sc,Y)         0.84      0.02      0.81      0.87 1.00     4377     3372
## rescor(Std,Be)       0.32      0.05      0.22      0.41 1.00     4023     3056
## rescor(Alt,Be)       0.28      0.05      0.17      0.38 1.00     4556     3750
## rescor(Ho2,Be)       0.36      0.05      0.26      0.45 1.00     4696     3860
## rescor(In,Be)        0.33      0.05      0.23      0.42 1.00     4049     3618
## rescor(Sc,Be)        0.27      0.05      0.17      0.37 1.00     4311     2970
## rescor(Y,Be)         0.34      0.05      0.25      0.43 1.00     4237     3382
## rescor(Std,Co)       0.50      0.04      0.42      0.58 1.00     2808     2975
## rescor(Alt,Co)       0.46      0.04      0.36      0.54 1.00     3320     3110
## rescor(Ho2,Co)       0.59      0.04      0.51      0.65 1.00     3394     3288
## rescor(In,Co)        0.69      0.03      0.63      0.74 1.00     3581     3131
## rescor(Sc,Co)        0.62      0.03      0.55      0.68 1.00     3482     3081
## rescor(Y,Co)         0.74      0.02      0.69      0.79 1.00     3758     3252
## rescor(Be,Co)        0.79      0.02      0.74      0.82 1.00     4696     3876
## rescor(Std,Th)       0.21      0.05      0.10      0.31 1.00     4516     3243
## rescor(Alt,Th)       0.20      0.05      0.10      0.31 1.00     4930     3321
## rescor(Ho2,Th)       0.14      0.05      0.03      0.25 1.00     4286     2953
## rescor(In,Th)        0.55      0.04      0.48      0.63 1.00     4488     3072
## rescor(Sc,Th)       -0.04      0.05     -0.15      0.06 1.00     4120     3400
## rescor(Y,Th)         0.36      0.05      0.26      0.45 1.00     4375     3285
## rescor(Be,Th)        0.54      0.04      0.46      0.62 1.00     4570     3697
## rescor(Co,Th)        0.56      0.04      0.48      0.63 1.00     4716     3438
## rescor(Std,Ho1)      0.29      0.05      0.19      0.39 1.00     4620     3225
## rescor(Alt,Ho1)      0.31      0.05      0.21      0.41 1.00     4997     3293
## rescor(Ho2,Ho1)      0.12      0.05      0.01      0.22 1.00     4212     3302
## rescor(In,Ho1)       0.57      0.04      0.49      0.64 1.00     4490     3418
## rescor(Sc,Ho1)      -0.02      0.05     -0.13      0.08 1.00     4119     3242
## rescor(Y,Ho1)        0.39      0.05      0.30      0.48 1.00     4425     3598
## rescor(Be,Ho1)       0.47      0.04      0.39      0.55 1.00     4246     3540
## rescor(Co,Ho1)       0.54      0.04      0.46      0.61 1.00     4321     3164
## rescor(Th,Ho1)       0.95      0.01      0.94      0.96 1.00     4613     3586
##
## 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).
```

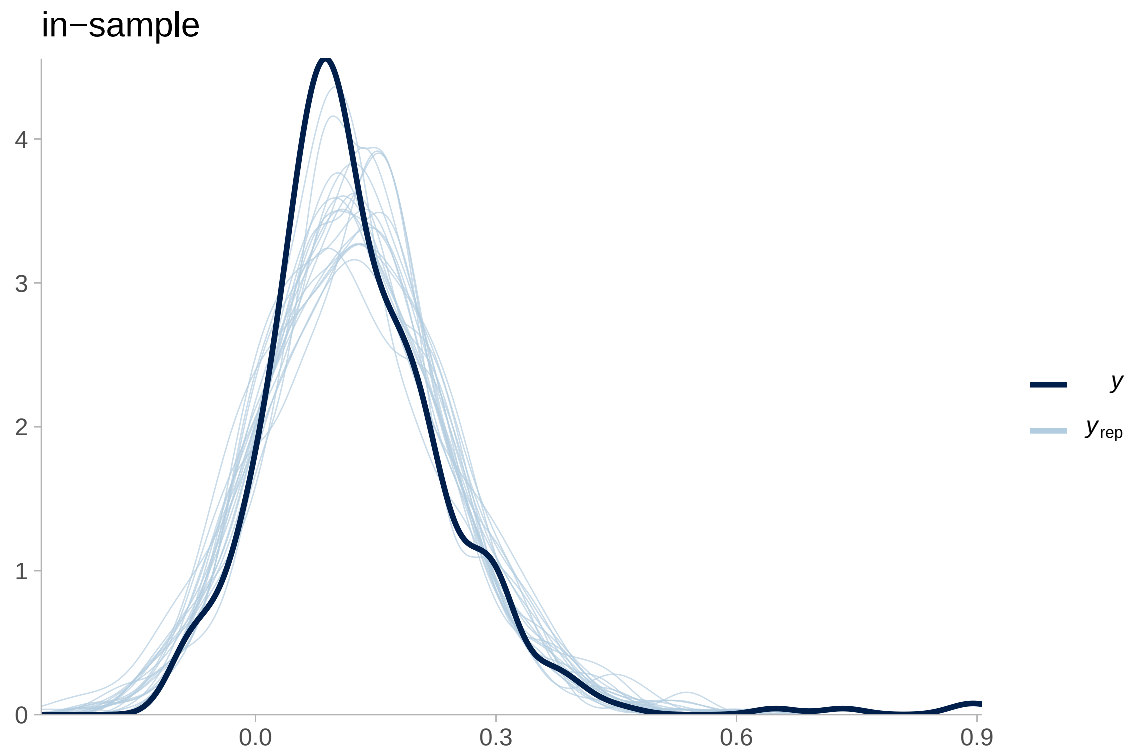
The tabular result above is helpful for looking at HMC convergence diagnostics (Rhat, ESS). There were no apparent issues with the HMC estimation procedure. The parameter summaries are also helpful, but it is more efficient to explore their implications graphically, as will be done for the final models below.

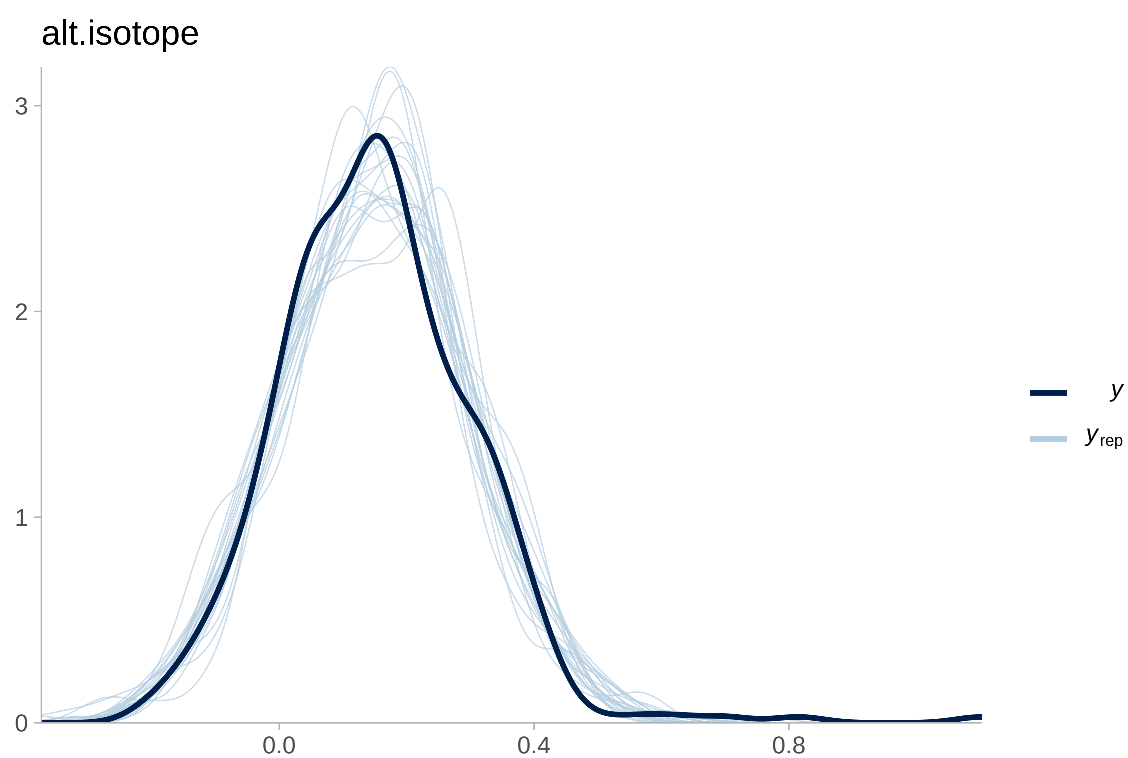
Model checks

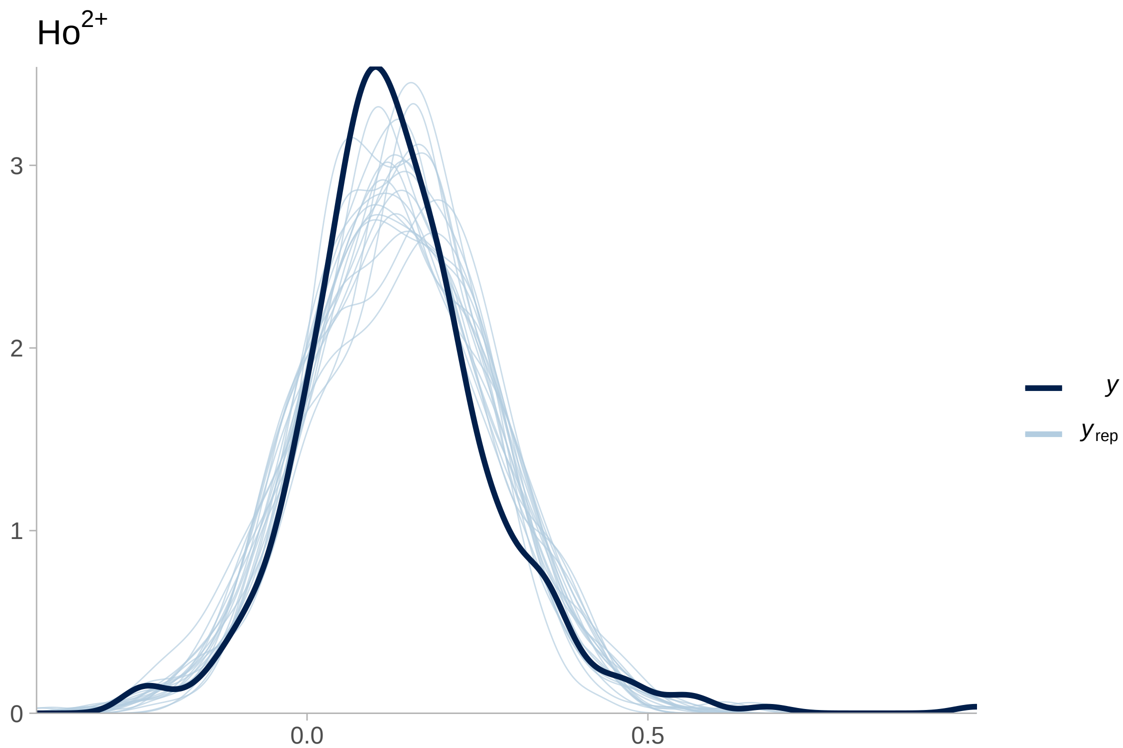
Posterior predictive checks are useful for visualizing the extent to which the fitted model generates replicate data that resembles the observed data.

Density overlay

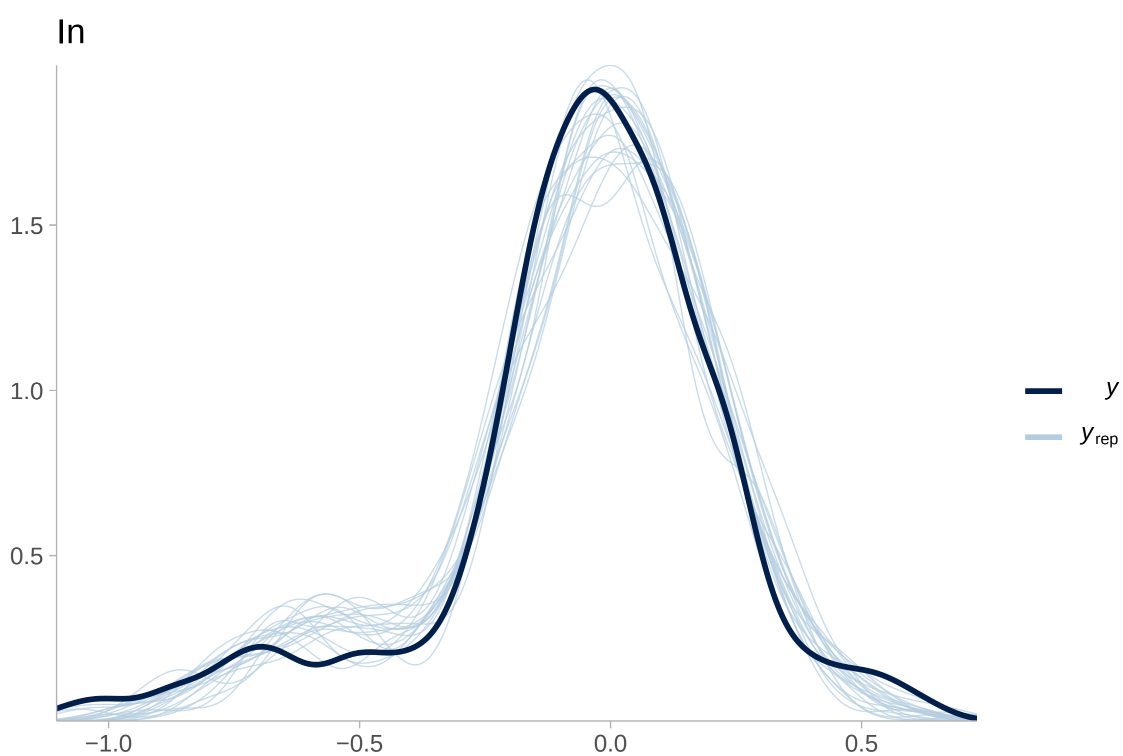
In the first check, 50 individual draws from the posterior predictive distribution of the fitted model were summarized using a density plot. The individual draws can be thought of as simulations of potential datasets from a data generating process the model is representing. The density plots of these hypothetical datasets (blue lines) are compared to the density plot for the observed data (black line) to gauge whether the observed data could have reasonably been drawn from the same data generating process.



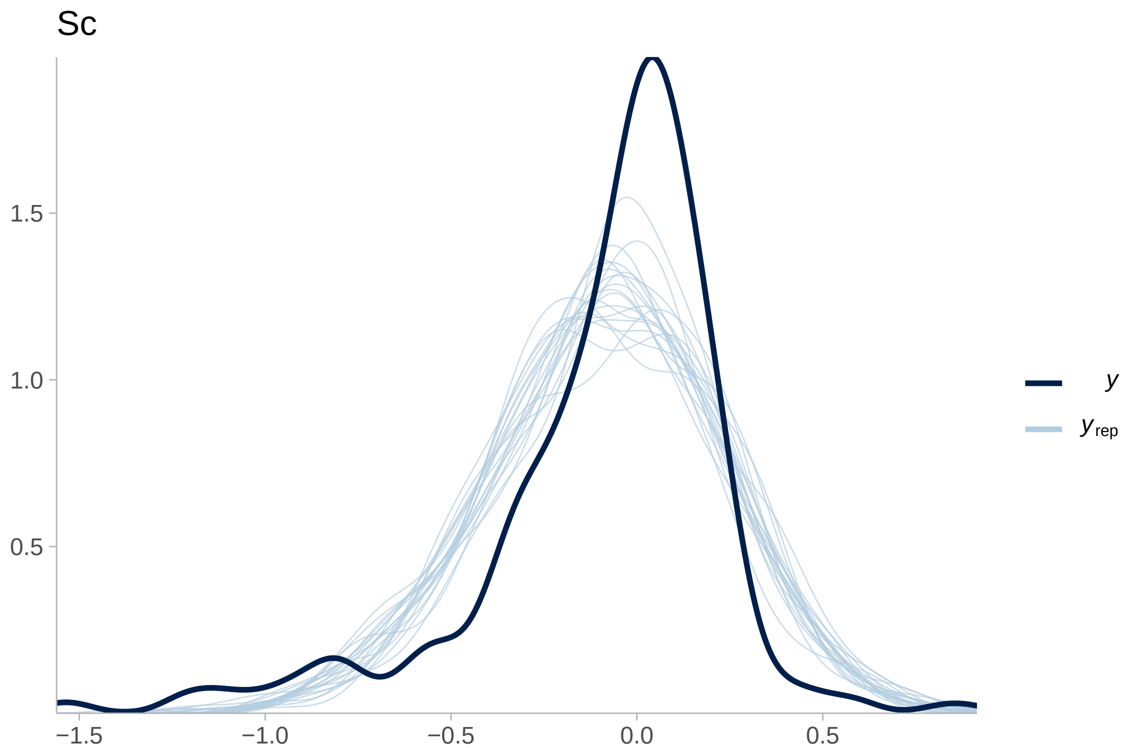


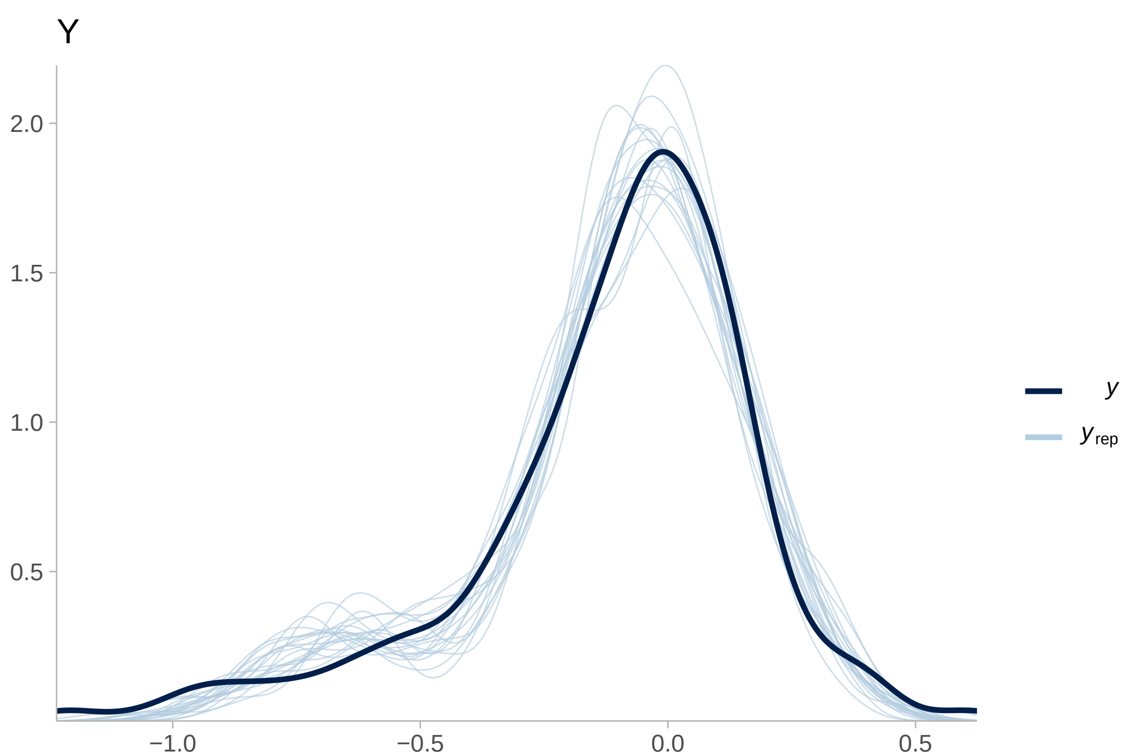


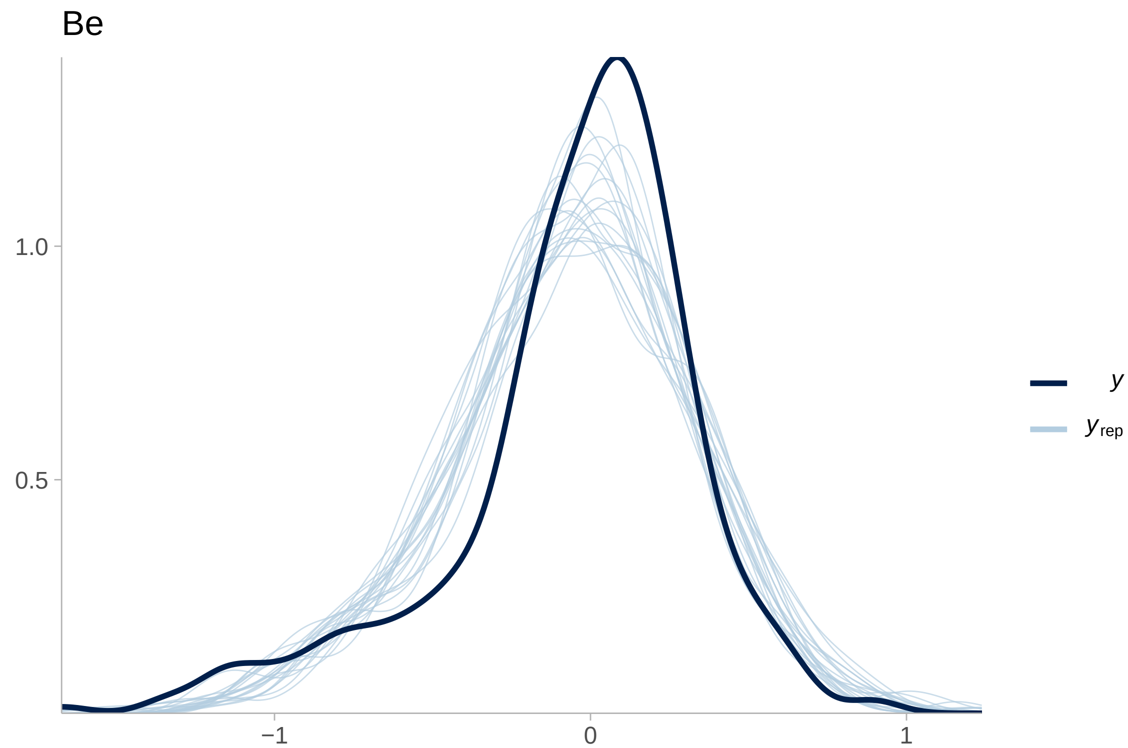
Ho^{2+}
3
2
1
0
0.0
0.5
y
y_{rep}

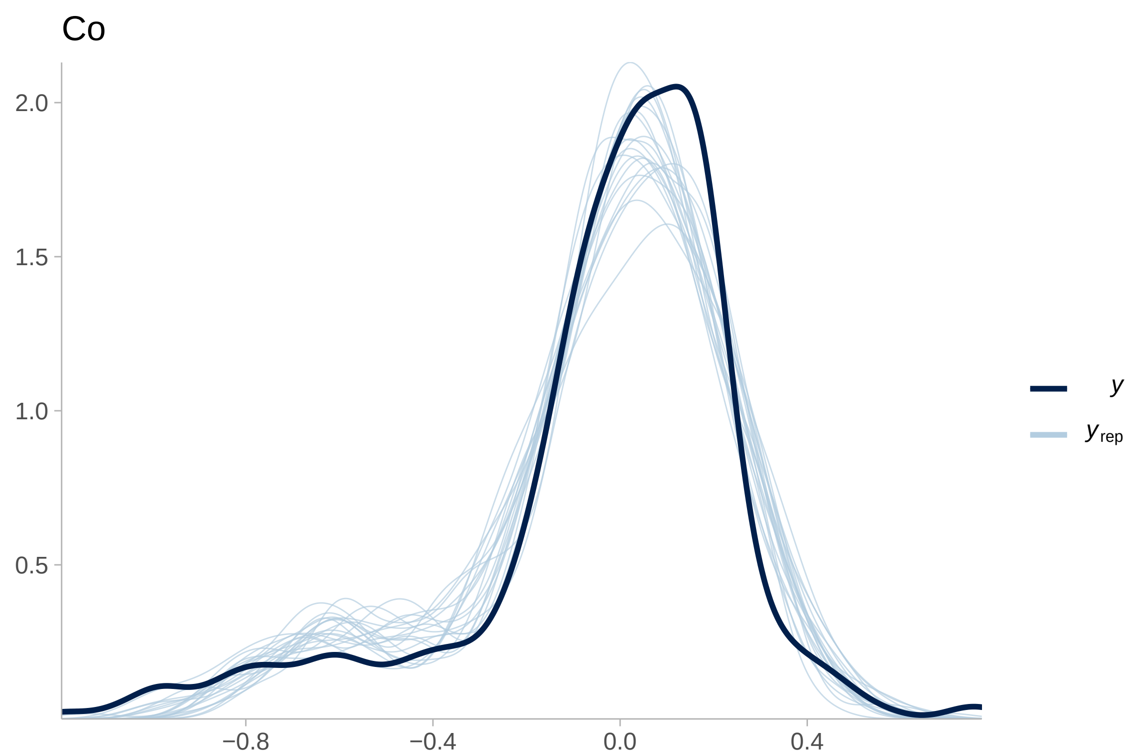


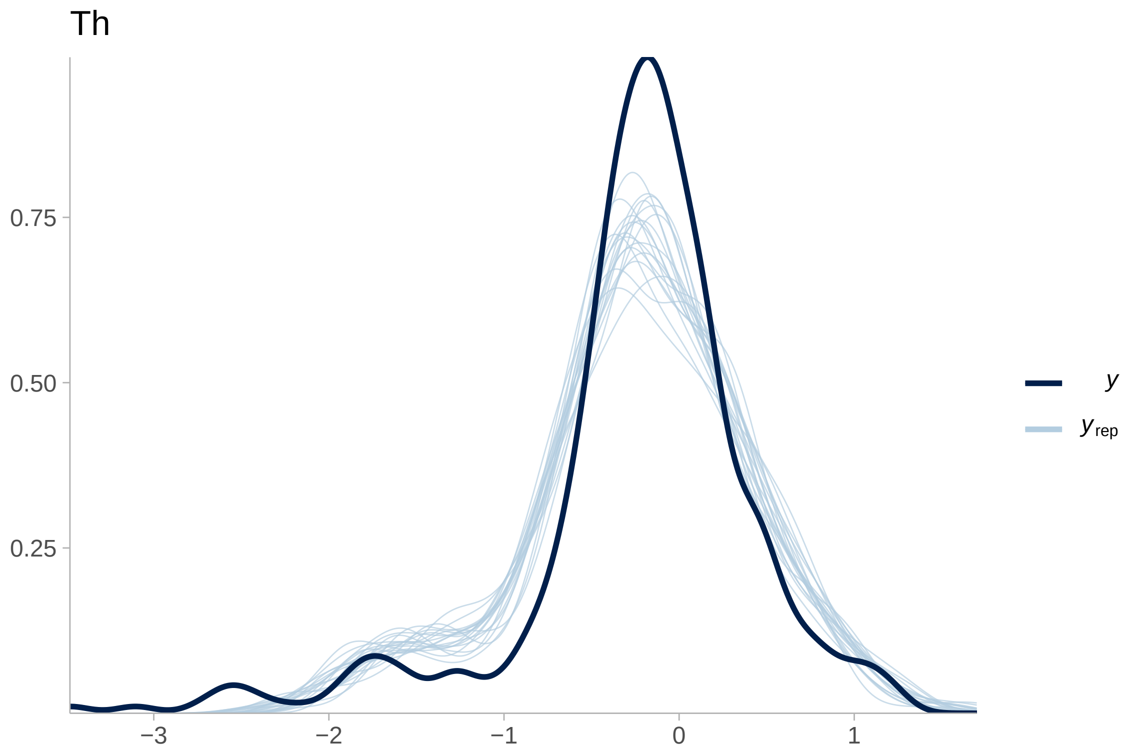
In
1.5
1.0
0.5
−1.0
−0.5
0.0
0.5
y
y_{rep}

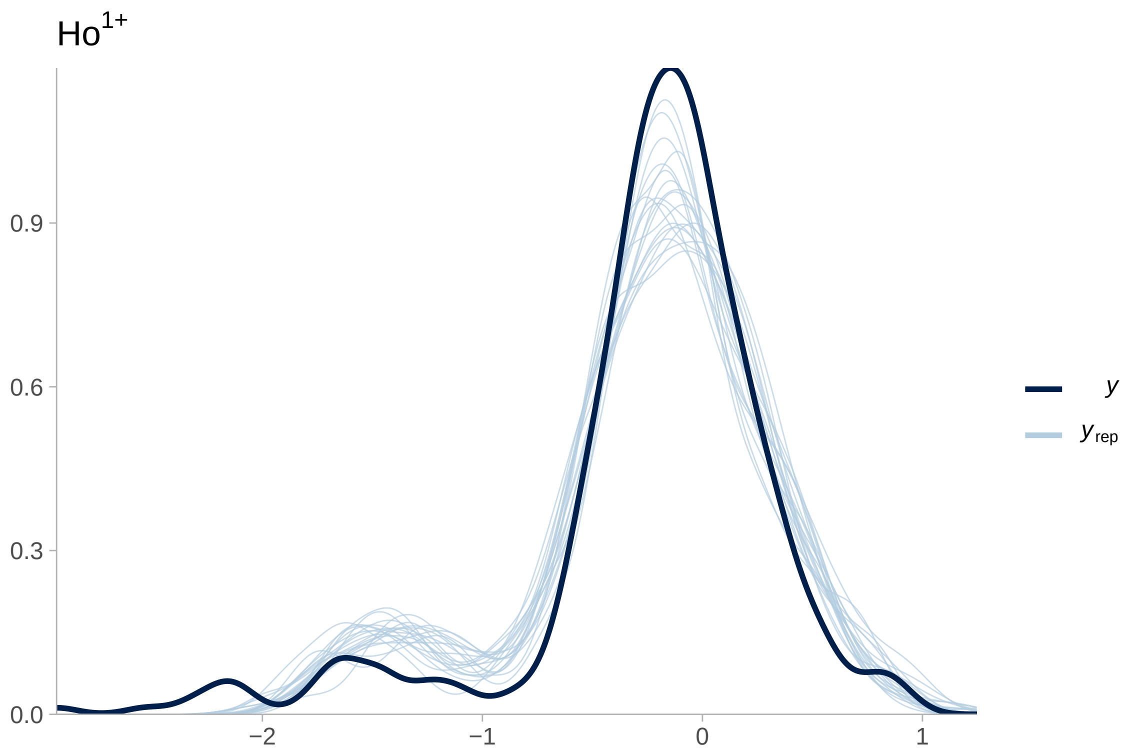

Sc
1.5
1.0
0.5
−1.5
−1.0
−0.5
0.0
0.5
y
yrep


Y
2.0
1.5
1.0
0.5
−1.0
−0.5
0.0
0.5
y
yrep


Be
1.0
0.5
−1
0
1
y
y rep


Co
2.0
1.5
1.0
0.5
−0.8
−0.4
0.0
0.4
y
y rep





There are several indications from the plots above that the fitted model should be improved. The simulated datasets don't match particularly well to the observed data for most of the methods, except for perhaps the alternative isotope, In, and Y. In most cases, it appears that the model predictions are underdispersed relative to the data, indicating that additional structure may be helpful.

No further checks on this model are necessary at this point because it is clear that improvement is needed. However, cross-validation is performed below in order to quantify potential improvements from this initial model moving forward.

Leave-one-out CV

Below, a Bayesian leave-one-out cross validation ('LOO-CV') assessment is conducted using the **loo** package in **R**, which implements fast and stable computations for approximate LOO-CV (and WAIC) (Vehtari, Gelman, and Gabry 2017). These computations estimate pointwise, out-of-sample prediction accuracy from a fitted model by using the log-likelihood evaluated at the posterior simulations of the parameters. The **brms** package automatically calculates the log-likelihood for all relevant models, making it simple to use in conjunction with the **loo** package. For additional information on **loo** and LOO-CV in general, see: http://mc-stan.org/loo/.

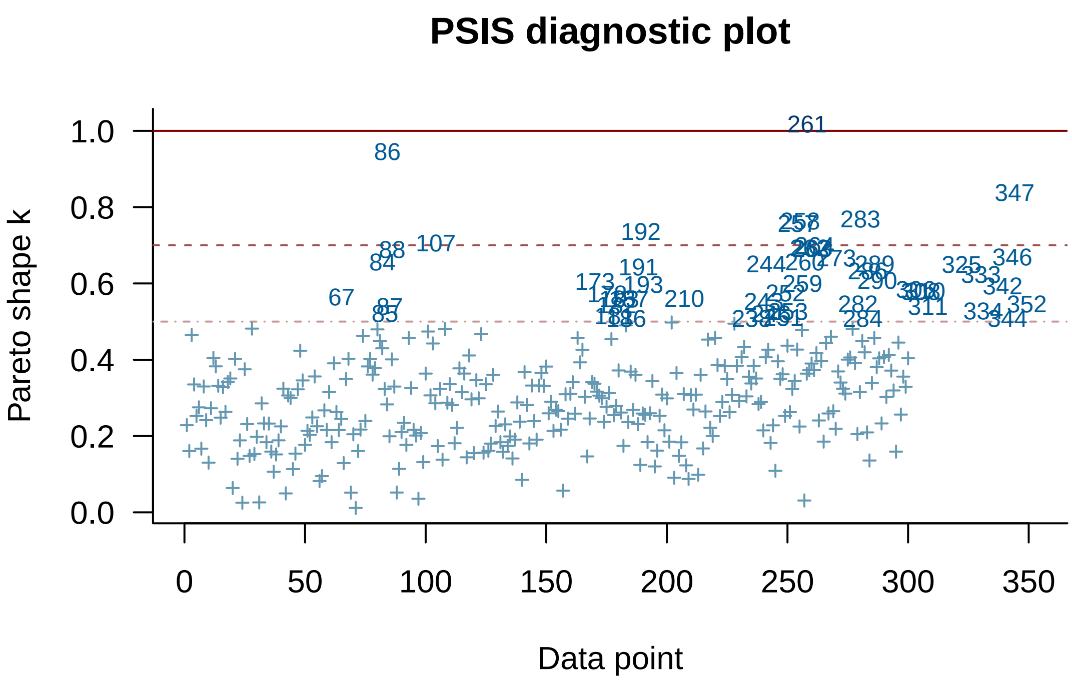
Pareto-smoothed importance sampling is the method used in **loo** for approximating the true leave-one-out posterior. The model is fit once, the log-likelihood for each data point is saved, and `loo()` utilizes that calculation to re-weight and approximate the leave-one-out posterior. This is convenient because it enables model evaluations via leave-one-out estimates without the need to fit the model N times. However, the approximation can fail, and the **loo** package provides some diagnostics that indicate when failure is more likely. In particular, each data point is assigned an importance ratio, k, which indicates how "influential" it may be. As such, too many data points with high (pareto-k > 0.7) or very high (pareto-k > 1) influence may indicate problems with the approximation. When there are too many such points, it is generally recommended to re-fit the model with a more traditional leave-k-out method, such as k-fold. Note that observations with high k are only problematic in the sense of trying to approximate the leave-one-out posterior, and there isn't *necessarily* anything inherently wrong with them. However, observations with high $\hat{k}$ may suggest an outsized influence on the posterior (Gabry et al. 2019).

```
##
## Computed from 4000 by 352 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo   3453.1  79.3
## p_loo       327.9  16.7
## looic     -6906.3 158.5
## ------
## Monte Carlo SE of elpd_loo is NA.
##
## Pareto k diagnostic values:
##                          Count Pct.    Min. n_eff
## (-Inf, 0.5]   (good)     300   85.2%   460
##  (0.5, 0.7]   (ok)        44   12.5%   87
##    (0.7, 1]   (bad)        7    2.0%   15
##    (1, Inf)   (very bad)   1    0.3%   28
## See help('pareto-k-diagnostic') for details.
```

The **loo** calculation resulted in 8 potentially "problematic" observations (i.e., Pareto k > 0.7). The next step is to refit the model 8 times in order to compute the exact cross validation for each of the 8 problematic observations ("reloo"). If the "reloo" pareto-k are all less than 0.7, then those 8 calculations can be substituted for computing the pointwise contributions to total $elpd_{loo}$. Otherwise, a k-fold or similar method should be used to compute a true leave-k-out posterior density.

With regard to the other outputs from the `summary(loo)` call, reference: https://mc-stan.org/loo/reference/loo-glossary.html. In short, the $elpd_{loo}$ is a way to compare models. For each observation, a calculation of how "surprised" the model is to see the left-out data point is made based on the predictive density at the observed value. The $elpd_{loo}$ is a sum of these individual contributions. The p_loo, by comparison, is the difference between the $elpd_{loo}$ and the non-CV log posterior predictive density, and may be interpreted as the effective number of parameters. The *looic*, finally, is just $-2^{elpd_{loo}}$ to provide output on the conventional scale of "deviance" or AIC.

The pareto-k can be plotted to get an indication of which specific observations are problematic and may have outsized influence.




```r
load("full-analysis-files/loo_1.rda")
load("full-analysis-files/df_mv_as.rda")
df_mv_as[which(loo_1$diagnostics$pareto_k > 0.7), ] %>% print()
```

```
## # A tibble: 8 x 14
##   ider      matrix day_expt tune       Alt      Ho2     In     Std      Sc       Y
##   <chr>     <fct>  <fct>    <fct>    <dbl>    <dbl>  <dbl>   <dbl>   <dbl>  <dbl>
## 1 HCl330LHe HCl    330      LHe     0.0385  0.142  -1.10    0.0848 -0.143  -0.833
## 2 SO4510LHe SO4    510      LHe     1.10    0.982   0.555  0.906   0.894   0.624
## 3 Mg316HHe  Mg     316      HHe    -0.0198  0.399   0.549  0.0373  0.0796  0.363
## 4 Na330HHe  Na     330      HHe     0.0002  0.122  -0.657  0.0799 -1.17   -0.944
## 5 Mg330HHe  Mg     330      HHe    -0.195  -0.0796 -0.877 -0.0838 -1.56   -1.20
## 6 SO4330HHe SO4    330      HHe     0.239   0.420  -0.905  0.349   0.181  -0.741
## 7 SO4510HHe SO4    510      HHe     0.816   0.576   0.382  0.884   0.598   0.424
## 8 K516HHe   K      516      HHe    -0.0162 -0.0628  0.266  0.0535 -1.06   -0.259
## # ... with 4 more variables: Be <dbl>, Co <dbl>, Th <dbl>, Ho1 <dbl>
```

All of the potentially problematic observations look to be associated with the 250ppm matrices and half of them on from 3/30, the date of the cone change.

In any case, a re-loo for these 8 observations is needed calculate the $elpd_{loo}$.

```
##
## Computed from 4000 by 352 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo   3452.0  79.6
## p_loo       329.0  17.1
## looic     -6904.0 159.2
## ------
## Monte Carlo SE of elpd_loo is 0.7.
##
## Pareto k diagnostic values:
##                          Count Pct.    Min. n_eff
## (-Inf, 0.5]   (good)     308   87.5%   15
##  (0.5, 0.7]   (ok)        44   12.5%   87
##    (0.7, 1]   (bad)        0    0.0%   <NA>
##    (1, Inf)   (very bad)   0    0.0%   <NA>
##
## All Pareto k estimates are ok (k < 0.7).
## See help('pareto-k-diagnostic') for details.
```

The reloo eliminated concerns with the potentially problematic observations with regard to the $elpd_{loo}$. This information can be used to compare this model with others to follow.

A more flexible arsenic model

For the next model, the σ term is also assigned a linear predictor. This is often referred to as a distributional model or a model for heterogeneous variances. From the exploratory plots above, this seemed like a reasonable next step, given apparent heterogeneity of variances for some methods across matrices, tunes, and days. In this model, σ is allowed to vary in a similar manner as μ. That is, σ is modeled as a linear function of a fixed tune effect and varying effects for matrix and day of the experiment. For more information on fitting distributional models in **brms** see (Bürkner 2017) and: https://cran.r-project.org/web/packages/brms/vignettes/brms_distreg.html

```
load("full-analysis-files/df_mv_as.rda")

bf_Std <- bf(Std ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Alt <- bf(Alt ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho2 <- bf(Ho2 ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_In <- bf(In ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Sc <- bf(Sc ~ tune + (1 | matrix) + (1 | day_expt),
```

```
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Y <- bf(Y ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Be <- bf(Be ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Co <- bf(Co ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Th <- bf(Th ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho1 <- bf(Ho1 ~ tune + (1 | matrix) + (1 | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

mod2 <- brm(bf_Std +
              bf_Alt +
              bf_Ho2 +
              bf_In +
              bf_Sc +
              bf_Y +
              bf_Be +
              bf_Co +
              bf_Th +
              bf_Ho1 +
              set_rescor(TRUE),
  data = df_mv_as,
  prior = c(prior(normal(0, 1), class = "Intercept", resp = "Std"),
            prior(normal(0, 1), class = "Intercept", resp = "Alt"),
            prior(normal(0, 1), class = "Intercept", resp = "Ho2"),
            prior(normal(0, 1), class = "Intercept", resp = "In"),
            prior(normal(0, 1), class = "Intercept", resp = "Sc"),
            prior(normal(0, 1), class = "Intercept", resp = "Y"),
            prior(normal(0, 1), class = "Intercept", resp = "Be"),
            prior(normal(0, 1), class = "Intercept", resp = "Co"),
            prior(normal(0, 1), class = "Intercept", resp = "Th"),
            prior(normal(0, 1), class = "Intercept", resp = "Ho1"),

            prior(normal(0, 1), class = "b", resp = "Std"),
            prior(normal(0, 1), class = "b", resp = "Alt"),
            prior(normal(0, 1), class = "b", resp = "Ho2"),
            prior(normal(0, 1), class = "b", resp = "In"),
            prior(normal(0, 1), class = "b", resp = "Sc"),
            prior(normal(0, 1), class = "b", resp = "Y"),
            prior(normal(0, 1), class = "b", resp = "Be"),
```

```r
          prior(normal(0, 1), class = "b", resp = "Co"),
          prior(normal(0, 1), class = "b", resp = "Th"),
          prior(normal(0, 1), class = "b", resp = "Ho1"),

          prior(normal(0, 1), class = "sd", resp = "Std"),
          prior(normal(0, 1), class = "sd", resp = "Alt"),
          prior(normal(0, 1), class = "sd", resp = "Ho2"),
          prior(normal(0, 1), class = "sd", resp = "In"),
          prior(normal(0, 1), class = "sd", resp = "Sc"),
          prior(normal(0, 1), class = "sd", resp = "Y"),
          prior(normal(0, 1), class = "sd", resp = "Be"),
          prior(normal(0, 1), class = "sd", resp = "Co"),
          prior(normal(0, 1), class = "sd", resp = "Th"),
          prior(normal(0, 1), class = "sd", resp = "Ho1"),

          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Std"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Alt"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Ho2"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "In"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Sc"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Y"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Be"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Co"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Th"),
          prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Ho1"),

          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Std"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Alt"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Ho2"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "In"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Sc"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Y"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Be"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Co"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Th"),
          prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Ho1"),

          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Std"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Alt"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Ho2"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "In"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Sc"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Y"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Be"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Co"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Th"),
          prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Ho1")#,

          prior(lkj(1), class = "rescor")
          ),
  control = list(adapt_delta = 0.95, max_treedepth = 14),
  init_r = 0.05,
  save_pars = save_pars(all = TRUE),
```

```
  seed = 65112,
  chains=4,
  iter=3000,
  cores=4 )

save(mod2, file = "full-analysis-files/mod2_As_mv.rda")
```

Tabular parameter estimates

Again, a summary of the posterior estimates.

```
##  Family: MV(gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian
##   Links: mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
## Formula: Std ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Alt ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho2 ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          In ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Sc ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Y ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Be ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Co ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Th ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho1 ~ tune + (1 | matrix) + (1 | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##    Data: df_mv_as (Number of observations: 352)
##   Draws: 4 chains, each with iter = 3000; warmup = 1500; thin = 1;
##          total post-warmup draws = 6000
##
## Priors:
## b_Alt ~ normal(0, 1)
## b_Alt_sigma ~ normal(0, 1)
## b_Be ~ normal(0, 1)
## b_Be_sigma ~ normal(0, 1)
## b_Co ~ normal(0, 1)
```

```
## b_Co_sigma ~ normal(0, 1)
## b_Ho1 ~ normal(0, 1)
## b_Ho1_sigma ~ normal(0, 1)
## b_Ho2 ~ normal(0, 1)
## b_Ho2_sigma ~ normal(0, 1)
## b_In ~ normal(0, 1)
## b_In_sigma ~ normal(0, 1)
## b_Sc ~ normal(0, 1)
## b_Sc_sigma ~ normal(0, 1)
## b_Std ~ normal(0, 1)
## b_Std_sigma ~ normal(0, 1)
## b_Th ~ normal(0, 1)
## b_Th_sigma ~ normal(0, 1)
## b_Y ~ normal(0, 1)
## b_Y_sigma ~ normal(0, 1)
## Intercept_Alt ~ normal(0, 1)
## Intercept_Alt_sigma ~ normal(-1, 2)
## Intercept_Be ~ normal(0, 1)
## Intercept_Be_sigma ~ normal(-1, 2)
## Intercept_Co ~ normal(0, 1)
## Intercept_Co_sigma ~ normal(-1, 2)
## Intercept_Ho1 ~ normal(0, 1)
## Intercept_Ho1_sigma ~ normal(-1, 2)
## Intercept_Ho2 ~ normal(0, 1)
## Intercept_Ho2_sigma ~ normal(-1, 2)
## Intercept_In ~ normal(0, 1)
## Intercept_In_sigma ~ normal(-1, 2)
## Intercept_Sc ~ normal(0, 1)
## Intercept_Sc_sigma ~ normal(-1, 2)
## Intercept_Std ~ normal(0, 1)
## Intercept_Std_sigma ~ normal(-1, 2)
## Intercept_Th ~ normal(0, 1)
## Intercept_Th_sigma ~ normal(-1, 2)
## Intercept_Y ~ normal(0, 1)
## Intercept_Y_sigma ~ normal(-1, 2)
## Lrescor ~ lkj_corr_cholesky(1)
## sd_Alt ~ normal(0, 1)
## sd_Alt_sigma ~ normal(0, 1)
## sd_Be ~ normal(0, 1)
## sd_Be_sigma ~ normal(0, 1)
## sd_Co ~ normal(0, 1)
## sd_Co_sigma ~ normal(0, 1)
## sd_Ho1 ~ normal(0, 1)
## sd_Ho1_sigma ~ normal(0, 1)
## sd_Ho2 ~ normal(0, 1)
## sd_Ho2_sigma ~ normal(0, 1)
## sd_In ~ normal(0, 1)
## sd_In_sigma ~ normal(0, 1)
## sd_Sc ~ normal(0, 1)
## sd_Sc_sigma ~ normal(0, 1)
## sd_Std ~ normal(0, 1)
## sd_Std_sigma ~ normal(0, 1)
## sd_Th ~ normal(0, 1)
## sd_Th_sigma ~ normal(0, 1)
```

```
## sd_Y ~ normal(0, 1)
## sd_Y_sigma ~ normal(0, 1)
##
## Group-Level Effects:
## ~day_expt (Number of levels: 8)
##                         Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## sd(Std_Intercept)           0.06      0.02     0.03     0.11 1.00     3615
## sd(sigma_Std_Intercept)     0.27      0.11     0.13     0.53 1.00     2765
## sd(Alt_Intercept)           0.11      0.04     0.06     0.21 1.00     3503
## sd(sigma_Alt_Intercept)     0.30      0.11     0.15     0.57 1.00     3511
## sd(Ho2_Intercept)           0.04      0.02     0.02     0.08 1.00     2503
## sd(sigma_Ho2_Intercept)     0.37      0.14     0.20     0.72 1.00     3580
## sd(In_Intercept)            0.28      0.10     0.16     0.54 1.00     4809
## sd(sigma_In_Intercept)      0.35      0.12     0.19     0.64 1.00     5012
## sd(Sc_Intercept)            0.17      0.06     0.09     0.34 1.00     4209
## sd(sigma_Sc_Intercept)      0.36      0.13     0.19     0.69 1.00     3751
## sd(Y_Intercept)             0.26      0.09     0.14     0.48 1.00     5100
## sd(sigma_Y_Intercept)       0.38      0.13     0.21     0.72 1.00     5461
## sd(Be_Intercept)            0.27      0.09     0.15     0.51 1.00     5045
## sd(sigma_Be_Intercept)      0.43      0.15     0.23     0.80 1.00     4138
## sd(Co_Intercept)            0.24      0.09     0.13     0.45 1.00     3982
## sd(sigma_Co_Intercept)      0.47      0.16     0.27     0.89 1.00     5742
## sd(Th_Intercept)            0.55      0.17     0.32     0.97 1.00     5602
## sd(sigma_Th_Intercept)      0.54      0.18     0.31     1.00 1.00     5673
## sd(Ho1_Intercept)           0.53      0.17     0.31     0.96 1.00     5675
## sd(sigma_Ho1_Intercept)     0.55      0.18     0.31     0.98 1.00     4744
##                         Tail_ESS
## sd(Std_Intercept)           4466
## sd(sigma_Std_Intercept)     4412
## sd(Alt_Intercept)           3864
## sd(sigma_Alt_Intercept)     4680
## sd(Ho2_Intercept)           3415
## sd(sigma_Ho2_Intercept)     4494
## sd(In_Intercept)            4057
## sd(sigma_In_Intercept)      4390
## sd(Sc_Intercept)            3799
## sd(sigma_Sc_Intercept)      3806
## sd(Y_Intercept)             4706
## sd(sigma_Y_Intercept)       3757
## sd(Be_Intercept)            4557
## sd(sigma_Be_Intercept)      4482
## sd(Co_Intercept)            4211
## sd(sigma_Co_Intercept)      4732
## sd(Th_Intercept)            4141
## sd(sigma_Th_Intercept)      4427
## sd(Ho1_Intercept)           4022
## sd(sigma_Ho1_Intercept)     3706
##
## ~matrix (Number of levels: 22)
##                         Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## sd(Std_Intercept)           0.05      0.01     0.04     0.08 1.00     1813
## sd(sigma_Std_Intercept)     0.10      0.05     0.01     0.19 1.00     2181
## sd(Alt_Intercept)           0.05      0.01     0.04     0.08 1.01     1921
## sd(sigma_Alt_Intercept)     0.07      0.04     0.00     0.16 1.00     2551
```

```
## sd(Ho2_Intercept)           0.07      0.01     0.05     0.09 1.00     2862
## sd(sigma_Ho2_Intercept)     0.07      0.04     0.00     0.17 1.00     2286
## sd(In_Intercept)            0.10      0.02     0.07     0.14 1.00     3518
## sd(sigma_In_Intercept)      0.06      0.03     0.01     0.12 1.01      894
## sd(Sc_Intercept)            0.09      0.02     0.06     0.13 1.00     3309
## sd(sigma_Sc_Intercept)      0.19      0.05     0.12     0.30 1.01     1163
## sd(Y_Intercept)             0.07      0.01     0.05     0.09 1.00     3371
## sd(sigma_Y_Intercept)       0.06      0.03     0.01     0.13 1.02      505
## sd(Be_Intercept)            0.09      0.02     0.06     0.12 1.00     3481
## sd(sigma_Be_Intercept)      0.12      0.05     0.03     0.21 1.00     1284
## sd(Co_Intercept)            0.04      0.01     0.03     0.06 1.00     2664
## sd(sigma_Co_Intercept)      0.08      0.03     0.02     0.15 1.00      694
## sd(Th_Intercept)            0.28      0.05     0.21     0.40 1.00     3082
## sd(sigma_Th_Intercept)      0.04      0.02     0.00     0.09 1.00     1204
## sd(Ho1_Intercept)           0.20      0.03     0.15     0.28 1.00     2981
## sd(sigma_Ho1_Intercept)     0.03      0.02     0.00     0.06 1.00     2258
##                         Tail_ESS
## sd(Std_Intercept)           2752
## sd(sigma_Std_Intercept)     2191
## sd(Alt_Intercept)           3385
## sd(sigma_Alt_Intercept)     2997
## sd(Ho2_Intercept)           3404
## sd(sigma_Ho2_Intercept)     3157
## sd(In_Intercept)            4200
## sd(sigma_In_Intercept)      1448
## sd(Sc_Intercept)            3995
## sd(sigma_Sc_Intercept)      3376
## sd(Y_Intercept)             3575
## sd(sigma_Y_Intercept)       1730
## sd(Be_Intercept)            4420
## sd(sigma_Be_Intercept)      1874
## sd(Co_Intercept)            4029
## sd(sigma_Co_Intercept)      1106
## sd(Th_Intercept)            4260
## sd(sigma_Th_Intercept)      2592
## sd(Ho1_Intercept)           3655
## sd(sigma_Ho1_Intercept)     3281
##
## Population-Level Effects:
##                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Std_Intercept           0.11      0.03     0.06     0.17 1.00     3033     3538
## sigma_Std_Intercept    -2.54      0.11    -2.77    -2.32 1.00     4052     3879
## Alt_Intercept           0.15      0.04     0.06     0.23 1.00     3174     3724
## sigma_Alt_Intercept    -2.35      0.13    -2.59    -2.09 1.00     4087     4131
## Ho2_Intercept           0.11      0.02     0.07     0.16 1.00     3036     3628
## sigma_Ho2_Intercept    -2.53      0.15    -2.83    -2.23 1.00     3808     4110
## In_Intercept           -0.04      0.11    -0.26     0.16 1.00     2489     3302
## sigma_In_Intercept     -2.36      0.14    -2.64    -2.08 1.00     3332     4004
## Sc_Intercept           -0.01      0.07    -0.15     0.13 1.00     2618     3312
## sigma_Sc_Intercept     -2.23      0.15    -2.52    -1.93 1.00     3661     4013
## Y_Intercept            -0.06      0.09    -0.25     0.13 1.00     2566     3332
## sigma_Y_Intercept      -2.38      0.15    -2.67    -2.07 1.00     2858     2805
## Be_Intercept            0.00      0.10    -0.20     0.20 1.00     2913     3808
## sigma_Be_Intercept     -1.96      0.17    -2.30    -1.62 1.00     3292     3494
```

```
## Co_Intercept             0.01      0.09    -0.18     0.19 1.00     2588     3070
## sigma_Co_Intercept      -2.30      0.18    -2.66    -1.92 1.00     2990     3307
## Th_Intercept            -0.22      0.20    -0.62     0.21 1.00     2863     3481
## sigma_Th_Intercept      -1.81      0.21    -2.25    -1.38 1.00     2904     3629
## Ho1_Intercept           -0.20      0.19    -0.59     0.20 1.00     2244     2965
## sigma_Ho1_Intercept     -2.04      0.21    -2.46    -1.61 1.00     3056     3598
## Std_tuneHHe              0.06      0.01     0.03     0.08 1.01      907     1508
## sigma_Std_tuneHHe        0.33      0.07     0.19     0.46 1.00     8229     5182
## Alt_tuneHHe              0.05      0.02     0.01     0.08 1.01      886     1479
## sigma_Alt_tuneHHe        0.33      0.08     0.18     0.49 1.00     7624     4970
## Ho2_tuneHHe              0.08      0.01     0.05     0.11 1.01      891     1595
## sigma_Ho2_tuneHHe        0.62      0.07     0.47     0.76 1.00     5943     4605
## In_tuneHHe               0.07      0.03     0.01     0.11 1.01      647      971
## sigma_In_tuneHHe         0.79      0.05     0.70     0.89 1.00     4889     5044
## Sc_tuneHHe              -0.02      0.03    -0.08     0.03 1.01      725     1601
## sigma_Sc_tuneHHe         0.92      0.06     0.80     1.03 1.00     6327     5232
## Y_tuneHHe                0.01      0.03    -0.05     0.06 1.01      617     1013
## sigma_Y_tuneHHe          0.90      0.05     0.81     0.99 1.00     5098     4938
## Be_tuneHHe               0.08      0.04    -0.02     0.15 1.01      787     1208
## sigma_Be_tuneHHe         1.02      0.06     0.90     1.14 1.00     6620     4862
## Co_tuneHHe               0.05      0.03    -0.01     0.10 1.01      615      935
## sigma_Co_tuneHHe         0.86      0.04     0.77     0.95 1.00     5689     4523
## Th_tuneHHe               0.12      0.05     0.01     0.21 1.01      774     1181
## sigma_Th_tuneHHe         0.87      0.06     0.74     0.98 1.00     5701     5192
## Ho1_tuneHHe              0.09      0.04    -0.01     0.15 1.01      715     1003
## sigma_Ho1_tuneHHe        0.90      0.05     0.80     1.01 1.00     5444     4524
##
## Residual Correlations:
##                  Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## rescor(Std,Alt)      0.69      0.03     0.63     0.75 1.00     4174     4478
## rescor(Std,Ho2)      0.65      0.03     0.58     0.71 1.00     4107     5092
## rescor(Alt,Ho2)      0.62      0.04     0.54     0.69 1.00     4736     5105
## rescor(Std,In)       0.61      0.04     0.53     0.68 1.00     3053     4368
## rescor(Alt,In)       0.60      0.04     0.52     0.67 1.00     3180     3792
## rescor(Ho2,In)       0.65      0.04     0.58     0.72 1.00     3409     4163
## rescor(Std,Sc)       0.61      0.04     0.52     0.68 1.00     2450     3786
## rescor(Alt,Sc)       0.59      0.04     0.50     0.67 1.00     1961     3439
## rescor(Ho2,Sc)       0.69      0.03     0.62     0.76 1.01     1796     4334
## rescor(In,Sc)        0.73      0.03     0.67     0.79 1.00     1710     3960
## rescor(Std,Y)        0.65      0.04     0.57     0.71 1.00     2817     4495
## rescor(Alt,Y)        0.63      0.04     0.55     0.70 1.00     2536     3823
## rescor(Ho2,Y)        0.72      0.03     0.65     0.77 1.01     2397     4278
## rescor(In,Y)         0.95      0.01     0.93     0.96 1.01     1192     3500
## rescor(Sc,Y)         0.87      0.02     0.84     0.90 1.00     3303     4691
## rescor(Std,Be)       0.49      0.05     0.39     0.58 1.00     3064     4036
## rescor(Alt,Be)       0.48      0.05     0.38     0.57 1.00     2924     4175
## rescor(Ho2,Be)       0.55      0.04     0.46     0.63 1.00     3025     4296
## rescor(In,Be)        0.64      0.04     0.55     0.71 1.00     1710     3442
## rescor(Sc,Be)        0.61      0.04     0.52     0.69 1.00     2191     4027
## rescor(Y,Be)         0.65      0.04     0.57     0.72 1.00     1901     3844
## rescor(Std,Co)       0.64      0.04     0.57     0.71 1.00     2855     4082
## rescor(Alt,Co)       0.62      0.04     0.54     0.69 1.00     2762     3805
## rescor(Ho2,Co)       0.73      0.03     0.67     0.79 1.00     2977     3654
## rescor(In,Co)        0.86      0.02     0.82     0.89 1.00     1542     3425
```

```
## rescor(Sc,Co)        0.83      0.02     0.78     0.87 1.01     1721     3908
## rescor(Y,Co)         0.91      0.01     0.88     0.93 1.01     1586     3370
## rescor(Be,Co)        0.87      0.02     0.83     0.90 1.00     2658     4560
## rescor(Std,Th)       0.34      0.05     0.23     0.44 1.00     3292     4497
## rescor(Alt,Th)       0.36      0.05     0.26     0.46 1.00     4114     4714
## rescor(Ho2,Th)       0.30      0.06     0.18     0.41 1.00     2509     3711
## rescor(In,Th)        0.75      0.03     0.69     0.80 1.00     2015     3855
## rescor(Sc,Th)        0.24      0.06     0.11     0.36 1.01     1191     2758
## rescor(Y,Th)         0.57      0.05     0.48     0.66 1.01     1357     3212
## rescor(Be,Th)        0.63      0.04     0.54     0.70 1.00     2373     4444
## rescor(Co,Th)        0.63      0.04     0.54     0.70 1.00     1725     3357
## rescor(Std,Ho1)      0.44      0.05     0.34     0.53 1.00     3077     4364
## rescor(Alt,Ho1)      0.48      0.05     0.38     0.56 1.00     3916     4466
## rescor(Ho2,Ho1)      0.35      0.06     0.23     0.46 1.00     2210     3481
## rescor(In,Ho1)       0.81      0.02     0.76     0.85 1.01     1511     2675
## rescor(Sc,Ho1)       0.31      0.06     0.19     0.43 1.01     1139     2518
## rescor(Y,Ho1)        0.65      0.04     0.56     0.72 1.01     1191     2499
## rescor(Be,Ho1)       0.60      0.04     0.52     0.67 1.00     2155     3825
## rescor(Co,Ho1)       0.66      0.04     0.57     0.72 1.01     1554     3213
## rescor(Th,Ho1)       0.97      0.00     0.96     0.98 1.00     2470     4121
##
## 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).
```

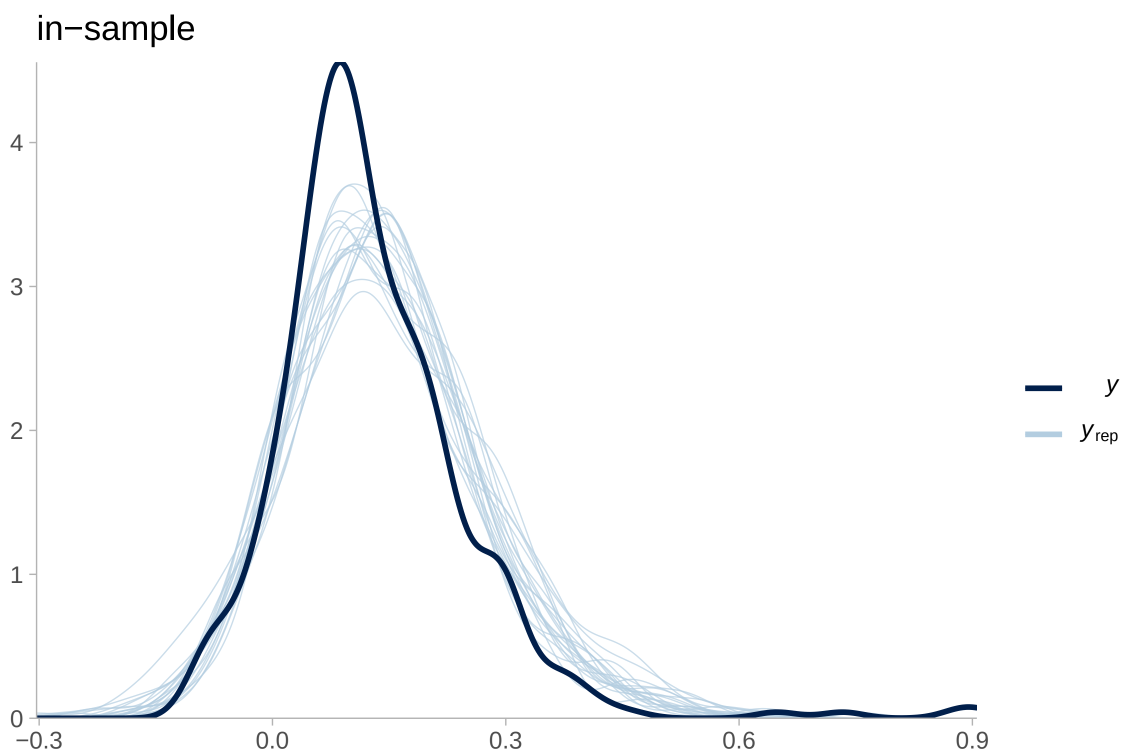
Again, no glaring issues with the HMC sampling.

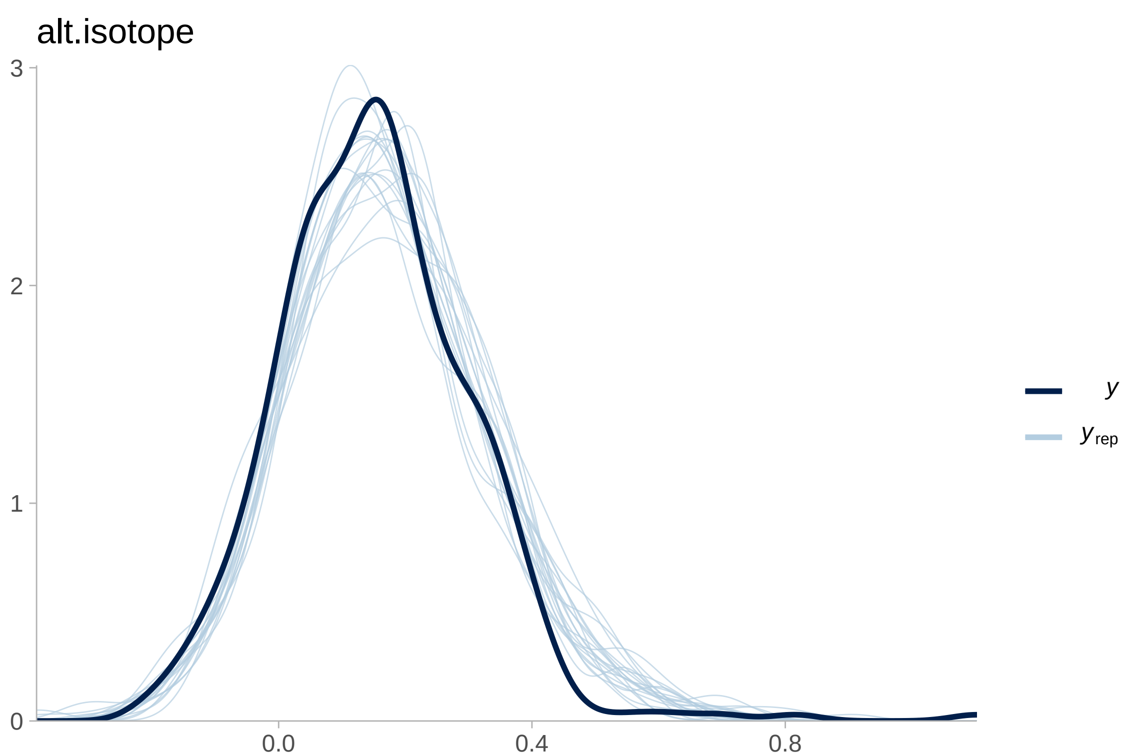
Model checks

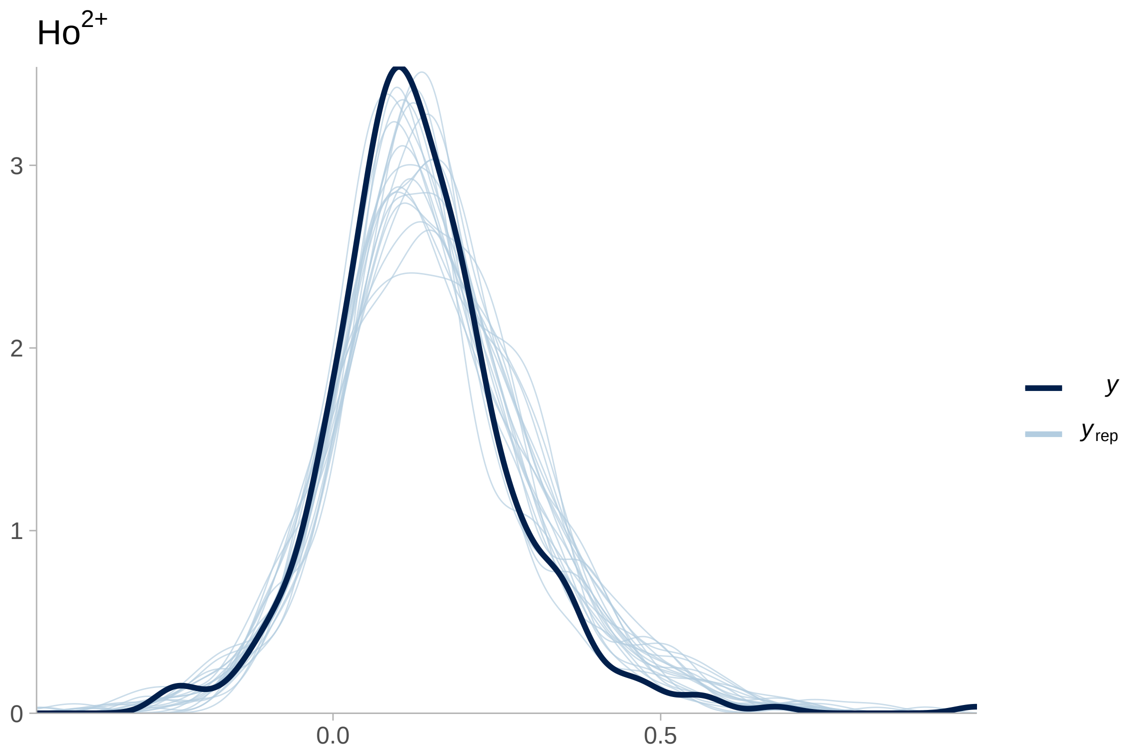
Posterior predictive checks are useful for visualizing the extent to which the fitted model generates replicate data that resembles the observed data.

Density overlay

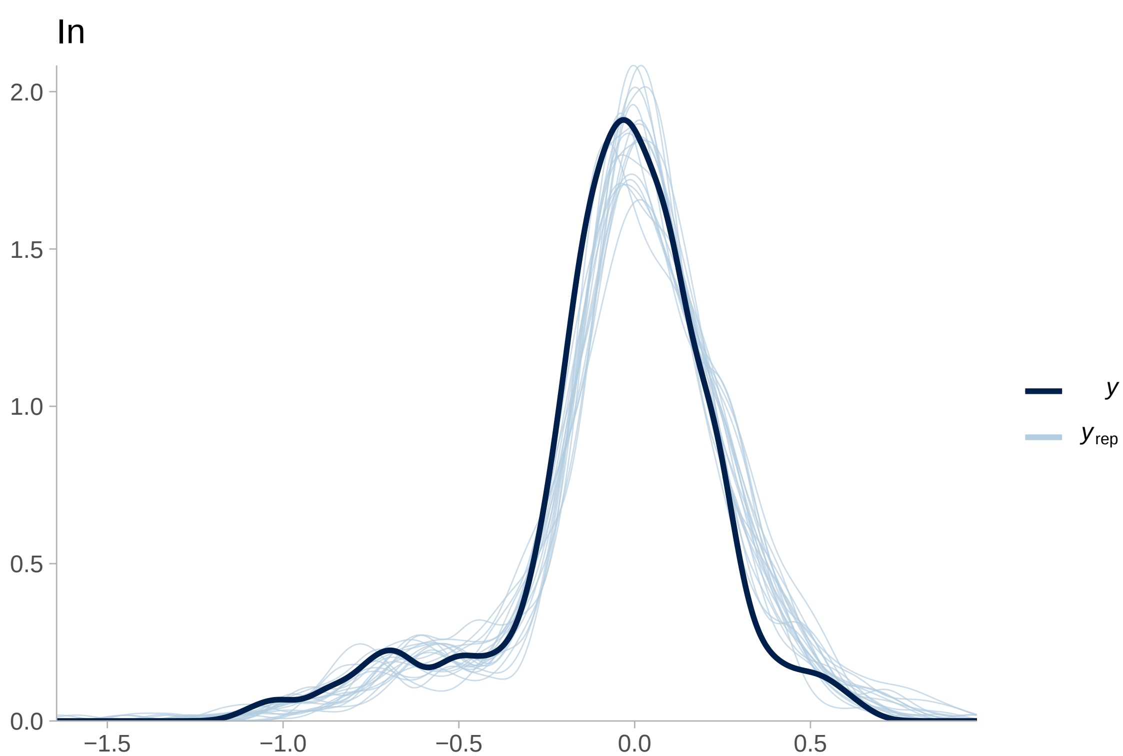
Another posterior predictive check using the density overlay.


in−sample
4
3
2
1
0
−0.3
0.0
0.3
0.6
0.9
y
y rep

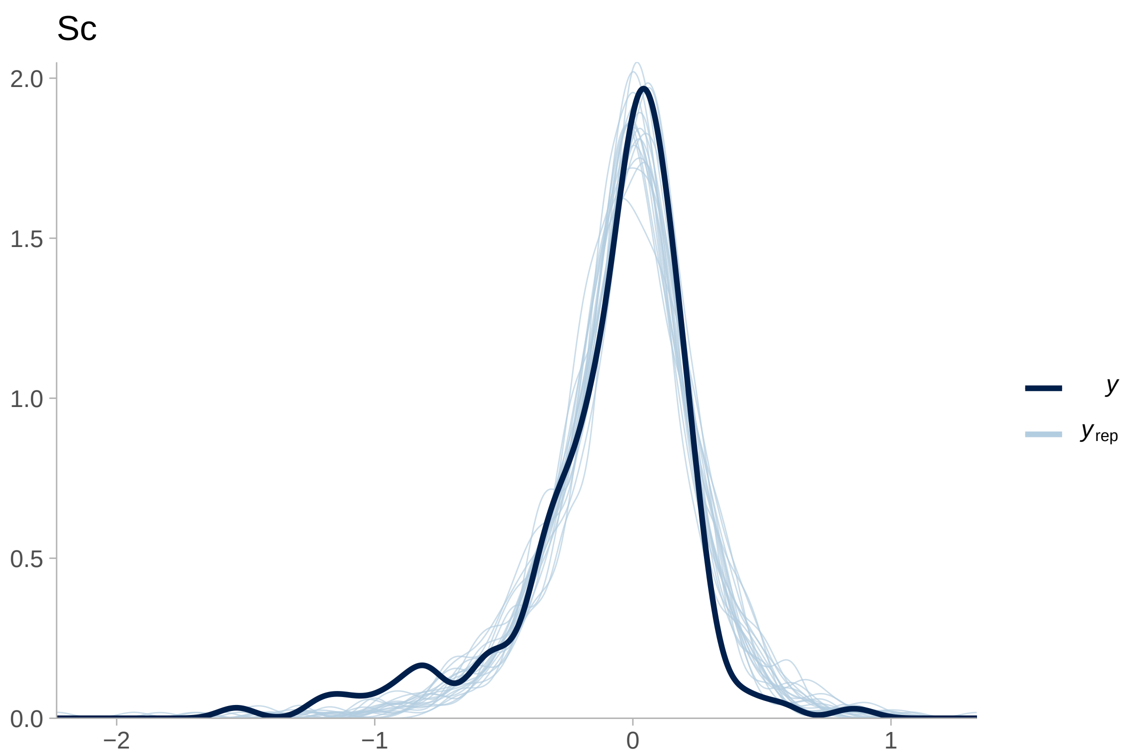

alt.isotope
3
2
1
0
0.0
0.4
0.8
y
y rep

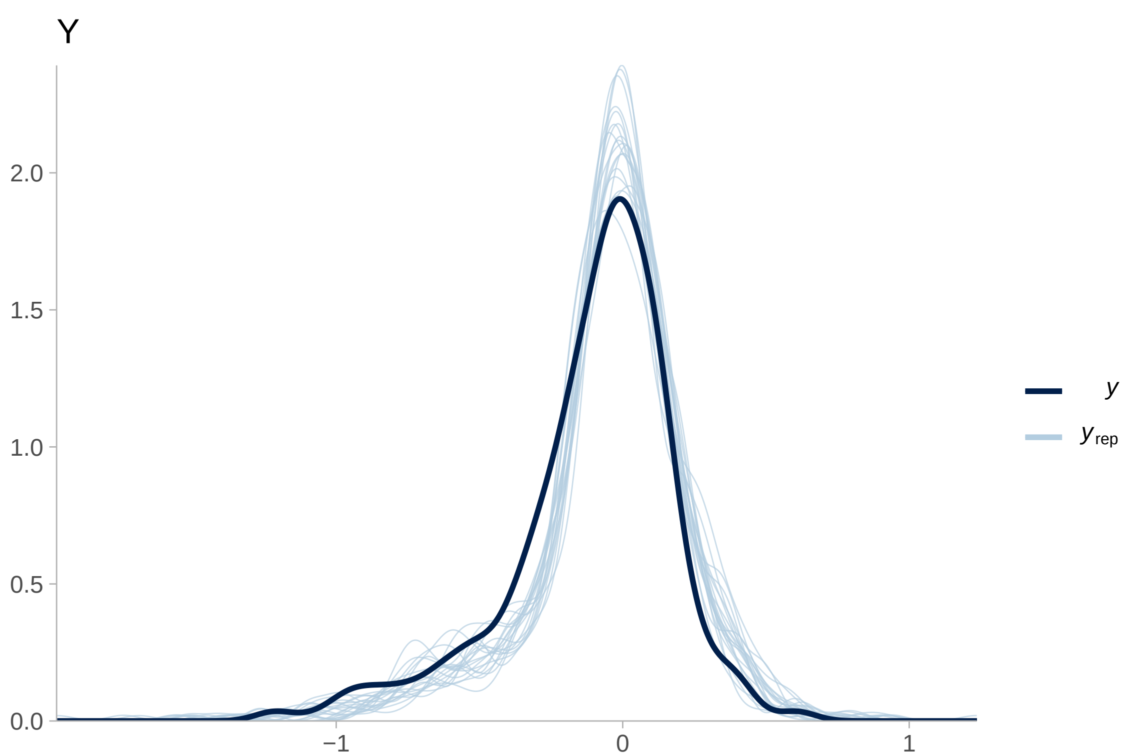


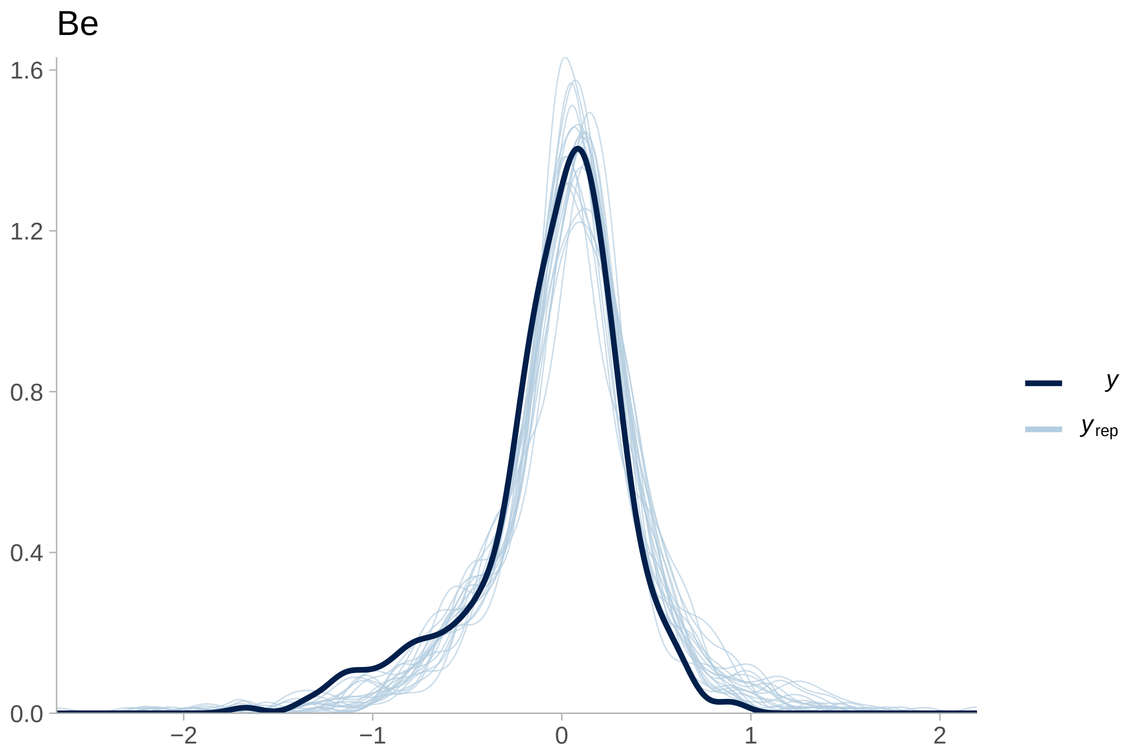
Ho^{2+}
3
2
1
0
0.0
0.5
y
y_{rep}

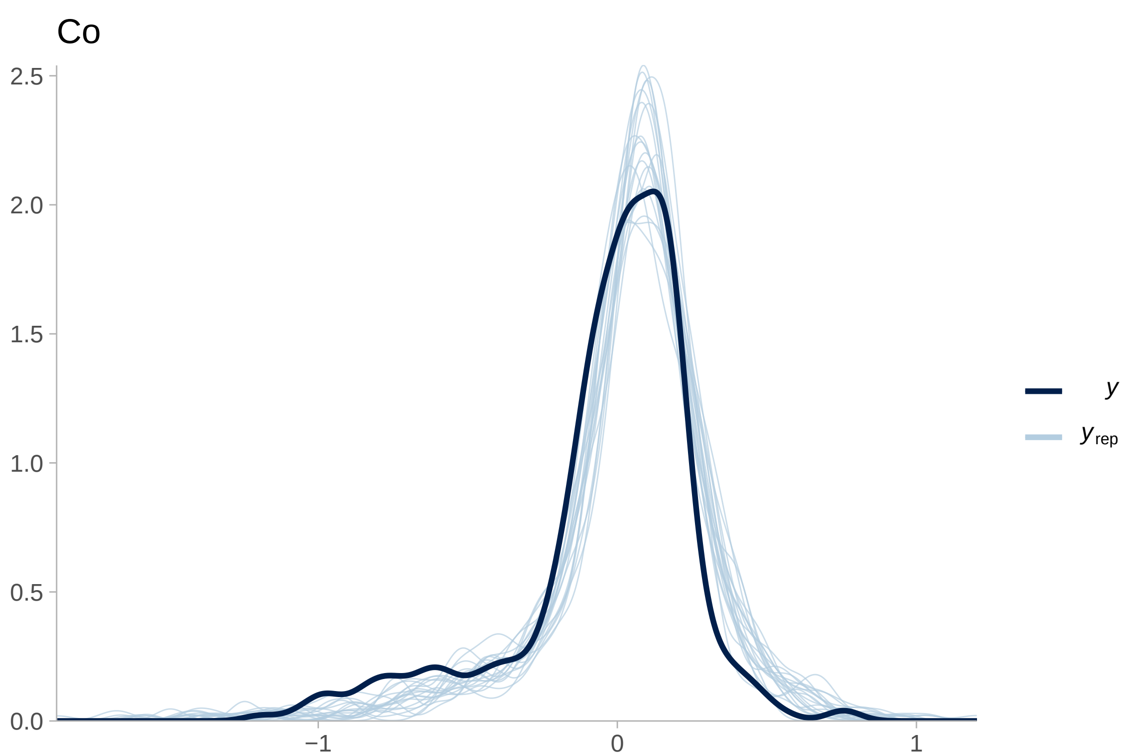


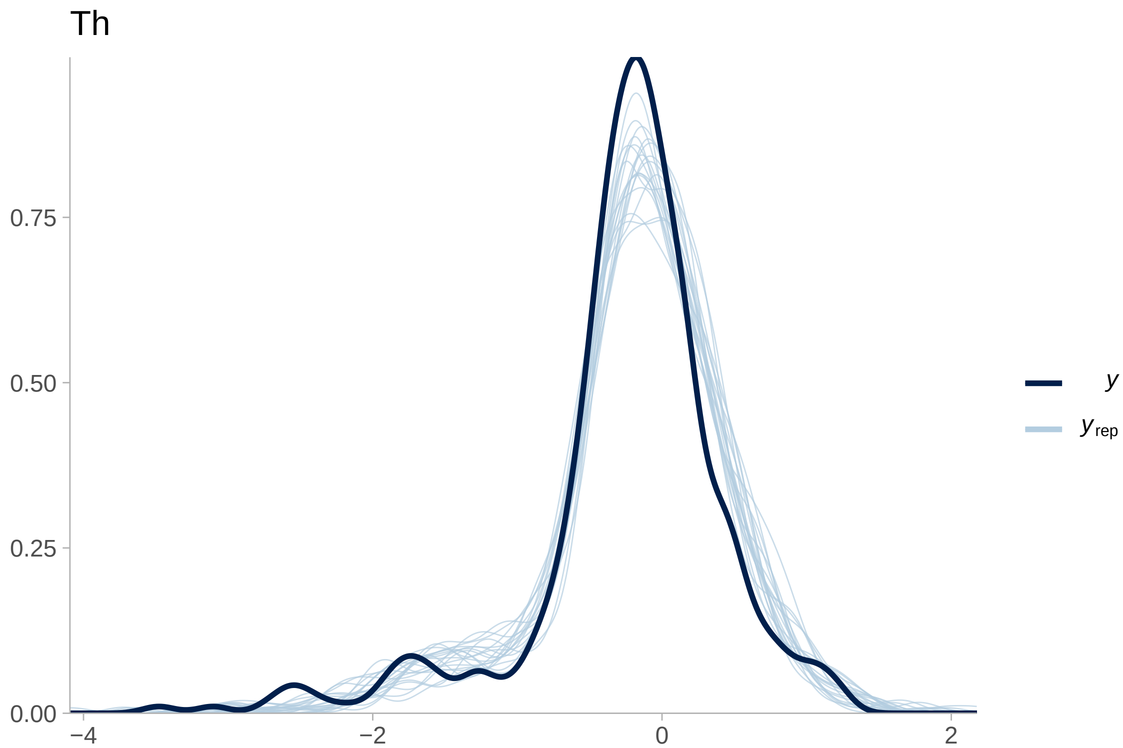
In
2.0
1.5
1.0
0.5
0.0
−1.5
−1.0
−0.5
0.0
0.5
y
y_{rep}

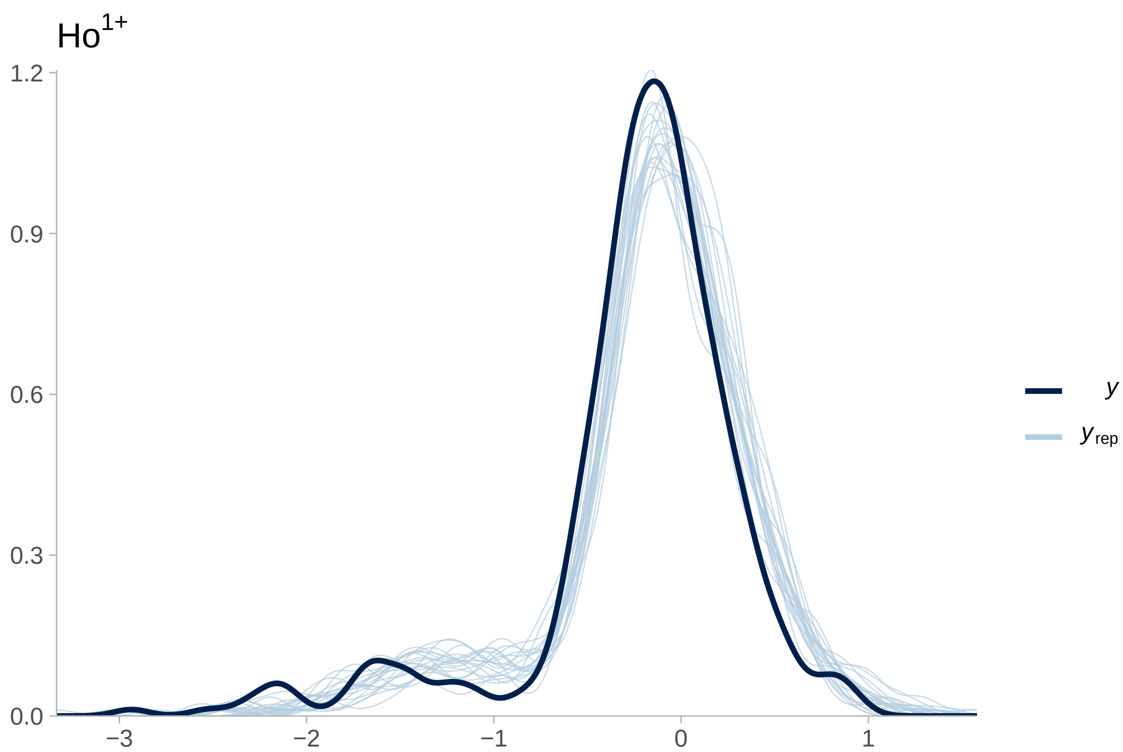

Sc
2.0
1.5
1.0
0.5
0.0
−2
−1
0
1
y
y_{rep}


Y
2.0
1.5
1.0
0.5
0.0
−1
0
1
y
y_{rep}


Be
1.6
1.2
0.8
0.4
0.0
−2
−1
0
1
2
y
yrep


Co
2.5
2.0
1.5
1.0
0.5
0.0
−1
0
1
y
yrep





Though a clear improvement over the first model, there are still indications that the model predictions are under-dispersed relative to the observed data for some of the methods. Adding more structure could be helpful.

Leave-one-out CV

The **loo** procedure is performed again, for comparison with the previous and later models.

```
load("full-analysis-files/mod2_As_mv.rda")

loo_2 <- loo(x = mod2,
             pointwise = TRUE,
             moment_match = TRUE,
             save_psis = TRUE,
             cores = 10)
save(loo_2, file = "full-analysis-files/loo_2.rda")
```

```
##
## Computed from 6000 by 352 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo   4123.9  98.0
## p_loo       416.2  18.8
## looic     -8247.7 195.9
## ------
## Monte Carlo SE of elpd_loo is NA.
##
## Pareto k diagnostic values:
##                          Count Pct.    Min. n_eff
## (-Inf, 0.5]   (good)     247   70.2%   473
##  (0.5, 0.7]   (ok)        75   21.3%   149
##    (0.7, 1]   (bad)       27    7.7%   17
##    (1, Inf)   (very bad)   3    0.9%   9
## See help('pareto-k-diagnostic') for details.
```

The **loo** calculation resulted in a large (30) number of potentially "problematic" observations.

```
## # A tibble: 30 x 14
##    ider    matrix day_expt tune      Alt     Ho2      In     Std      Sc       Y
##    <chr>   <fct>  <fct>    <fct>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 water_~ water~ 316      LHe   -0.0553 -0.0113 -0.204  -0.0593 -0.262  -0.240
##  2 water_~ water~ 316      LHe   -0.047  -0.0199 -0.112  -0.0881 -0.169  -0.122
##  3 water_~ water~ 316      LHe   -0.104  -0.0816 -0.274  -0.0951 -0.367  -0.322
##  4 water_~ water~ 316      LHe    0.0505  0.112  -0.0572  0.0225  0.0469 -0.0167
##  5 PO4316~ PO4    316      LHe   -0.0479  0.0836 -0.215   0.0733 -0.0739 -0.184
##  6 water_~ water~ 320      LHe    0.181   0.004  -0.0972  0.102  -0.141  -0.109
##  7 neat_1~ neat_1 320      LHe    0.227   0.0949  0.0259  0.136   0.0539  0.0357
##  8 water_~ water~ 320      LHe    0.0467 -0.0148 -0.0861  0.0962 -0.116  -0.101
##  9 Mg320L~ Mg     320      LHe    0.199   0.0731  0.411   0.153   0.171   0.277
## 10 K320LHe K      320      LHe    0.131   0.0409  0.300   0.0956  0.132   0.192
## # ... with 20 more rows, and 4 more variables: Be <dbl>, Co <dbl>, Th <dbl>,
## #   Ho1 <dbl>
```

Most of these, again, look to be observations in the 250ppm matrices. However, it is not all, so there may be other issues to consider. With so many k > 0.7, it is important to check that the model isn't just terribly mis-specified. Comparing the number of model parameters ($p = 745$) to p_{loo} can be helpful in this regard. In this case, $p_{loo} << p$, suggesting that the model may be mispecified, which would maybe agree with the posterior predictive checks above (see: https://mc-stan.org/loo/reference/loo-glossary.html). However, another reasonable explanation would be that, with so many latent parameters over the different groupings (i.e, varying effects) with relatively few observations per group, leaving out any observations from any particular group can make for a noisy CV process. A solution, in that case, may be to add stronger priors over the standard deviations for the varying effects, but the current priors should already be reasonably informative to stabilize estimates. Therefore, the approach taken below will be to just add more structure to the model, particularly since the posterior predictive checks also hinted at some potential for mis-specification.

With regard to just calculating the CV criteria for this model, given the high number of potentially problematic observations it would be too computationally expensive to reloo, so a k-fold CV is in order. A k-fold CV is also needed for model 1 because it isn't recommended to compare loo-CV and k-fold CV metrics.

K-fold CV

K-fold CV can be performed using the **brms** or **loo** package, and involves refitting the model K times, each time leaving out one-Kth of the original data (http://paul-buerkner.github.io/brms/reference/kfold.html).

First, the k-fold for model 1, where k = 10.

```
load("full-analysis-files/mod1_As_mv.rda")

library(future)
plan(multisession(workers = 10), gc = TRUE)
kfold_1 <- kfold(x = mod1, save_fits = TRUE, K = 10, chains = 4)
save(kfold_1, file = "full-analysis-files/kfold_1.rda")
plan(sequential)
```

Then, for model 2.

```
load("full-analysis-files/mod2_As_mv.rda")

library(future)
plan(multisession(workers = 10), gc = TRUE)
kfold_2 <- kfold(x = mod2, save_fits = TRUE, K = 10, chains = 4)
save(kfold_2, file = "full-analysis-files/kfold_2.rda")
plan(sequential)
```

The results for model 1.

```
##
## Based on 10-fold cross-validation
##
##            Estimate    SE
## elpd_kfold   3397.0  84.1
## p_kfold       384.1  24.3
## kfoldic     -6793.9 168.2
```

And for model 2.

```
##
## Based on 10-fold cross-validation
```

```
##
##            Estimate    SE
## elpd_kfold   4145.8 100.5
## p_kfold       522.5  24.3
## kfoldic     -8291.6 200.9
```

The direct comparison.

```
##      elpd_diff se_diff
## mod3    0.0       0.0
## mod1 -748.8      58.7
```

Clearly, model 2 is preferred by this metric. It has a much larger $elpd_{kfold}$ and the difference from mod1 is much larger than standard error of the difference. It would be expected to perform better out-of-sample. However, the potential mis-specification remains, so another model is constructed below with additional structure.

A final model for arsenic

For the next model, additional structure is added to the μ term. In this model, the tune effect is allowed to vary by matrix and day of the experiment. That is, the difference between between the HHe tune and LHe tune is allowed to vary according to matrix and/or day of the experiment.

```r
load("full-analysis-files/df_mv_as.rda")

bf_Std <- bf(Std ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Alt <- bf(Alt ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho2 <- bf(Ho2 ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_In <- bf(In ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Sc <- bf(Sc ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Y <- bf(Y ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Be <- bf(Be ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
```

```
             family = gaussian())

bf_Co <- bf(Co ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Th <- bf(Th ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho1 <- bf(Ho1 ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

mod3 <- brm(bf_Std +
              bf_Alt +
              bf_Ho2 +
              bf_In +
              bf_Sc +
              bf_Y +
              bf_Be +
              bf_Co +
              bf_Th +
              bf_Ho1 +
              set_rescor(TRUE),
  data = df_mv_as,
  prior = c(prior(normal(0, 1), class = "Intercept", resp = "Std"),
            prior(normal(0, 1), class = "Intercept", resp = "Alt"),
            prior(normal(0, 1), class = "Intercept", resp = "Ho2"),
            prior(normal(0, 1), class = "Intercept", resp = "In"),
            prior(normal(0, 1), class = "Intercept", resp = "Sc"),
            prior(normal(0, 1), class = "Intercept", resp = "Y"),
            prior(normal(0, 1), class = "Intercept", resp = "Be"),
            prior(normal(0, 1), class = "Intercept", resp = "Co"),
            prior(normal(0, 1), class = "Intercept", resp = "Th"),
            prior(normal(0, 1), class = "Intercept", resp = "Ho1"),

            prior(normal(0, 1), class = "b", resp = "Std"),
            prior(normal(0, 1), class = "b", resp = "Alt"),
            prior(normal(0, 1), class = "b", resp = "Ho2"),
            prior(normal(0, 1), class = "b", resp = "In"),
            prior(normal(0, 1), class = "b", resp = "Sc"),
            prior(normal(0, 1), class = "b", resp = "Y"),
            prior(normal(0, 1), class = "b", resp = "Be"),
            prior(normal(0, 1), class = "b", resp = "Co"),
            prior(normal(0, 1), class = "b", resp = "Th"),
            prior(normal(0, 1), class = "b", resp = "Ho1"),

            prior(normal(0, 1), class = "sd", resp = "Std"),
            prior(normal(0, 1), class = "sd", resp = "Alt"),
            prior(normal(0, 1), class = "sd", resp = "Ho2"),
            prior(normal(0, 1), class = "sd", resp = "In"),
            prior(normal(0, 1), class = "sd", resp = "Sc"),
```

```
            prior(normal(0, 1), class = "sd", resp = "Y"),
            prior(normal(0, 1), class = "sd", resp = "Be"),
            prior(normal(0, 1), class = "sd", resp = "Co"),
            prior(normal(0, 1), class = "sd", resp = "Th"),
            prior(normal(0, 1), class = "sd", resp = "Ho1"),

            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Std"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Alt"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Ho2"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "In"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Sc"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Y"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Be"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Co"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Th"),
            prior(normal(-1, 2), class = "Intercept", dpar = "sigma", resp = "Ho1"),

            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Std"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Alt"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Ho2"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "In"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Sc"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Y"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Be"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Co"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Th"),
            prior(normal(0, 1), class = "b", dpar = "sigma", resp = "Ho1"),

            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Std"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Alt"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Ho2"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "In"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Sc"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Y"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Be"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Co"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Th"),
            prior(normal(0, 1), class = "sd", dpar = "sigma", resp = "Ho1"),

            prior(lkj(1), class = "rescor")
            ),
  control = list(adapt_delta = 0.95, max_treedepth = 14),
  init_r = 0.05,
  save_pars = save_pars(all = TRUE),
  seed = 5214,
  chains=4,
  iter=3000,
  cores=4 )

save(mod3, file = "full-analysis-files/mod3_As_mv.rda")
```

Tabular parameter estimates

Again, a summary of the posterior estimates.

```
##  Family: MV(gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian
##   Links: mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
## Formula: Std ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Alt ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho2 ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          In ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Sc ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Y ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Be ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Co ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Th ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho1 ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##    Data: df_mv_as (Number of observations: 352)
##   Draws: 4 chains, each with iter = 3000; warmup = 1500; thin = 1;
##          total post-warmup draws = 6000
##
## Priors:
## b_Alt ~ normal(0, 1)
## b_Alt_sigma ~ normal(0, 1)
## b_Be ~ normal(0, 1)
## b_Be_sigma ~ normal(0, 1)
## b_Co ~ normal(0, 1)
## b_Co_sigma ~ normal(0, 1)
## b_Ho1 ~ normal(0, 1)
## b_Ho1_sigma ~ normal(0, 1)
## b_Ho2 ~ normal(0, 1)
## b_Ho2_sigma ~ normal(0, 1)
## b_In ~ normal(0, 1)
## b_In_sigma ~ normal(0, 1)
## b_Sc ~ normal(0, 1)
## b_Sc_sigma ~ normal(0, 1)
```

```
## b_Std ~ normal(0, 1)
## b_Std_sigma ~ normal(0, 1)
## b_Th ~ normal(0, 1)
## b_Th_sigma ~ normal(0, 1)
## b_Y ~ normal(0, 1)
## b_Y_sigma ~ normal(0, 1)
## Intercept_Alt ~ normal(0, 1)
## Intercept_Alt_sigma ~ normal(-1, 2)
## Intercept_Be ~ normal(0, 1)
## Intercept_Be_sigma ~ normal(-1, 2)
## Intercept_Co ~ normal(0, 1)
## Intercept_Co_sigma ~ normal(-1, 2)
## Intercept_Ho1 ~ normal(0, 1)
## Intercept_Ho1_sigma ~ normal(-1, 2)
## Intercept_Ho2 ~ normal(0, 1)
## Intercept_Ho2_sigma ~ normal(-1, 2)
## Intercept_In ~ normal(0, 1)
## Intercept_In_sigma ~ normal(-1, 2)
## Intercept_Sc ~ normal(0, 1)
## Intercept_Sc_sigma ~ normal(-1, 2)
## Intercept_Std ~ normal(0, 1)
## Intercept_Std_sigma ~ normal(-1, 2)
## Intercept_Th ~ normal(0, 1)
## Intercept_Th_sigma ~ normal(-1, 2)
## Intercept_Y ~ normal(0, 1)
## Intercept_Y_sigma ~ normal(-1, 2)
## L ~ lkj_corr_cholesky(1)
## Lrescor ~ lkj_corr_cholesky(1)
## sd_Alt ~ normal(0, 1)
## sd_Alt_sigma ~ normal(0, 1)
## sd_Be ~ normal(0, 1)
## sd_Be_sigma ~ normal(0, 1)
## sd_Co ~ normal(0, 1)
## sd_Co_sigma ~ normal(0, 1)
## sd_Ho1 ~ normal(0, 1)
## sd_Ho1_sigma ~ normal(0, 1)
## sd_Ho2 ~ normal(0, 1)
## sd_Ho2_sigma ~ normal(0, 1)
## sd_In ~ normal(0, 1)
## sd_In_sigma ~ normal(0, 1)
## sd_Sc ~ normal(0, 1)
## sd_Sc_sigma ~ normal(0, 1)
## sd_Std ~ normal(0, 1)
## sd_Std_sigma ~ normal(0, 1)
## sd_Th ~ normal(0, 1)
## sd_Th_sigma ~ normal(0, 1)
## sd_Y ~ normal(0, 1)
## sd_Y_sigma ~ normal(0, 1)
##
## Group-Level Effects:
## ~day_expt (Number of levels: 8)
##                               Estimate Est.Error l-95% CI u-95% CI Rhat
## sd(Std_Intercept)                 0.05      0.02     0.03     0.10 1.00
## sd(Std_tuneHHe)                   0.01      0.01     0.00     0.04 1.00
```

```
## sd(sigma_Std_Intercept)          0.25      0.10     0.12     0.51 1.00
## sd(Alt_Intercept)                0.13      0.05     0.07     0.26 1.00
## sd(Alt_tuneHHe)                  0.12      0.05     0.07     0.24 1.00
## sd(sigma_Alt_Intercept)          0.22      0.09     0.10     0.45 1.00
## sd(Ho2_Intercept)                0.05      0.02     0.03     0.10 1.00
## sd(Ho2_tuneHHe)                  0.09      0.03     0.05     0.18 1.00
## sd(sigma_Ho2_Intercept)          0.35      0.13     0.18     0.66 1.00
## sd(In_Intercept)                 0.26      0.09     0.15     0.48 1.00
## sd(In_tuneHHe)                   0.05      0.03     0.01     0.11 1.00
## sd(sigma_In_Intercept)           0.35      0.12     0.19     0.65 1.00
## sd(Sc_Intercept)                 0.17      0.06     0.09     0.33 1.00
## sd(Sc_tuneHHe)                   0.08      0.04     0.03     0.17 1.00
## sd(sigma_Sc_Intercept)           0.34      0.13     0.18     0.66 1.00
## sd(Y_Intercept)                  0.26      0.10     0.14     0.51 1.00
## sd(Y_tuneHHe)                    0.05      0.02     0.02     0.11 1.00
## sd(sigma_Y_Intercept)            0.37      0.13     0.21     0.71 1.00
## sd(Be_Intercept)                 0.26      0.10     0.14     0.53 1.00
## sd(Be_tuneHHe)                   0.22      0.09     0.11     0.45 1.00
## sd(sigma_Be_Intercept)           0.38      0.14     0.20     0.72 1.00
## sd(Co_Intercept)                 0.23      0.08     0.13     0.43 1.00
## sd(Co_tuneHHe)                   0.11      0.04     0.05     0.22 1.00
## sd(sigma_Co_Intercept)           0.41      0.14     0.23     0.78 1.00
## sd(Th_Intercept)                 0.55      0.18     0.31     1.00 1.00
## sd(Th_tuneHHe)                   0.12      0.05     0.05     0.24 1.00
## sd(sigma_Th_Intercept)           0.52      0.18     0.30     0.97 1.00
## sd(Ho1_Intercept)                0.53      0.18     0.30     0.97 1.00
## sd(Ho1_tuneHHe)                  0.12      0.05     0.06     0.24 1.00
## sd(sigma_Ho1_Intercept)          0.47      0.16     0.27     0.86 1.00
## cor(Std_Intercept,Std_tuneHHe)   0.12      0.56    -0.91     0.96 1.00
## cor(Alt_Intercept,Alt_tuneHHe)  -0.51      0.29    -0.90     0.22 1.00
## cor(Ho2_Intercept,Ho2_tuneHHe)  -0.65      0.26    -0.95     0.04 1.00
## cor(In_Intercept,In_tuneHHe)    -0.78      0.26    -1.00    -0.03 1.00
## cor(Sc_Intercept,Sc_tuneHHe)    -0.31      0.40    -0.91     0.54 1.00
## cor(Y_Intercept,Y_tuneHHe)      -0.31      0.43    -0.93     0.62 1.00
## cor(Be_Intercept,Be_tuneHHe)     0.04      0.37    -0.67     0.73 1.00
## cor(Co_Intercept,Co_tuneHHe)     0.49      0.33    -0.32     0.92 1.00
## cor(Th_Intercept,Th_tuneHHe)    -0.09      0.43    -0.80     0.73 1.00
## cor(Ho1_Intercept,Ho1_tuneHHe)   0.07      0.40    -0.70     0.77 1.00
##                                Bulk_ESS Tail_ESS
## sd(Std_Intercept)                  3817     3909
## sd(Std_tuneHHe)                    4030     3866
## sd(sigma_Std_Intercept)            3390     4321
## sd(Alt_Intercept)                  4062     4051
## sd(Alt_tuneHHe)                    4136     3984
## sd(sigma_Alt_Intercept)            3274     4162
## sd(Ho2_Intercept)                  3943     3977
## sd(Ho2_tuneHHe)                    4616     4436
## sd(sigma_Ho2_Intercept)            4093     4383
## sd(In_Intercept)                   5454     4250
## sd(In_tuneHHe)                     3181     3483
## sd(sigma_In_Intercept)             5855     4609
## sd(Sc_Intercept)                   5864     4477
## sd(Sc_tuneHHe)                     3411     3764
## sd(sigma_Sc_Intercept)             4974     4432
```

```
## sd(Y_Intercept)                 5051     3333
## sd(Y_tuneHHe)                   3772     2611
## sd(sigma_Y_Intercept)           6371     4577
## sd(Be_Intercept)                5129     4426
## sd(Be_tuneHHe)                  4361     4423
## sd(sigma_Be_Intercept)          4589     4465
## sd(Co_Intercept)                4966     4367
## sd(Co_tuneHHe)                  3936     3936
## sd(sigma_Co_Intercept)          5707     4972
## sd(Th_Intercept)                6235     4958
## sd(Th_tuneHHe)                  6281     4695
## sd(sigma_Th_Intercept)          6429     4403
## sd(Ho1_Intercept)               6134     4400
## sd(Ho1_tuneHHe)                 5060     3954
## sd(sigma_Ho1_Intercept)         6571     4685
## cor(Std_Intercept,Std_tuneHHe)  8554     4471
## cor(Alt_Intercept,Alt_tuneHHe)  4996     4178
## cor(Ho2_Intercept,Ho2_tuneHHe)  5014     4847
## cor(In_Intercept,In_tuneHHe)    4760     4405
## cor(Sc_Intercept,Sc_tuneHHe)    4639     4175
## cor(Y_Intercept,Y_tuneHHe)      4027     3430
## cor(Be_Intercept,Be_tuneHHe)    6245     4211
## cor(Co_Intercept,Co_tuneHHe)    4675     4137
## cor(Th_Intercept,Th_tuneHHe)    6662     4624
## cor(Ho1_Intercept,Ho1_tuneHHe)  6495     4306
##
## ~matrix (Number of levels: 22)
##                               Estimate Est.Error l-95% CI u-95% CI Rhat
## sd(Std_Intercept)                 0.06      0.01     0.04     0.09 1.00
## sd(Std_tuneHHe)                   0.01      0.01     0.00     0.03 1.00
## sd(sigma_Std_Intercept)           0.09      0.05     0.01     0.19 1.00
## sd(Alt_Intercept)                 0.06      0.01     0.04     0.09 1.00
## sd(Alt_tuneHHe)                   0.02      0.01     0.00     0.04 1.00
## sd(sigma_Alt_Intercept)           0.11      0.05     0.02     0.21 1.00
## sd(Ho2_Intercept)                 0.07      0.01     0.05     0.11 1.00
## sd(Ho2_tuneHHe)                   0.01      0.01     0.00     0.03 1.00
## sd(sigma_Ho2_Intercept)           0.06      0.04     0.00     0.15 1.00
## sd(In_Intercept)                  0.07      0.01     0.05     0.11 1.00
## sd(In_tuneHHe)                    0.05      0.01     0.03     0.08 1.00
## sd(sigma_In_Intercept)            0.14      0.03     0.08     0.21 1.00
## sd(Sc_Intercept)                  0.09      0.02     0.07     0.13 1.00
## sd(Sc_tuneHHe)                    0.10      0.02     0.07     0.15 1.00
## sd(sigma_Sc_Intercept)            0.28      0.06     0.19     0.41 1.00
## sd(Y_Intercept)                   0.06      0.01     0.04     0.08 1.00
## sd(Y_tuneHHe)                     0.05      0.01     0.03     0.07 1.00
## sd(sigma_Y_Intercept)             0.16      0.04     0.09     0.24 1.00
## sd(Be_Intercept)                  0.08      0.01     0.05     0.11 1.00
## sd(Be_tuneHHe)                    0.07      0.02     0.04     0.12 1.00
## sd(sigma_Be_Intercept)            0.16      0.05     0.07     0.27 1.00
## sd(Co_Intercept)                  0.04      0.01     0.02     0.06 1.00
## sd(Co_tuneHHe)                    0.03      0.01     0.02     0.05 1.00
## sd(sigma_Co_Intercept)            0.09      0.04     0.02     0.17 1.00
## sd(Th_Intercept)                  0.25      0.04     0.18     0.35 1.01
## sd(Th_tuneHHe)                    0.20      0.04     0.14     0.28 1.00
```

```
## sd(sigma_Th_Intercept)            0.05      0.03     0.01     0.11 1.00
## sd(Ho1_Intercept)                 0.18      0.03     0.13     0.24 1.00
## sd(Ho1_tuneHHe)                   0.13      0.02     0.09     0.18 1.00
## sd(sigma_Ho1_Intercept)           0.04      0.02     0.00     0.10 1.00
## cor(Std_Intercept,Std_tuneHHe)   -0.20      0.49    -0.94     0.85 1.00
## cor(Alt_Intercept,Alt_tuneHHe)   -0.13      0.44    -0.88     0.81 1.00
## cor(Ho2_Intercept,Ho2_tuneHHe)   -0.17      0.50    -0.95     0.86 1.00
## cor(In_Intercept,In_tuneHHe)      0.97      0.04     0.86     1.00 1.00
## cor(Sc_Intercept,Sc_tuneHHe)      0.97      0.03     0.88     1.00 1.00
## cor(Y_Intercept,Y_tuneHHe)        0.96      0.06     0.79     1.00 1.00
## cor(Be_Intercept,Be_tuneHHe)      0.87      0.13     0.53     1.00 1.00
## cor(Co_Intercept,Co_tuneHHe)      0.81      0.15     0.44     0.99 1.00
## cor(Th_Intercept,Th_tuneHHe)      0.97      0.02     0.92     0.99 1.00
## cor(Ho1_Intercept,Ho1_tuneHHe)    0.99      0.01     0.96     1.00 1.00
##                                Bulk_ESS Tail_ESS
## sd(Std_Intercept)                 2525     3411
## sd(Std_tuneHHe)                   2622     4229
## sd(sigma_Std_Intercept)           2024     3018
## sd(Alt_Intercept)                 2942     3942
## sd(Alt_tuneHHe)                   1655     3050
## sd(sigma_Alt_Intercept)           1992     3199
## sd(Ho2_Intercept)                 3103     4201
## sd(Ho2_tuneHHe)                   2551     3633
## sd(sigma_Ho2_Intercept)           2630     4061
## sd(In_Intercept)                  2829     4125
## sd(In_tuneHHe)                    3091     4009
## sd(sigma_In_Intercept)            2221     3939
## sd(Sc_Intercept)                  2364     4159
## sd(Sc_tuneHHe)                    2713     4193
## sd(sigma_Sc_Intercept)            2557     3449
## sd(Y_Intercept)                   3564     4392
## sd(Y_tuneHHe)                     3874     4651
## sd(sigma_Y_Intercept)             1892     3595
## sd(Be_Intercept)                  4304     4698
## sd(Be_tuneHHe)                    5212     4921
## sd(sigma_Be_Intercept)            1811     2540
## sd(Co_Intercept)                  2990     4109
## sd(Co_tuneHHe)                    4302     4755
## sd(sigma_Co_Intercept)            1172     1263
## sd(Th_Intercept)                  2101     3407
## sd(Th_tuneHHe)                    2257     3325
## sd(sigma_Th_Intercept)            1448     1746
## sd(Ho1_Intercept)                 2512     3304
## sd(Ho1_tuneHHe)                   3007     4360
## sd(sigma_Ho1_Intercept)           1779     3218
## cor(Std_Intercept,Std_tuneHHe)    8799     4347
## cor(Alt_Intercept,Alt_tuneHHe)    7412     3495
## cor(Ho2_Intercept,Ho2_tuneHHe)    8524     4384
## cor(In_Intercept,In_tuneHHe)      2374     2826
## cor(Sc_Intercept,Sc_tuneHHe)      3099     4159
## cor(Y_Intercept,Y_tuneHHe)        1536     2037
## cor(Be_Intercept,Be_tuneHHe)      3663     3619
## cor(Co_Intercept,Co_tuneHHe)      2921     3824
## cor(Th_Intercept,Th_tuneHHe)      2990     4270
```

```
## cor(Ho1_Intercept,Ho1_tuneHHe)     2732     4045
##
## Population-Level Effects:
##                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Std_Intercept           0.12      0.03     0.07     0.17 1.00     3286     3706
## sigma_Std_Intercept    -2.57      0.11    -2.79    -2.35 1.00     4873     4364
## Alt_Intercept           0.15      0.05     0.04     0.25 1.00     3220     3919
## sigma_Alt_Intercept    -2.44      0.10    -2.64    -2.25 1.00     5595     4840
## Ho2_Intercept           0.12      0.03     0.07     0.17 1.00     2832     3621
## sigma_Ho2_Intercept    -2.54      0.14    -2.82    -2.26 1.00     4250     3717
## In_Intercept           -0.04      0.10    -0.23     0.16 1.00     2583     2795
## sigma_In_Intercept     -2.31      0.14    -2.59    -2.03 1.00     2841     3490
## Sc_Intercept           -0.01      0.07    -0.14     0.13 1.00     2668     3506
## sigma_Sc_Intercept     -2.26      0.15    -2.56    -1.96 1.00     3696     3941
## Y_Intercept            -0.06      0.10    -0.26     0.14 1.00     2305     2154
## sigma_Y_Intercept      -2.36      0.15    -2.64    -2.06 1.00     3126     3638
## Be_Intercept            0.02      0.10    -0.17     0.22 1.00     2722     3328
## sigma_Be_Intercept     -2.01      0.16    -2.32    -1.69 1.00     3918     4227
## Co_Intercept            0.02      0.09    -0.16     0.20 1.00     2173     2584
## sigma_Co_Intercept     -2.32      0.16    -2.65    -2.00 1.00     2989     3515
## Th_Intercept           -0.20      0.21    -0.60     0.22 1.00     2353     3034
## sigma_Th_Intercept     -1.82      0.20    -2.23    -1.42 1.00     3038     3534
## Ho1_Intercept          -0.18      0.20    -0.58     0.22 1.00     2620     3357
## sigma_Ho1_Intercept    -2.02      0.18    -2.38    -1.66 1.00     2908     3574
## Std_tuneHHe             0.03      0.01     0.01     0.05 1.00     6905     4670
## sigma_Std_tuneHHe       0.33      0.07     0.20     0.46 1.00    10187     4753
## Alt_tuneHHe             0.01      0.05    -0.08     0.11 1.00     3988     4333
## sigma_Alt_tuneHHe       0.27      0.06     0.15     0.40 1.00     7973     4962
## Ho2_tuneHHe             0.04      0.04    -0.03     0.12 1.00     4772     4382
## sigma_Ho2_tuneHHe       0.55      0.07     0.42     0.68 1.00     9311     4910
## In_tuneHHe              0.01      0.03    -0.04     0.07 1.00     3152     3648
## sigma_In_tuneHHe        0.61      0.05     0.51     0.71 1.00     4638     5286
## Sc_tuneHHe             -0.11      0.04    -0.19    -0.03 1.00     2868     3970
## sigma_Sc_tuneHHe        0.67      0.06     0.56     0.78 1.00     7130     5445
## Y_tuneHHe              -0.06      0.03    -0.11    -0.00 1.00     2983     3899
## sigma_Y_tuneHHe         0.62      0.05     0.53     0.71 1.00     5023     4959
## Be_tuneHHe             -0.09      0.09    -0.27     0.08 1.00     3900     3722
## sigma_Be_tuneHHe        0.78      0.06     0.66     0.90 1.00     8885     5410
## Co_tuneHHe             -0.05      0.05    -0.14     0.04 1.00     2804     3595
## sigma_Co_tuneHHe        0.63      0.04     0.55     0.72 1.00     6926     5328
## Th_tuneHHe              0.02      0.07    -0.11     0.16 1.00     2117     3480
## sigma_Th_tuneHHe        0.57      0.05     0.46     0.67 1.00     6026     5020
## Ho1_tuneHHe            -0.03      0.06    -0.14     0.08 1.00     2913     3660
## sigma_Ho1_tuneHHe       0.53      0.05     0.42     0.63 1.00     5263     4724
##
## Residual Correlations:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## rescor(Std,Alt)     0.73      0.03     0.67     0.78 1.00     5851     5287
## rescor(Std,Ho2)     0.67      0.03     0.60     0.73 1.00     5052     5524
## rescor(Alt,Ho2)     0.69      0.03     0.63     0.75 1.00     5396     5187
## rescor(Std,In)      0.62      0.04     0.55     0.69 1.00     3679     4260
## rescor(Alt,In)      0.63      0.04     0.55     0.69 1.00     3682     4338
## rescor(Ho2,In)      0.70      0.03     0.63     0.76 1.00     4367     4654
## rescor(Std,Sc)      0.59      0.04     0.51     0.66 1.00     4492     4685
```

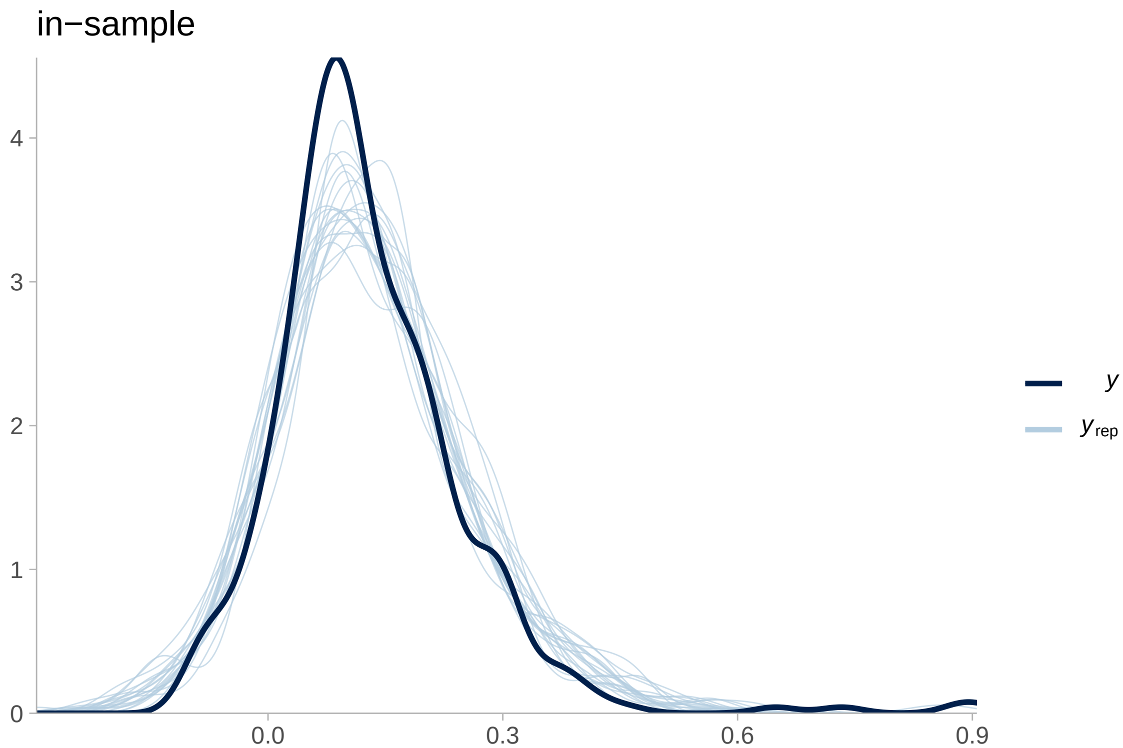
```
## rescor(Alt,Sc)       0.60      0.04      0.51      0.67 1.00      4470      4696
## rescor(Ho2,Sc)       0.73      0.03      0.67      0.79 1.00      5067      4779
## rescor(In,Sc)        0.75      0.03      0.69      0.80 1.00      6573      5213
## rescor(Std,Y)        0.65      0.03      0.58      0.71 1.00      3723      4511
## rescor(Alt,Y)        0.66      0.04      0.59      0.72 1.00      3506      4384
## rescor(Ho2,Y)        0.76      0.03      0.70      0.81 1.00      4398      4777
## rescor(In,Y)         0.96      0.00      0.95      0.97 1.00      6389      5136
## rescor(Sc,Y)         0.86      0.02      0.82      0.89 1.00      6516      5237
## rescor(Std,Be)       0.47      0.05      0.38      0.56 1.00      4344      4661
## rescor(Alt,Be)       0.47      0.05      0.37      0.56 1.00      4765      5164
## rescor(Ho2,Be)       0.56      0.04      0.47      0.63 1.00      4735      4790
## rescor(In,Be)        0.64      0.04      0.56      0.71 1.00      5659      5329
## rescor(Sc,Be)        0.62      0.04      0.53      0.70 1.00      4082      4769
## rescor(Y,Be)         0.65      0.04      0.58      0.72 1.00      4899      4905
## rescor(Std,Co)       0.62      0.04      0.55      0.69 1.00      3772      4196
## rescor(Alt,Co)       0.64      0.04      0.56      0.70 1.00      3720      4298
## rescor(Ho2,Co)       0.75      0.03      0.70      0.80 1.00      4098      4419
## rescor(In,Co)        0.87      0.02      0.84      0.90 1.00      5771      4653
## rescor(Sc,Co)        0.82      0.02      0.78      0.87 1.00      3817      4688
## rescor(Y,Co)         0.90      0.01      0.88      0.92 1.00      4565      4533
## rescor(Be,Co)        0.86      0.02      0.83      0.89 1.00      6113      5223
## rescor(Std,Th)       0.39      0.05      0.29      0.48 1.00      4019      5015
## rescor(Alt,Th)       0.40      0.05      0.29      0.49 1.00      4455      4342
## rescor(Ho2,Th)       0.39      0.05      0.29      0.49 1.00      4591      4468
## rescor(In,Th)        0.81      0.02      0.77      0.85 1.00      5820      4889
## rescor(Sc,Th)        0.36      0.05      0.25      0.46 1.00      4966      4828
## rescor(Y,Th)         0.69      0.03      0.63      0.75 1.00      5480      4857
## rescor(Be,Th)        0.64      0.03      0.57      0.71 1.00      8038      5227
## rescor(Co,Th)        0.71      0.03      0.65      0.77 1.00      7804      5329
## rescor(Std,Ho1)      0.47      0.05      0.37      0.56 1.00      3925      5261
## rescor(Alt,Ho1)      0.48      0.05      0.38      0.56 1.00      4600      5181
## rescor(Ho2,Ho1)      0.41      0.05      0.31      0.51 1.00      4881      4391
## rescor(In,Ho1)       0.82      0.02      0.77      0.86 1.00      5553      4986
## rescor(Sc,Ho1)       0.35      0.05      0.24      0.45 1.00      4986      5042
## rescor(Y,Ho1)        0.70      0.03      0.63      0.76 1.00      5457      5043
## rescor(Be,Ho1)       0.60      0.04      0.52      0.67 1.00      8190      5253
## rescor(Co,Ho1)       0.69      0.03      0.63      0.75 1.00      7823      5297
## rescor(Th,Ho1)       0.97      0.00      0.97      0.98 1.00      6505      5365
##
## 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).
```

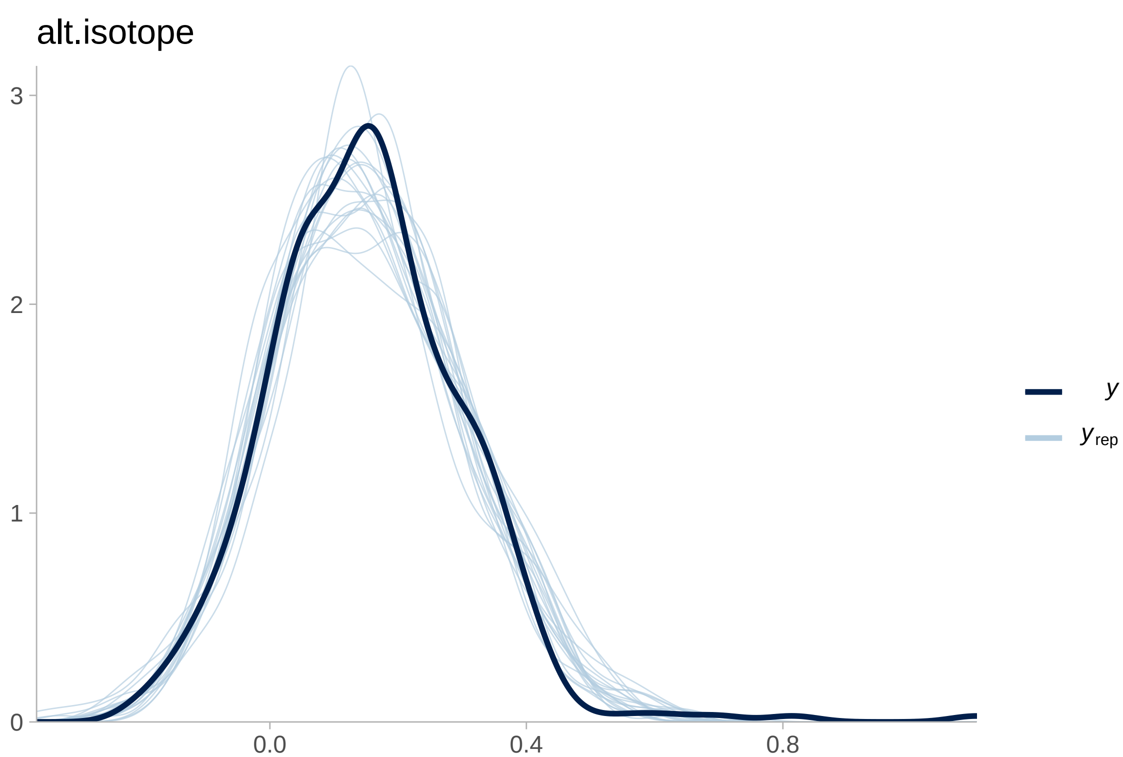
Again, the HMC sampling looks to have gone well.

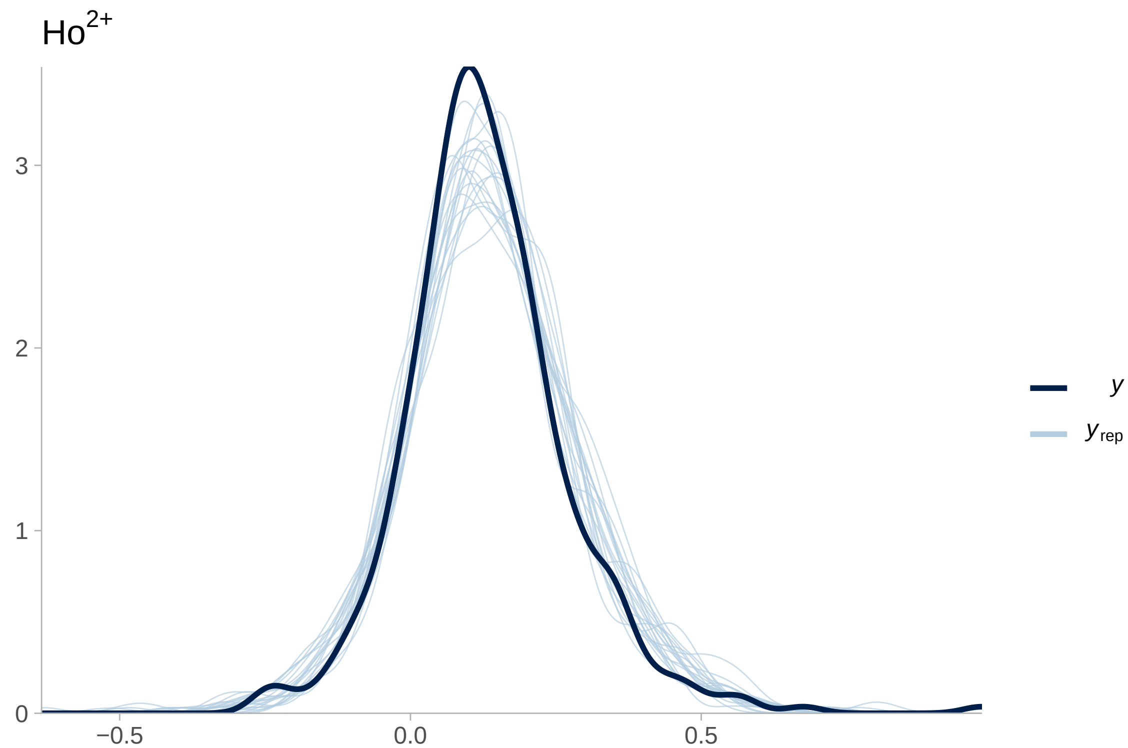
Model checks

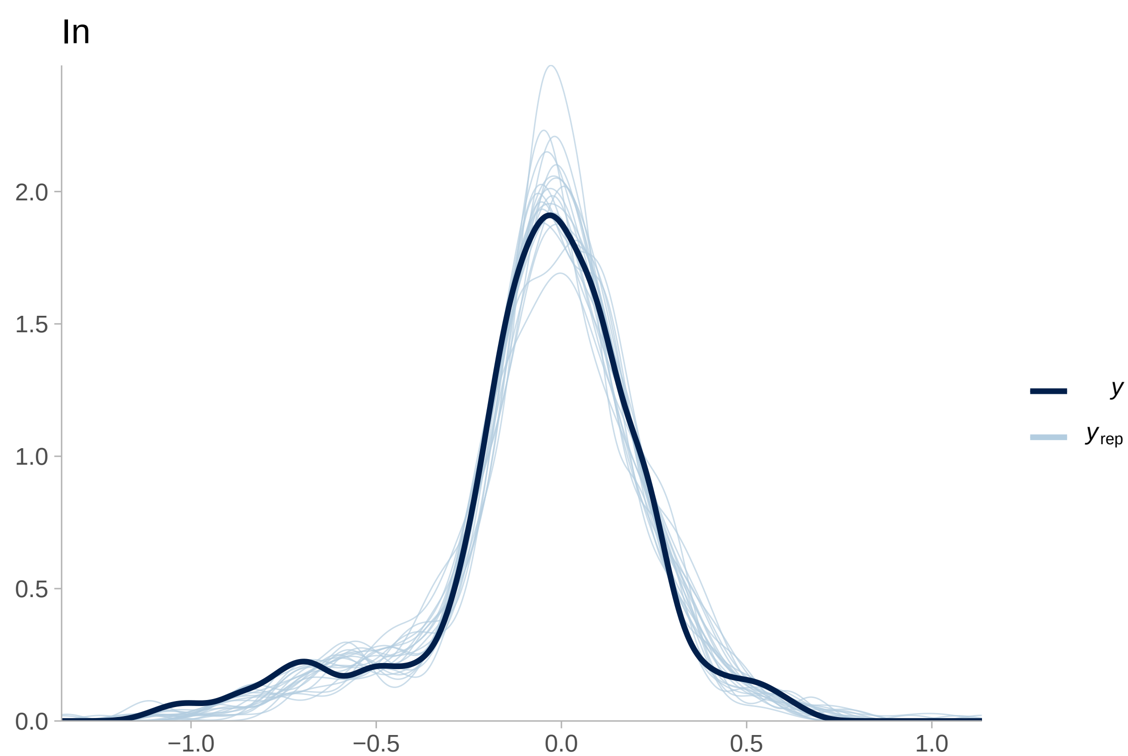
Next, the density checks.

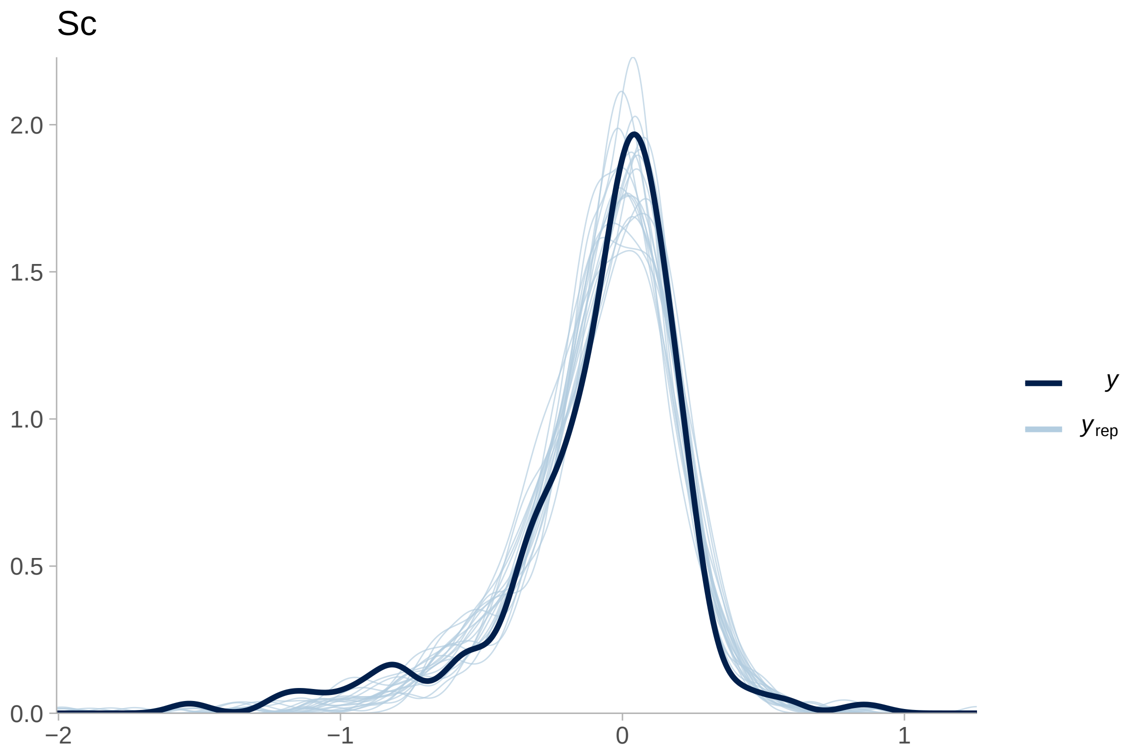
Density overlay

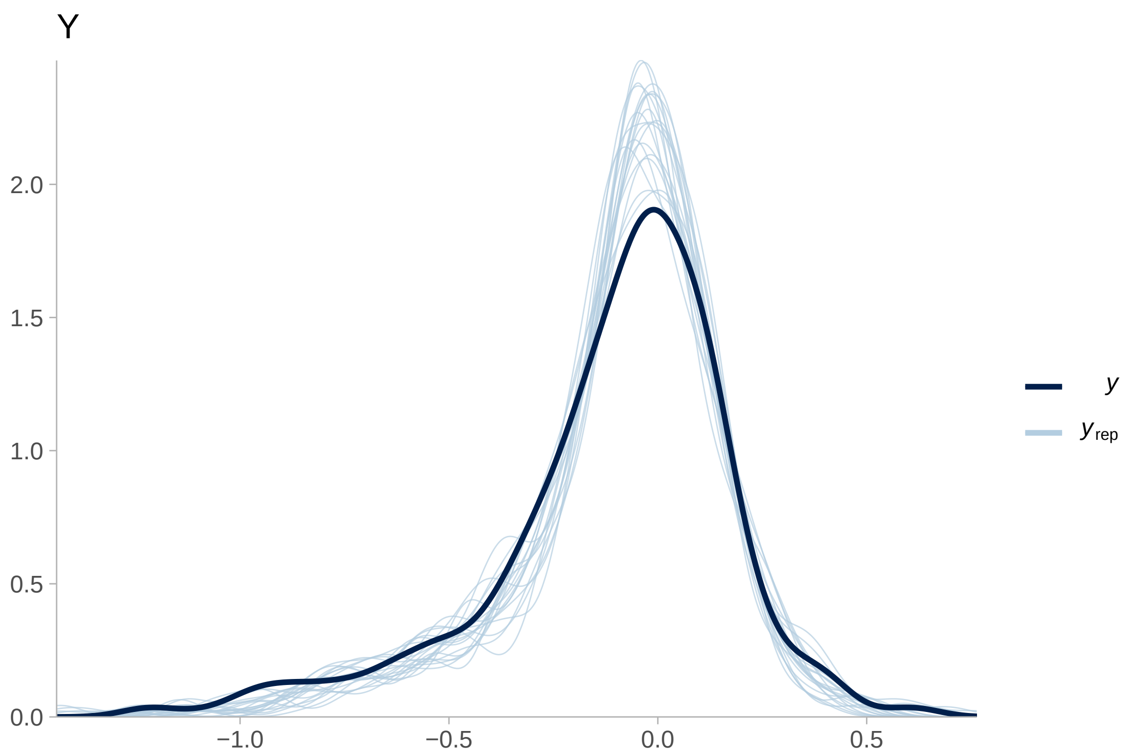


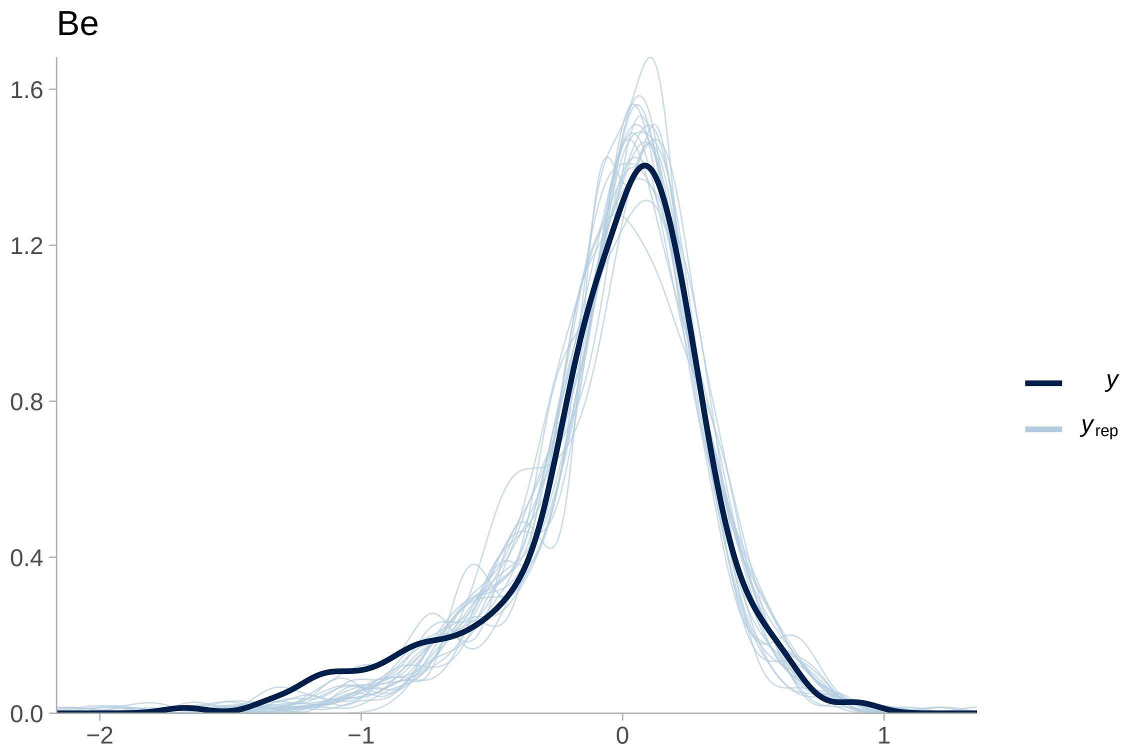


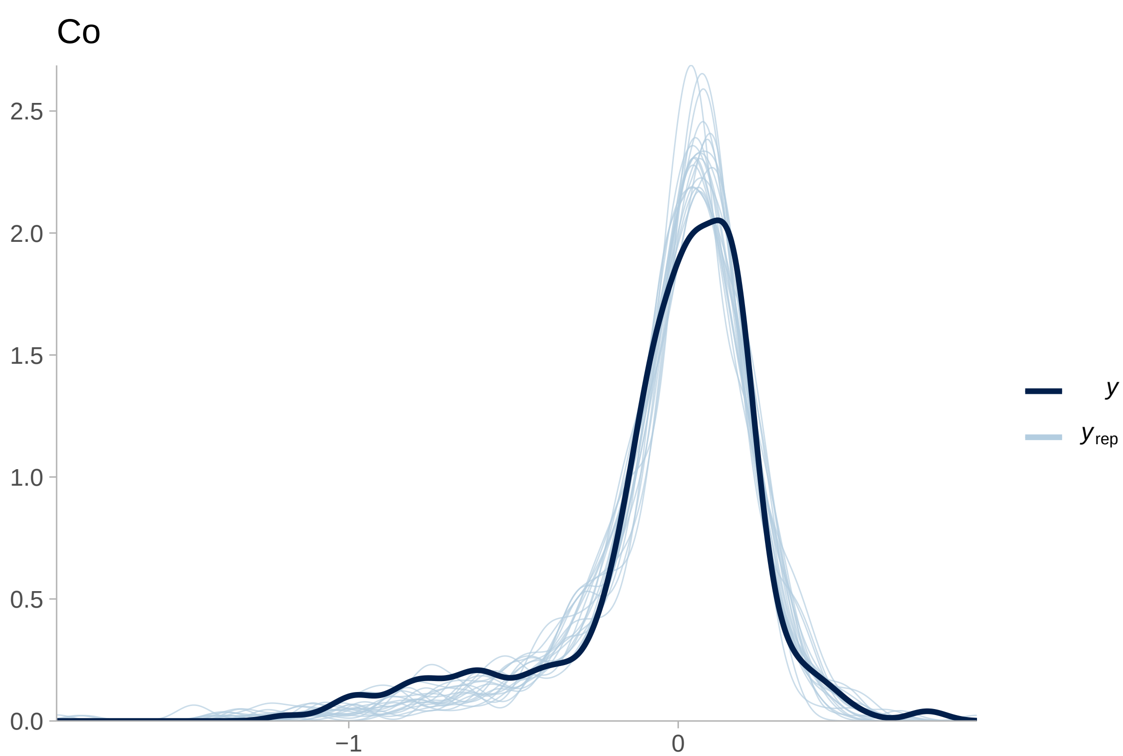

Ho2+
0
1
2
3
−0.5
0.0
0.5
y
yrep

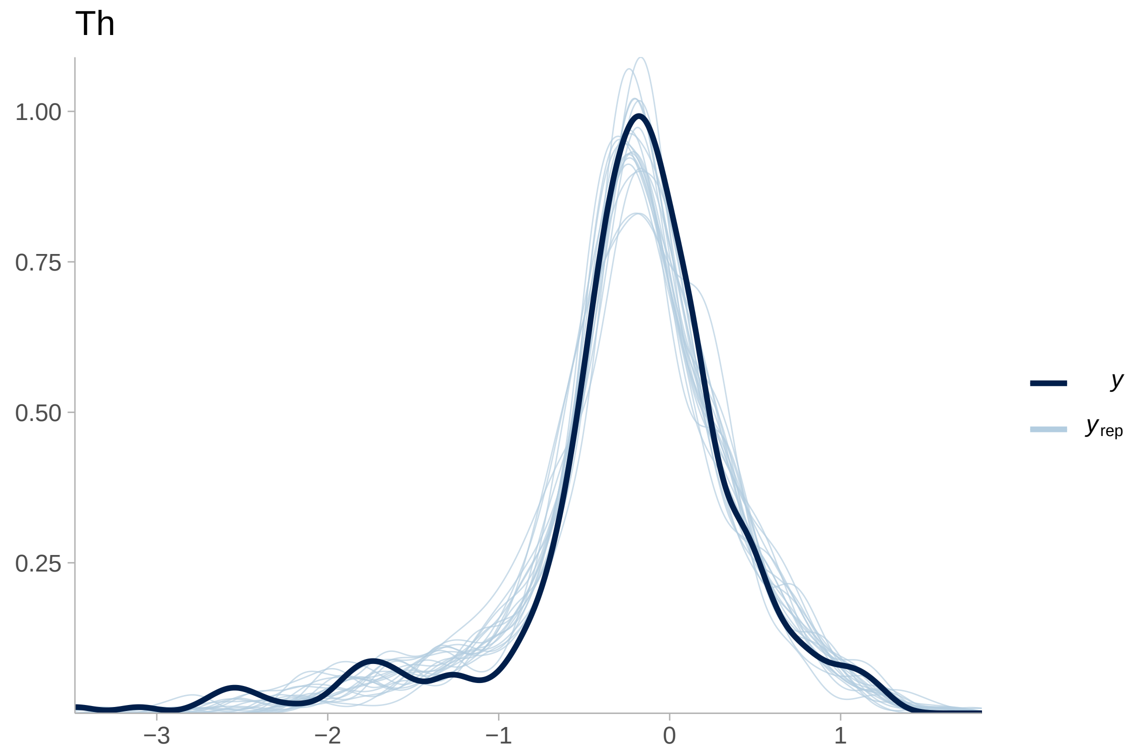

In
0.0
0.5
1.0
1.5
2.0
−1.0
−0.5
0.0
0.5
1.0
y
yrep

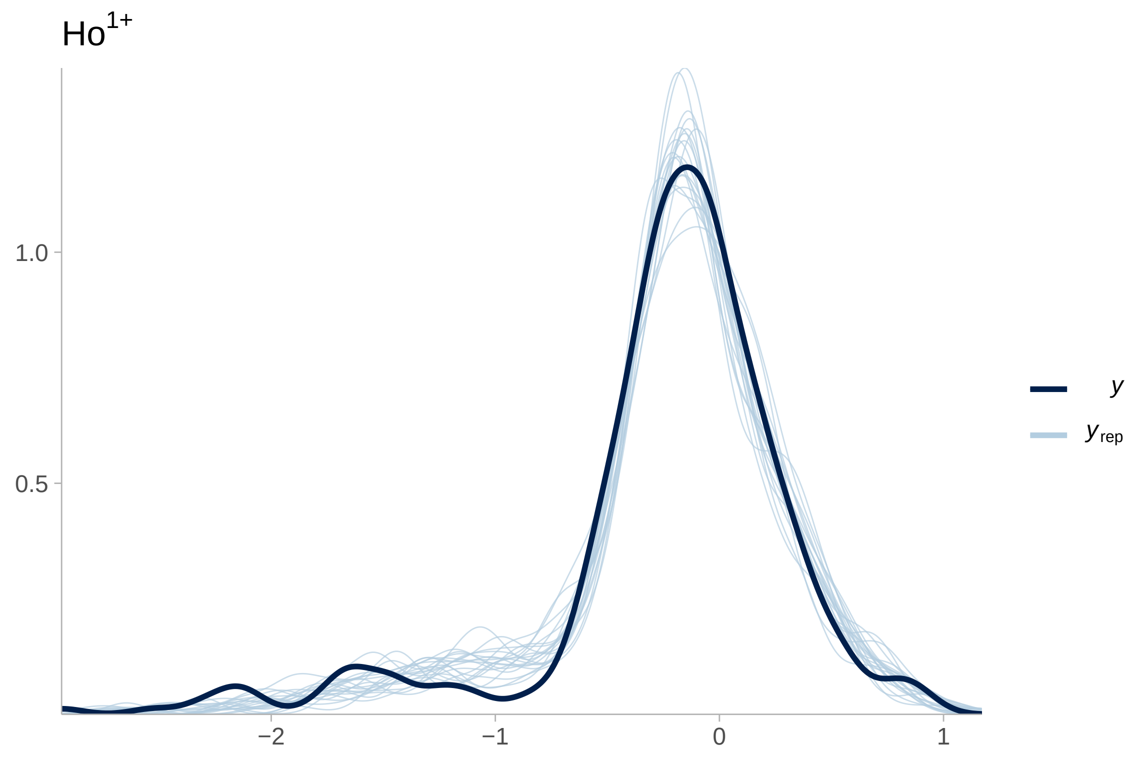

Sc
2.0
1.5
1.0
0.5
0.0
−2
−1
0
1
y
yrep


Y
2.0
1.5
1.0
0.5
0.0
−1.0
−0.5
0.0
0.5
y
yrep


Be
1.6
1.2
0.8
0.4
0.0
−2
−1
0
1
y
yrep


Co
2.5
2.0
1.5
1.0
0.5
0.0
−1
0
y
yrep

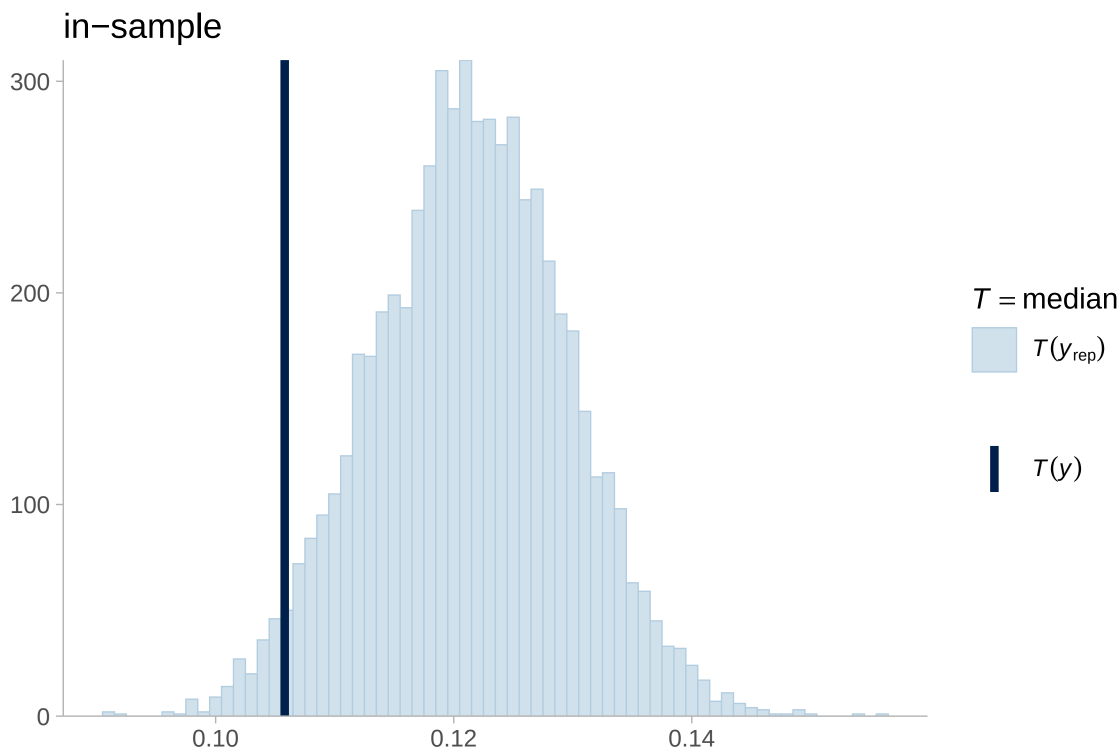


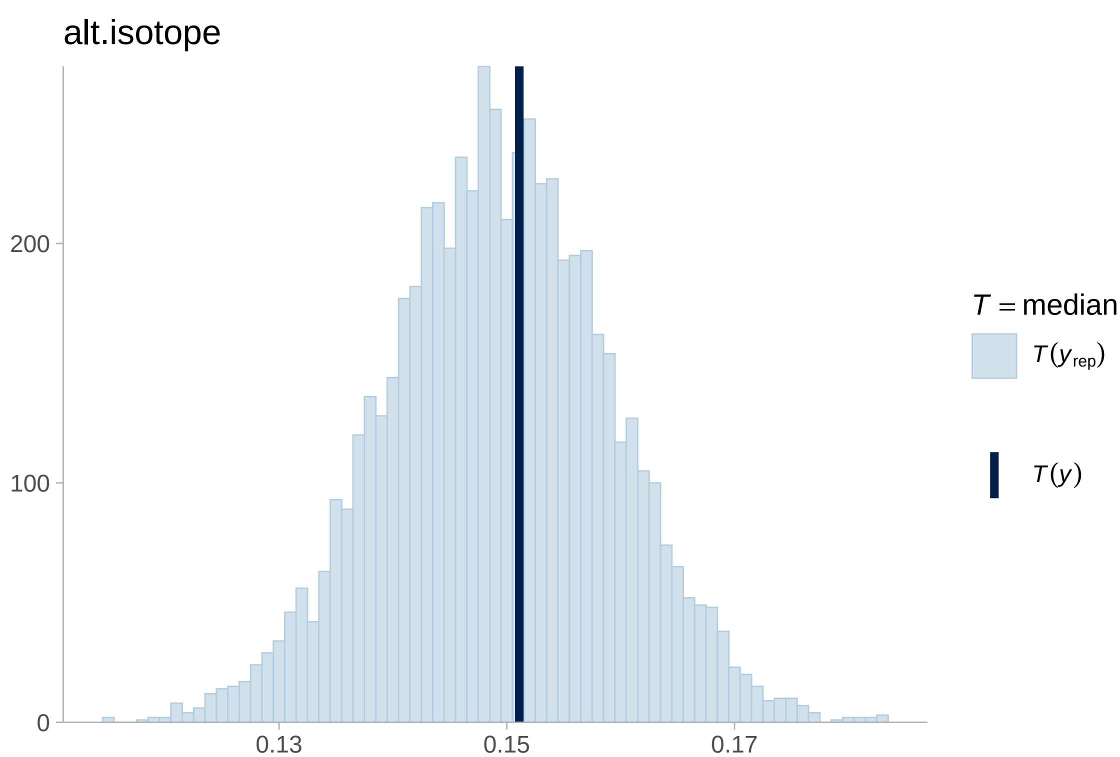


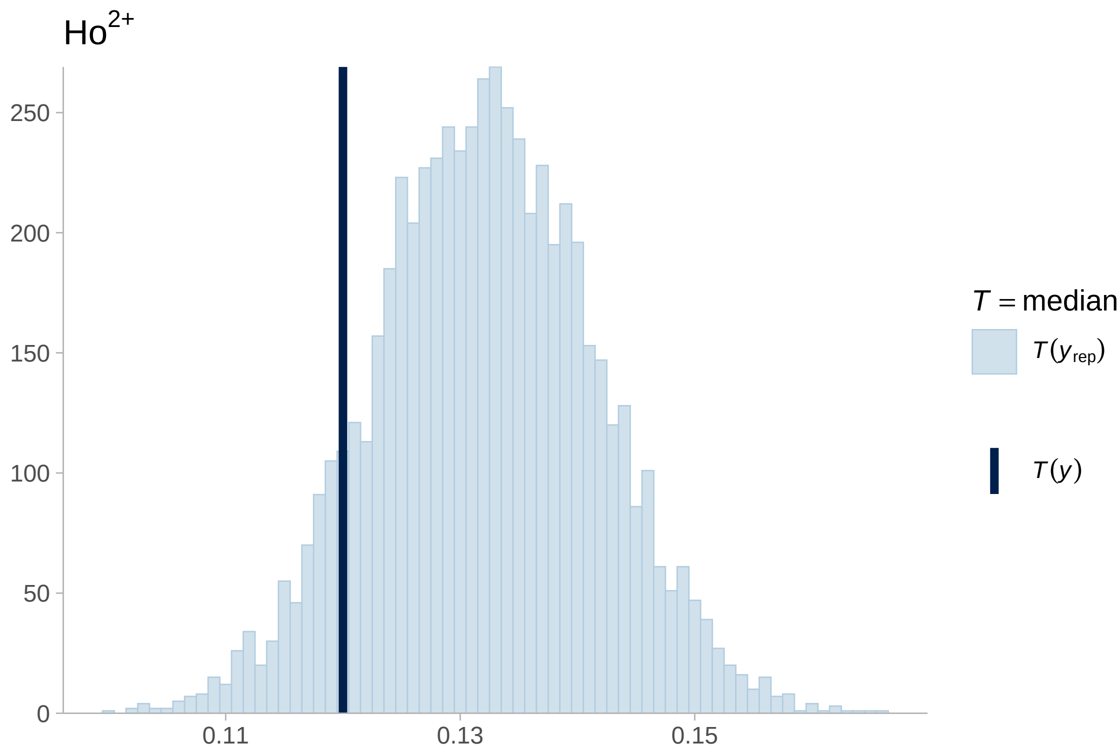
These checks suggest an improvement on the previous models. The check looks reasonable across most of the methods, though clearly the in-sample observations again proved more difficult to replicate. It can be helpful to look at some additional checks (below) to get a better handle on which aspects of the observed data are reasonably approximated by the model and which may not be.

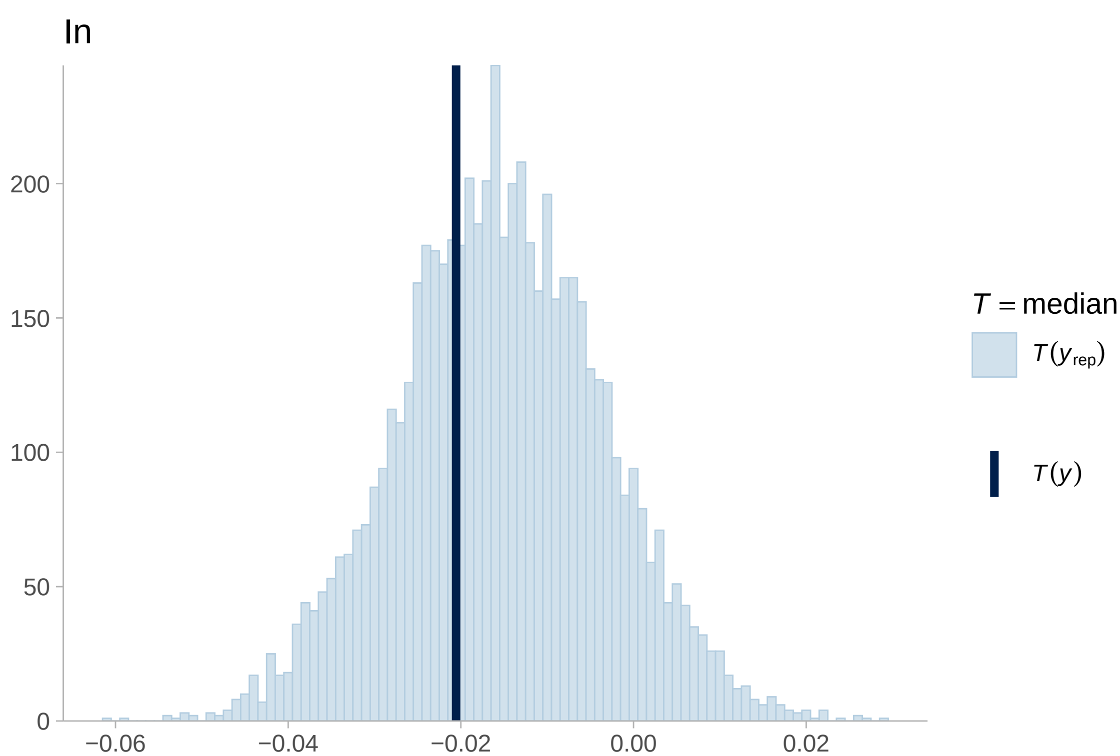
Median

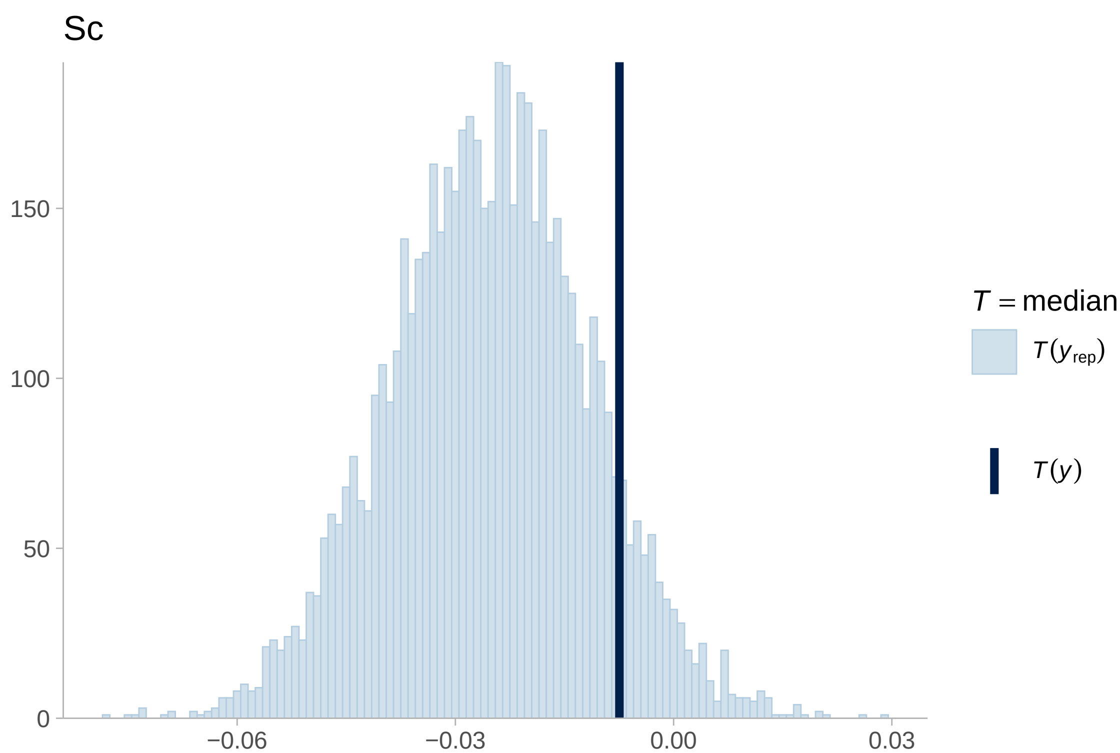
Next is a check comparing replicated medians to the observed medians.

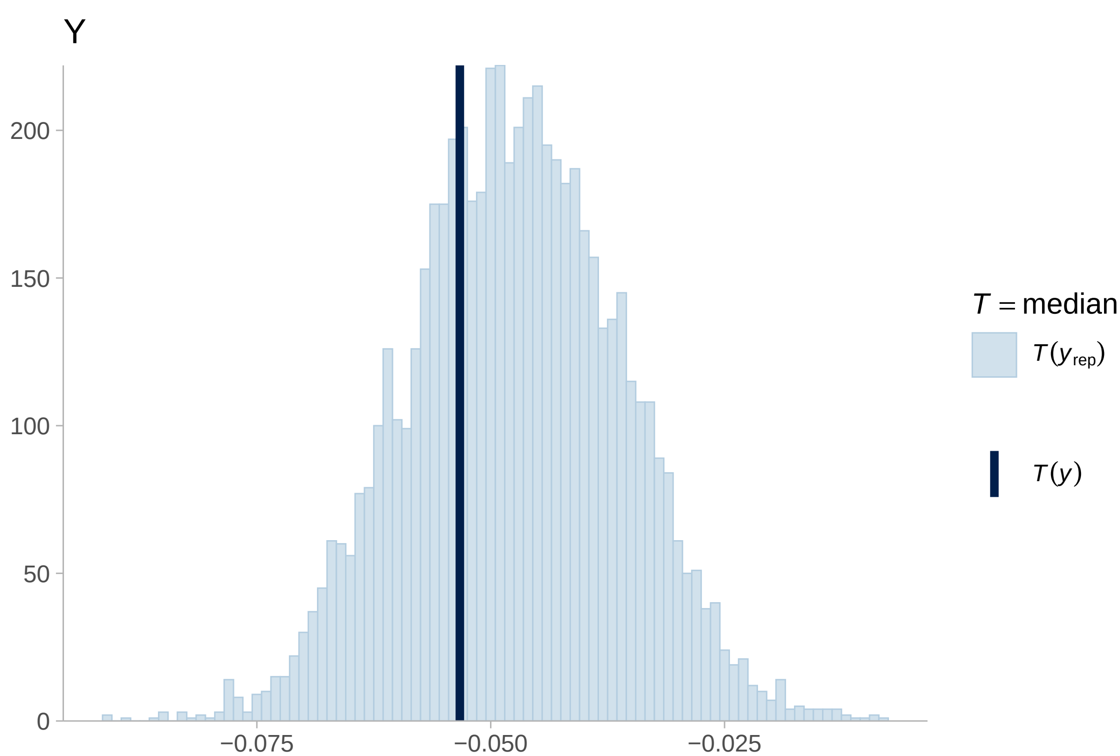


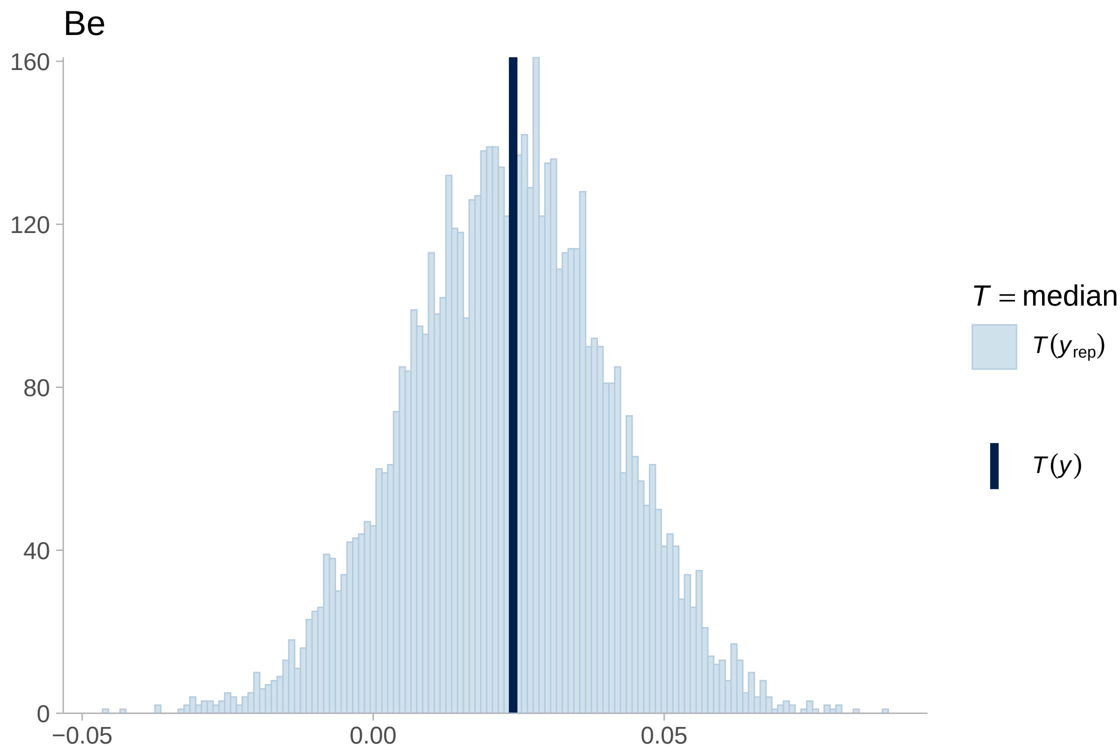


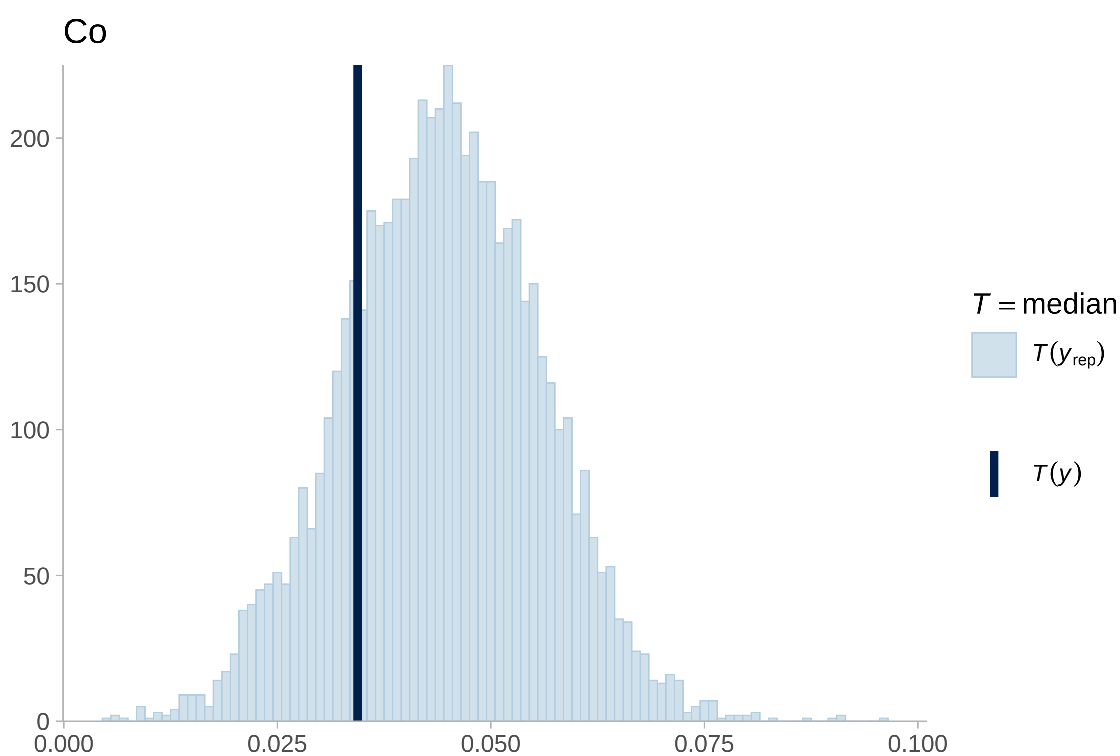

Ho2+
250
200
150
100
50
0
0.11
0.13
0.15
T = median
T(yrep)
T(y)

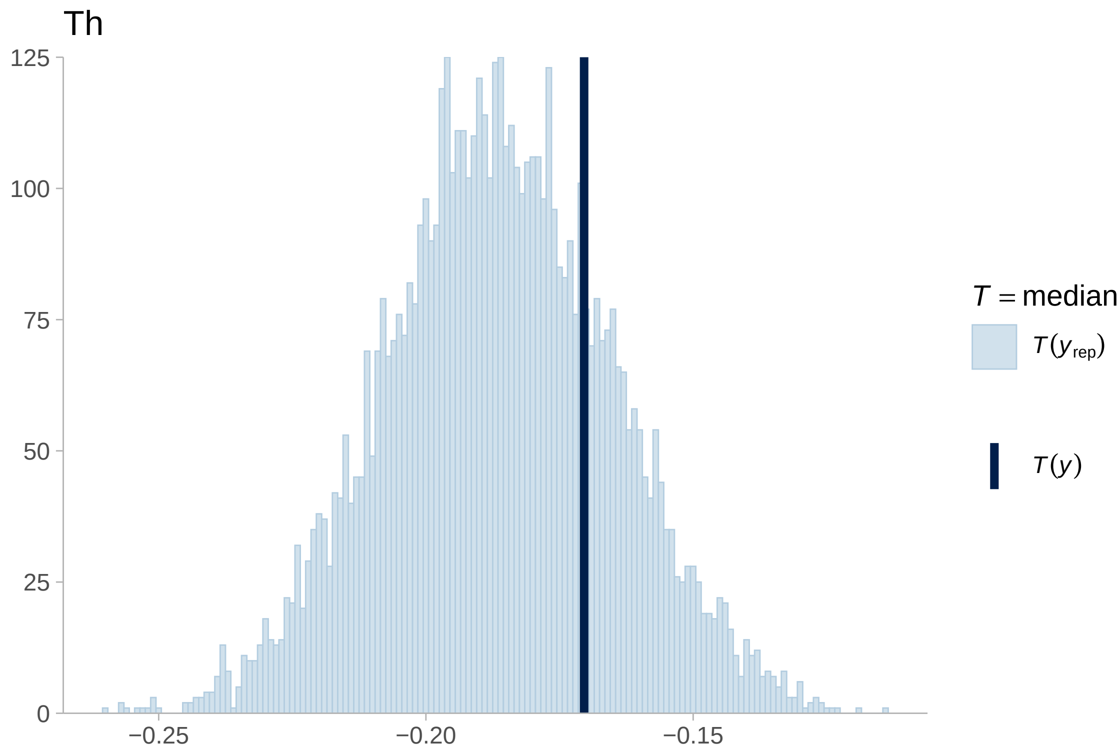

In
200
150
100
50
0
−0.06
−0.04
−0.02
0.00
0.02
T = median
T(yrep)
T(y)

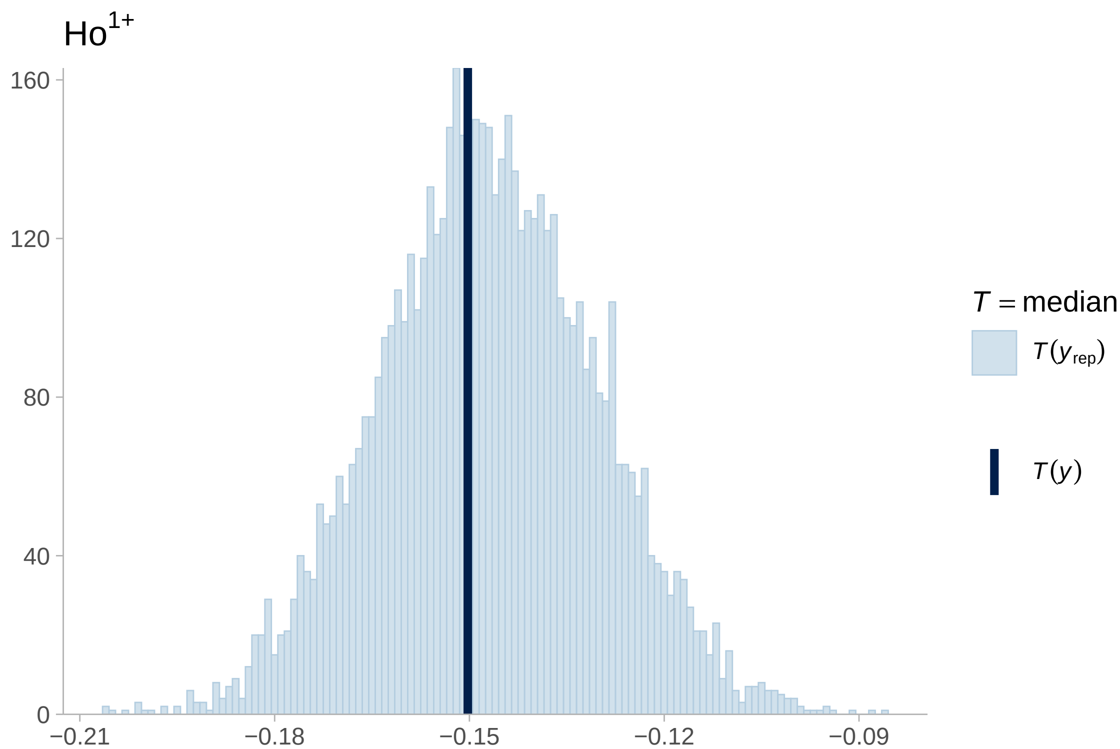

Sc
150
100
50
0
−0.06
−0.03
0.00
0.03
T = median
T(y_rep)
T(y)


Y
200
150
100
50
0
−0.075
−0.050
−0.025
T = median
T(y_rep)
T(y)


Be
160
120
80
40
0
−0.05
0.00
0.05
T = median
$T(y_{rep})$
$T(y)$


Co
200
150
100
50
0
0.000
0.025
0.050
0.075
0.100
T = median
$T(y_{rep})$
$T(y)$

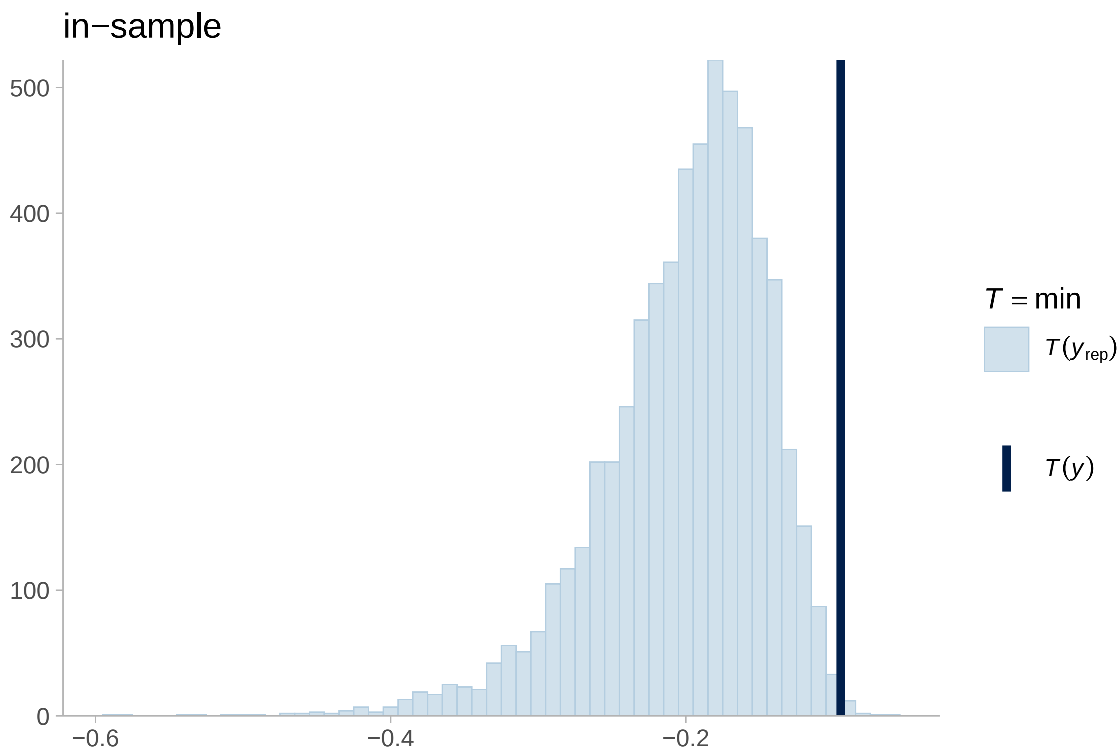


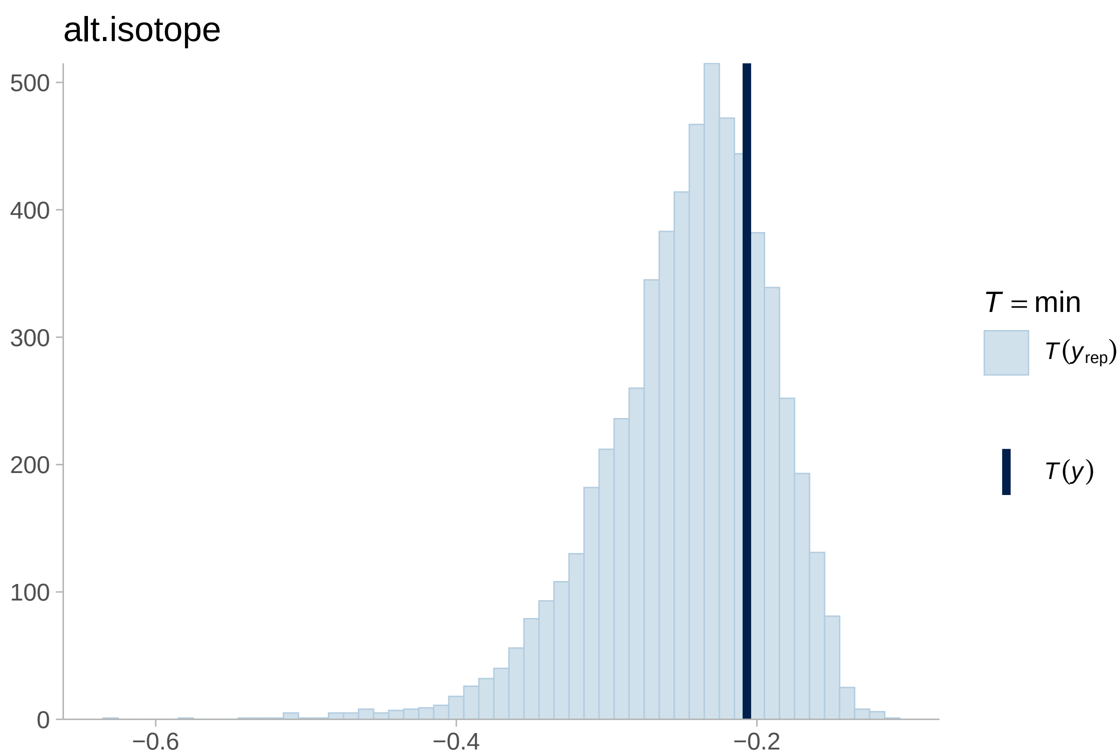


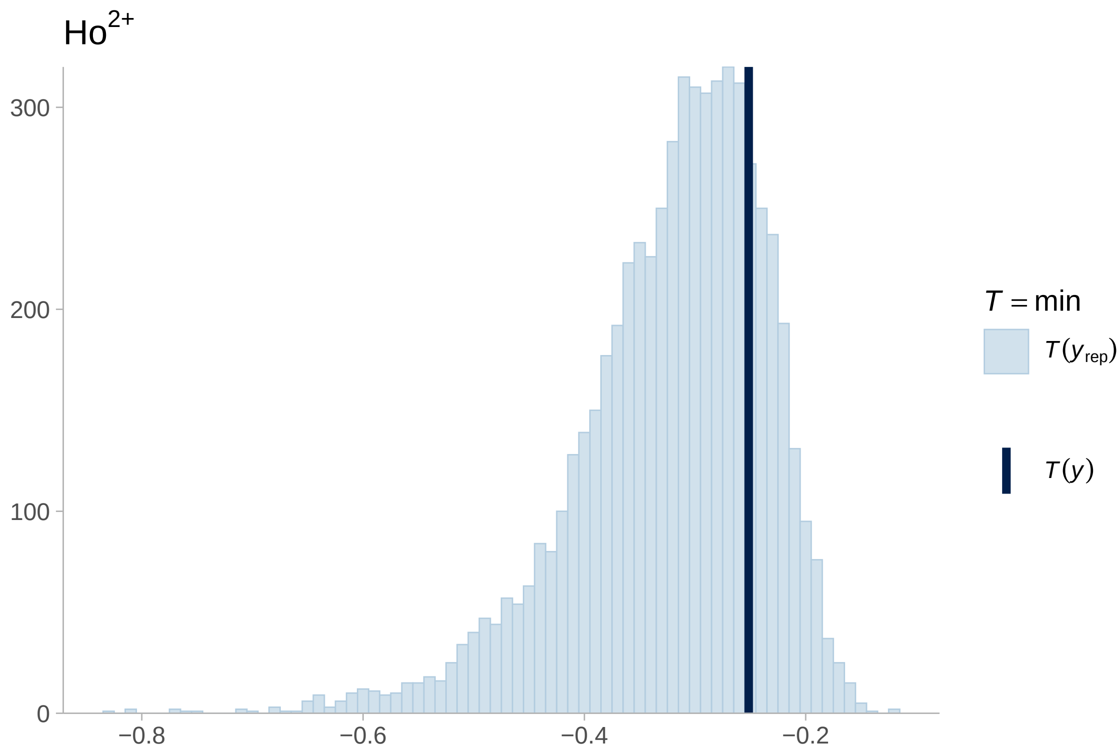
In the above check, the observed median (black vertical line) should fall in the bulk of the replicated medians (blue histogram). The plot for the in-sample suggests that the model may be consistently overestimating the observed median for that method. This would agree with the density plot for the in-sample method above. The scale of the bias is on the order of around 100 ppt, which may be less meaningful in the practical sense.

Min

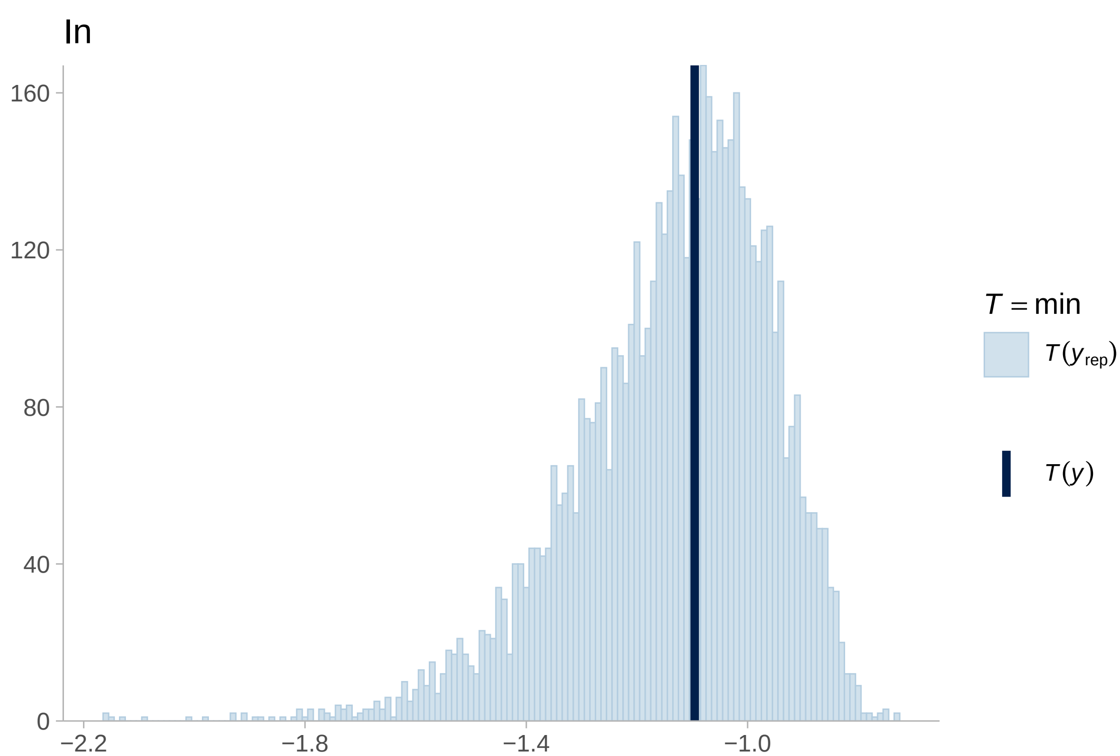
Next is a check comparing replicated mins to the observed mins.



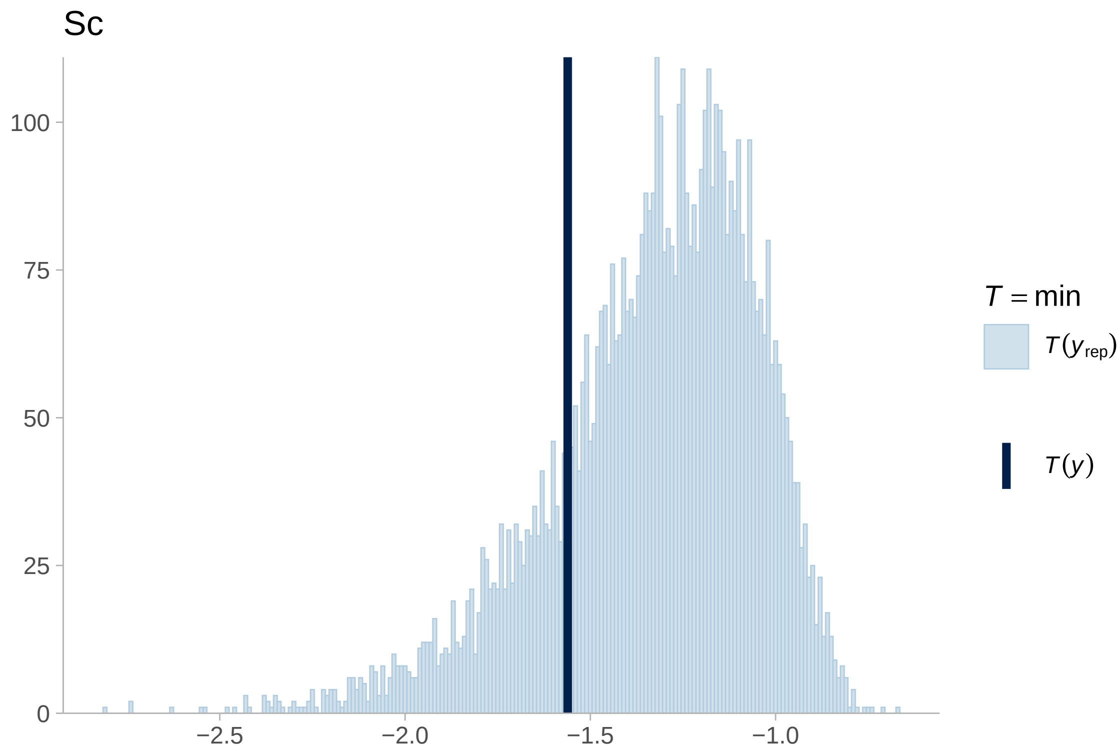


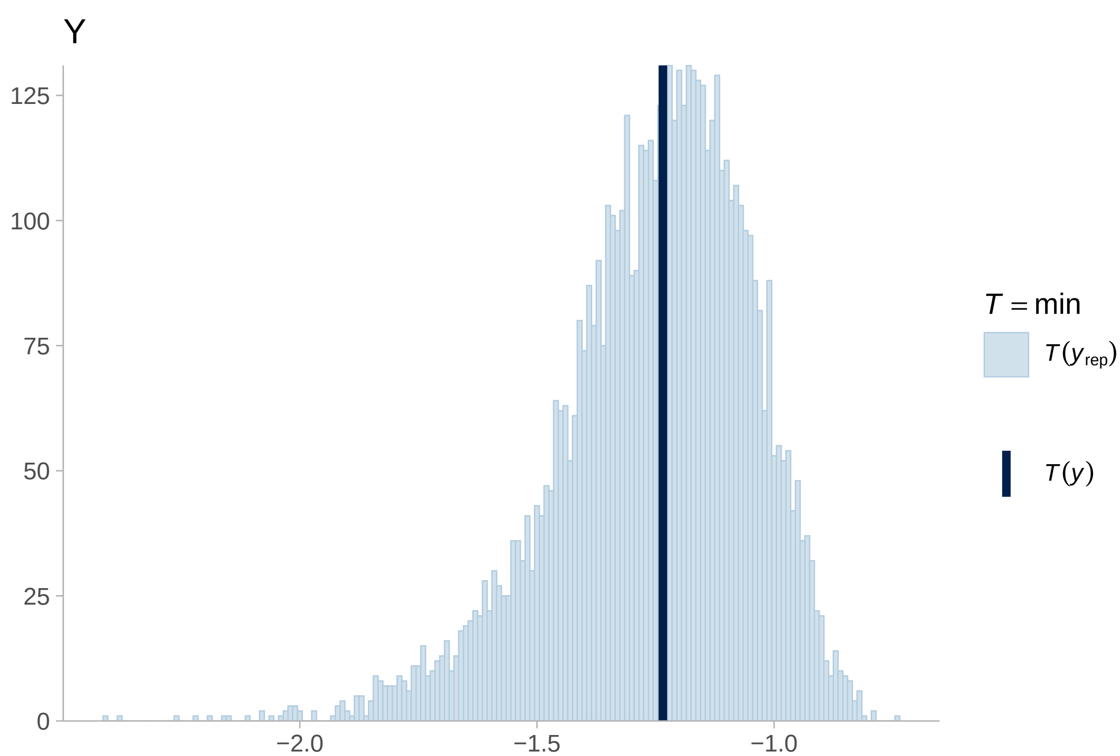


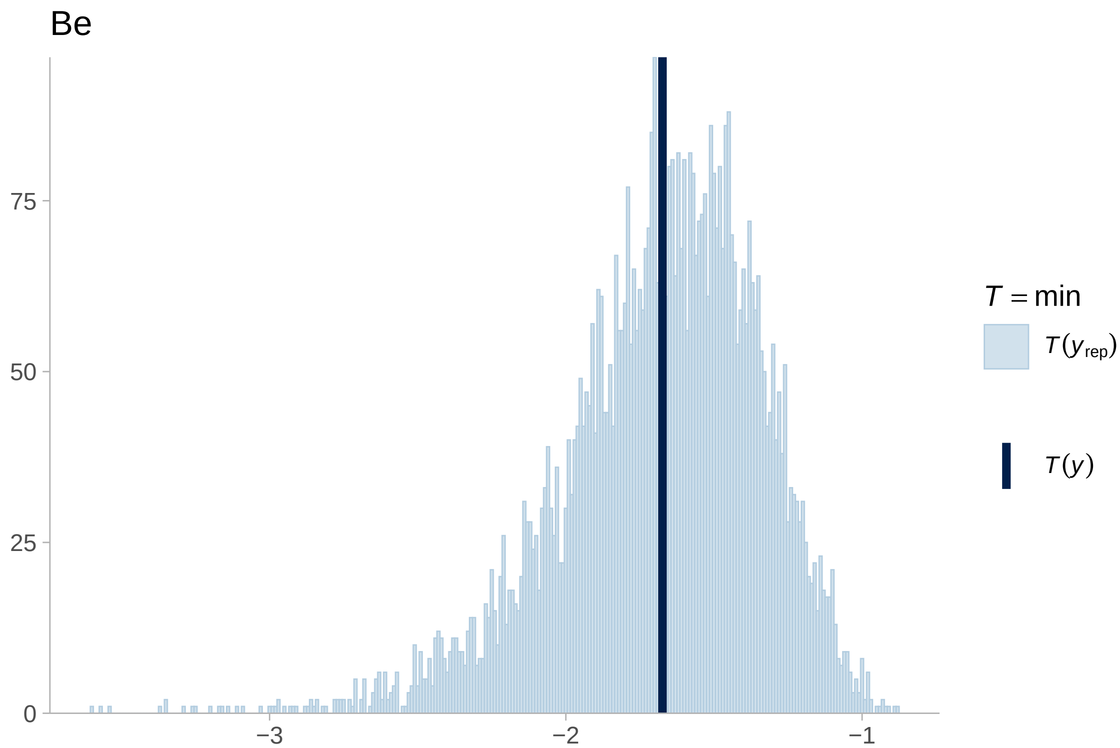
Ho2+
300
200
100
0
−0.8
−0.6
−0.4
−0.2
T = min
T(y_rep)
T(y)

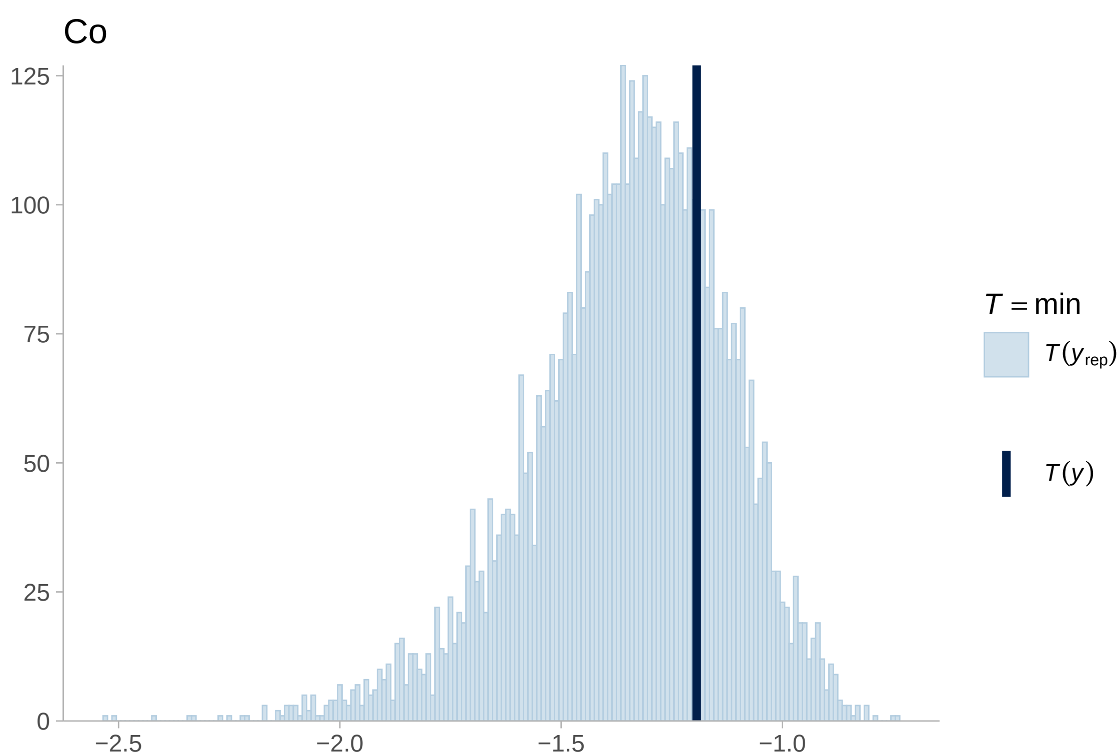


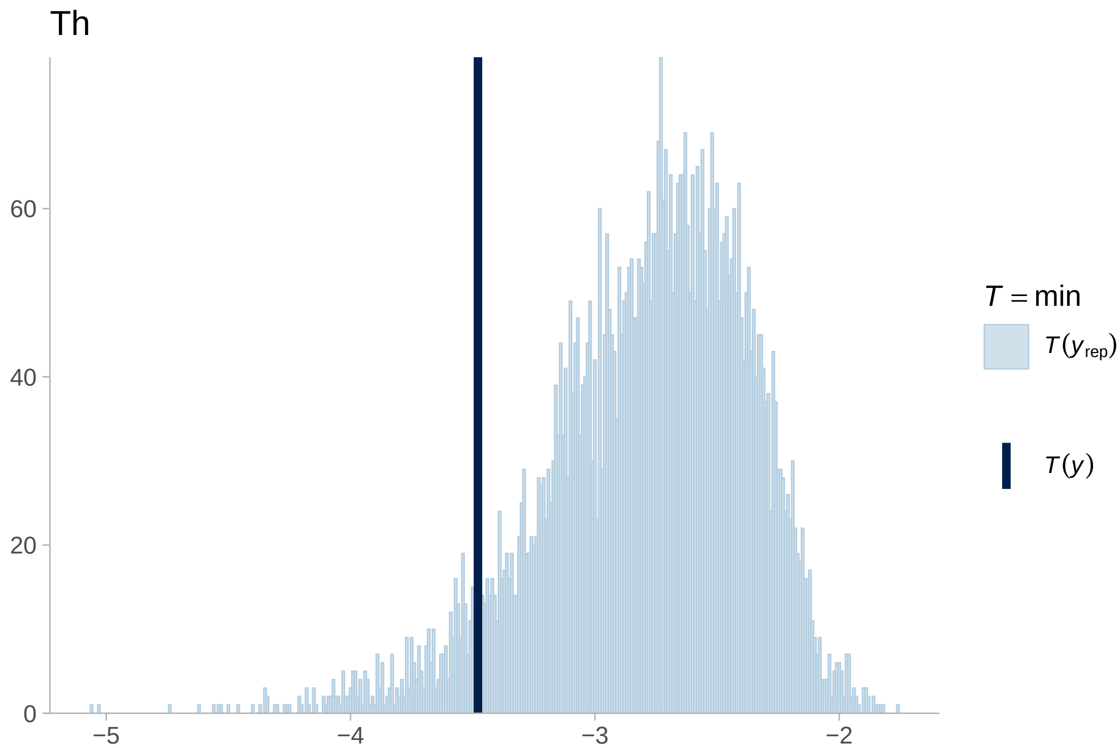
In
160
120
80
40
0
−2.2
−1.8
−1.4
−1.0
T = min
T(y_rep)
T(y)

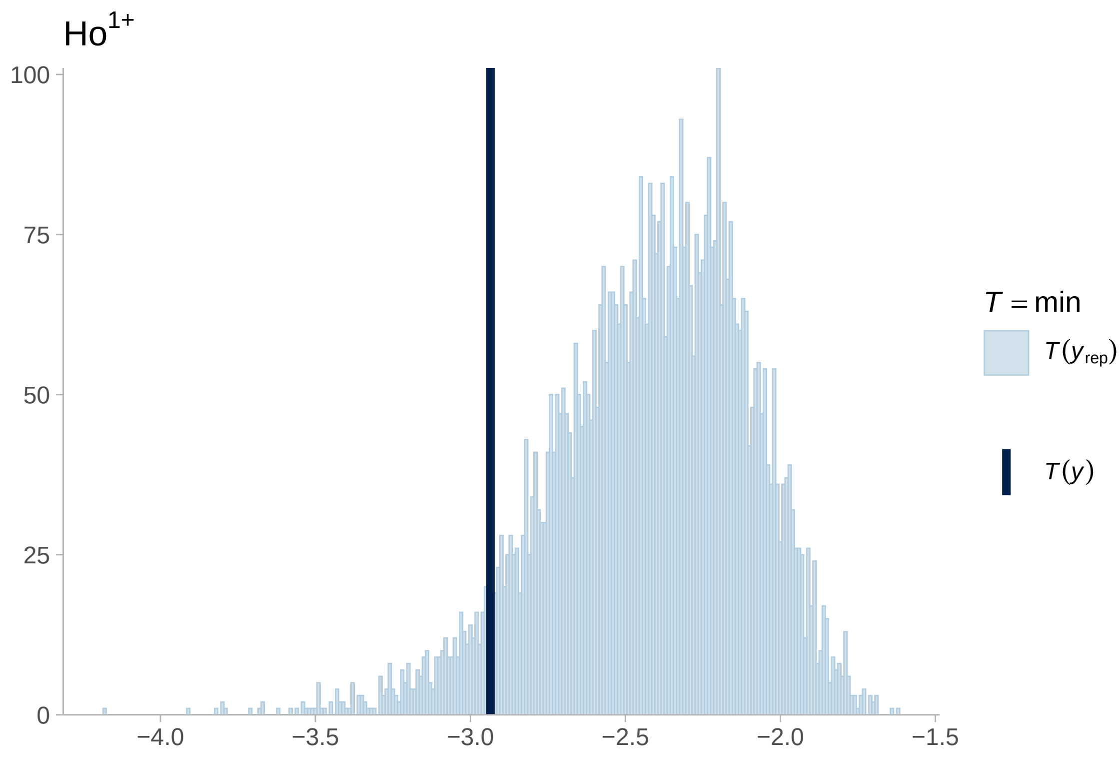

Sc
100
75
50
25
0
−2.5
−2.0
−1.5
−1.0
T = min
T(y_rep)
T(y)


Y
125
100
75
50
25
0
−2.0
−1.5
−1.0
T = min
T(y_rep)
T(y)


Be
75
50
25
0
−3
−2
−1
T = min
T(y_rep)
T(y)


Co
125
100
75
50
25
0
−2.5
−2.0
−1.5
−1.0
T = min
T(y_rep)
T(y)

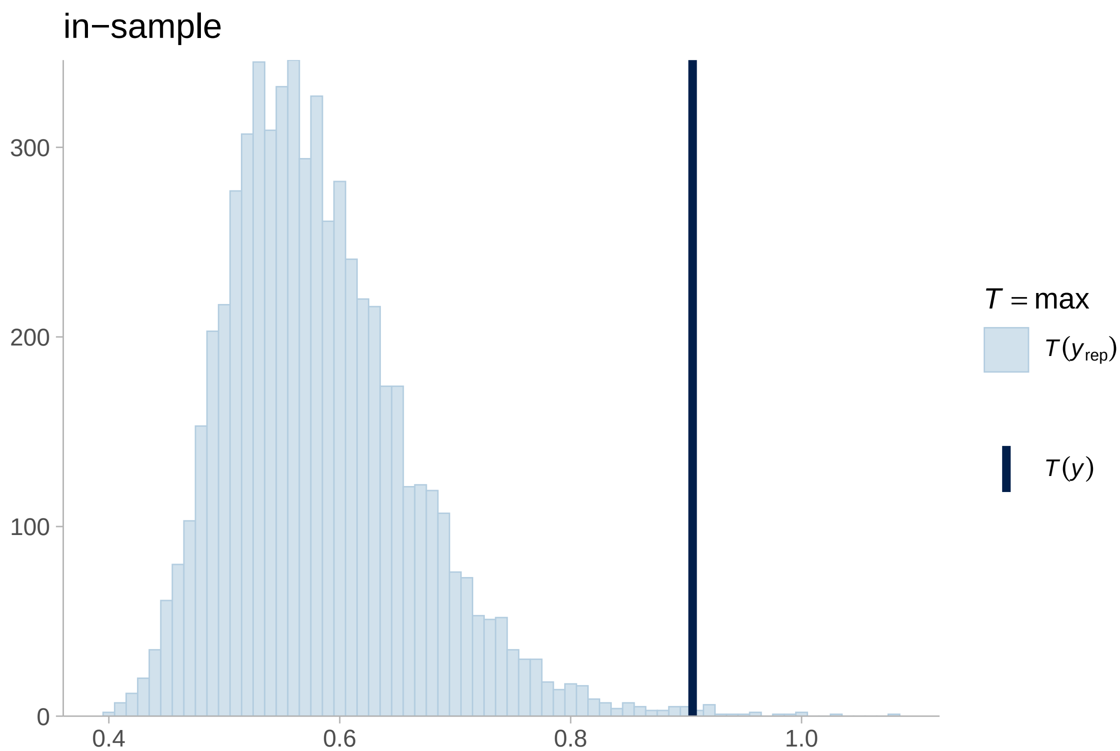


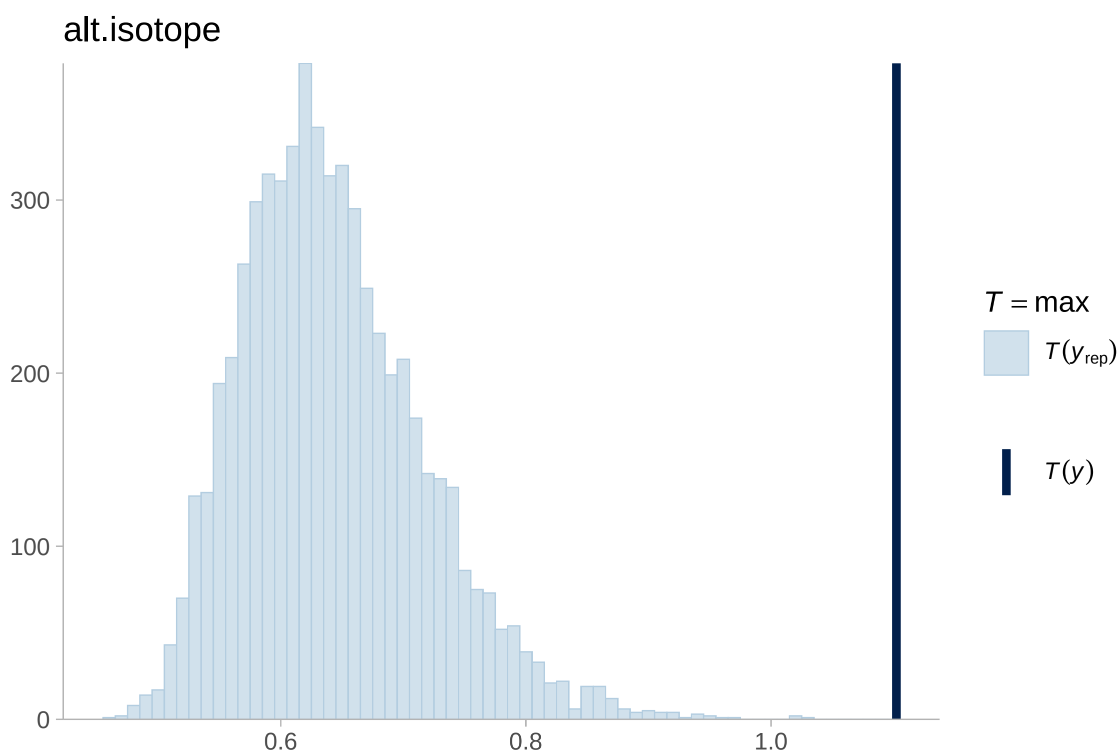


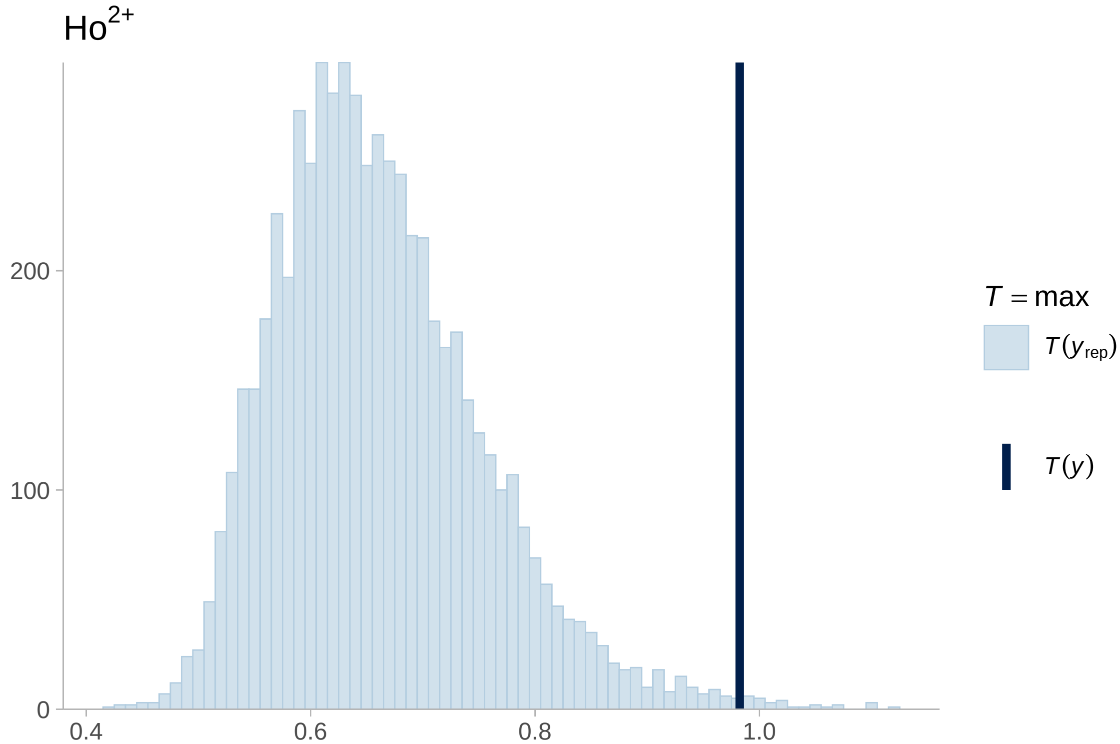
The model is consistently "over-shooting" the min for the in-sample method. This would also agree with the density plot above, where the left tails of the replicated datasets consistently are left of the observed density, if only slightly. On the other hand, the model may be somewhat under-estimating the left tails for Th and Ho^{1+} on average.

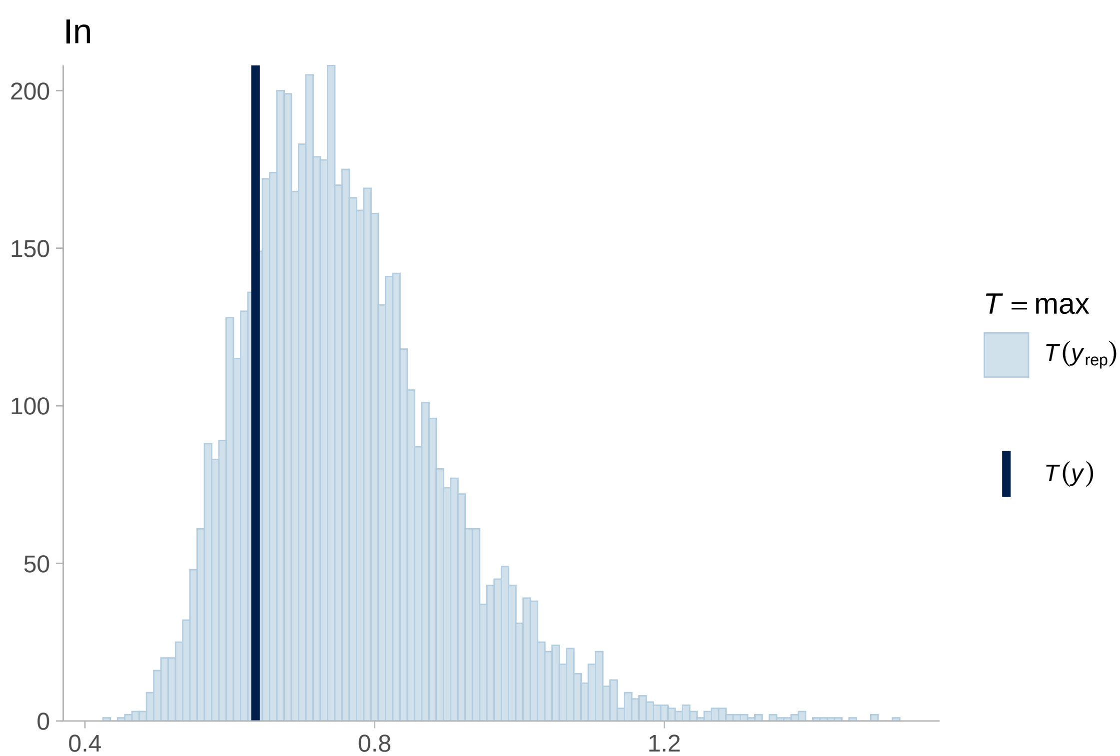
Max

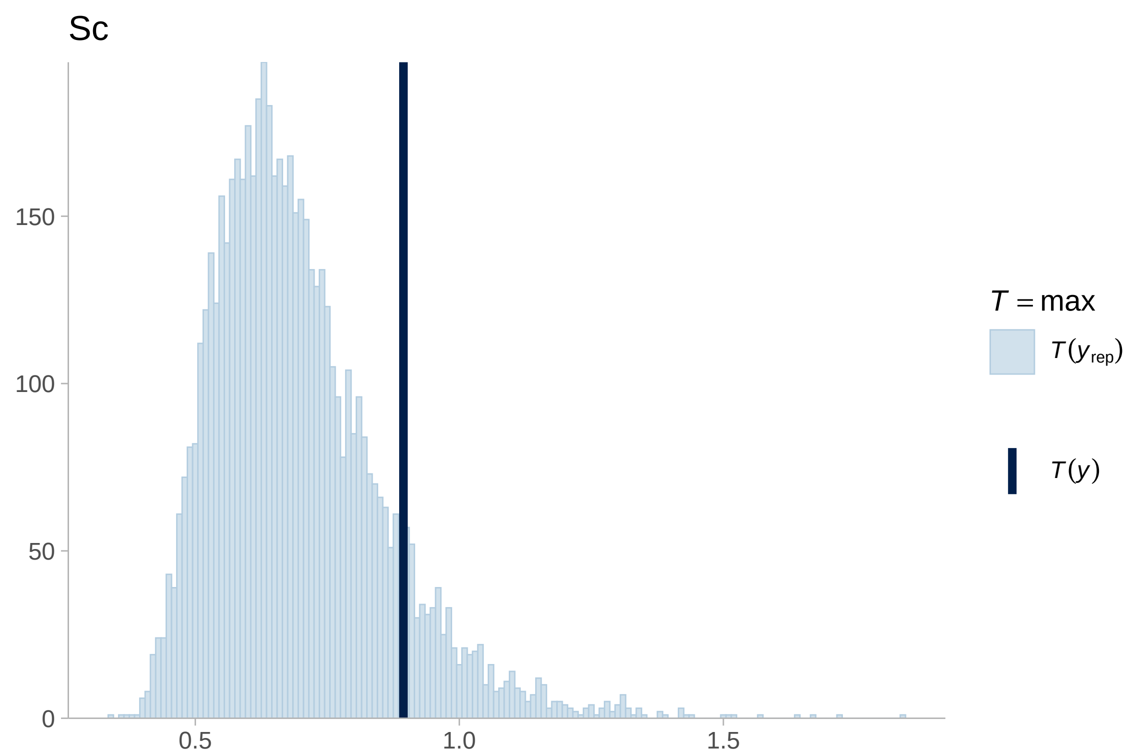
Next is a check comparing replicated maxs to the observed maxs.

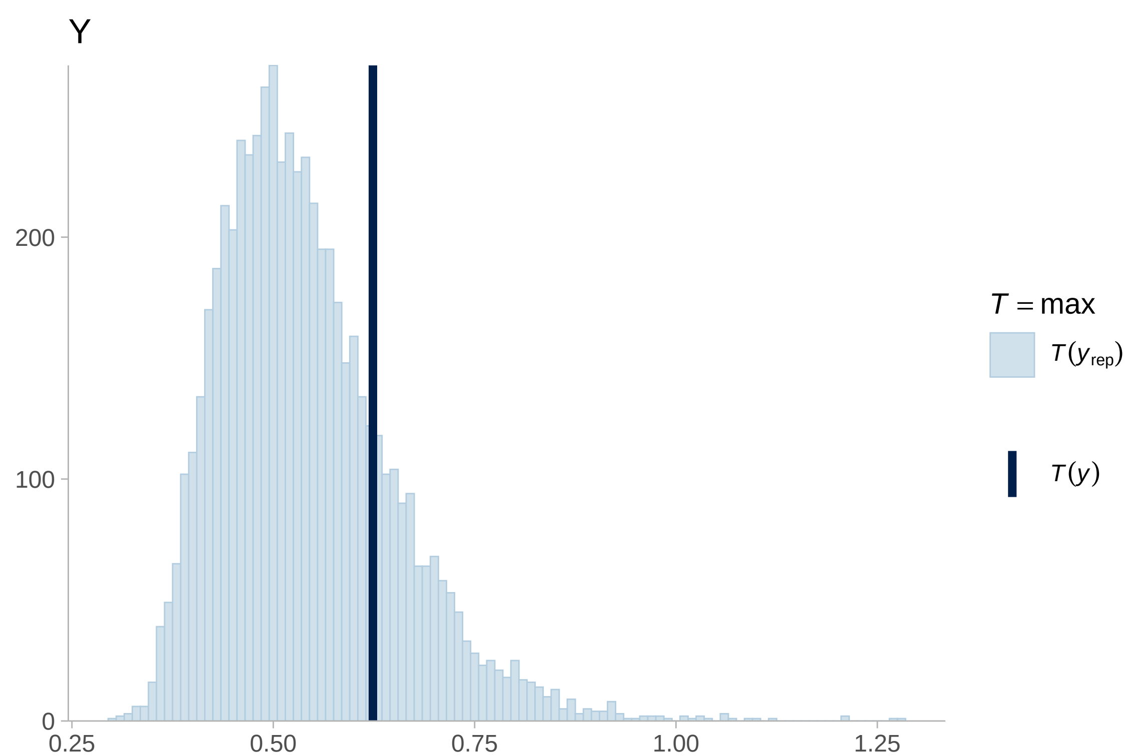


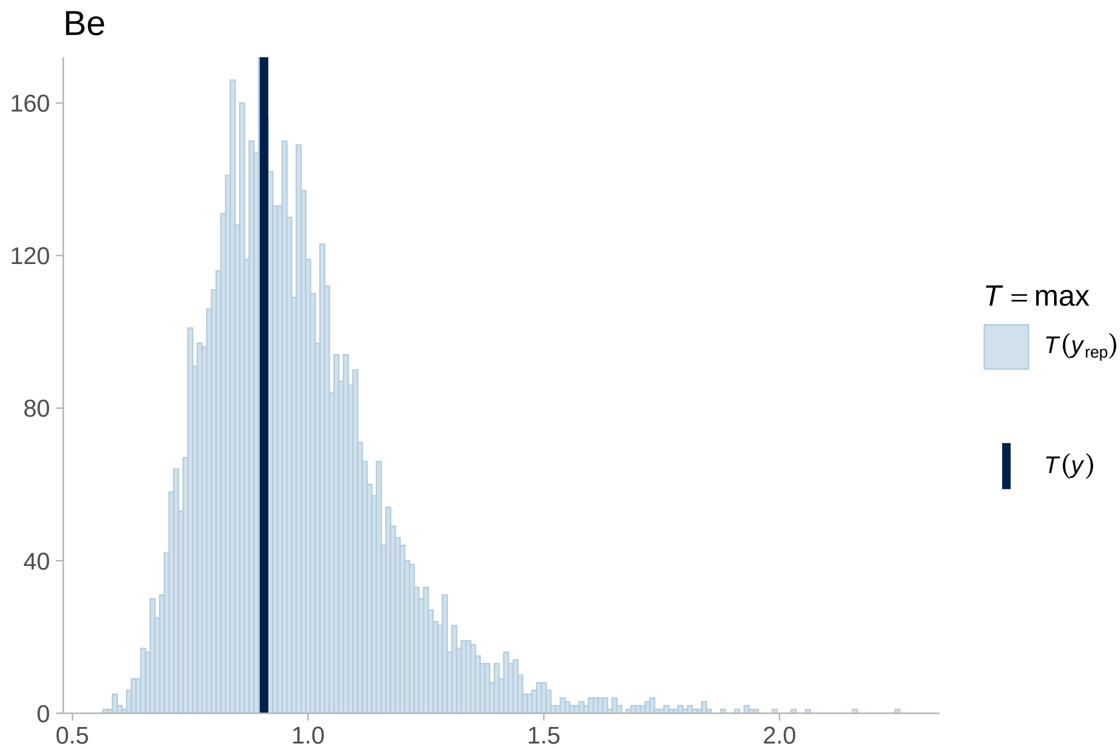



Ho^{2+}
200
100
0
0.4
0.6
0.8
1.0
$T = \max$
$T(y_{rep})$
$T(y)$

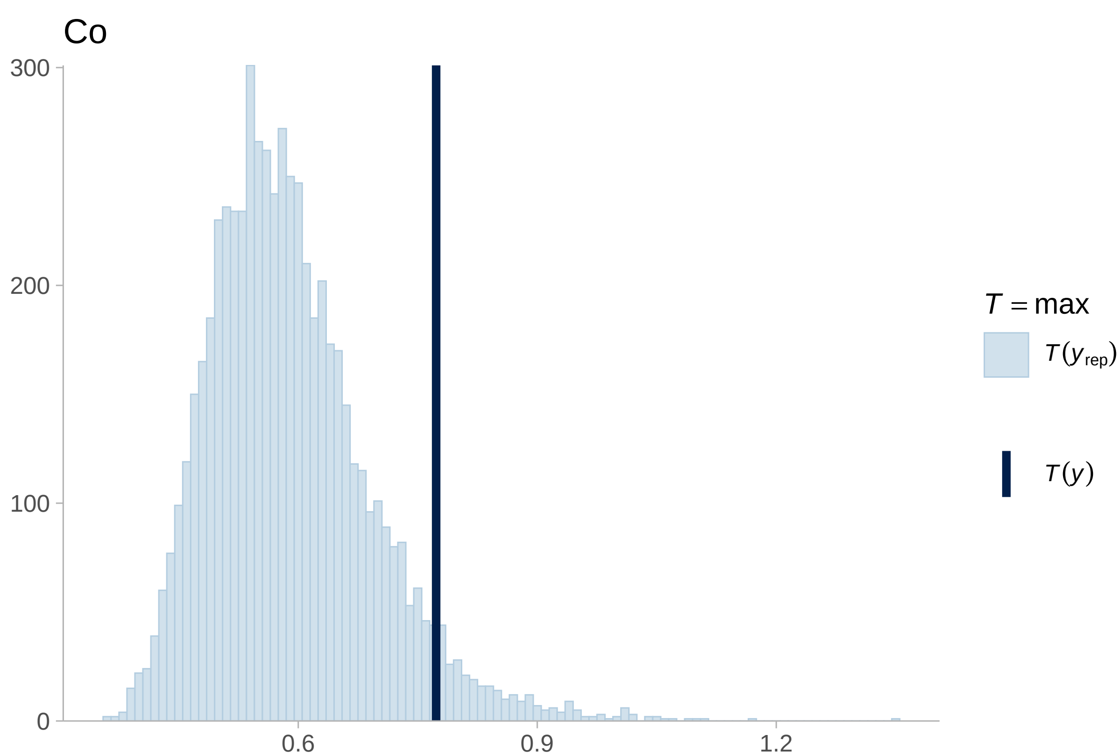

In
200
150
100
50
0
0.4
0.8
1.2
$T = \max$
$T(y_{rep})$
$T(y)$


Sc
150
100
50
0
0.5
1.0
1.5
T = max
T(y_rep)
T(y)

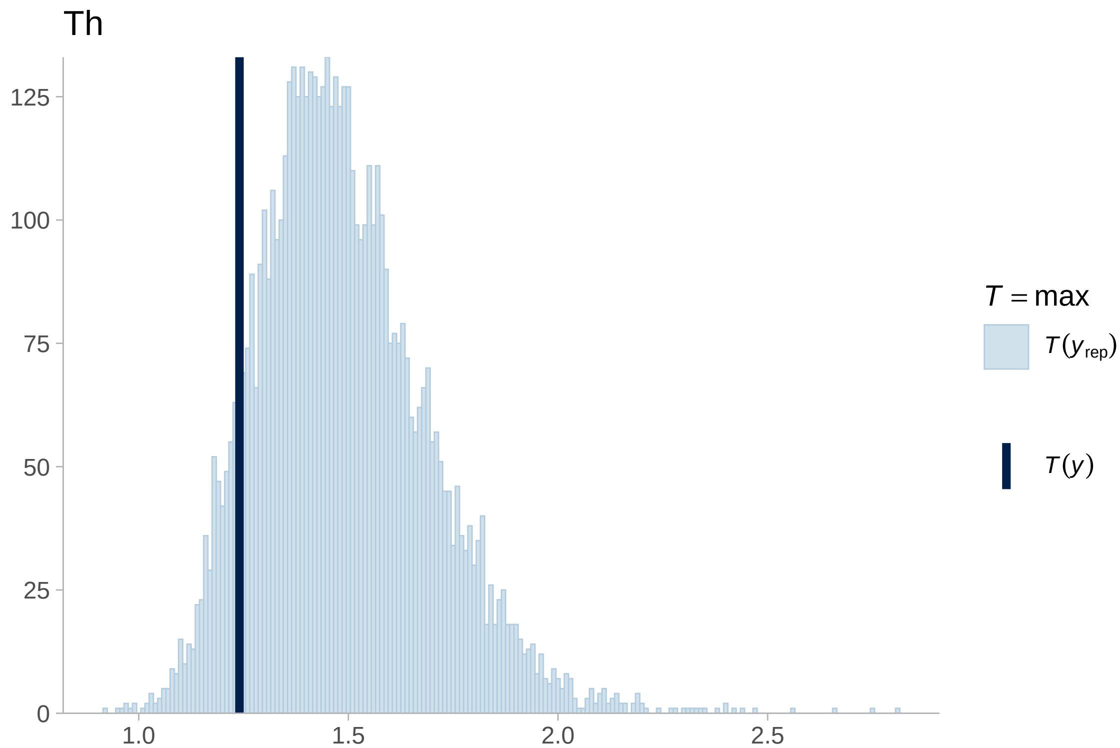

Y
200
100
0
0.25
0.50
0.75
1.00
1.25
T = max
T(y_rep)
T(y)

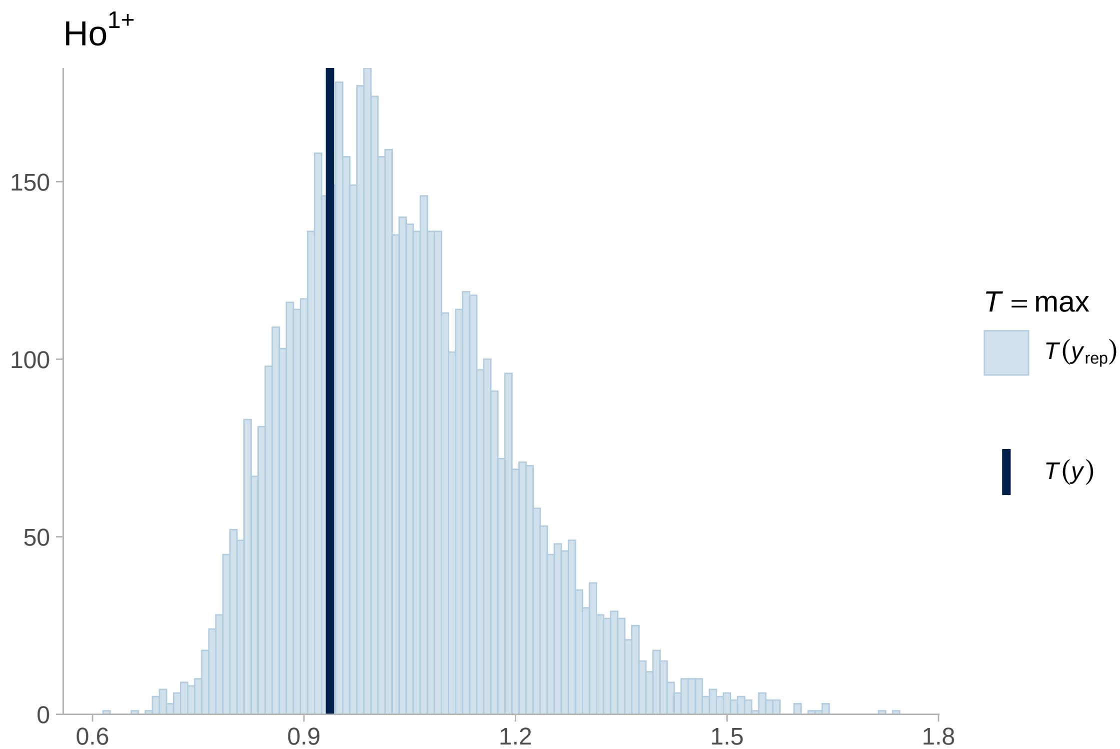


Be
160
120
80
40
0
0.5
1.0
1.5
2.0
T = max
T(y_rep)
T(y)



Co
300
200
100
0
0.6
0.9
1.2
T = max
T(y_rep)
T(y)





Clearly, the model is underestimating the observed max for all of the +2 methods. There is some important aspect of the true data generating process that isn't captured by the model. One potential place to look for explanations for this discrepancy may be the 250ppm matrices, which resulted in fairly extreme under-corrections for all three of the +2 methods, relative to other matrices (see data visualization section above).

K-fold CV

Finally, the k-fold for model 3, with k = 10.

```r
load("full-analysis-files/mod3_As_mv.rda")

library(future)
plan(multisession)
kfold_3 <- kfold(mod3, K = 10, save_fits = TRUE)
save(kfold_3, file = "full-analysis-files/kfold_3.rda")
plan(sequential)
```

The result.

```
##
## Based on 10-fold cross-validation
##
##            Estimate    SE
## elpd_kfold   4536.3  90.0
## p_kfold       699.4  21.9
## kfoldic     -9072.6 180.1
```

The comparison.

```r
load("full-analysis-files/kfold_1.rda")
load("full-analysis-files/kfold_2.rda")
load("full-analysis-files/kfold_3.rda")
loo_compare(kfold_1, kfold_2, kfold_3)
```

```
##      elpd_diff se_diff
## mod3     0.0       0.0
## mod3  -390.5      37.0
## mod1 -1139.3      54.3
```

The more flexible model 3 is clearly favored.

Posterior inferences

With the potential limitations to the model suggested by the predictive checks loo-CV diagnostics, the focus moves next to posterior inferences, which will be based on the last model.

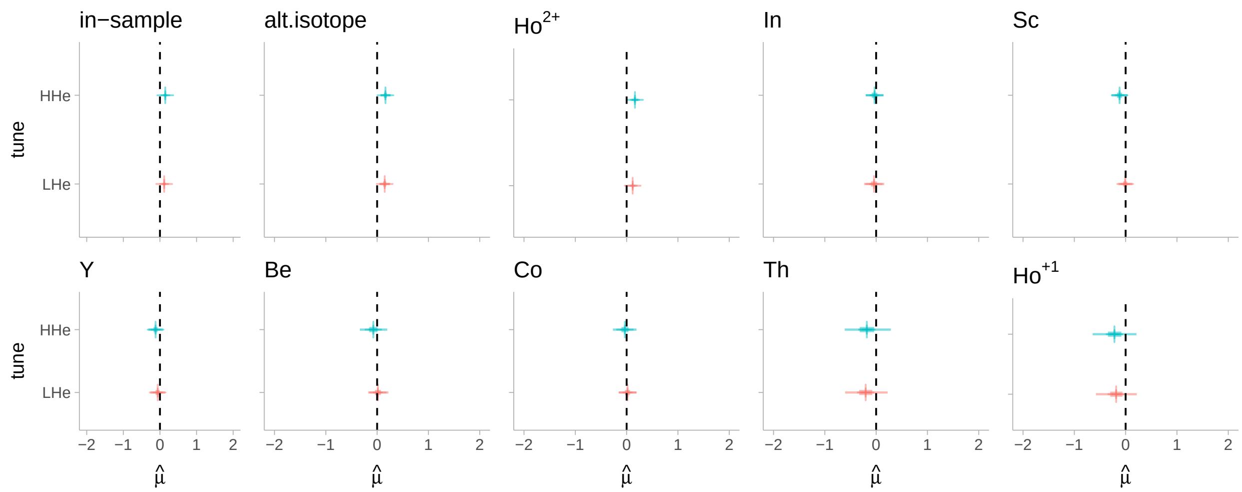
Conditional means

Below are plots of the fitted values of the linear predictors for both μ and σ across the different experimental groupings. These estimates pertain to expected population averages, conditional on the model, data, and priors, and include uncertainty in the estimation of each of the parameters comprising the linear predictors.

μ First, the estimated conditional means for the μ component of the model.

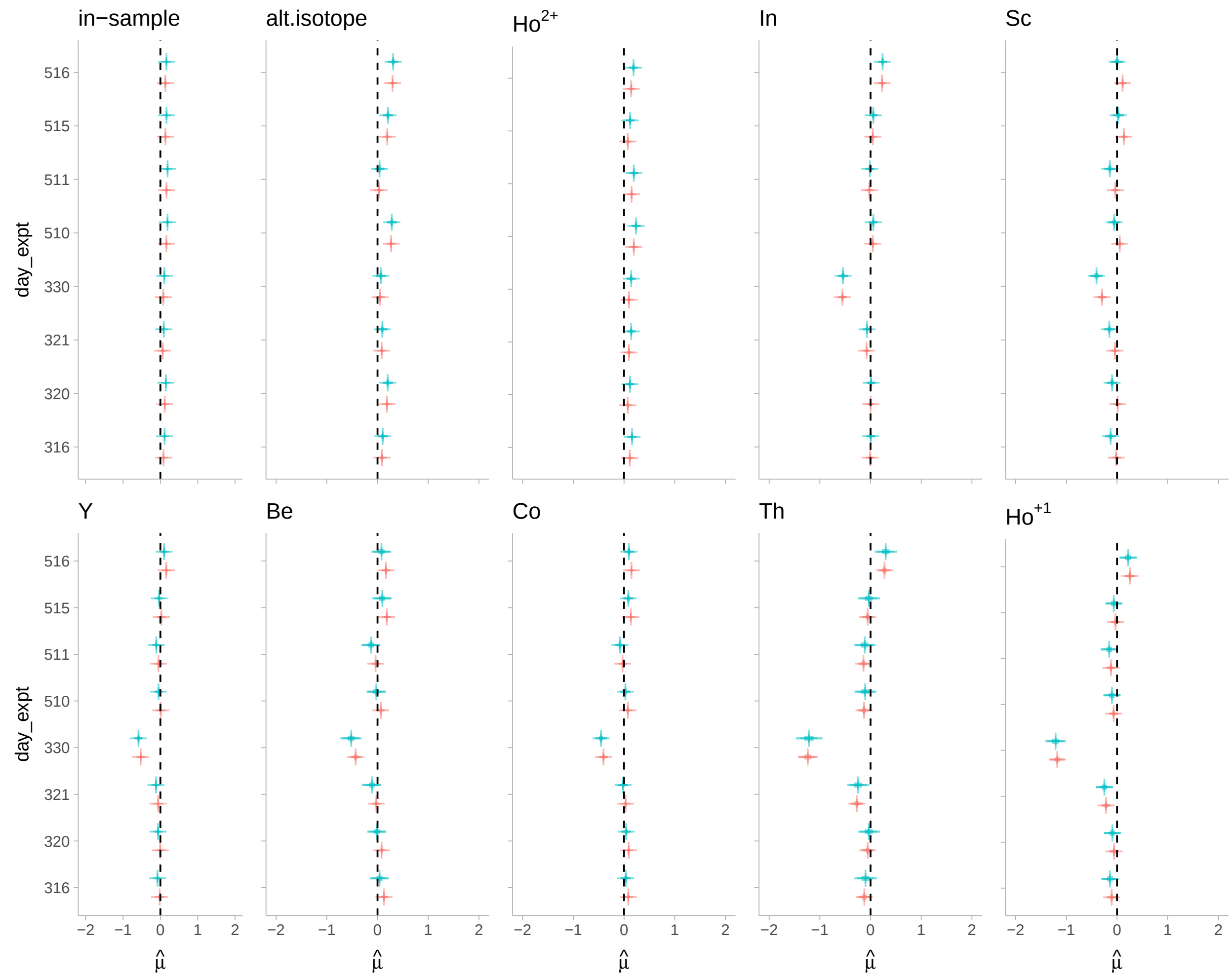
Tune Estimate means conditional on method and tune while marginalizing over matrix and day.

```
load("full-analysis-files/df_mv_as.rda")
load("full-analysis-files/mod3_As_mv.rda")
fitted_method_tune <- df_mv_as %>%
  add_fitted_draws(mod3,
                   dpar = FALSE,
                   re_formula = NA,
                   cores = 1)
save(fitted_method_tune, file = "full-analysis-files/fitted_method_tune.rda")
```



Day Estimate means conditional on method and day while marginalizing over matrix.

```
load("full-analysis-files/df_mv_as.rda")
load("full-analysis-files/mod3_As_mv.rda")
fitted_method_day <- df_mv_as %>%
  add_fitted_draws(mod3,
                   dpar = FALSE,
                   re_formula = ~ (1 | day_expt),
                   cores = 1)
save(fitted_method_day, file = "full-analysis-files/fitted_method_day.rda")
```

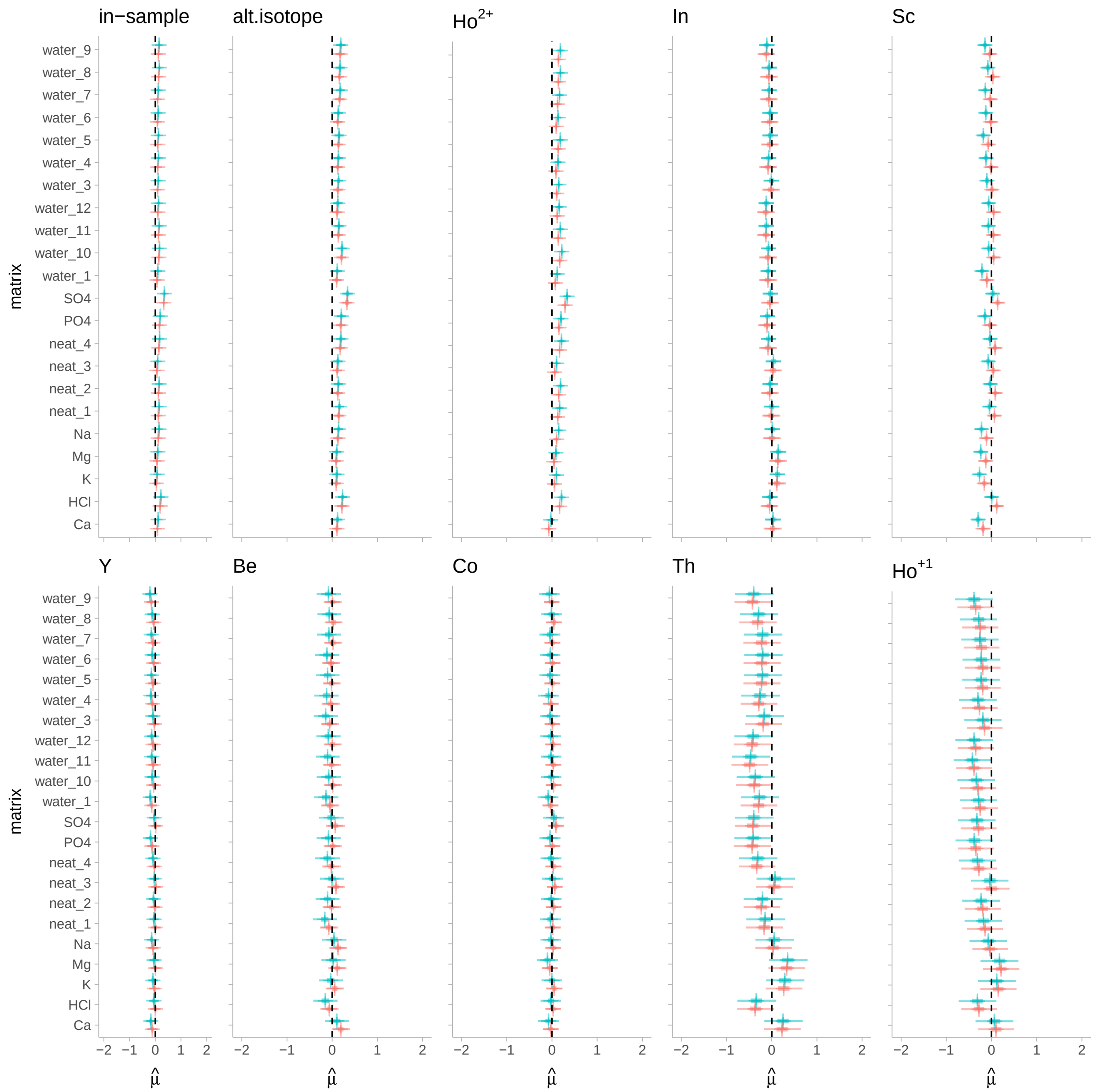


This figure illustrates the large over-correction (i.e, negative bias) expected on 3/30 for all of the +1 methods. By comparison, the means for the +2 methods are relatively consistent across the days of the experiment. Also note the consistent, if slight tendency towards under-correction (i.e., positive bias) estimated across most days for the +2 methods. Within days, there were no clear differences in means due to tune setting.

Matrix Estimate means conditional on method and matrix while marginalizing over day.

```
load("full-analysis-files/df_mv_as.rda")
load("full-analysis-files/mod3_As_mv.rda")

fitted_method_matrix <- df_mv_as %>%
  add_fitted_draws(mod3,
                   dpar = FALSE,
                   re_formula = ~ (1 | matrix),
                   cores = 1)
save(fitted_method_matrix, file = "full-analysis-files/fitted_method_matrix.rda")
```



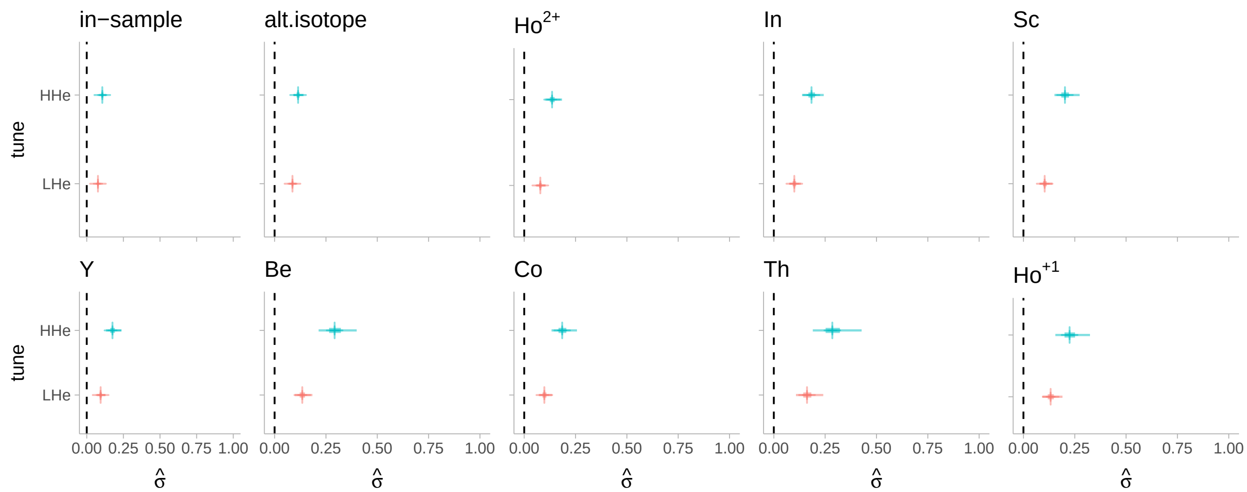
The estimated means for the +2 methods are again largely in the direction of slight under-correction, whereas most of the +1 methods vary around zero bias, with the exception of Th and Ho^{+1}, which tend more towards over-correction, though the inferences are uncertain (credible intervals overlaps zero for most part). Matrix to matrix variability for all of the methods is expected to be relatively less substantial compared to day to day variation above. The tune effect also doesn't appear to be particularly large, if important at all, for most methods.

σ Next, the conditional means for the σ component of the model.

```
load("full-analysis-files/df_mv_as.rda")
load("full-analysis-files/mod3_As_mv.rda")
```

```r
fitted_sigma_tune <- df_mv_as %>%
  add_fitted_draws(mod3,
                   dpar = "sigma",
                   re_formula = NA,
                   cores = 1)
save(fitted_sigma_tune, file = "full-analysis-files/fitted_sigma_tune.rda")
```

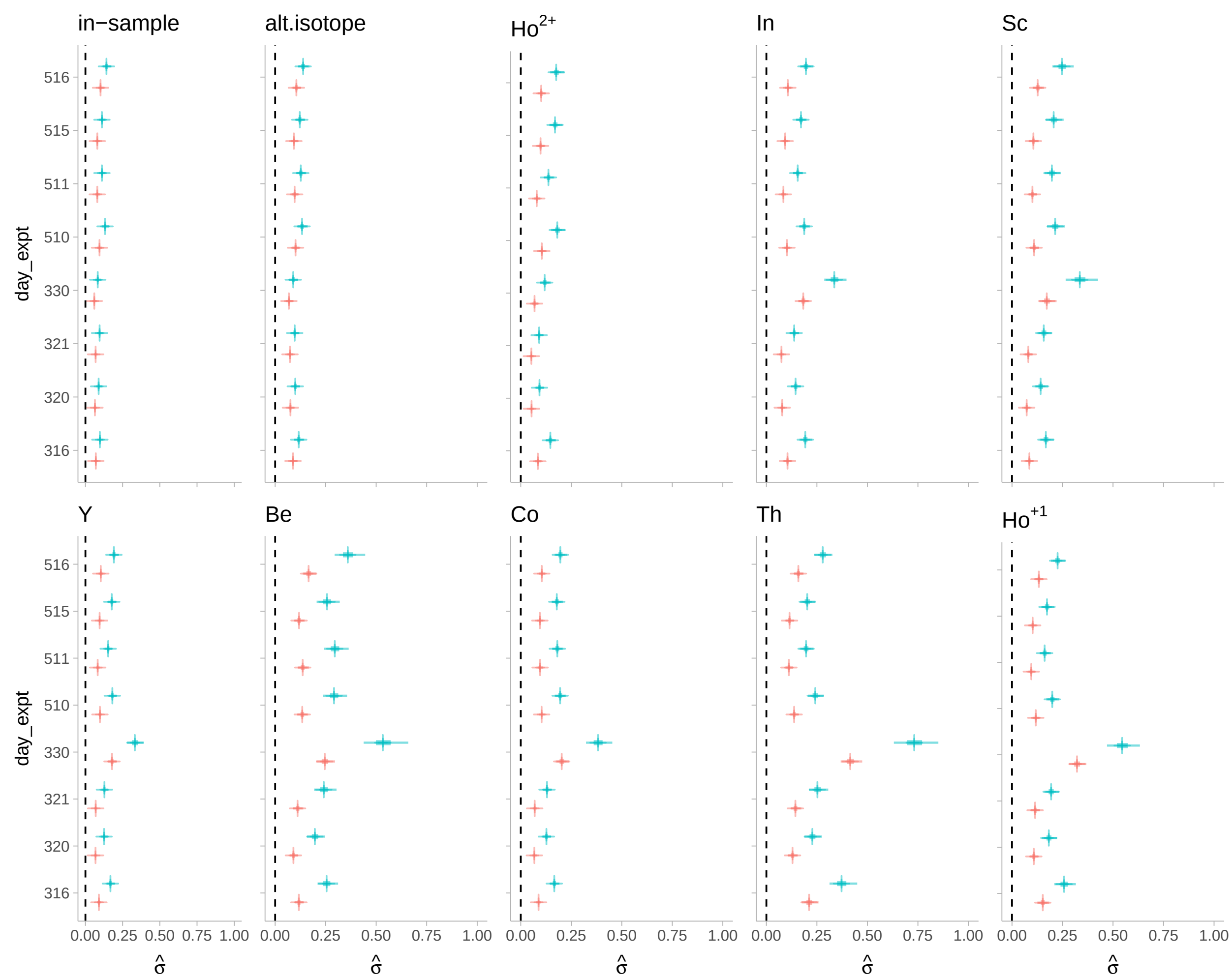
Tune The expected standard deviations by tune after marginalizing over matrix are below.




```r
load("full-analysis-files/df_mv_as.rda")
load("full-analysis-files/mod3_As_mv.rda")

fitted_sigma_day <- df_mv_as %>%
  add_fitted_draws(mod3,
                   dpar = "sigma",
                   re_formula = sigma ~ (1 | day_expt),
                   cores = 1)
save(fitted_sigma_day, file = "full-analysis-files/fitted_sigma_day.rda")
```

Day The expected standard deviations by day after marginalizing over matrix are below.

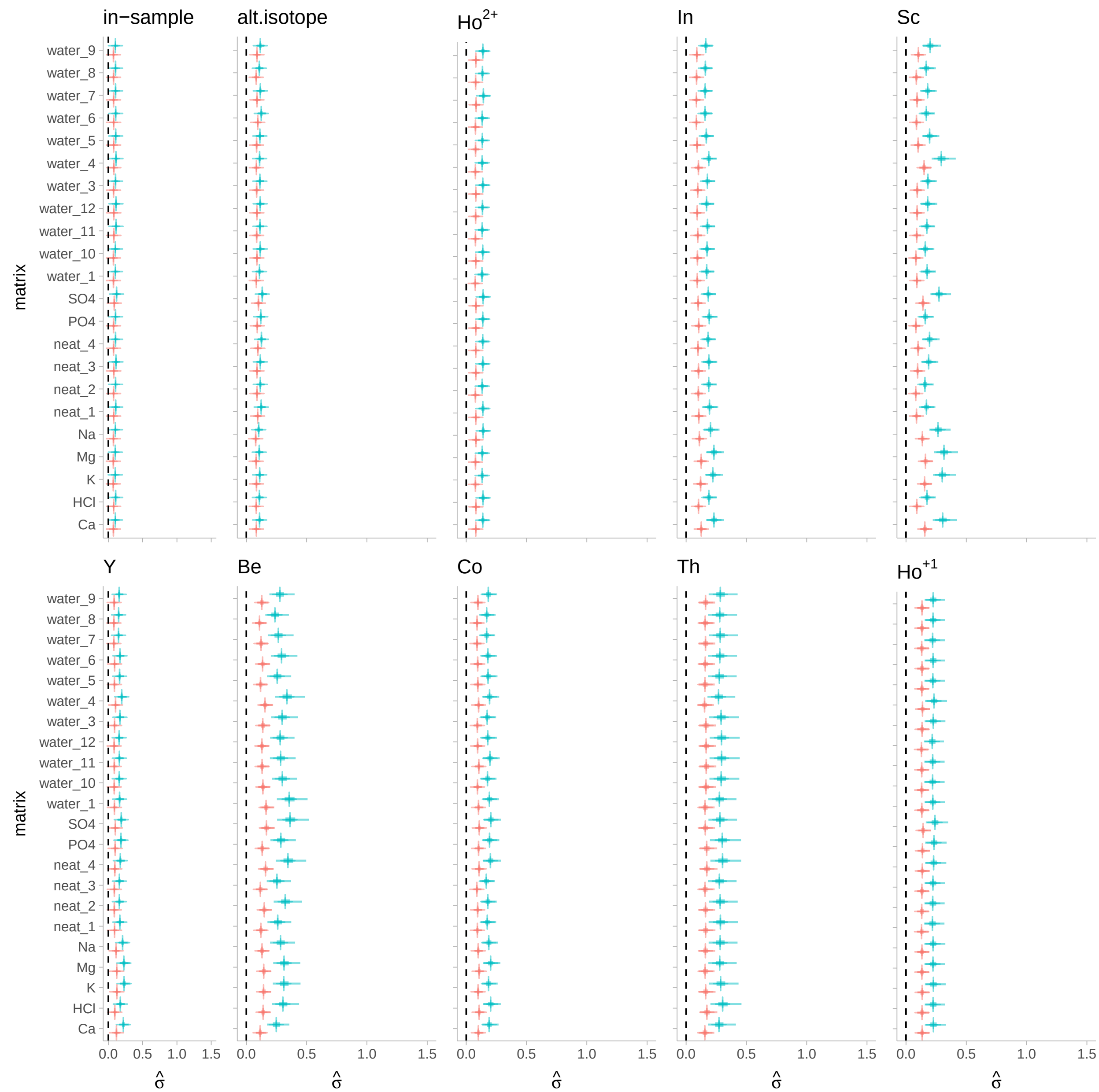


This figures suggest some differences in standard deviation due to tune setting, particularly for the +1 methods. The standard deviation was also estimated to vary fairly substantially day to day, no matter the tune setting, for most of the +1 methods. The largest standard deviation is estimated for 3/30 (cone change day) for all of those methods. Standard deviation may vary slightly day to day for the +2 methods, but the large effect on 3/30 is not apparent. Differences due to tune setting also aren't as clear for the +2 methods.

```r
load("full-analysis-files/df_mv_as.rda")
load("full-analysis-files/mod3_As_mv.rda")

fitted_sigma_matrix <- df_mv_as %>%
  add_fitted_draws(mod3,
                   dpar = "sigma",
                   re_formula = sigma ~ (1 | matrix),
                   cores = 1)
save(fitted_sigma_matrix, file = "full-analysis-files/fitted_sigma_matrix.rda")
```

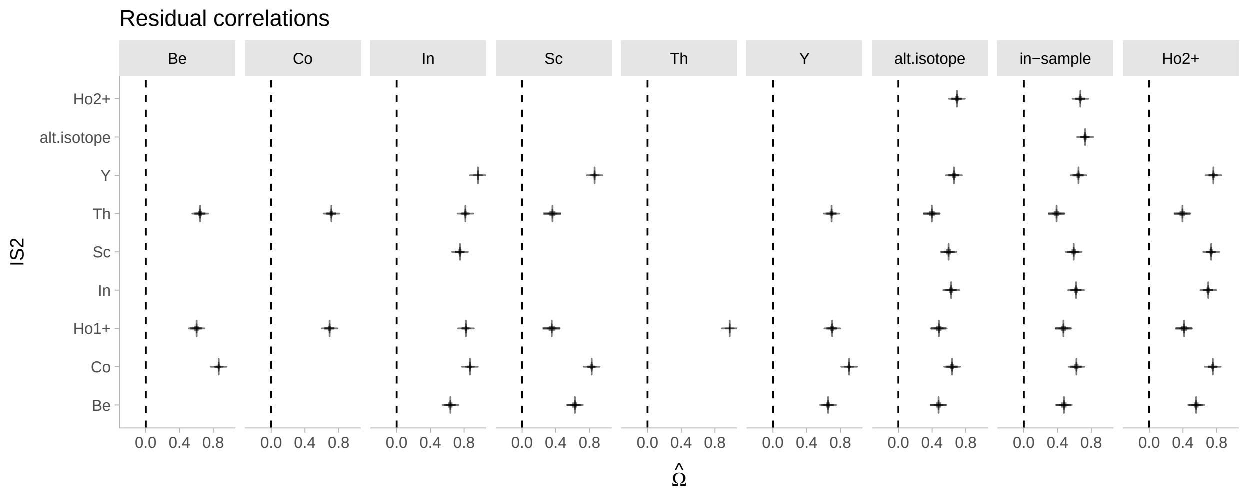
Matrix Finally, the expected standard deviations after marginalizing over day.



Overall, excepting perhaps the *Sc* method, matrix to matrix variability in standard deviation was estimated to be very low to negligible.

Residual Correlations

Next, the estimated residual correlations among IS methods. Note that not all pairs are shown under each method, since many are redundant, but all non-redundant pairs are shown.



The residual correlations are fairly large (e.g., > 0.8) in many cases, particularly between the +1 methods. These correlations may suggest where sources of unexplained variability may be common between methods.

Predictions

Next, the model was used to make out-of-sample predictions to a "new" matrix, or day, or both. Here, "new" refers to an out-of-sample matrix and/or day from the hypothetical broader population of matrices or days. When considering predictions to a new matrix, for example, the prediction is for an average matrix estimated from the observed matrices, and includes the uncertainty estimated from the observed matrix to matrix variation. In contrast to the conditional *means* above, the predictions below incorporates both the μ (expectation) and σ (standard deviation) components of the fitted model. That is, the predictions, y_{new}, are drawn from

$$y_{new} \sim N(\hat{\mu}, \hat{\sigma})$$

so they include both parameter and sampling uncertainty.

Hypothetical data Hypothetical data are created below to hold the conditions that the model will predict to. The first condition is for a "new" day, but with the same matrix and tune conditions as observed. The new day can be considered as any day that might belong to the same population as the observed days. An average day, assuming the days of this experiment were drawn from a larger population of days; a future or past day, for example.

```
load("full-analysis-files/df_mv_as.rda")

new_day_dat <- expand_grid(matrix = factor(levels(df_mv_as$matrix)),
                           day_expt = "new_day",
                           tune = factor(c("LHe", "HHe"))) %>%
  mutate(tune = relevel(tune, ref = "LHe"))

save(new_day_dat, file = "full-analysis-files/new_day_dat.rda")
```

```
## # A tibble: 44 x 3
##    matrix day_expt tune
##    <fct>  <chr>    <fct>
##  1 Ca     new_day  LHe
##  2 Ca     new_day  HHe
```

```
##   3 HCl    new_day  LHe
##   4 HCl    new_day  HHe
##   5 K      new_day  LHe
##   6 K      new_day  HHe
##   7 Mg     new_day  LHe
##   8 Mg     new_day  HHe
##   9 Na     new_day  LHe
##  10 Na     new_day  HHe
## # ... with 34 more rows
```

Next, a hypothetical new matrix for the observed days and tune settings.

```
load("full-analysis-files/df_mv_as.rda")

new_matrix_dat <- expand_grid(matrix = factor("new_matrix"),
                              day_expt = factor(levels(df_mv_as$day_expt)),
                              tune = factor(c("LHe", "HHe"))) %>%
  mutate(tune = relevel(tune, ref = "LHe"))

save(new_matrix_dat, file = "full-analysis-files/new_matrix_dat.rda")
```

```
## # A tibble: 16 x 3
##    matrix     day_expt tune
##    <fct>      <fct>    <fct>
##  1 new_matrix 316      LHe
##  2 new_matrix 316      HHe
##  3 new_matrix 320      LHe
##  4 new_matrix 320      HHe
##  5 new_matrix 321      LHe
##  6 new_matrix 321      HHe
##  7 new_matrix 330      LHe
##  8 new_matrix 330      HHe
##  9 new_matrix 510      LHe
## 10 new_matrix 510      HHe
## 11 new_matrix 511      LHe
## 12 new_matrix 511      HHe
## 13 new_matrix 515      LHe
## 14 new_matrix 515      HHe
## 15 new_matrix 516      LHe
## 16 new_matrix 516      HHe
```

Finally, a new day and matrix for the observed tune settings.

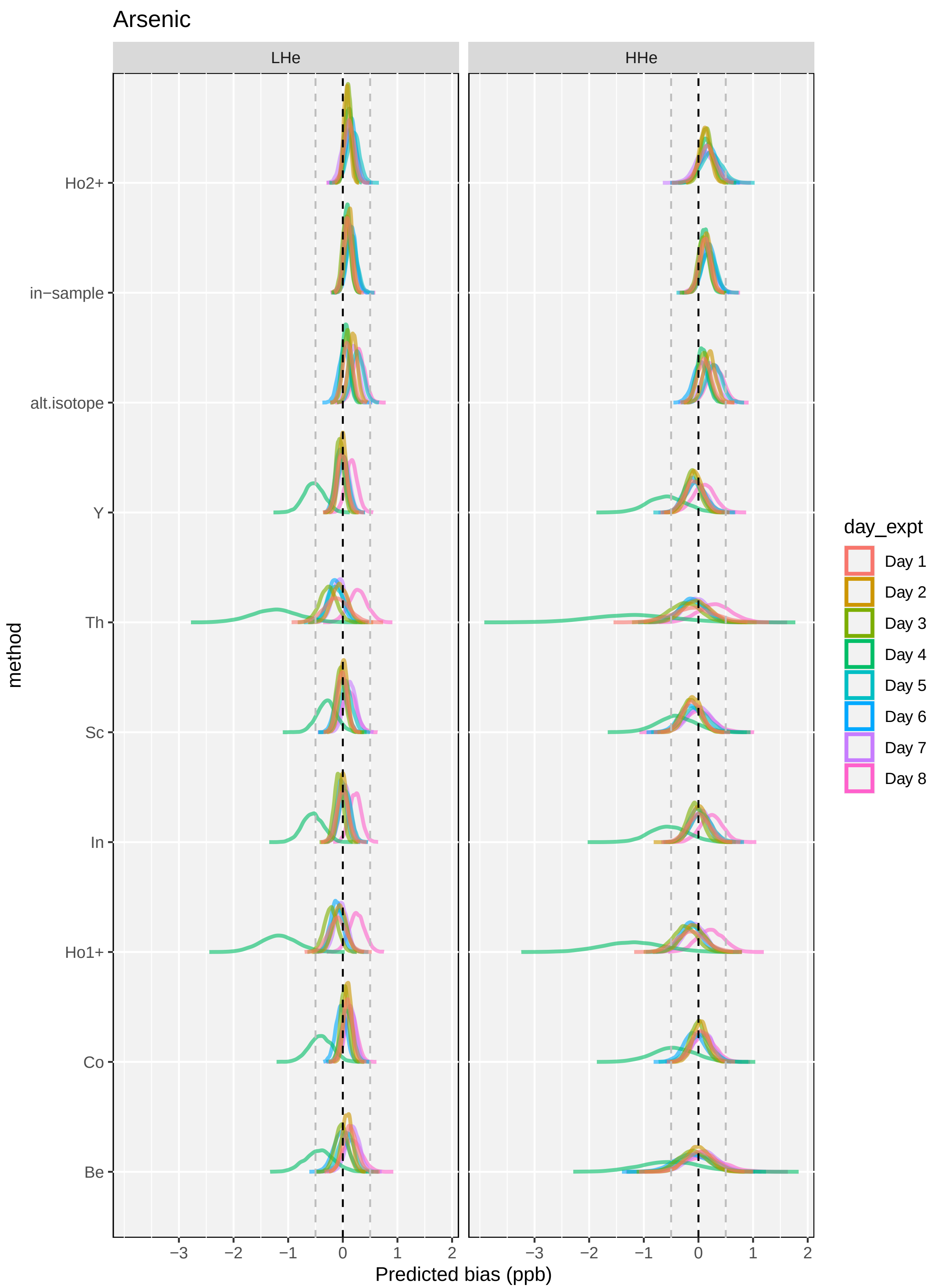
```
load("full-analysis-files/df_mv_as.rda")
# Include replicate observations for estimating uncertainty in probability to over-correct
new_matrix_day_dat <- expand_grid(matrix = factor("new_matrix"),
                                  day_expt = factor("new_day"),
                                  tune = factor( c("LHe", "HHe"))) %>%
  mutate(tune = relevel(tune, ref = "LHe"))

save(new_matrix_day_dat, file = "full-analysis-files/new_matrix_day_dat.rda")
```

```
## # A tibble: 2 x 3
```

```
##   matrix     day_expt tune
##   <fct>      <fct>    <fct>
## 1 new_matrix new_day  LHe
## 2 new_matrix new_day  HHe
```

Day The predictions for the observed days for a new matrix are below.

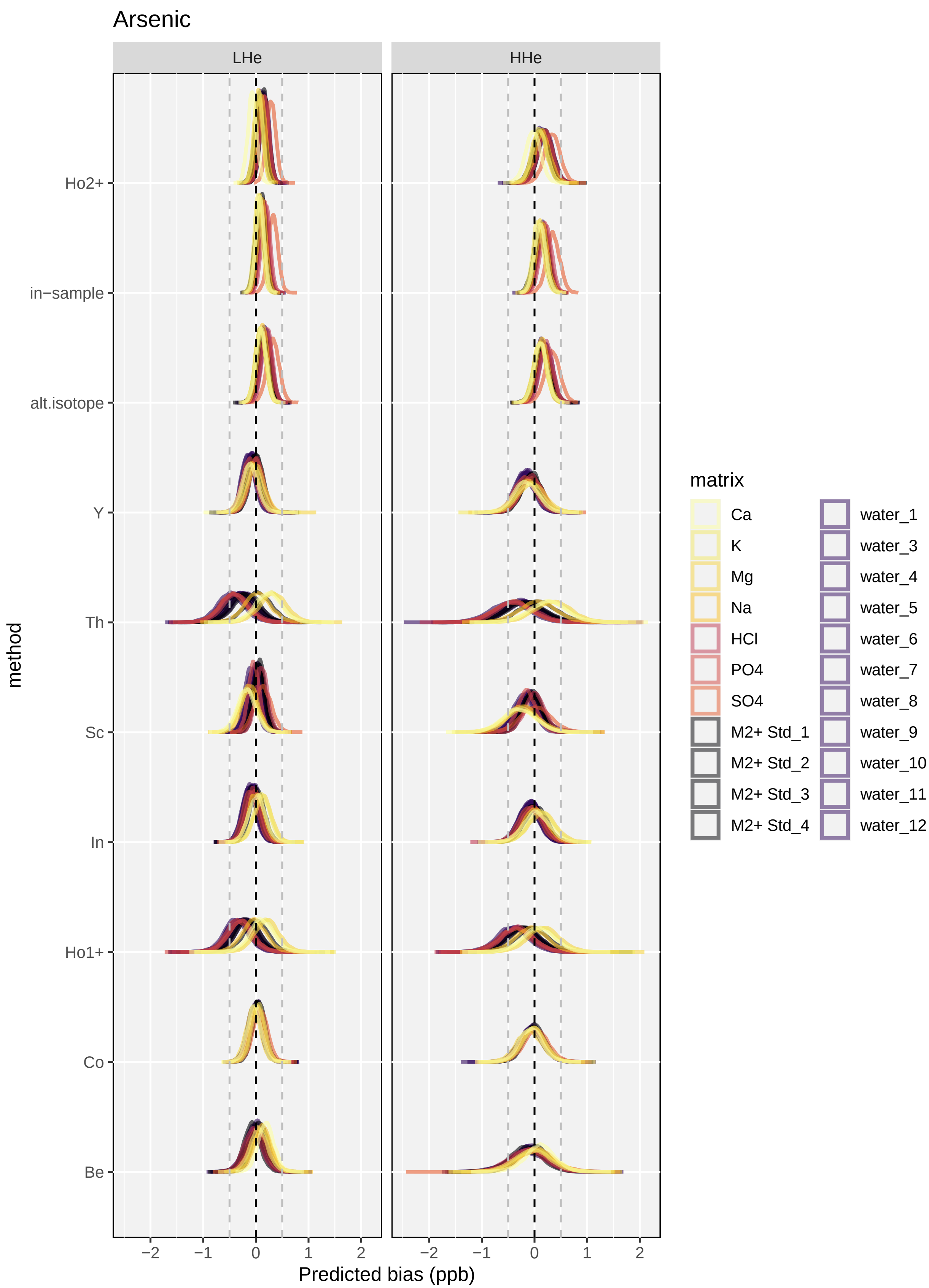


Arsenic
LHe
HHe
Ho2+
in−sample
alt.isotope
Y
Th
Sc
In
Ho1+
Co
Be
method
−3
−2
−1
0
1
2
Predicted bias (ppb)
day_expt
Day 1
Day 2
Day 3
Day 4
Day 5
Day 6
Day 7
Day 8

As expected from examining the conditional means for μ, the model predicts a high probability of relatively extreme over-corrections on 3/30 for all of the +1 methods. None of the +2 method's predictions included much probability in that region. The predictions for the +1 methods also suggest much more day to day variability in general.

As suggested in the conditional means for σ, the standard deviation of the +1 and +2 methods are generally predicted to be lower for the LHe tune, though the difference is more extreme for the +1 methods. The standard deviation was generally lower on 3/30 for the +2 methods as well, compared to other days within method.

Matrix The predictions to the observed matrices for a new day is below.



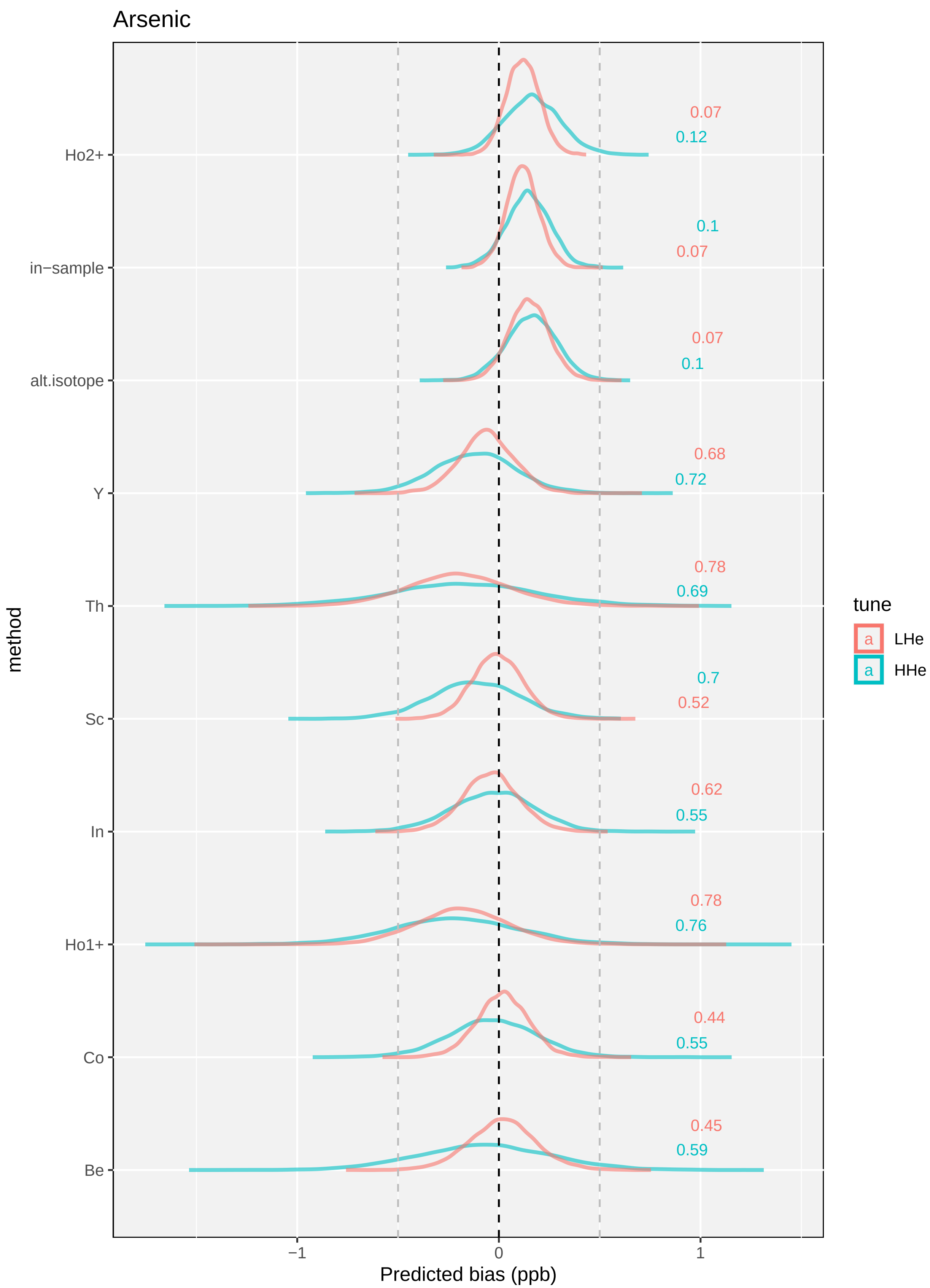
Arsenic
LHe
HHe
Ho2+
in−sample
alt.isotope
Y
Th
Sc
In
Ho1+
Co
Be
method
−2
−1
0
1
2
Predicted bias (ppb)
matrix
Ca
K
Mg
Na
HCl
PO4
SO4
M2+ Std_1
M2+ Std_2
M2+ Std_3
M2+ Std_4
water_1
water_3
water_4
water_5
water_6
water_7
water_8
water_9
water_10
water_11
water_12

Matrix to matrix variability is predicted to be lower than day to day, in general across all methods. With regard to any particular matrix, the predictions for some of the 250ppm matrices consistently suggest positive bias for Th, Ho^{+1}, and In relative to the other matrices in those methods. For the +2 methods, the 250ppm SO_4 matrix is consistently predicted with somewhat higher probability to under-correct relative to other matrices. Overall differences in standard deviation by method are again clear in this figure, with Th and Ho^{+1} being the least precise predictions. Within method, differences due to tune are again apparent, with the HHe tune generally predicted to result in more variable bias.

New day and matrix The predictions for a new day and new matrix for the observed tune settings are below.

The proportion of posterior prediction samples with over-corrections for each case.

```
## # A tibble: 20 x 4
## # Groups:   .category [10]
##    .category method     tune  p_over
##    <fct>     <fct>      <fct>  <dbl>
##  1 Std       in-sample  LHe     0.07
##  2 Std       in-sample  HHe     0.1
##  3 Alt       alt.isotope LHe    0.07
##  4 Alt       alt.isotope HHe    0.1
##  5 Ho2       Ho2+       LHe     0.07
##  6 Ho2       Ho2+       HHe     0.12
##  7 In        In         LHe     0.62
##  8 In        In         HHe     0.55
##  9 Sc        Sc         LHe     0.52
## 10 Sc        Sc         HHe     0.7
## 11 Y         Y          LHe     0.68
## 12 Y         Y          HHe     0.72
## 13 Be        Be         LHe     0.45
## 14 Be        Be         HHe     0.59
## 15 Co        Co         LHe     0.44
## 16 Co        Co         HHe     0.55
## 17 Th        Th         LHe     0.78
## 18 Th        Th         HHe     0.69
## 19 Ho1       Ho1+       LHe     0.78
## 20 Ho1       Ho1+       HHe     0.76
```


Arsenic
Ho2+
in−sample
alt.isotope
Y
Th
Sc
In
Ho1+
Co
Be
method
0.07
0.12
0.1
0.07
0.07
0.1
0.68
0.72
0.78
0.69
0.7
0.52
0.62
0.55
0.78
0.76
0.44
0.55
0.45
0.59
−1
0
1
Predicted bias (ppb)
tune
LHe
HHe

In this last set of predictions above, the most clear takeaways are (1) the variation in precision of the predictions across the different internal standard methods and tunes; and (2) the variations in the proportion of over- vs. under-corrected draws from the posterior predictive distribution. The predictions for the +2 methods are clearly more precise and their tendency to under-correct, on average, is also clear. The predictions for the +1 methods, by comparison, are much less precise and the bias is typically either minimal to more in the direction of over-correction. Overall, the HHe tune setting is consistently predicted with less precision across all of the methods.

A final model for selenium

The final model for arsenic is also applied to the selenium observations in the following analyses. In this model, the priors on fixed interceps and the standard deviations for the varying intercepts were adjusted upward by a factor of 10 to account for the difference in scale.

```
load("full-analysis-files/df_mv_se.rda")

bf_Std <- bf(Std ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Alt <- bf(Alt ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Ho2 <- bf(Ho2 ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_In <- bf(In ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Sc <- bf(Sc ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Y <- bf(Y ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Be <- bf(Be ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Co <- bf(Co ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

bf_Th <- bf(Th ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())
```

```r
bf_Ho1 <- bf(Ho1 ~ tune + (tune | matrix) + (tune | day_expt),
             sigma ~ tune + (1 | matrix) + (1 | day_expt),
             family = gaussian())

mod3 <- brm(bf_Std +
              bf_Alt +
              bf_Ho2 +
              bf_In +
              bf_Sc +
              bf_Y +
              bf_Be +
              bf_Co +
              bf_Th +
              bf_Ho1 +
              set_rescor(TRUE),
  data = df_mv_se,
  prior = c(prior(normal(0, 10), class = "Intercept", resp = "Std"),
            prior(normal(0, 10), class = "Intercept", resp = "Alt"),
            prior(normal(0, 10), class = "Intercept", resp = "Ho2"),
            prior(normal(0, 10), class = "Intercept", resp = "In"),
            prior(normal(0, 10), class = "Intercept", resp = "Sc"),
            prior(normal(0, 10), class = "Intercept", resp = "Y"),
            prior(normal(0, 10), class = "Intercept", resp = "Be"),
            prior(normal(0, 10), class = "Intercept", resp = "Co"),
            prior(normal(0, 10), class = "Intercept", resp = "Th"),
            prior(normal(0, 10), class = "Intercept", resp = "Ho1"),

            prior(normal(0, 10), class = "b", resp = "Std"),
            prior(normal(0, 10), class = "b", resp = "Alt"),
            prior(normal(0, 10), class = "b", resp = "Ho2"),
            prior(normal(0, 10), class = "b", resp = "In"),
            prior(normal(0, 10), class = "b", resp = "Sc"),
            prior(normal(0, 10), class = "b", resp = "Y"),
            prior(normal(0, 10), class = "b", resp = "Be"),
            prior(normal(0, 10), class = "b", resp = "Co"),
            prior(normal(0, 10), class = "b", resp = "Th"),
            prior(normal(0, 10), class = "b", resp = "Ho1"),

            prior(normal(0, 10), class = "sd", resp = "Std"),
            prior(normal(0, 10), class = "sd", resp = "Alt"),
            prior(normal(0, 10), class = "sd", resp = "Ho2"),
            prior(normal(0, 10), class = "sd", resp = "In"),
            prior(normal(0, 10), class = "sd", resp = "Sc"),
            prior(normal(0, 10), class = "sd", resp = "Y"),
            prior(normal(0, 10), class = "sd", resp = "Be"),
            prior(normal(0, 10), class = "sd", resp = "Co"),
            prior(normal(0, 10), class = "sd", resp = "Th"),
            prior(normal(0, 10), class = "sd", resp = "Ho1"),

            prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Std"),
            prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Alt"),
            prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Ho2"),
            prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "In"),
```

```r
          prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Sc"),
          prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Y"),
          prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Be"),
          prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Co"),
          prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Th"),
          prior(normal(2, 1), class = "Intercept", dpar = "sigma", resp = "Ho1"),

          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Std"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Alt"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Ho2"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "In"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Sc"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Y"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Be"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Co"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Th"),
          prior(normal(0, 10), class = "b", dpar = "sigma", resp = "Ho1"),

          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Std"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Alt"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Ho2"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "In"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Sc"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Y"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Be"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Co"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Th"),
          prior(normal(0, 10), class = "sd", dpar = "sigma", resp = "Ho1"),

          prior(lkj(1), class = "rescor")
          ),
  control = list(adapt_delta = 0.95, max_treedepth = 14),
  init_r = 0.05,
  save_pars = save_pars(all = TRUE),
  seed = 5214,
  chains=4,
  iter=3000,
  cores=4 )

save(mod3, file = "full-analysis-files/mod3_Se_mv.rda")
```

Tabular parameter estimates

Again, a summary of the posterior estimates.

```
##  Family: MV(gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian, gaussian
##   Links: mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
```

```
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
##          mu = identity; sigma = log
## Formula: Std ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Alt ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho2 ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          In ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Sc ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Y ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Be ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Co ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Th ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##          Ho1 ~ tune + (tune | matrix) + (tune | day_expt)
##          sigma ~ tune + (1 | matrix) + (1 | day_expt)
##    Data: df_mv_se (Number of observations: 352)
##   Draws: 4 chains, each with iter = 3000; warmup = 1500; thin = 1;
##          total post-warmup draws = 6000
##
## Priors:
## b_Alt ~ normal(0, 10)
## b_Alt_sigma ~ normal(0, 10)
## b_Be ~ normal(0, 10)
## b_Be_sigma ~ normal(0, 10)
## b_Co ~ normal(0, 10)
## b_Co_sigma ~ normal(0, 10)
## b_Ho1 ~ normal(0, 10)
## b_Ho1_sigma ~ normal(0, 10)
## b_Ho2 ~ normal(0, 10)
## b_Ho2_sigma ~ normal(0, 10)
## b_In ~ normal(0, 10)
## b_In_sigma ~ normal(0, 10)
## b_Sc ~ normal(0, 10)
## b_Sc_sigma ~ normal(0, 10)
## b_Std ~ normal(0, 10)
## b_Std_sigma ~ normal(0, 10)
## b_Th ~ normal(0, 10)
## b_Th_sigma ~ normal(0, 10)
## b_Y ~ normal(0, 10)
## b_Y_sigma ~ normal(0, 10)
## Intercept_Alt ~ normal(0, 10)
## Intercept_Alt_sigma ~ normal(2, 1)
## Intercept_Be ~ normal(0, 10)
## Intercept_Be_sigma ~ normal(2, 1)
## Intercept_Co ~ normal(0, 10)
```

```
## Intercept_Co_sigma ~ normal(2, 1)
## Intercept_Ho1 ~ normal(0, 10)
## Intercept_Ho1_sigma ~ normal(2, 1)
## Intercept_Ho2 ~ normal(0, 10)
## Intercept_Ho2_sigma ~ normal(2, 1)
## Intercept_In ~ normal(0, 10)
## Intercept_In_sigma ~ normal(2, 1)
## Intercept_Sc ~ normal(0, 10)
## Intercept_Sc_sigma ~ normal(2, 1)
## Intercept_Std ~ normal(0, 10)
## Intercept_Std_sigma ~ normal(2, 1)
## Intercept_Th ~ normal(0, 10)
## Intercept_Th_sigma ~ normal(2, 1)
## Intercept_Y ~ normal(0, 10)
## Intercept_Y_sigma ~ normal(2, 1)
## L ~ lkj_corr_cholesky(1)
## Lrescor ~ lkj_corr_cholesky(1)
## sd_Alt ~ normal(0, 10)
## sd_Alt_sigma ~ normal(0, 10)
## sd_Be ~ normal(0, 10)
## sd_Be_sigma ~ normal(0, 10)
## sd_Co ~ normal(0, 10)
## sd_Co_sigma ~ normal(0, 10)
## sd_Ho1 ~ normal(0, 10)
## sd_Ho1_sigma ~ normal(0, 10)
## sd_Ho2 ~ normal(0, 10)
## sd_Ho2_sigma ~ normal(0, 10)
## sd_In ~ normal(0, 10)
## sd_In_sigma ~ normal(0, 10)
## sd_Sc ~ normal(0, 10)
## sd_Sc_sigma ~ normal(0, 10)
## sd_Std ~ normal(0, 10)
## sd_Std_sigma ~ normal(0, 10)
## sd_Th ~ normal(0, 10)
## sd_Th_sigma ~ normal(0, 10)
## sd_Y ~ normal(0, 10)
## sd_Y_sigma ~ normal(0, 10)
##
## Group-Level Effects:
## ~day_expt (Number of levels: 8)
##                               Estimate Est.Error l-95% CI u-95% CI Rhat
## sd(Std_Intercept)                 0.20      0.15     0.01     0.56 1.00
## sd(Std_tuneHHe)                   0.73      0.43     0.09     1.78 1.00
## sd(sigma_Std_Intercept)           0.26      0.11     0.12     0.55 1.00
## sd(Alt_Intercept)                 1.99      0.76     1.07     3.98 1.00
## sd(Alt_tuneHHe)                   3.06      1.11     1.64     5.94 1.00
## sd(sigma_Alt_Intercept)           0.29      0.12     0.13     0.59 1.00
## sd(Ho2_Intercept)                 1.50      0.57     0.80     2.98 1.00
## sd(Ho2_tuneHHe)                   3.08      1.13     1.64     5.86 1.00
## sd(sigma_Ho2_Intercept)           0.41      0.16     0.21     0.80 1.00
## sd(In_Intercept)                  3.59      1.22     2.00     6.76 1.00
## sd(In_tuneHHe)                    2.14      0.83     0.96     4.18 1.00
## sd(sigma_In_Intercept)            0.24      0.09     0.13     0.48 1.00
## sd(Sc_Intercept)                  2.56      0.88     1.43     4.83 1.00
```

```
## sd(Sc_tuneHHe)                    4.03      1.33     2.21     7.45 1.00
## sd(sigma_Sc_Intercept)            0.31      0.13     0.15     0.62 1.00
## sd(Y_Intercept)                   3.37      1.10     1.93     6.08 1.00
## sd(Y_tuneHHe)                     3.19      1.13     1.71     5.95 1.00
## sd(sigma_Y_Intercept)             0.26      0.10     0.14     0.52 1.00
## sd(Be_Intercept)                  3.41      1.15     1.87     6.33 1.00
## sd(Be_tuneHHe)                    5.49      1.83     3.04    10.20 1.00
## sd(sigma_Be_Intercept)            0.40      0.14     0.22     0.74 1.00
## sd(Co_Intercept)                  3.11      0.99     1.75     5.64 1.00
## sd(Co_tuneHHe)                    4.06      1.35     2.24     7.41 1.00
## sd(sigma_Co_Intercept)            0.38      0.14     0.21     0.75 1.00
## sd(Th_Intercept)                  6.61      2.14     3.61    11.83 1.00
## sd(Th_tuneHHe)                    2.20      1.36     0.17     5.41 1.00
## sd(sigma_Th_Intercept)            0.34      0.13     0.18     0.65 1.00
## sd(Ho1_Intercept)                 6.12      1.95     3.40    11.04 1.00
## sd(Ho1_tuneHHe)                   2.73      1.28     0.92     5.89 1.00
## sd(sigma_Ho1_Intercept)           0.31      0.11     0.17     0.60 1.00
## cor(Std_Intercept,Std_tuneHHe)   -0.14      0.54    -0.95     0.90 1.00
## cor(Alt_Intercept,Alt_tuneHHe)   -0.03      0.36    -0.69     0.65 1.00
## cor(Ho2_Intercept,Ho2_tuneHHe)    0.05      0.35    -0.61     0.68 1.00
## cor(In_Intercept,In_tuneHHe)      0.55      0.31    -0.21     0.95 1.00
## cor(Sc_Intercept,Sc_tuneHHe)      0.64      0.24     0.02     0.94 1.00
## cor(Y_Intercept,Y_tuneHHe)        0.61      0.26    -0.05     0.94 1.00
## cor(Be_Intercept,Be_tuneHHe)      0.54      0.29    -0.16     0.92 1.00
## cor(Co_Intercept,Co_tuneHHe)      0.81      0.17     0.32     0.98 1.00
## cor(Th_Intercept,Th_tuneHHe)      0.34      0.51    -0.75     0.98 1.00
## cor(Ho1_Intercept,Ho1_tuneHHe)    0.60      0.33    -0.24     0.97 1.00
##                                Bulk_ESS Tail_ESS
## sd(Std_Intercept)                  2023     2491
## sd(Std_tuneHHe)                    1895     2130
## sd(sigma_Std_Intercept)            2161     2976
## sd(Alt_Intercept)                  2048     2663
## sd(Alt_tuneHHe)                    2586     3202
## sd(sigma_Alt_Intercept)            1964     3133
## sd(Ho2_Intercept)                  1975     2886
## sd(Ho2_tuneHHe)                    2561     3374
## sd(sigma_Ho2_Intercept)            1736     2943
## sd(In_Intercept)                   3047     3031
## sd(In_tuneHHe)                     2311     3087
## sd(sigma_In_Intercept)             1872     2293
## sd(Sc_Intercept)                   2428     3474
## sd(Sc_tuneHHe)                     2705     3566
## sd(sigma_Sc_Intercept)             1788     3289
## sd(Y_Intercept)                    2632     3656
## sd(Y_tuneHHe)                      2602     2807
## sd(sigma_Y_Intercept)              2572     3179
## sd(Be_Intercept)                   2539     2813
## sd(Be_tuneHHe)                     2913     3291
## sd(sigma_Be_Intercept)             2361     3635
## sd(Co_Intercept)                   2150     2914
## sd(Co_tuneHHe)                     2139     2661
## sd(sigma_Co_Intercept)             2448     2828
## sd(Th_Intercept)                   3006     4080
## sd(Th_tuneHHe)                     1377     1433
```

```
## sd(sigma_Th_Intercept)            2676     2993
## sd(Ho1_Intercept)                 3029     3498
## sd(Ho1_tuneHHe)                   1798     1566
## sd(sigma_Ho1_Intercept)           2821     3178
## cor(Std_Intercept,Std_tuneHHe)    1616     2705
## cor(Alt_Intercept,Alt_tuneHHe)    2285     2654
## cor(Ho2_Intercept,Ho2_tuneHHe)    2641     3307
## cor(In_Intercept,In_tuneHHe)      2179     2552
## cor(Sc_Intercept,Sc_tuneHHe)      2476     3196
## cor(Y_Intercept,Y_tuneHHe)        2170     2806
## cor(Be_Intercept,Be_tuneHHe)      2634     2910
## cor(Co_Intercept,Co_tuneHHe)      2504     3642
## cor(Th_Intercept,Th_tuneHHe)      2310     3105
## cor(Ho1_Intercept,Ho1_tuneHHe)    2140     2577
##
## ~matrix (Number of levels: 22)
##                                 Estimate Est.Error l-95% CI u-95% CI Rhat
## sd(Std_Intercept)                   0.09      0.07     0.00     0.26 1.00
## sd(Std_tuneHHe)                     0.29      0.20     0.01     0.75 1.00
## sd(sigma_Std_Intercept)             0.05      0.04     0.00     0.13 1.00
## sd(Alt_Intercept)                   0.17      0.10     0.01     0.39 1.00
## sd(Alt_tuneHHe)                     0.33      0.21     0.01     0.80 1.00
## sd(sigma_Alt_Intercept)             0.09      0.06     0.00     0.22 1.00
## sd(Ho2_Intercept)                   0.44      0.10     0.25     0.68 1.00
## sd(Ho2_tuneHHe)                     0.82      0.25     0.32     1.33 1.00
## sd(sigma_Ho2_Intercept)             0.04      0.03     0.00     0.12 1.00
## sd(In_Intercept)                    0.08      0.05     0.00     0.19 1.00
## sd(In_tuneHHe)                      0.32      0.14     0.04     0.62 1.01
## sd(sigma_In_Intercept)              0.03      0.02     0.00     0.06 1.00
## sd(Sc_Intercept)                    0.76      0.14     0.53     1.07 1.00
## sd(Sc_tuneHHe)                      1.63      0.36     1.05     2.45 1.00
## sd(sigma_Sc_Intercept)              0.14      0.04     0.07     0.22 1.00
## sd(Y_Intercept)                     0.48      0.09     0.34     0.69 1.00
## sd(Y_tuneHHe)                       1.00      0.21     0.67     1.49 1.00
## sd(sigma_Y_Intercept)               0.04      0.02     0.00     0.08 1.00
## sd(Be_Intercept)                    0.88      0.18     0.60     1.29 1.00
## sd(Be_tuneHHe)                      1.83      0.42     1.12     2.77 1.00
## sd(sigma_Be_Intercept)              0.11      0.04     0.03     0.19 1.00
## sd(Co_Intercept)                    0.50      0.10     0.34     0.73 1.00
## sd(Co_tuneHHe)                      0.67      0.16     0.40     1.04 1.00
## sd(sigma_Co_Intercept)              0.02      0.02     0.00     0.06 1.00
## sd(Th_Intercept)                    0.22      0.07     0.09     0.38 1.00
## sd(Th_tuneHHe)                      0.92      0.23     0.54     1.46 1.00
## sd(sigma_Th_Intercept)              0.01      0.01     0.00     0.04 1.00
## sd(Ho1_Intercept)                   0.05      0.04     0.00     0.15 1.00
## sd(Ho1_tuneHHe)                     0.20      0.16     0.01     0.58 1.00
## sd(sigma_Ho1_Intercept)             0.01      0.01     0.00     0.03 1.00
## cor(Std_Intercept,Std_tuneHHe)     -0.05      0.57    -0.96     0.94 1.00
## cor(Alt_Intercept,Alt_tuneHHe)     -0.05      0.55    -0.95     0.93 1.00
## cor(Ho2_Intercept,Ho2_tuneHHe)     -0.49      0.25    -0.90     0.09 1.00
## cor(In_Intercept,In_tuneHHe)        0.38      0.50    -0.78     0.98 1.01
## cor(Sc_Intercept,Sc_tuneHHe)        0.79      0.13     0.47     0.97 1.00
## cor(Y_Intercept,Y_tuneHHe)          0.98      0.03     0.91     1.00 1.00
## cor(Be_Intercept,Be_tuneHHe)        0.86      0.12     0.55     0.99 1.00
```

```
## cor(Co_Intercept,Co_tuneHHe)       0.91      0.09     0.66     1.00 1.00
## cor(Th_Intercept,Th_tuneHHe)       0.68      0.24     0.09     0.98 1.00
## cor(Ho1_Intercept,Ho1_tuneHHe)     0.11      0.58    -0.94     0.97 1.00
##                                Bulk_ESS Tail_ESS
## sd(Std_Intercept)                  3309     2993
## sd(Std_tuneHHe)                    1919     2078
## sd(sigma_Std_Intercept)            3790     3349
## sd(Alt_Intercept)                  1623     2427
## sd(Alt_tuneHHe)                    2135     2694
## sd(sigma_Alt_Intercept)            1878     2328
## sd(Ho2_Intercept)                  1495     2113
## sd(Ho2_tuneHHe)                    1398     1712
## sd(sigma_Ho2_Intercept)            2848     2750
## sd(In_Intercept)                    933     2282
## sd(In_tuneHHe)                      965     1218
## sd(sigma_In_Intercept)              970     1189
## sd(Sc_Intercept)                   1544     2460
## sd(Sc_tuneHHe)                     1514     2851
## sd(sigma_Sc_Intercept)             2141     2887
## sd(Y_Intercept)                    1212     1979
## sd(Y_tuneHHe)                       944     2347
## sd(sigma_Y_Intercept)              1194     1901
## sd(Be_Intercept)                   2024     2490
## sd(Be_tuneHHe)                     2448     3547
## sd(sigma_Be_Intercept)             1395      826
## sd(Co_Intercept)                   1626     3535
## sd(Co_tuneHHe)                     1928     3857
## sd(sigma_Co_Intercept)             2209     3193
## sd(Th_Intercept)                   1763     2230
## sd(Th_tuneHHe)                     2108     3338
## sd(sigma_Th_Intercept)             1639     1765
## sd(Ho1_Intercept)                  2237     3005
## sd(Ho1_tuneHHe)                    1923     3223
## sd(sigma_Ho1_Intercept)            2963     3266
## cor(Std_Intercept,Std_tuneHHe)     3294     3908
## cor(Alt_Intercept,Alt_tuneHHe)     2995     3500
## cor(Ho2_Intercept,Ho2_tuneHHe)     2690     2734
## cor(In_Intercept,In_tuneHHe)        856     1623
## cor(Sc_Intercept,Sc_tuneHHe)       1362     2027
## cor(Y_Intercept,Y_tuneHHe)         2207     2840
## cor(Be_Intercept,Be_tuneHHe)       1969     2476
## cor(Co_Intercept,Co_tuneHHe)       2184     3588
## cor(Th_Intercept,Th_tuneHHe)       1070     2002
## cor(Ho1_Intercept,Ho1_tuneHHe)     2505     3963
##
## Population-Level Effects:
##                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Std_Intercept          -0.43      0.12    -0.68    -0.18 1.00     4631     4148
## sigma_Std_Intercept     0.04      0.11    -0.18     0.27 1.00     2280     2943
## Alt_Intercept          -0.84      0.74    -2.33     0.55 1.00     1278     2176
## sigma_Alt_Intercept    -0.00      0.12    -0.24     0.25 1.00     2179     2536
## Ho2_Intercept          -1.45      0.59    -2.64    -0.30 1.00     1405     2406
## sigma_Ho2_Intercept     0.10      0.17    -0.20     0.47 1.00     1721     1755
## In_Intercept           -3.53      1.28    -6.03    -0.93 1.00     1297     2100
```

```
## sigma_In_Intercept      0.87      0.10     0.67     1.08 1.00     1572     2649
## Sc_Intercept           -2.99      0.96    -4.83    -1.03 1.00     1271     1997
## sigma_Sc_Intercept      0.26      0.13     0.00     0.53 1.00     1680     2802
## Y_Intercept            -3.72      1.24    -6.13    -1.17 1.00     1108     1810
## sigma_Y_Intercept       0.55      0.11     0.35     0.77 1.00     1511     2666
## Be_Intercept           -2.70      1.25    -5.23    -0.13 1.00     1148     1915
## sigma_Be_Intercept      0.78      0.16     0.47     1.10 1.00     1558     2316
## Co_Intercept           -2.71      1.13    -4.93    -0.49 1.00      982     1556
## sigma_Co_Intercept      0.50      0.15     0.20     0.80 1.00     1361     2244
## Th_Intercept           -5.65      2.40   -10.44    -0.81 1.00     1175     2167
## sigma_Th_Intercept      1.61      0.13     1.35     1.88 1.00     1642     2409
## Ho1_Intercept          -5.19      2.22    -9.40    -0.61 1.01     1217     2025
## sigma_Ho1_Intercept     1.38      0.12     1.13     1.63 1.00     1242     2307
## Std_tuneHHe            -0.20      0.37    -0.95     0.55 1.00     2614     2555
## sigma_Std_tuneHHe       0.72      0.07     0.57     0.86 1.00     7538     4530
## Alt_tuneHHe             0.33      1.14    -1.95     2.66 1.00     1693     2236
## sigma_Alt_tuneHHe       0.75      0.08     0.59     0.91 1.00     7881     4342
## Ho2_tuneHHe             0.40      1.21    -1.98     2.85 1.00     1758     2286
## sigma_Ho2_tuneHHe       1.09      0.07     0.95     1.22 1.00     4108     4355
## In_tuneHHe             -1.36      0.91    -3.13     0.46 1.00     2036     2492
## sigma_In_tuneHHe        0.85      0.04     0.77     0.94 1.00     2085     3435
## Sc_tuneHHe             -3.87      1.56    -7.00    -0.75 1.00     1483     2442
## sigma_Sc_tuneHHe        1.04      0.05     0.93     1.14 1.00     3680     4271
## Y_tuneHHe              -2.93      1.23    -5.39    -0.40 1.00     1374     2370
## sigma_Y_tuneHHe         0.90      0.04     0.82     0.98 1.00     2302     3511
## Be_tuneHHe             -2.70      2.06    -6.80     1.27 1.00     1474     2113
## sigma_Be_tuneHHe        1.29      0.05     1.19     1.40 1.00     3878     3928
## Co_tuneHHe             -1.99      1.52    -4.95     1.06 1.00     1094     1785
## sigma_Co_tuneHHe        1.13      0.04     1.05     1.20 1.00     2706     3871
## Th_tuneHHe             -1.35      1.34    -3.97     1.27 1.00     3117     3413
## sigma_Th_tuneHHe        0.96      0.04     0.88     1.05 1.00     2097     3126
## Ho1_tuneHHe            -2.39      1.31    -4.99     0.18 1.00     1912     2479
## sigma_Ho1_tuneHHe       0.97      0.04     0.89     1.06 1.00     2112     3310
##
## Residual Correlations:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## rescor(Std,Alt)     0.33      0.05     0.24     0.43 1.00     7984     5239
## rescor(Std,Ho2)     0.33      0.05     0.24     0.43 1.00     1765     3392
## rescor(Alt,Ho2)     0.33      0.05     0.23     0.43 1.00     2126     3028
## rescor(Std,In)      0.13      0.05     0.03     0.24 1.00     1308     2325
## rescor(Alt,In)      0.14      0.06     0.03     0.26 1.00     1445     1731
## rescor(Ho2,In)      0.70      0.04     0.62     0.77 1.00     1001     2253
## rescor(Std,Sc)      0.33      0.05     0.23     0.42 1.00     4229     4413
## rescor(Alt,Sc)      0.32      0.05     0.22     0.42 1.00     4442     3852
## rescor(Ho2,Sc)      0.55      0.05     0.44     0.64 1.00     2424     4012
## rescor(In,Sc)       0.34      0.07     0.21     0.48 1.00     1916     2593
## rescor(Std,Y)       0.21      0.05     0.11     0.31 1.00     1299     2000
## rescor(Alt,Y)       0.22      0.06     0.11     0.33 1.00     1468     1755
## rescor(Ho2,Y)       0.77      0.03     0.70     0.82 1.00      984     2149
## rescor(In,Y)        0.97      0.01     0.96     0.98 1.00     2194     3996
## rescor(Sc,Y)        0.53      0.06     0.42     0.64 1.00     2396     2850
## rescor(Std,Be)      0.20      0.05     0.09     0.30 1.00     1426     2485
## rescor(Alt,Be)      0.15      0.06     0.04     0.26 1.00     1688     2093
## rescor(Ho2,Be)      0.73      0.03     0.66     0.79 1.00     1391     2515
```

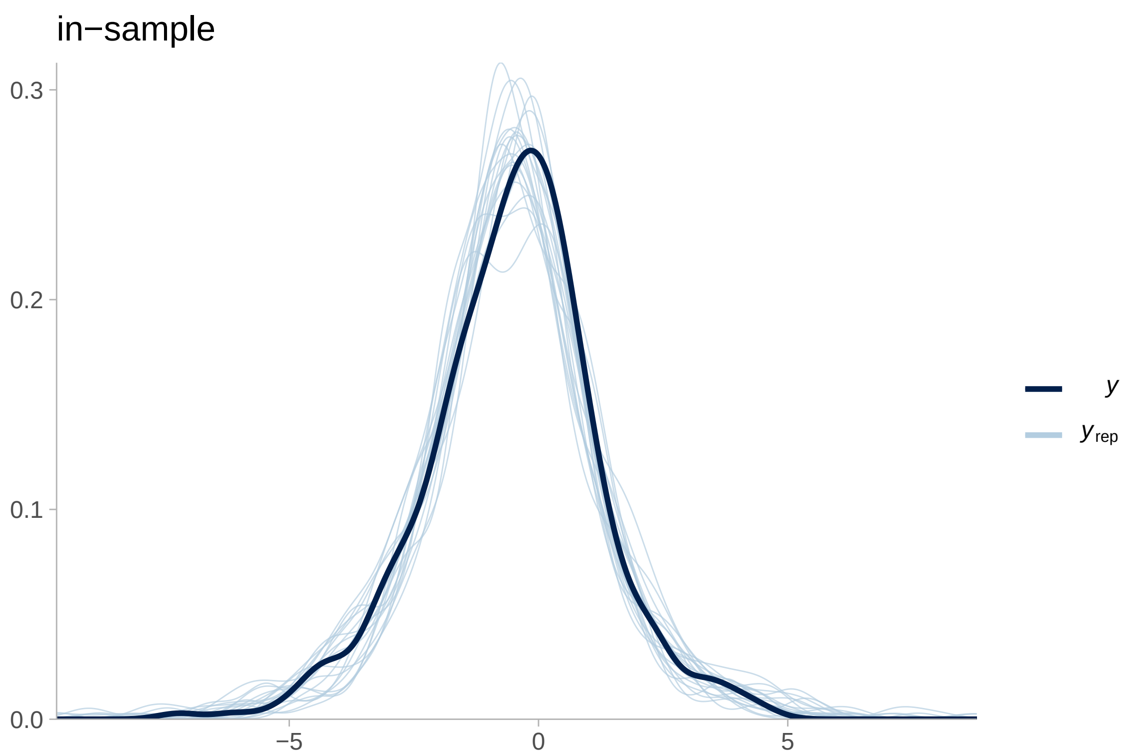
```
## rescor(In,Be)        0.78      0.03     0.72     0.83 1.00     2817     4456
## rescor(Sc,Be)        0.56      0.06     0.45     0.66 1.00     2119     3098
## rescor(Y,Be)         0.82      0.02     0.78     0.86 1.00     2971     4579
## rescor(Std,Co)       0.24      0.05     0.14     0.34 1.00     1301     2211
## rescor(Alt,Co)       0.21      0.06     0.10     0.32 1.00     1476     1741
## rescor(Ho2,Co)       0.82      0.03     0.76     0.86 1.00     1064     2150
## rescor(In,Co)        0.90      0.01     0.88     0.93 1.00     2640     4016
## rescor(Sc,Co)        0.60      0.05     0.49     0.70 1.00     2272     3110
## rescor(Y,Co)         0.95      0.01     0.93     0.96 1.00     2928     4582
## rescor(Be,Co)        0.93      0.01     0.91     0.95 1.00     3523     4767
## rescor(Std,Th)       0.04      0.05    -0.07     0.14 1.00     1384     2181
## rescor(Alt,Th)       0.04      0.06    -0.07     0.17 1.00     1490     1849
## rescor(Ho2,Th)       0.60      0.05     0.51     0.69 1.00     1097     2326
## rescor(In,Th)        0.96      0.00     0.95     0.97 1.00     3448     4487
## rescor(Sc,Th)        0.14      0.08    -0.01     0.29 1.00     1708     2289
## rescor(Y,Th)         0.88      0.02     0.85     0.91 1.00     2102     3407
## rescor(Be,Th)        0.74      0.03     0.67     0.80 1.00     2779     4175
## rescor(Co,Th)        0.83      0.02     0.79     0.87 1.00     2907     3799
## rescor(Std,Ho1)      0.06      0.05    -0.05     0.16 1.00     1397     2271
## rescor(Alt,Ho1)      0.06      0.06    -0.06     0.18 1.00     1508     1740
## rescor(Ho2,Ho1)      0.60      0.05     0.50     0.68 1.00     1124     2342
## rescor(In,Ho1)       0.96      0.00     0.95     0.97 1.00     3376     4708
## rescor(Sc,Ho1)       0.12      0.08    -0.03     0.27 1.00     1722     2412
## rescor(Y,Ho1)        0.88      0.02     0.85     0.91 1.00     2134     3432
## rescor(Be,Ho1)       0.73      0.03     0.65     0.79 1.00     2776     4333
## rescor(Co,Ho1)       0.82      0.02     0.78     0.86 1.00     2966     3909
## rescor(Th,Ho1)       1.00      0.00     1.00     1.00 1.00     6443     5604
##
## 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).
```

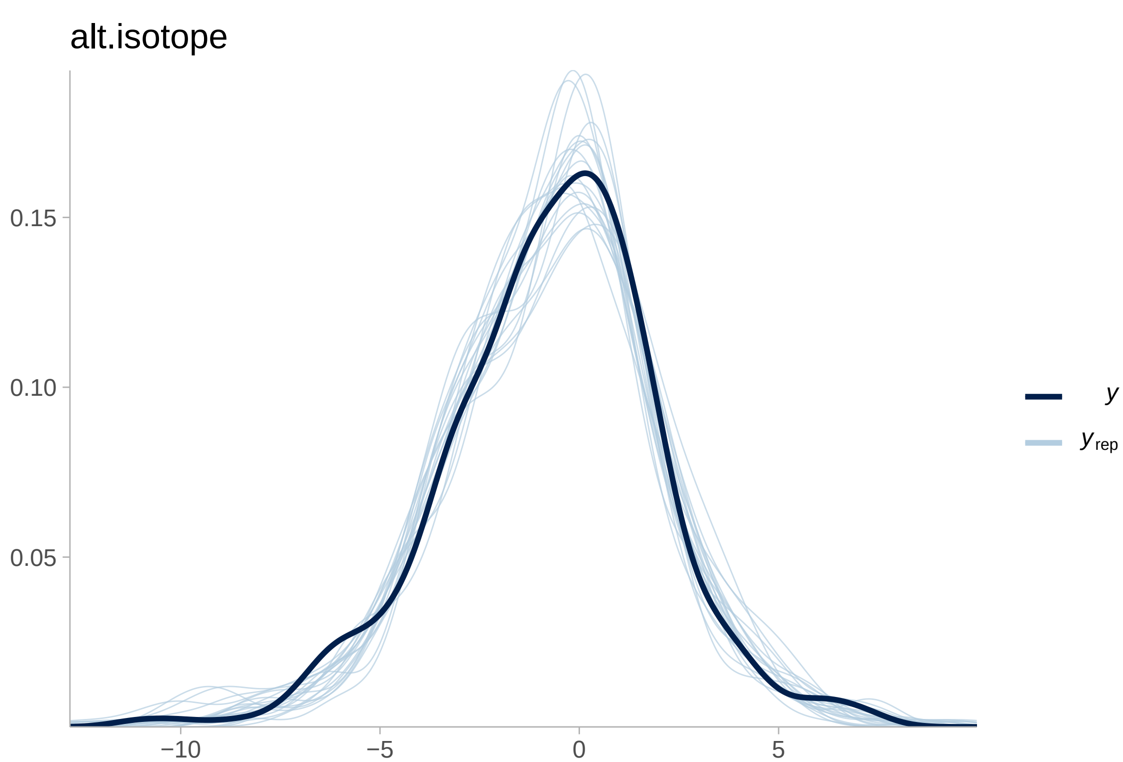
Again, the HMC sampling looks to have gone well.

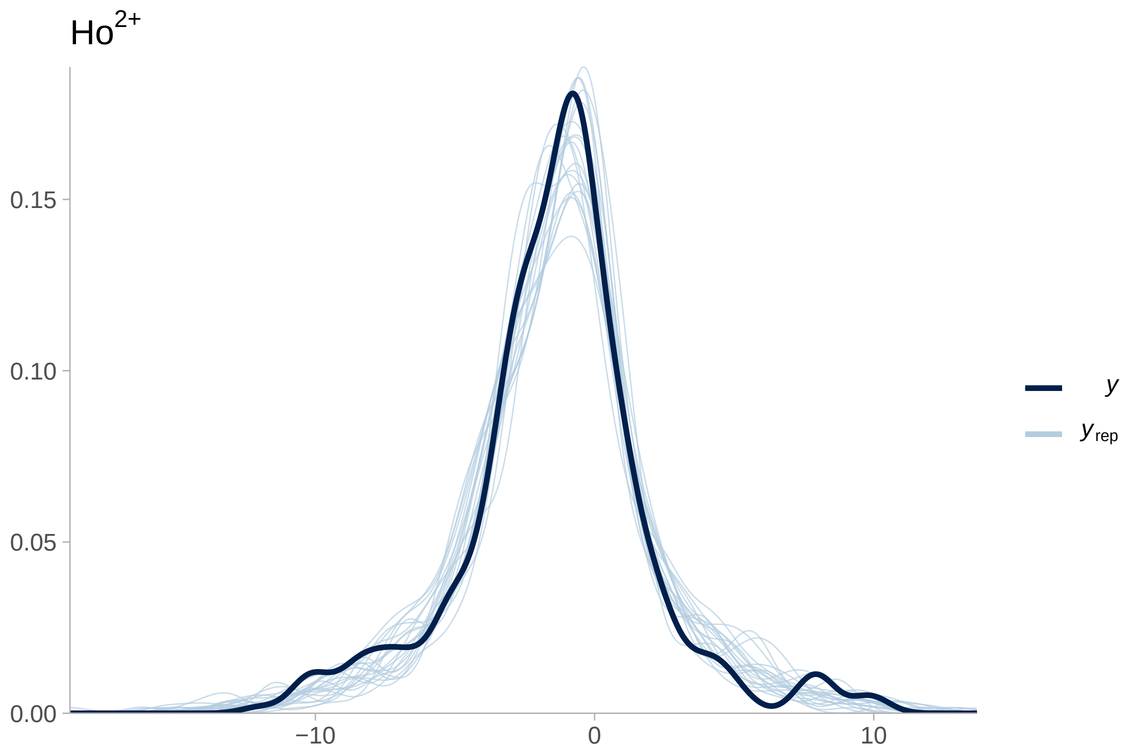
Model checks

Next, the checks for the selenium model.

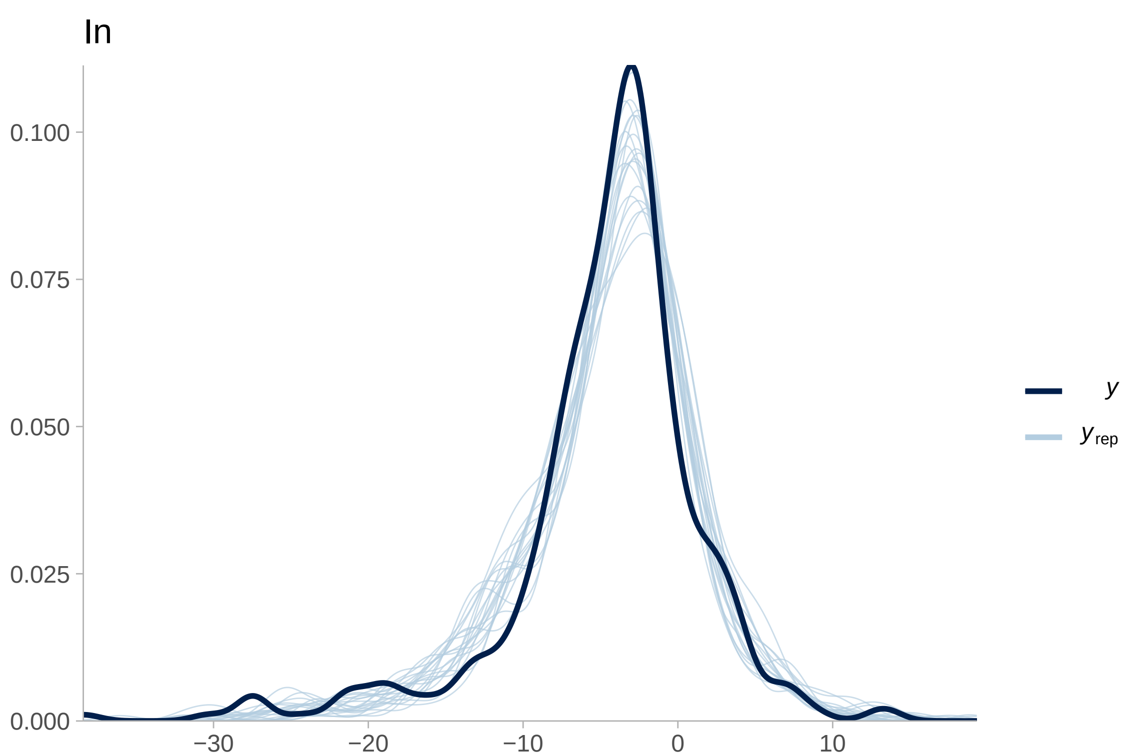
Density overlay



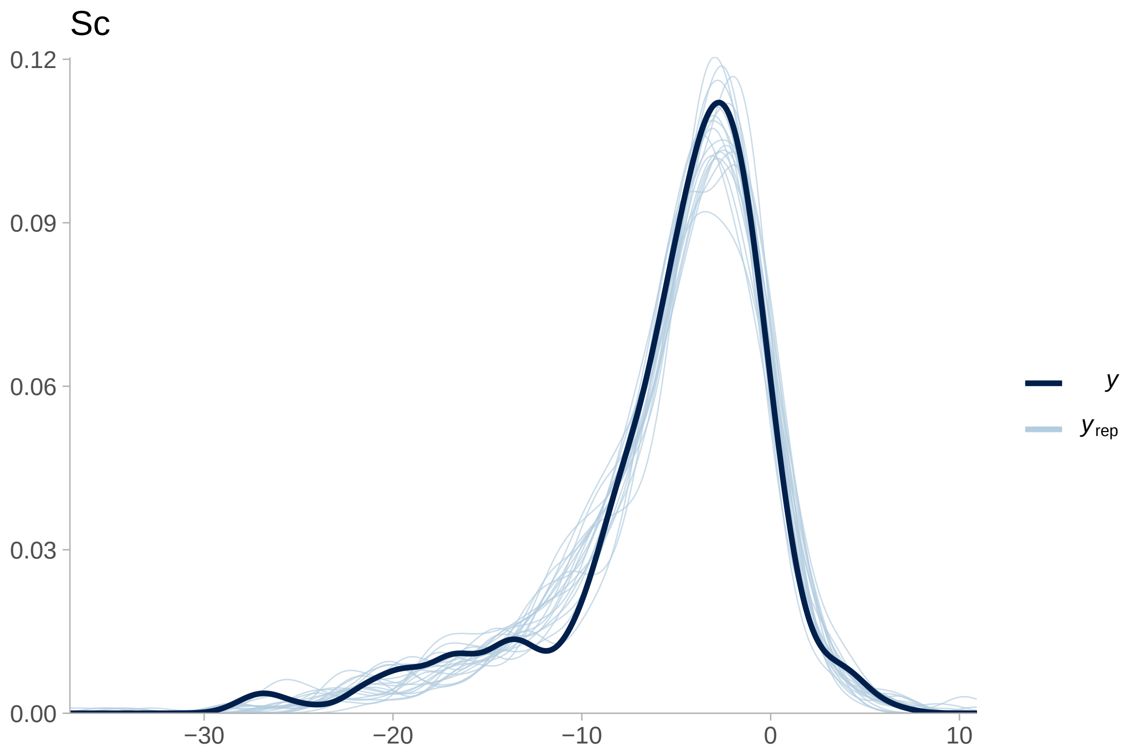


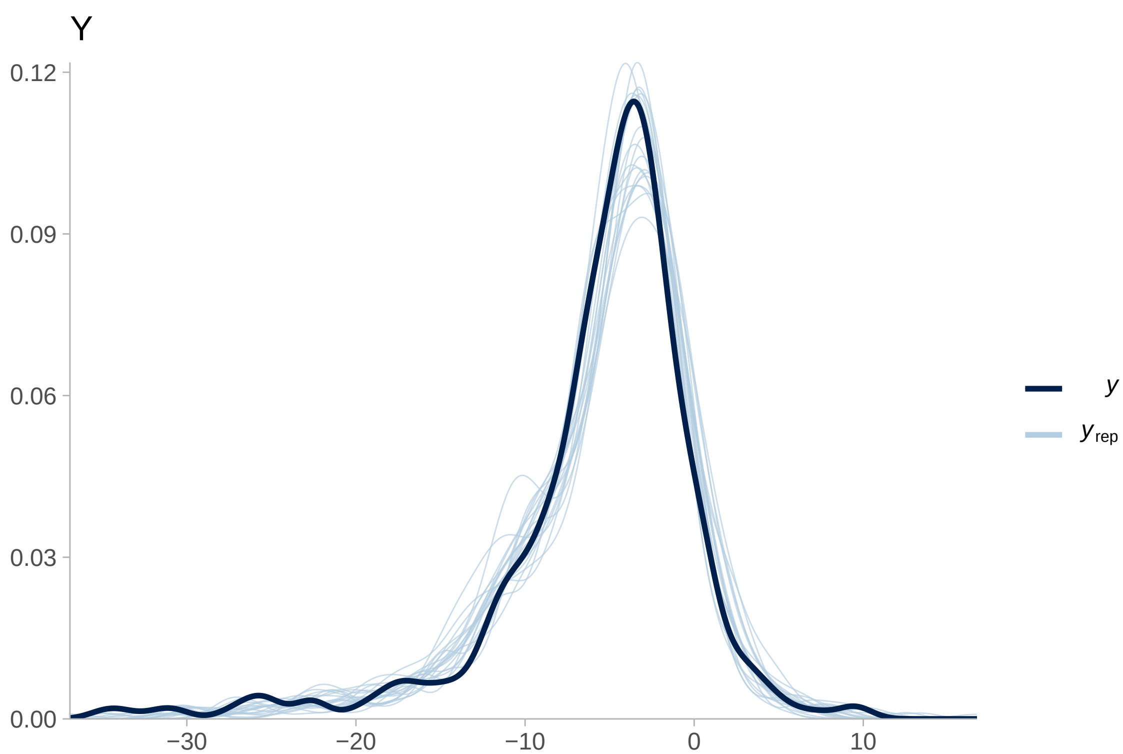


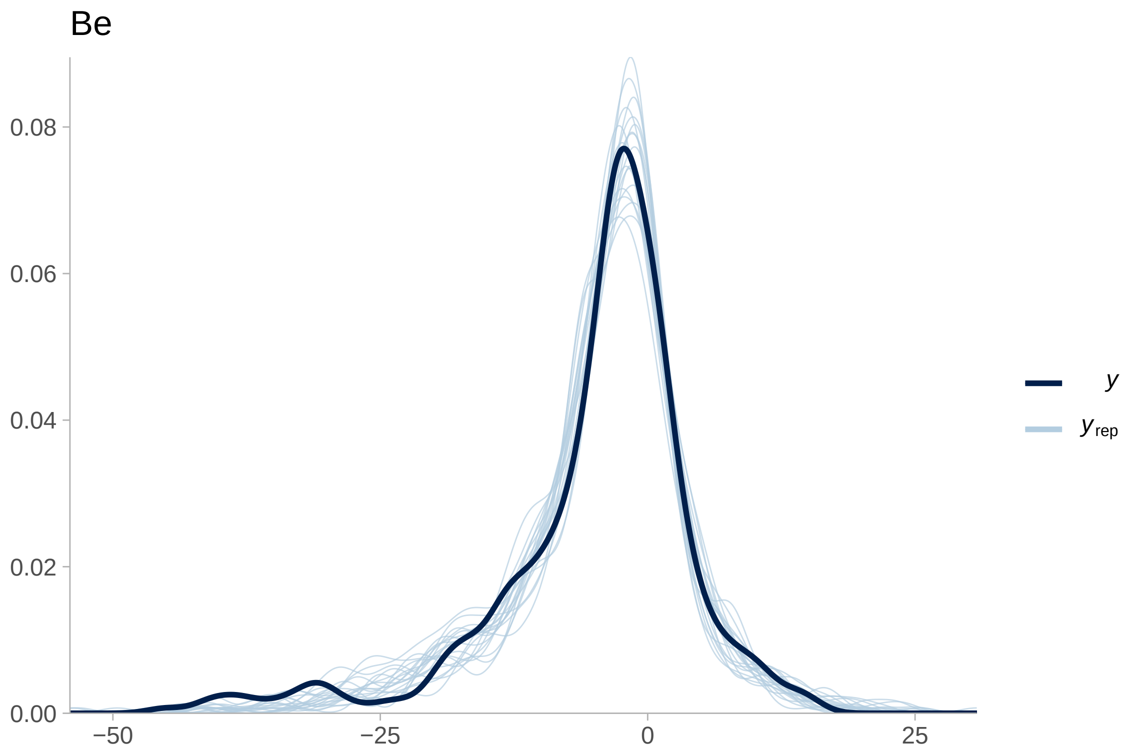
Ho^{2+}
0.15
0.10
0.05
0.00
−10
0
10
y
y_{rep}



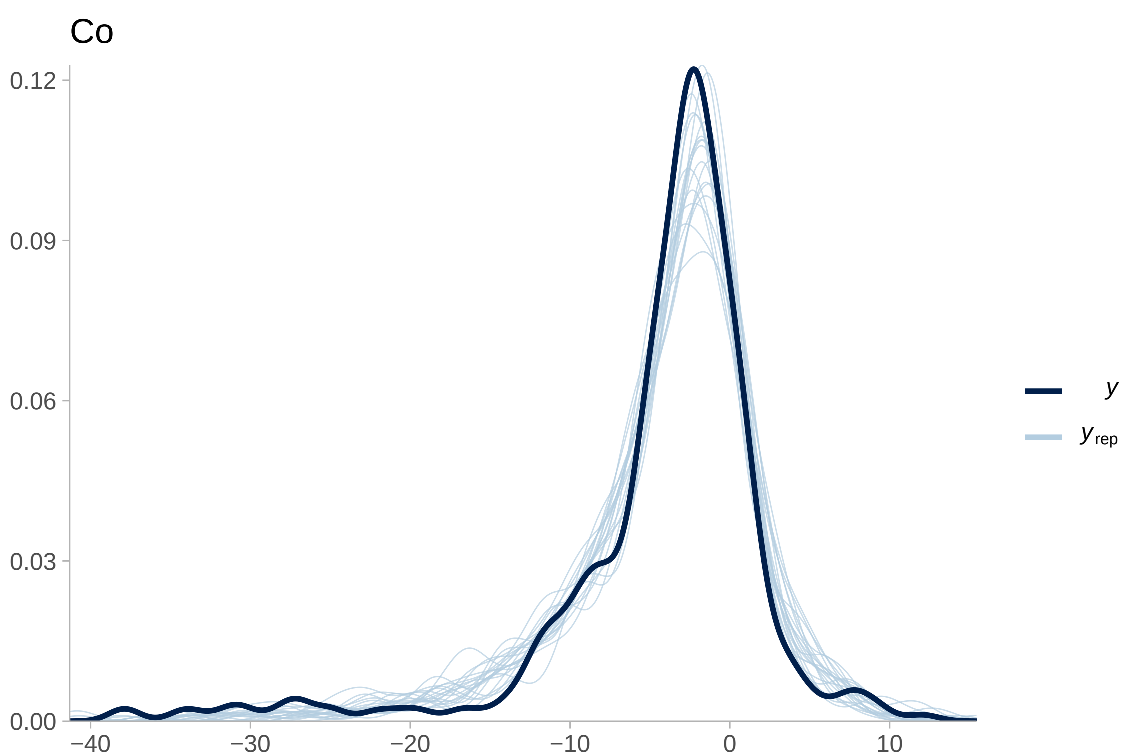
In
0.100
0.075
0.050
0.025
0.000
−30
−20
−10
0
10
y
y_{rep}


Sc
0.12
0.09
0.06
0.03
0.00
−30
−20
−10
0
10
y
y rep

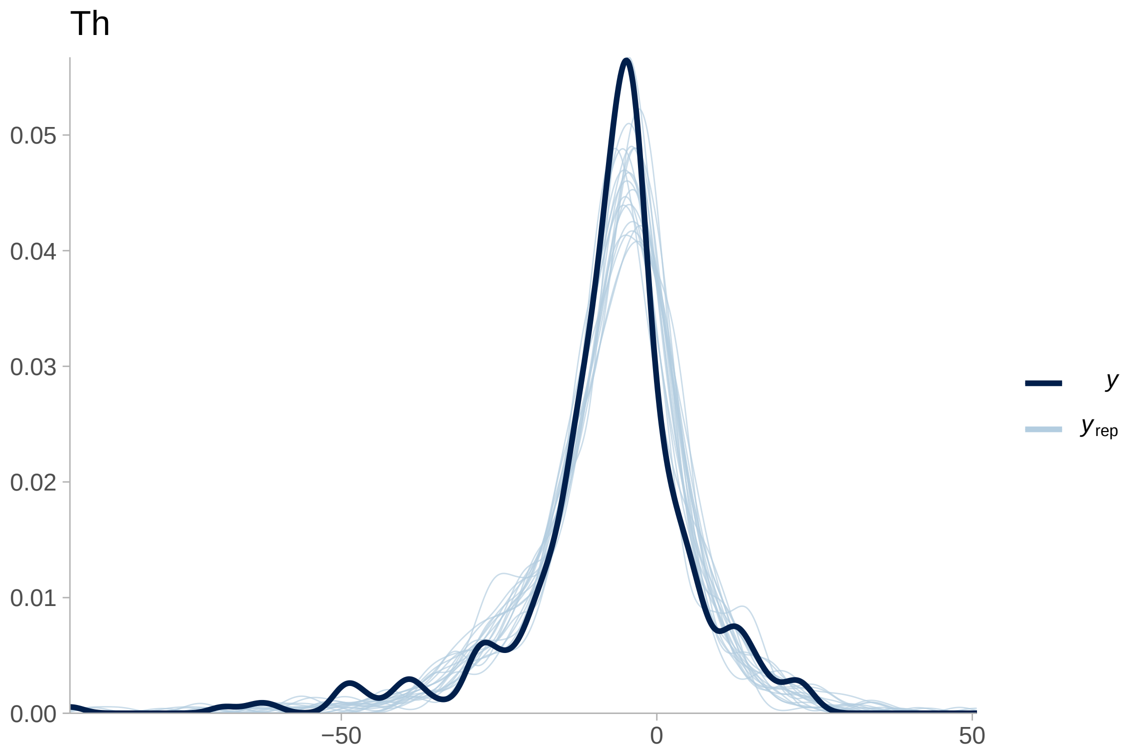

Y
0.12
0.09
0.06
0.03
0.00
−30
−20
−10
0
10
y
y rep

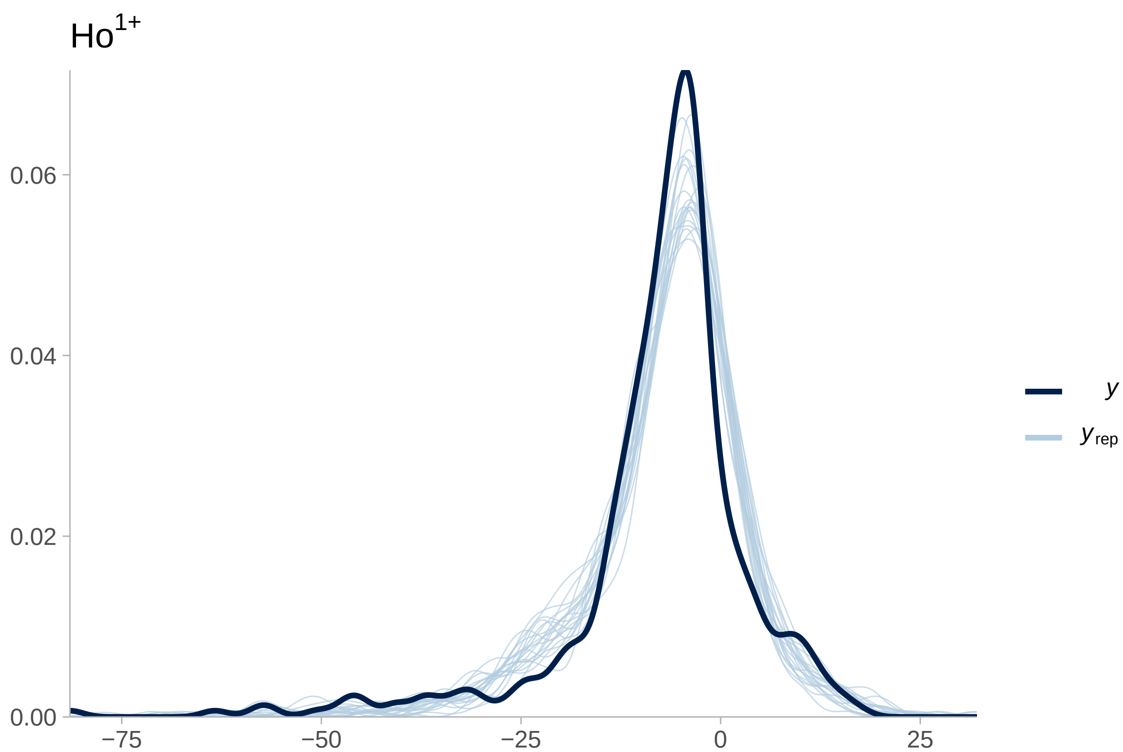


Be
0.08
0.06
0.04
0.02
0.00
−50
−25
0
25
y
y rep



Co
0.12
0.09
0.06
0.03
0.00
−40
−30
−20
−10
0
10
y
y rep

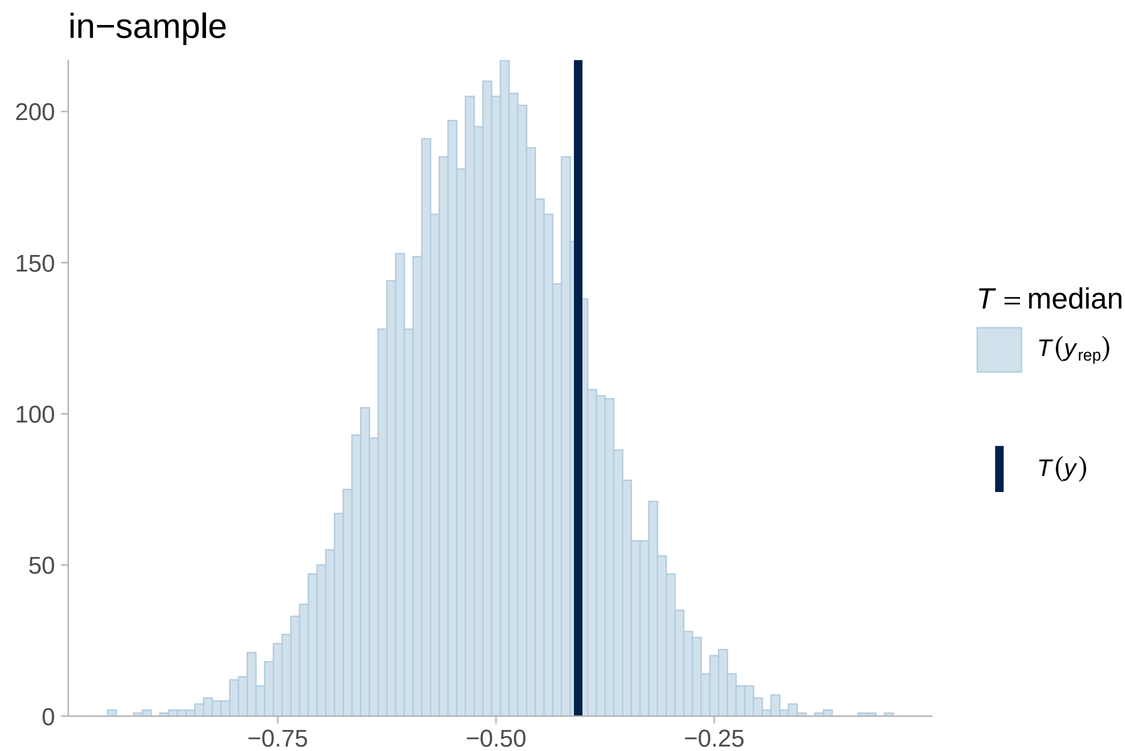


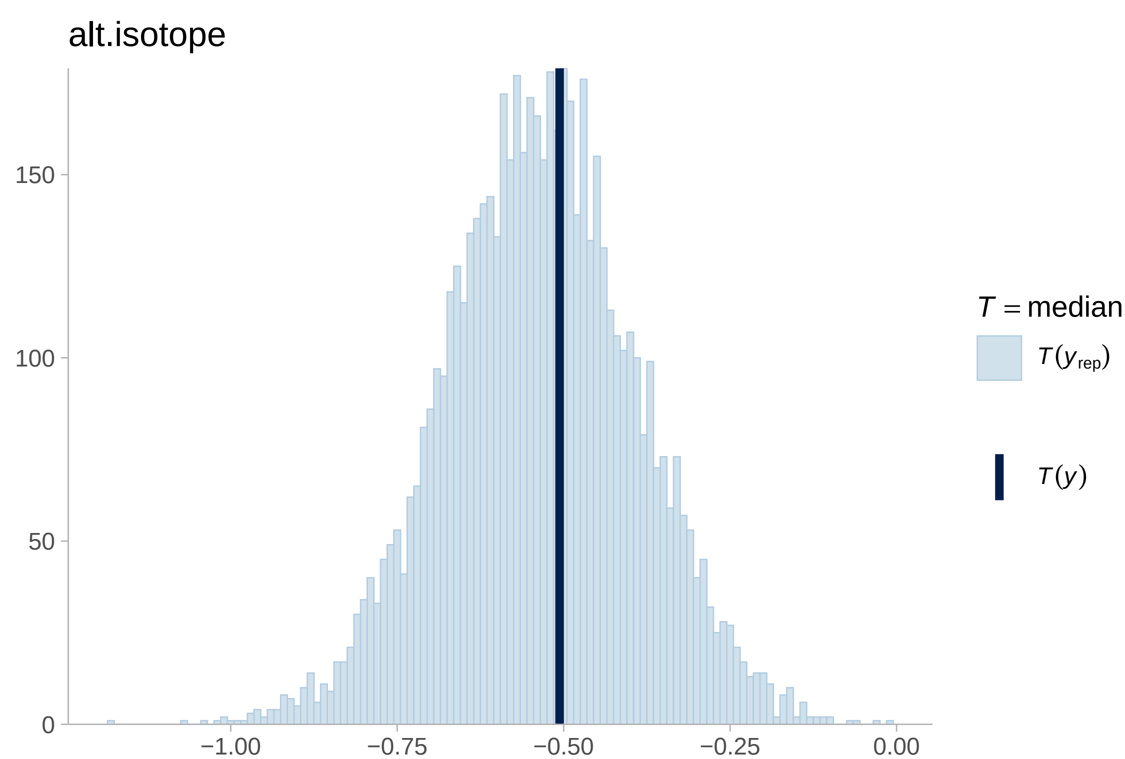


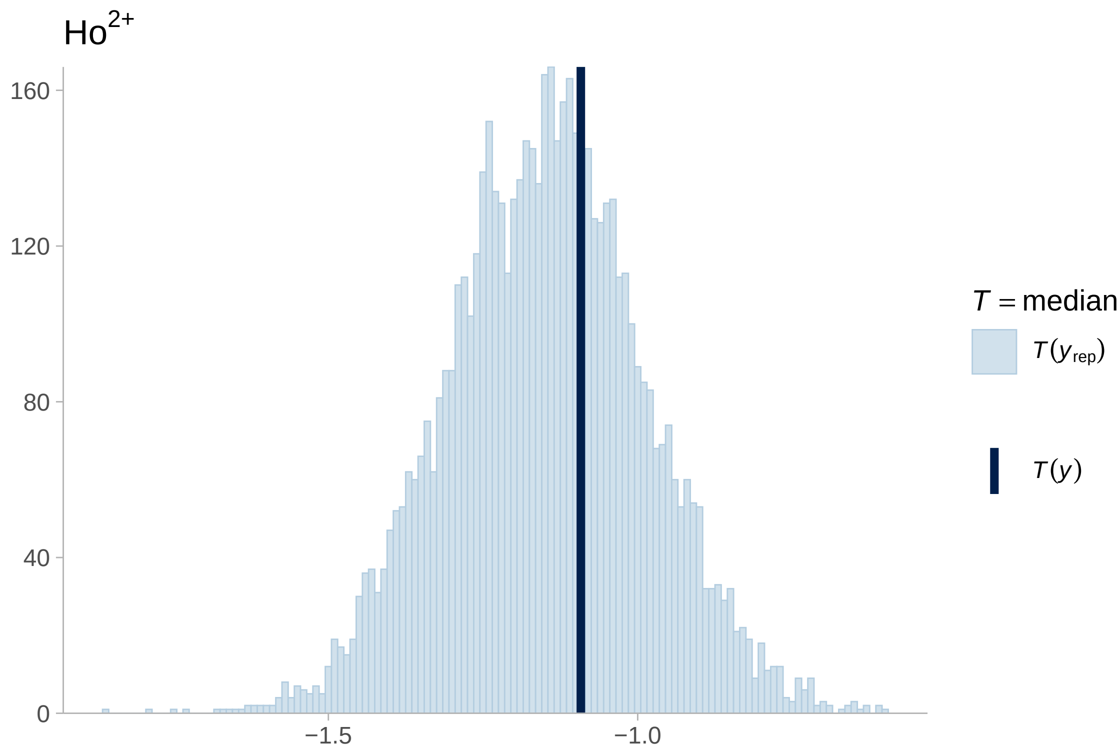
In this check, compared to the same one for arsenic, the same model structure appears to be doing a better job of replicating the observed data.

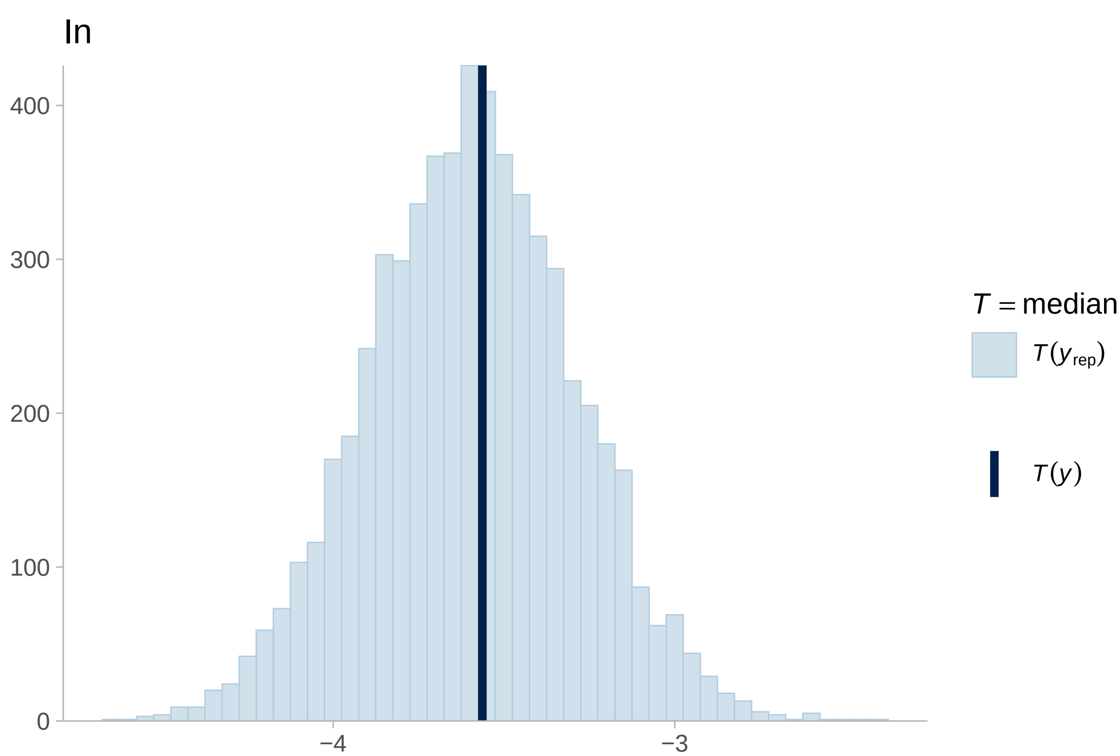
Median

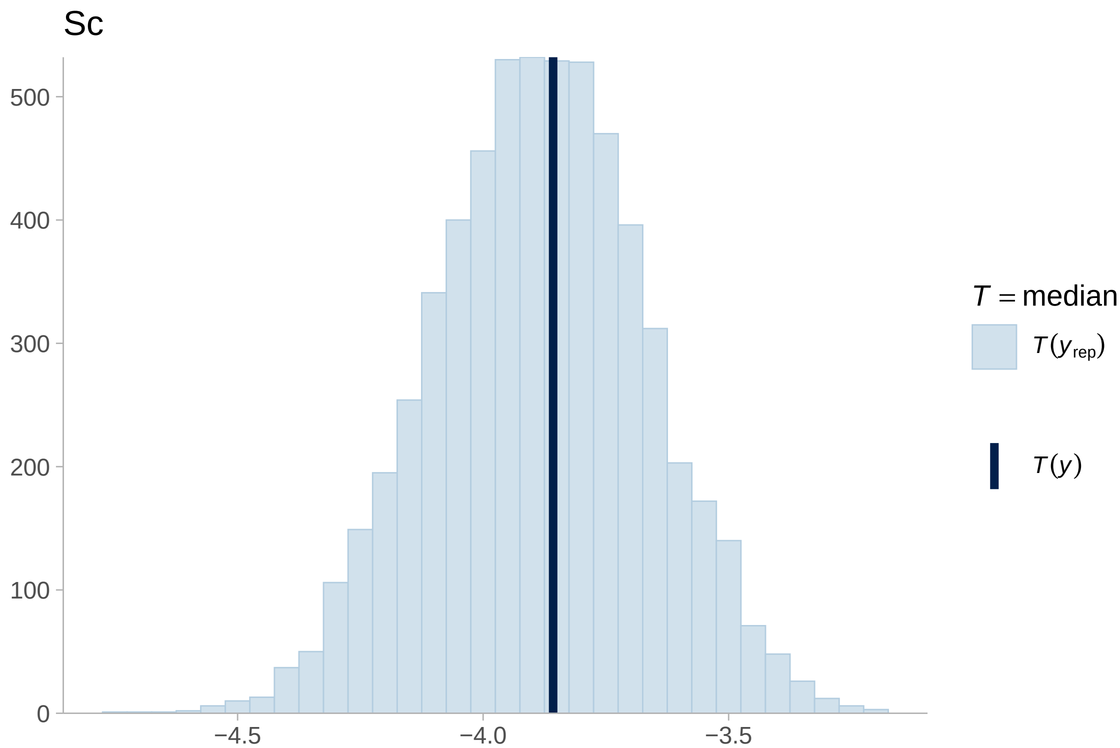
Next is a check comparing replicated medians to the observed medians.

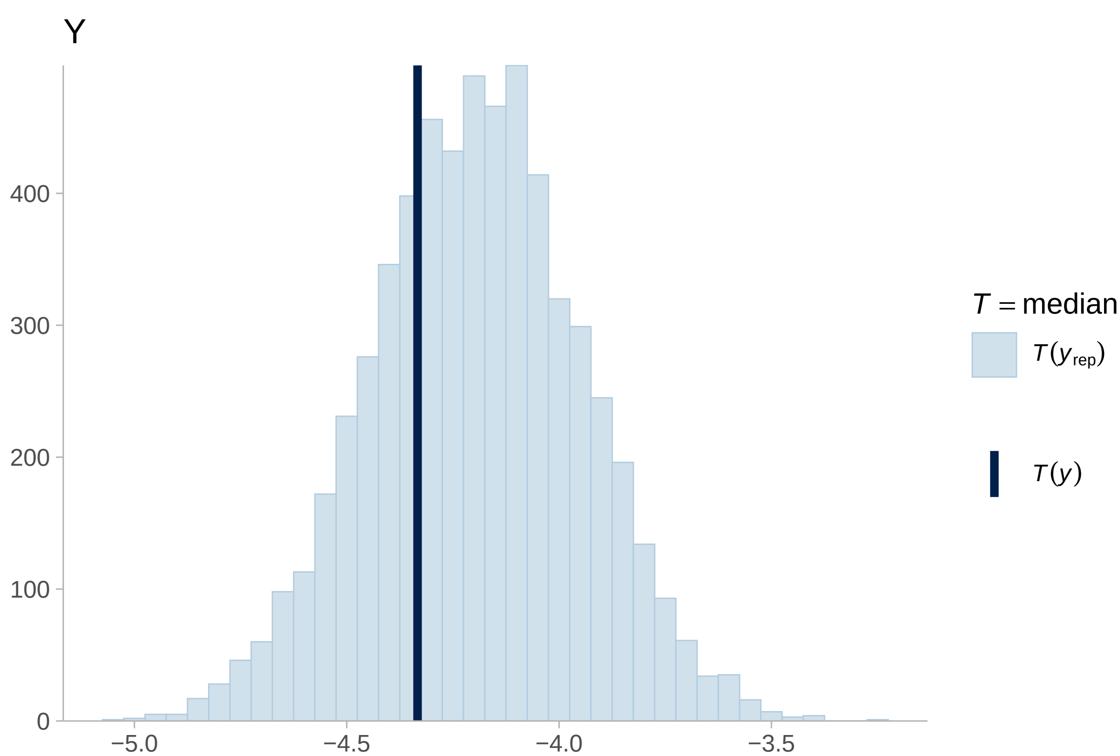


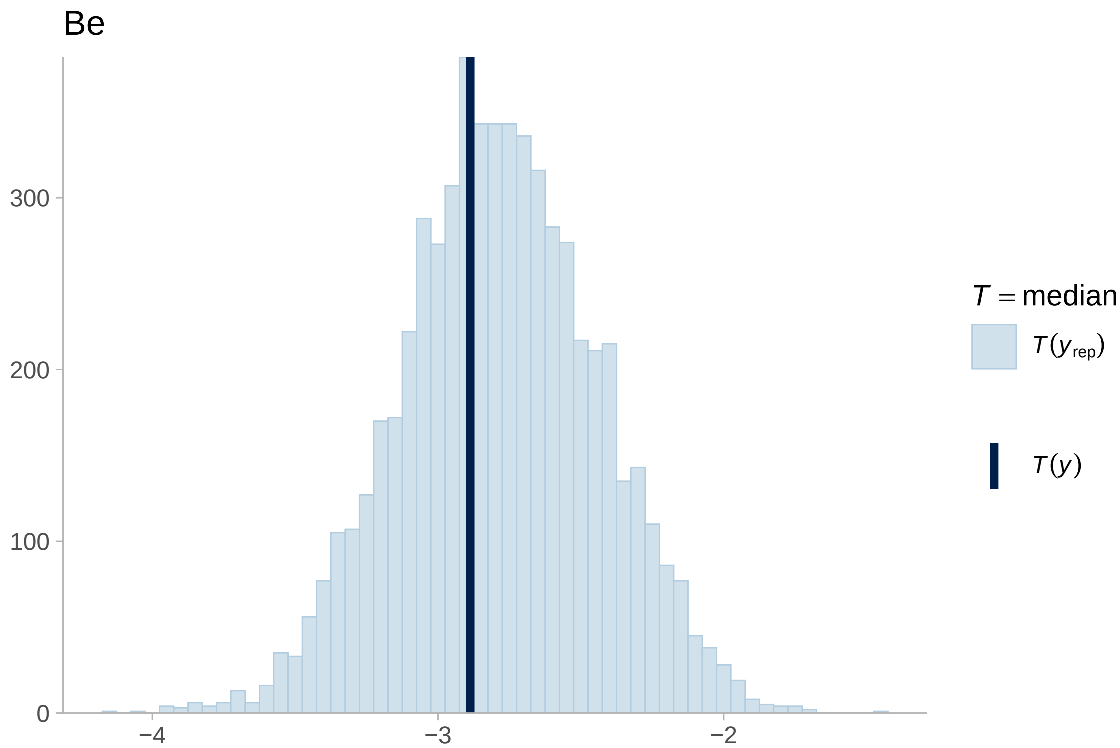



Ho2+
160
120
80
40
0
−1.5
−1.0
T = median
T(yrep)
T(y)

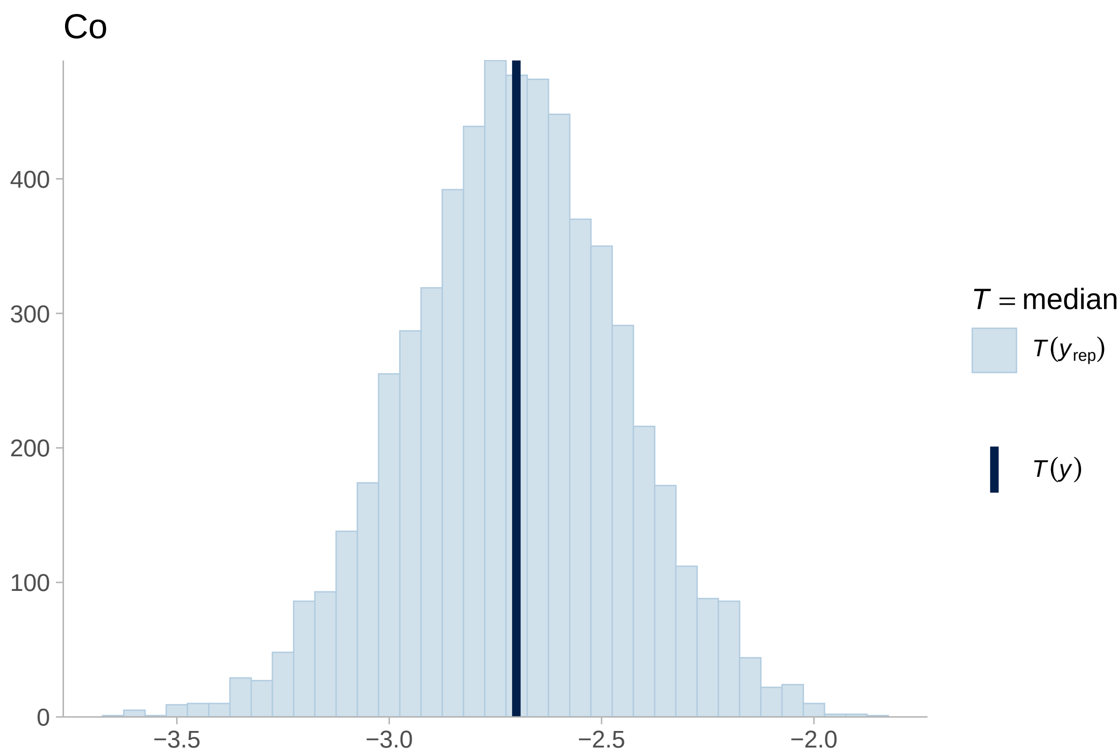

In
400
300
200
100
0
−4
−3
T = median
T(yrep)
T(y)


Sc
500
400
300
200
100
0
−4.5
−4.0
−3.5
T = median
T(y_rep)
T(y)

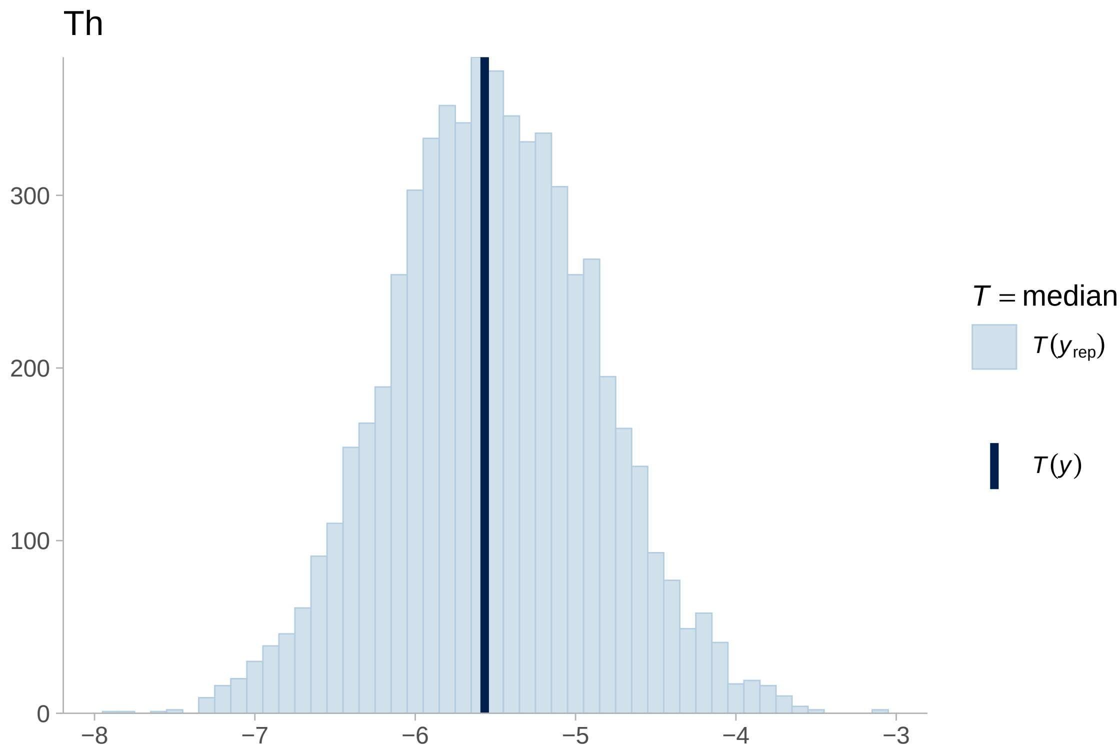

Y
400
300
200
100
0
−5.0
−4.5
−4.0
−3.5
T = median
T(y_rep)
T(y)

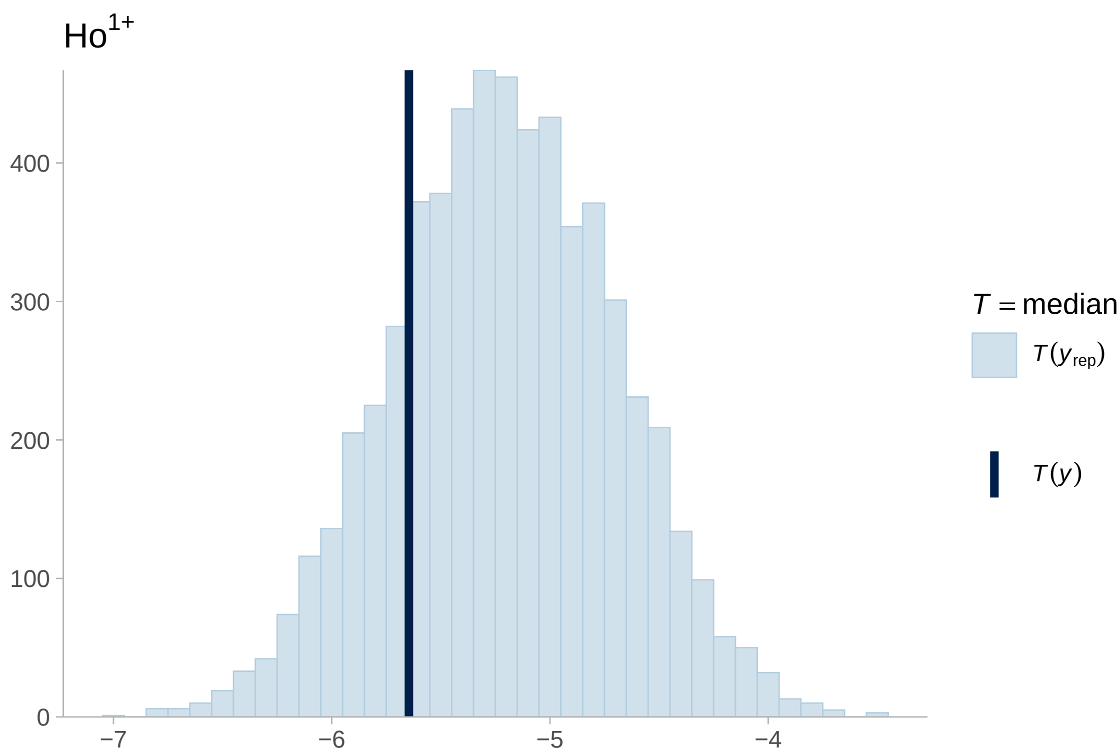


Be
300
200
100
0
−4
−3
−2
T = median
T(y_rep)
T(y)



Co
400
300
200
100
0
−3.5
−3.0
−2.5
−2.0
T = median
T(y_rep)
T(y)

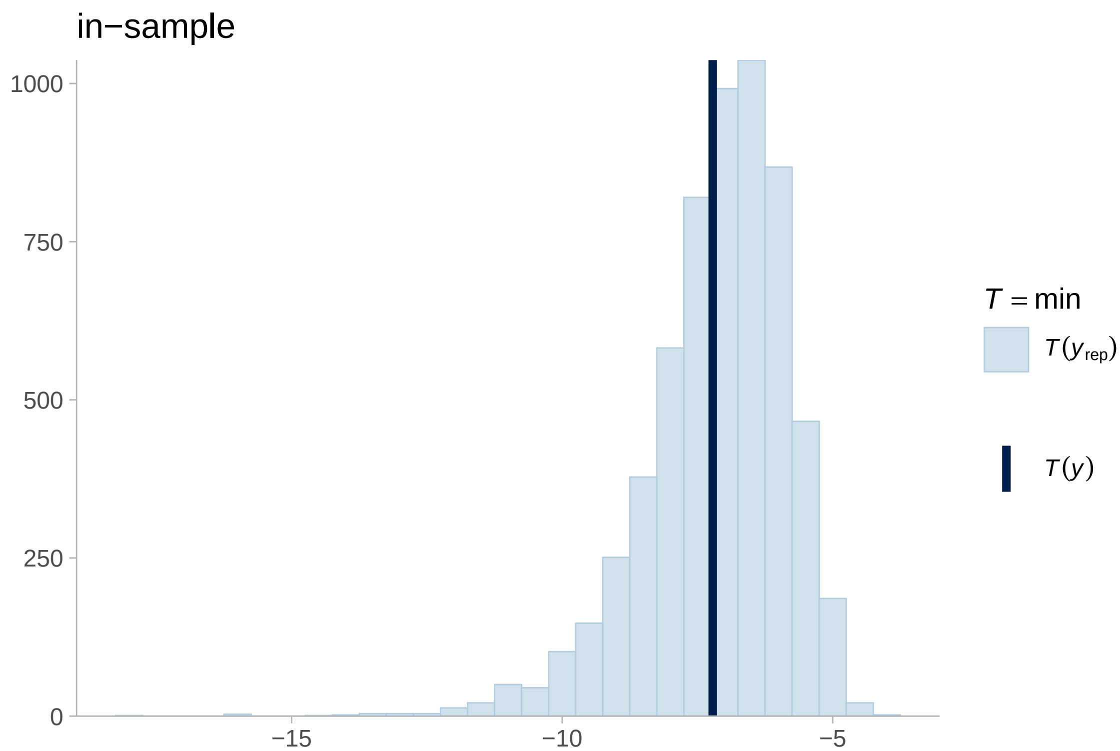


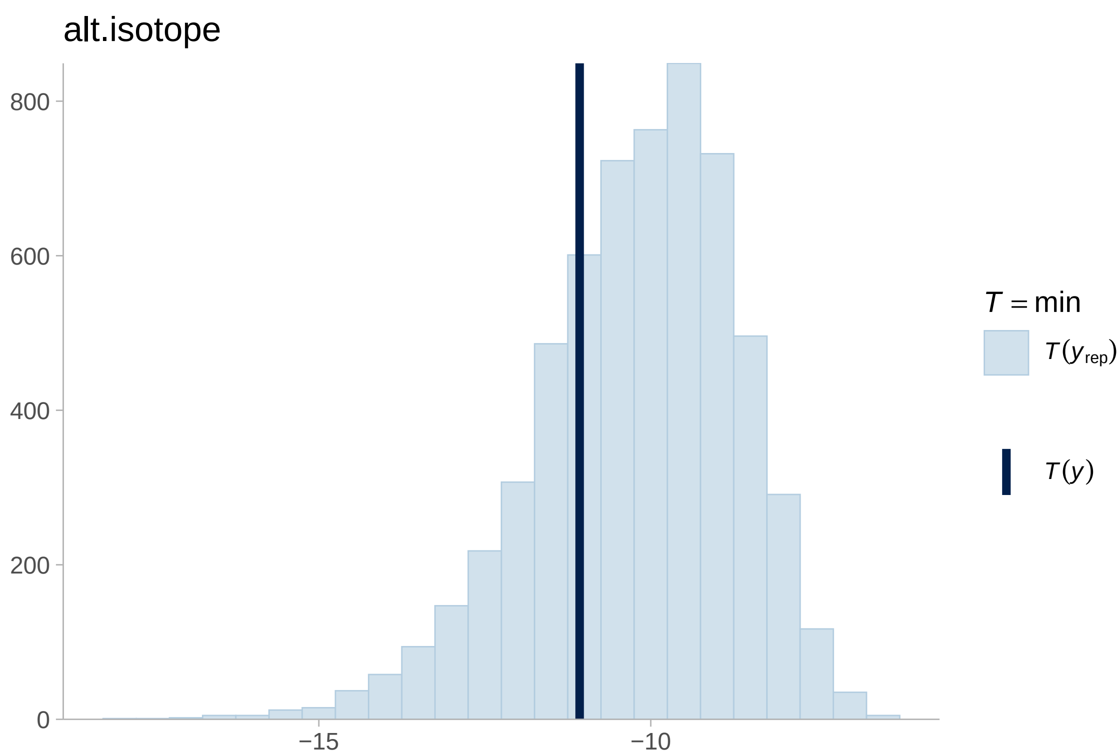


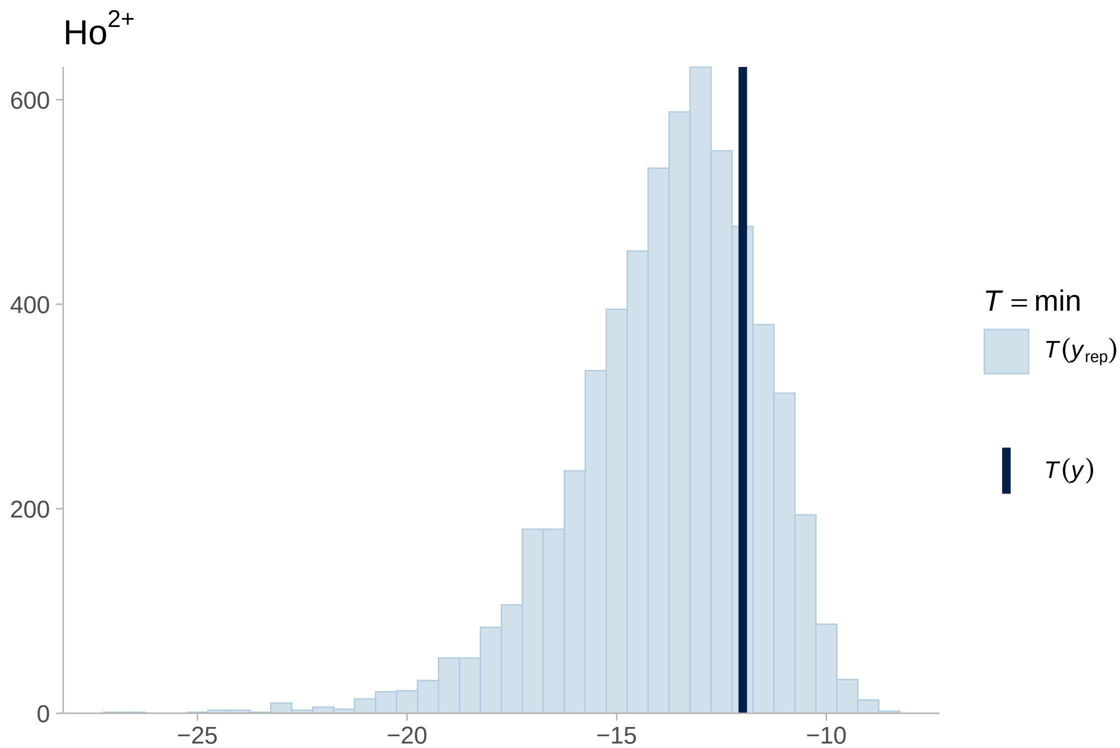
This check doesn't suggest any glaring issues.

Min

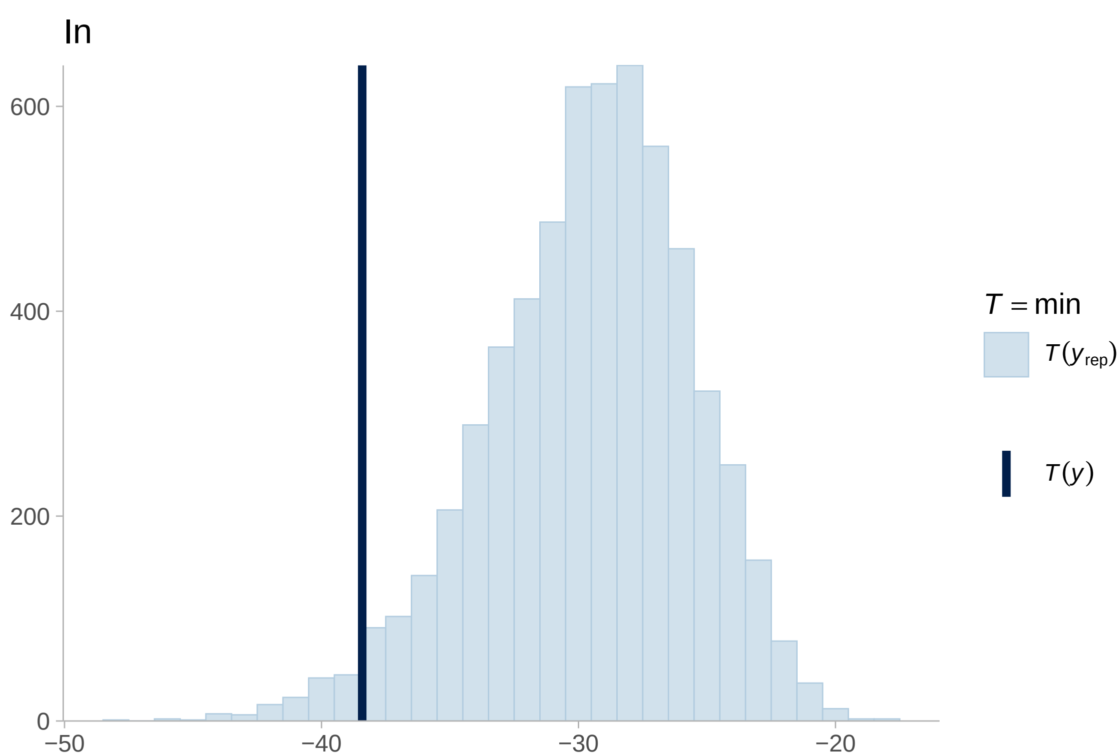
Next is a check comparing replicated mins to the observed mins.



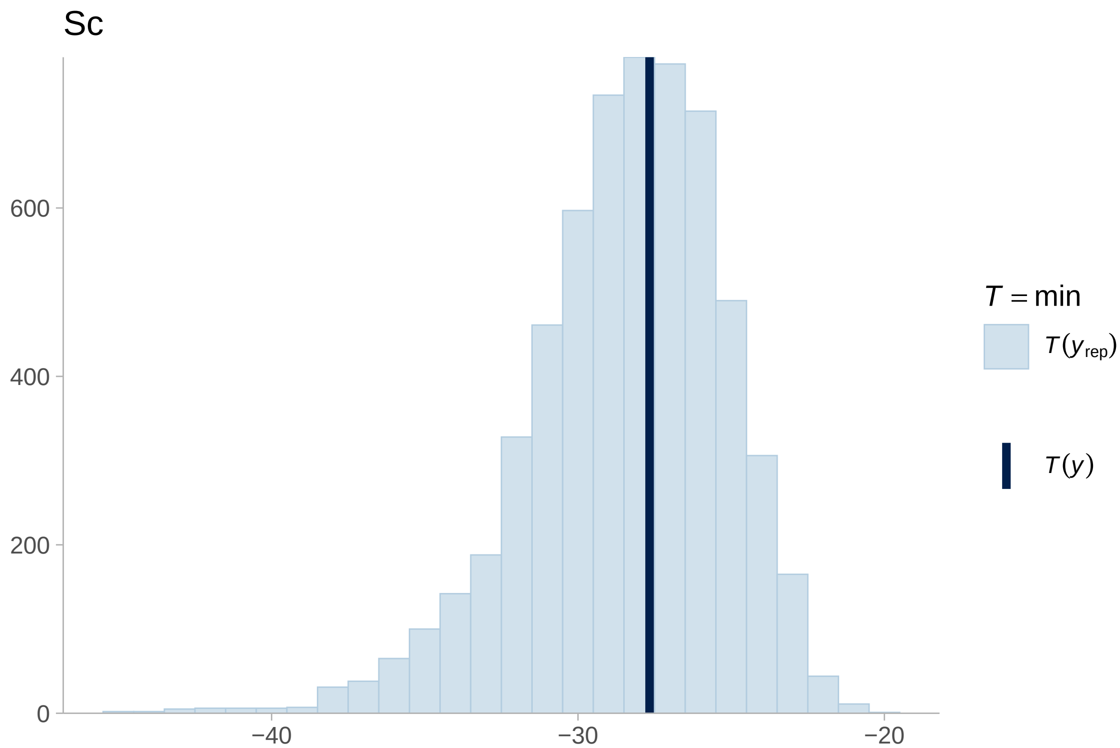


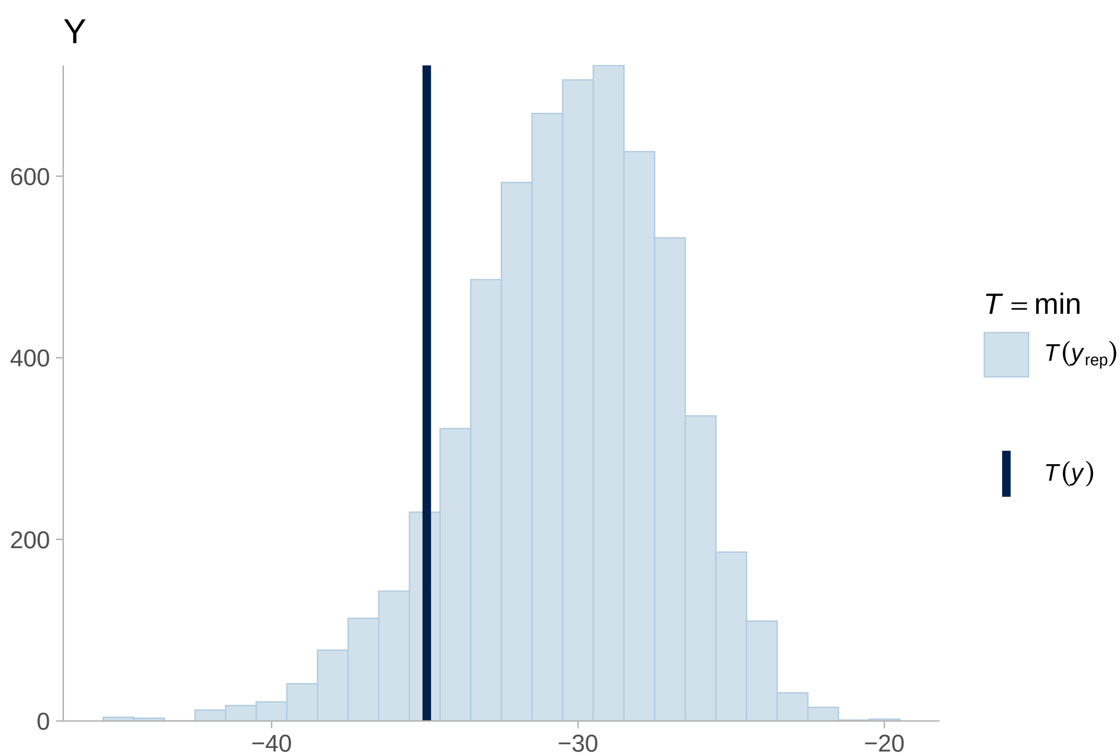


Ho^{2+}
600
400
200
0
−25
−20
−15
−10
$T = \min$
$T(y_{rep})$
$T(y)$

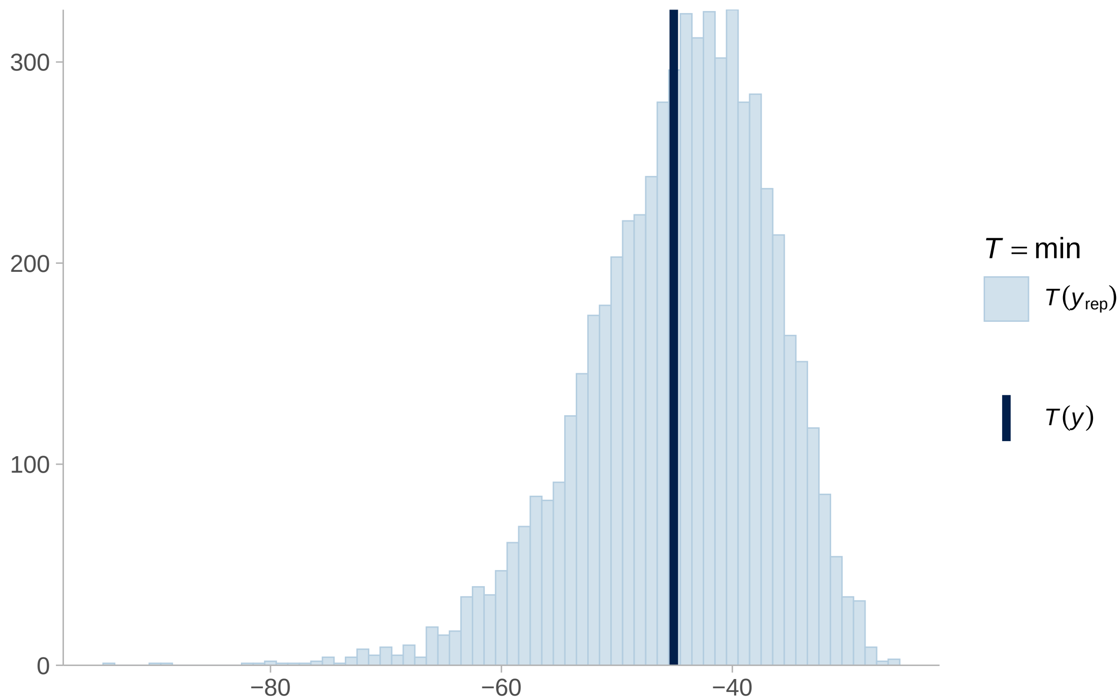


In
600
400
200
0
−50
−40
−30
−20
$T = \min$
$T(y_{rep})$
$T(y)$


Sc
600
400
200
0
−40
−30
−20
$T = \text{min}$
$T(y_{\text{rep}})$
$T(y)$

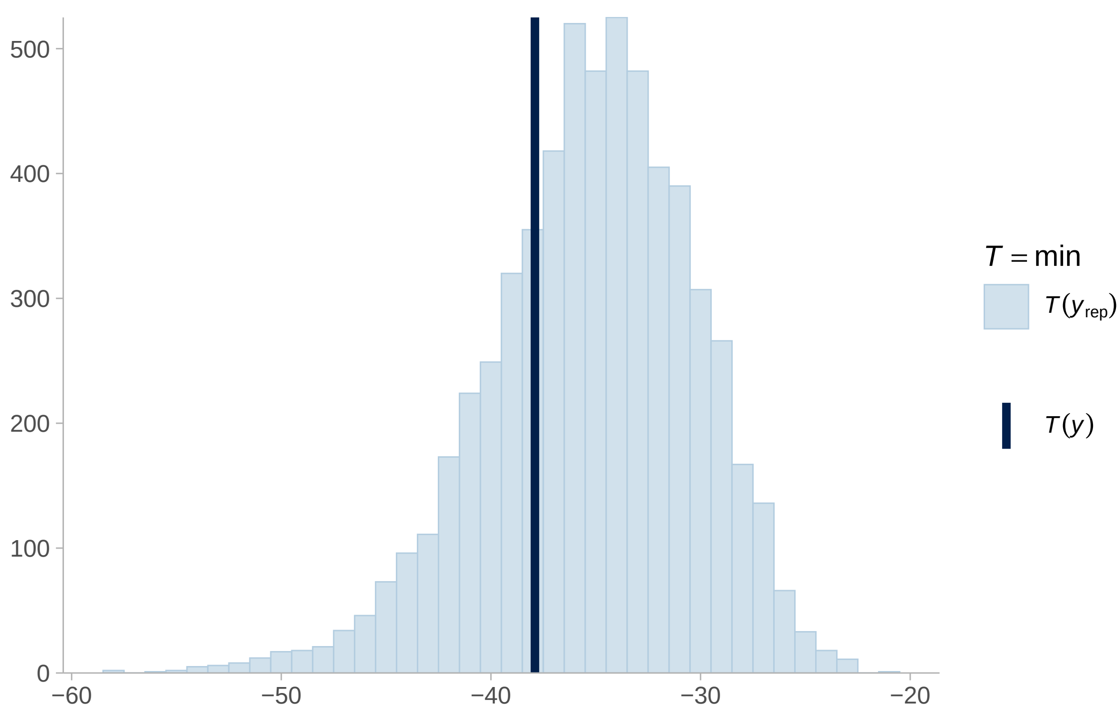

Y
600
400
200
0
−40
−30
−20
$T = \text{min}$
$T(y_{\text{rep}})$
$T(y)$

Be

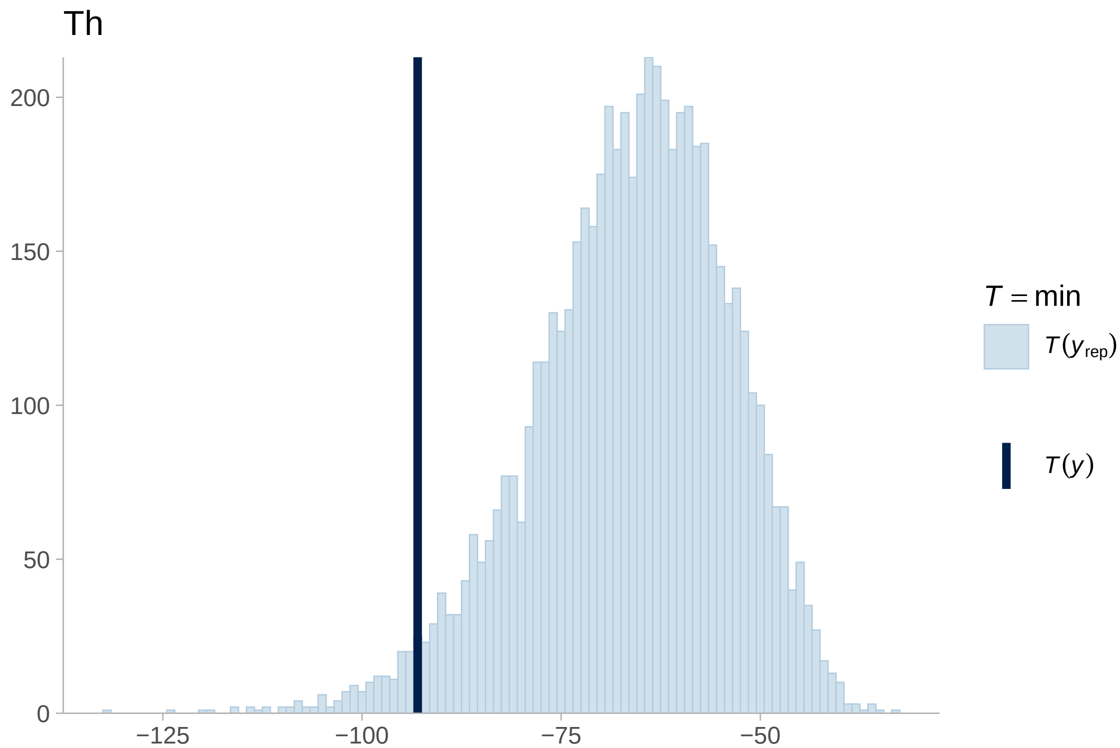


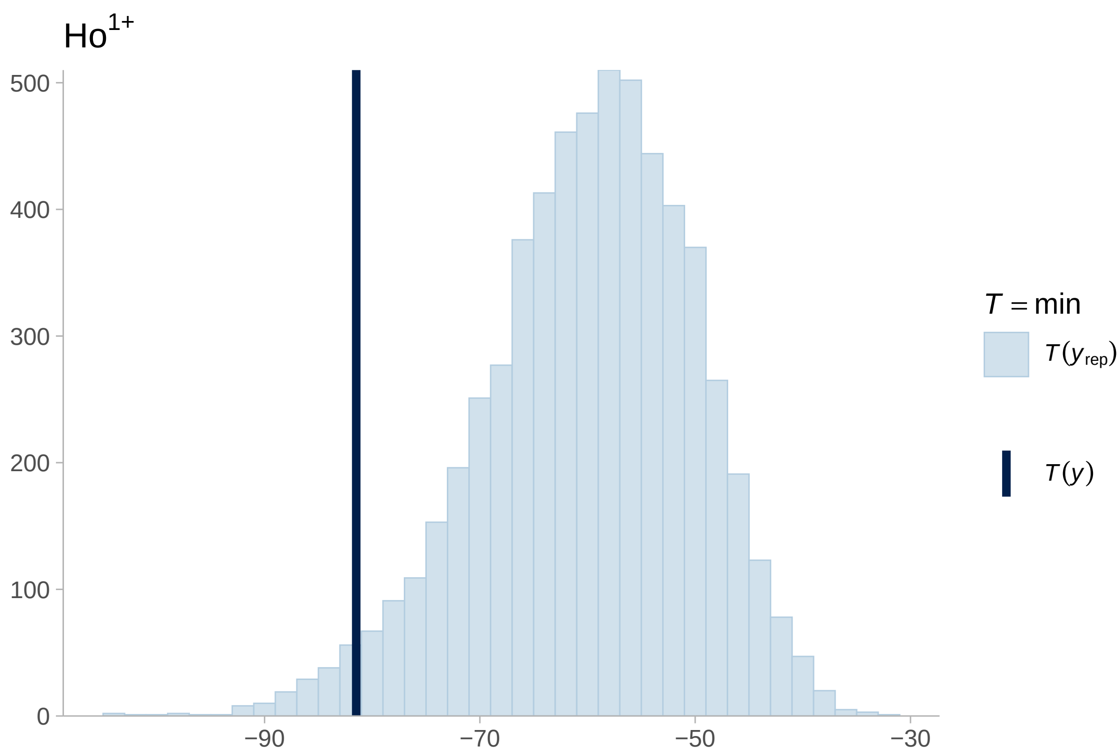
300
200
100
0
−80
−60
−40
$T = \min$
$T(y_{rep})$
$T(y)$

Co



500
400
300
200
100
0
−60
−50
−40
−30
−20
$T = \min$
$T(y_{rep})$
$T(y)$

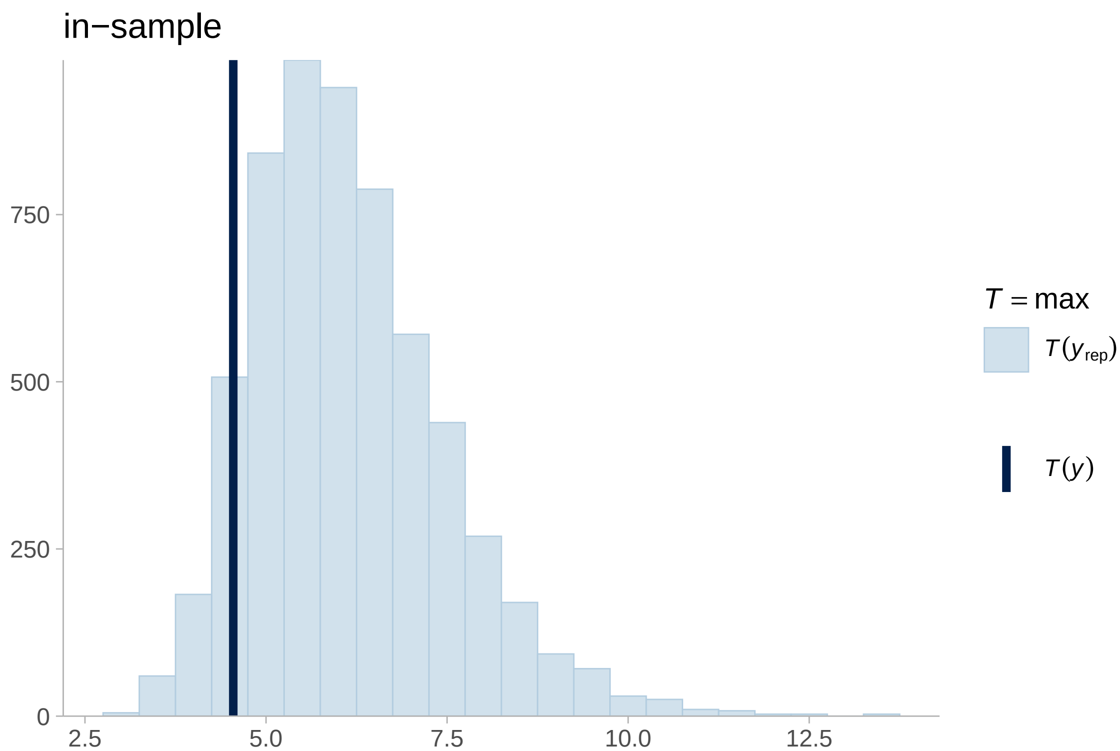


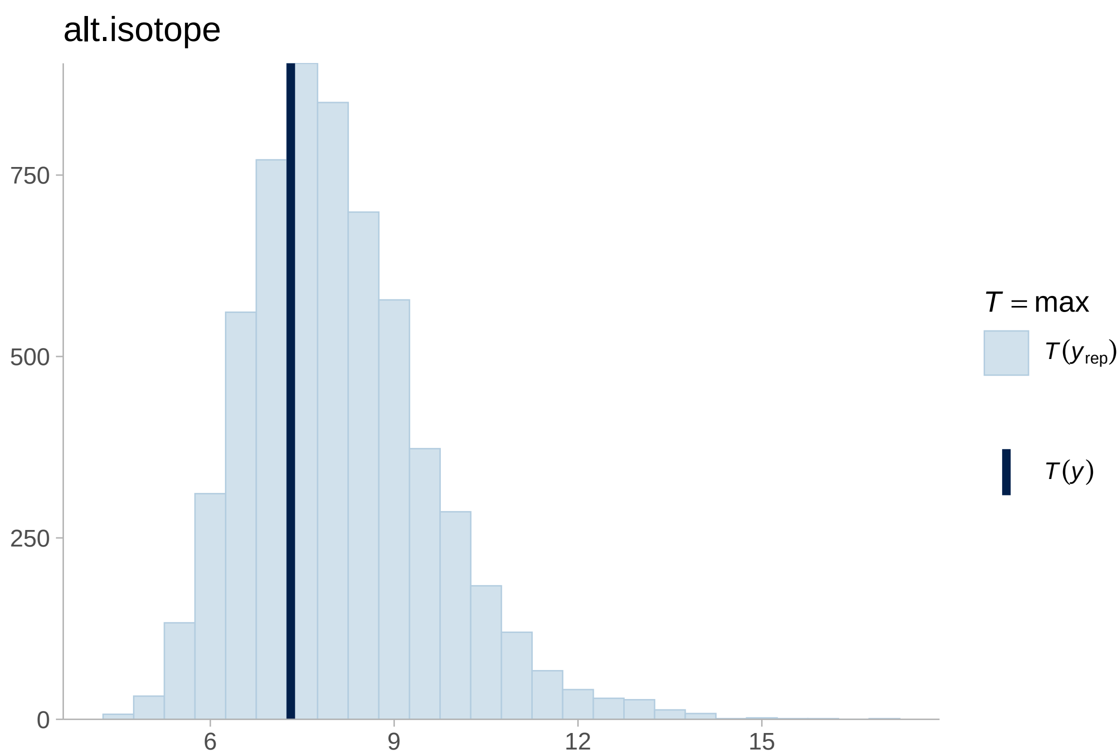


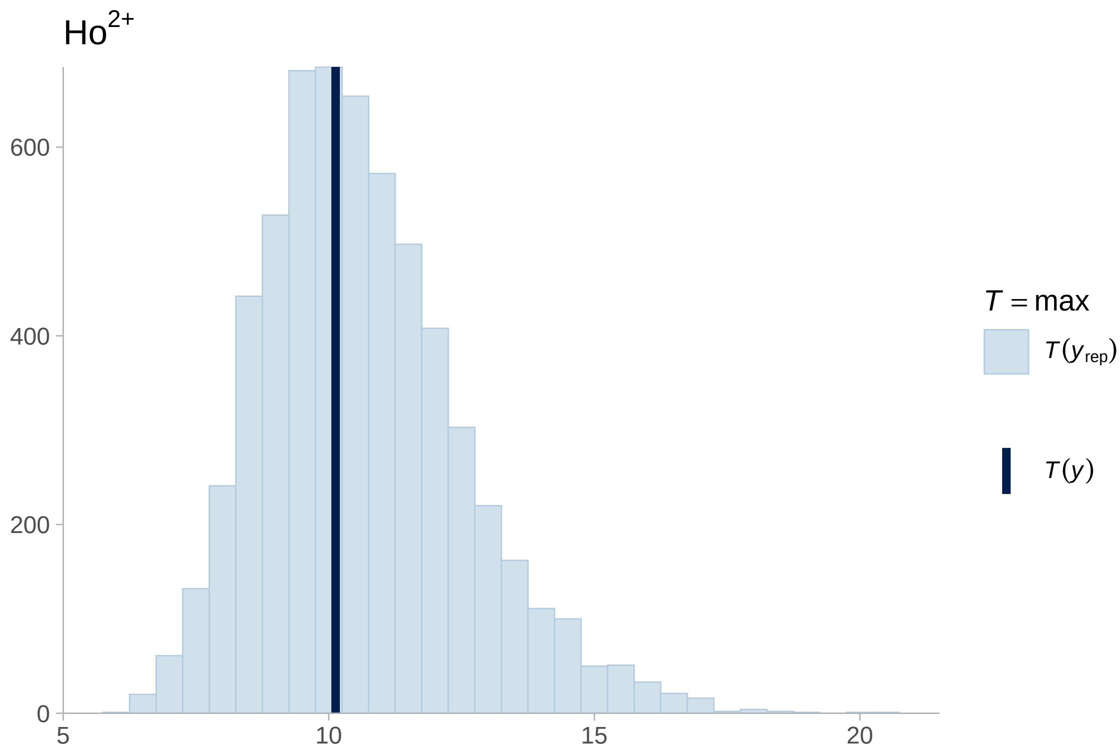
This check look good for most of the methods, but the checks for In, Y, Th, and Ho^{+1} may suggest that the model is consistently generating a lighter left tail compared to the observed data.

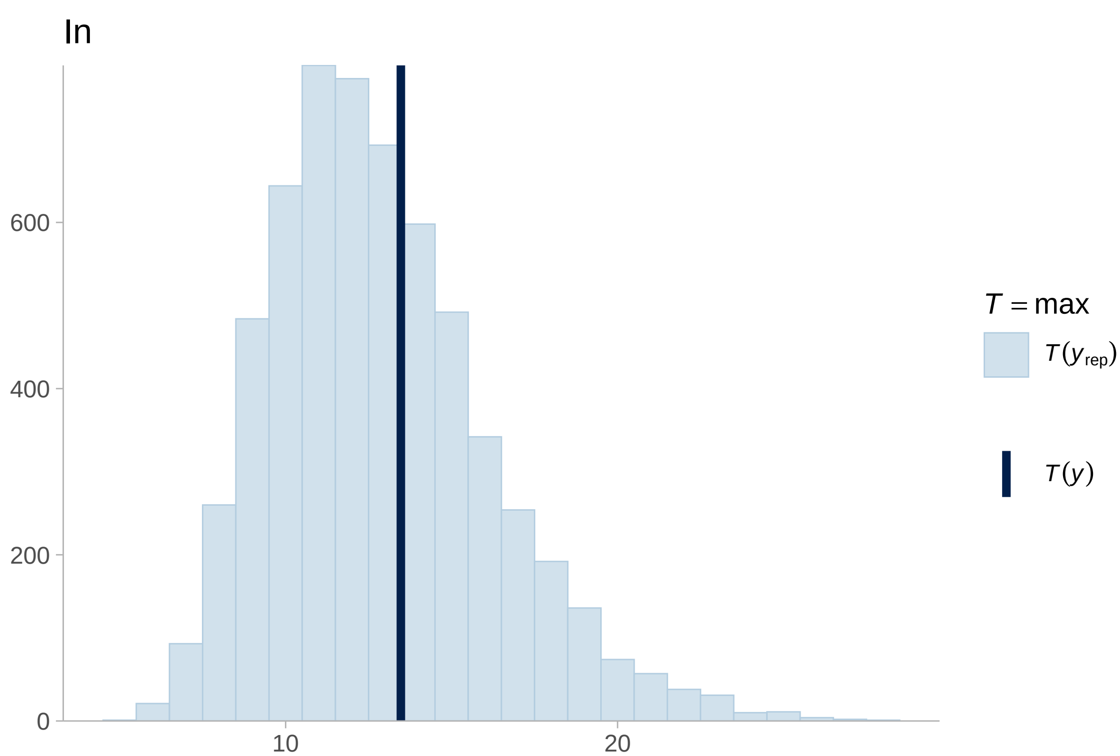
Max

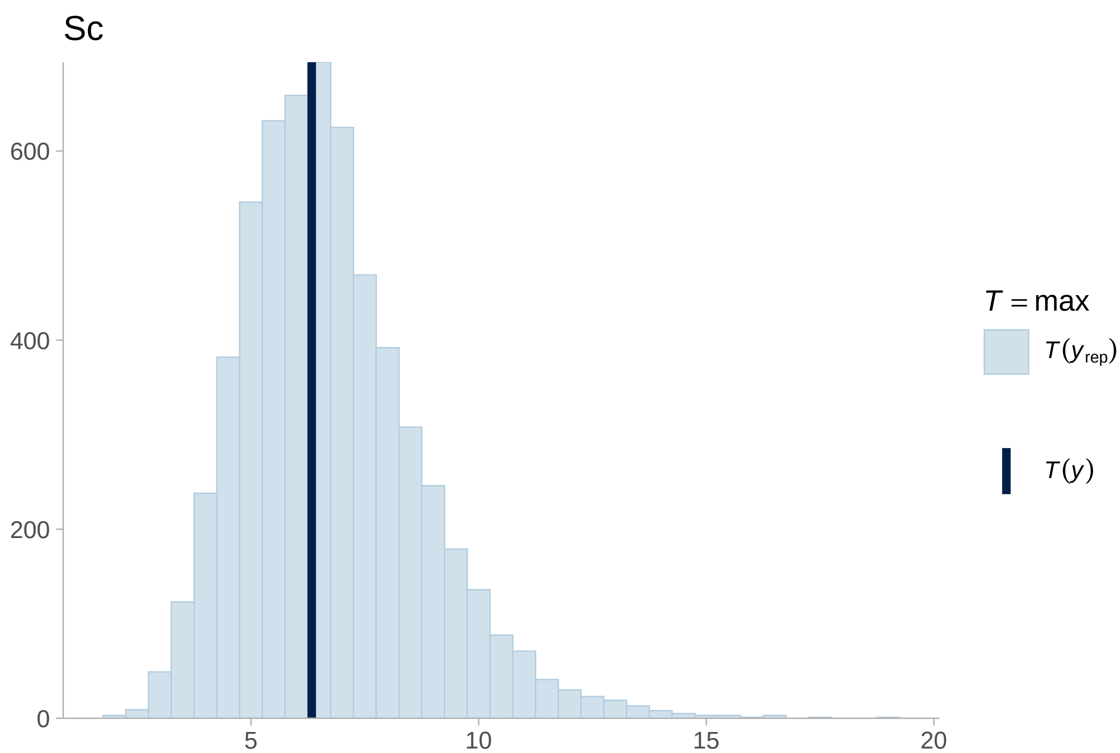
Next is a check comparing replicated maxs to the observed maxs.

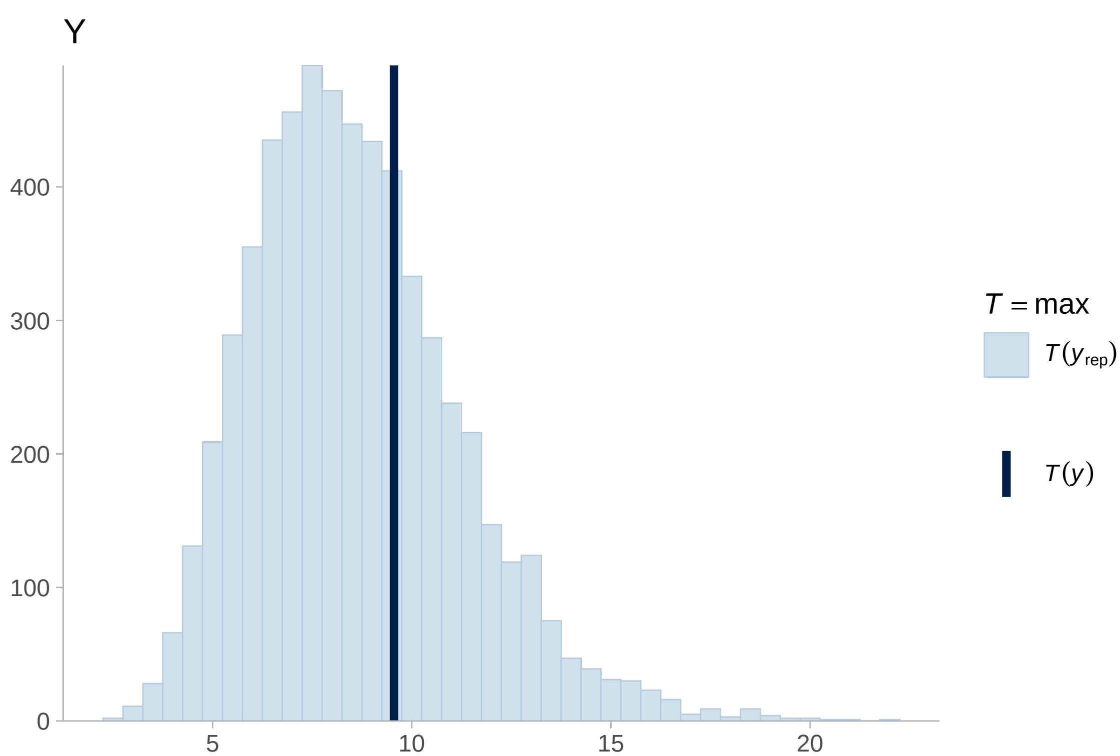


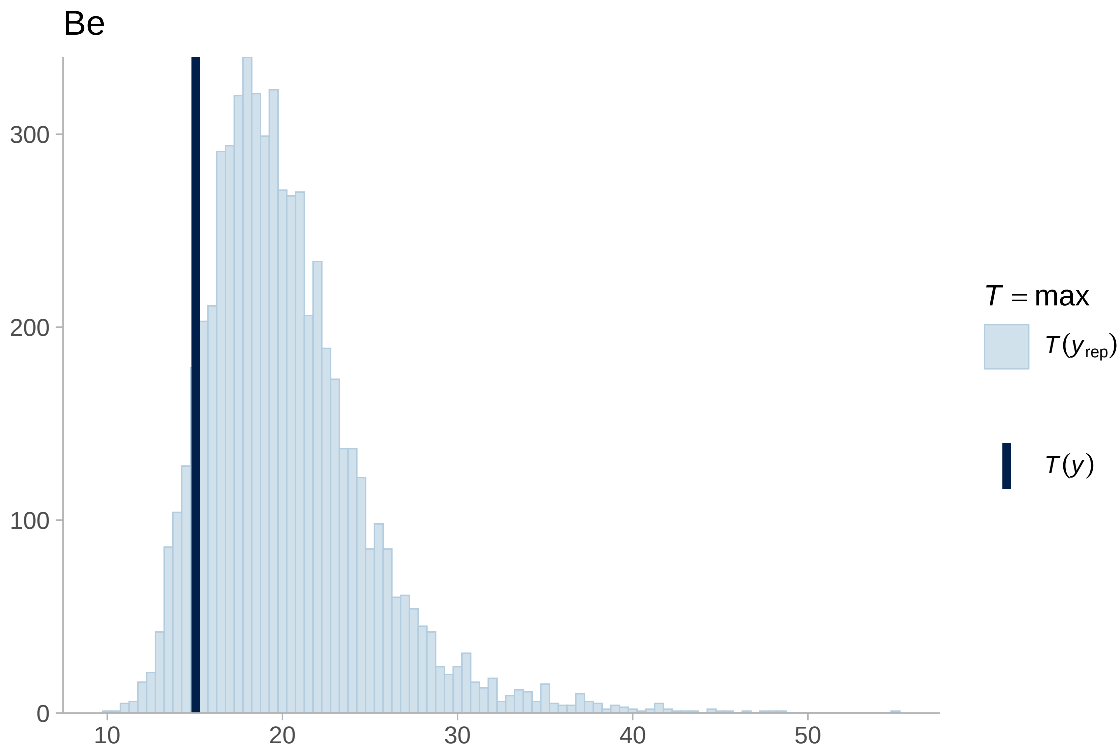


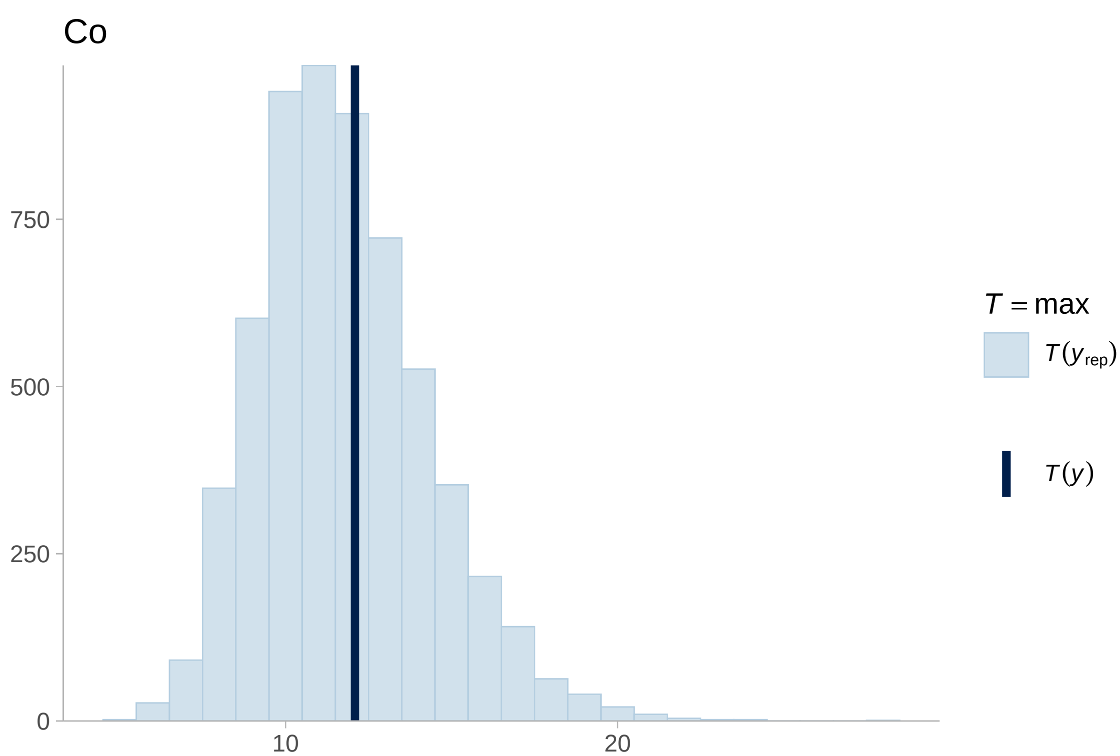

Ho2+
600
400
200
0
5
10
15
20
T = max
T(yrep)
T(y)

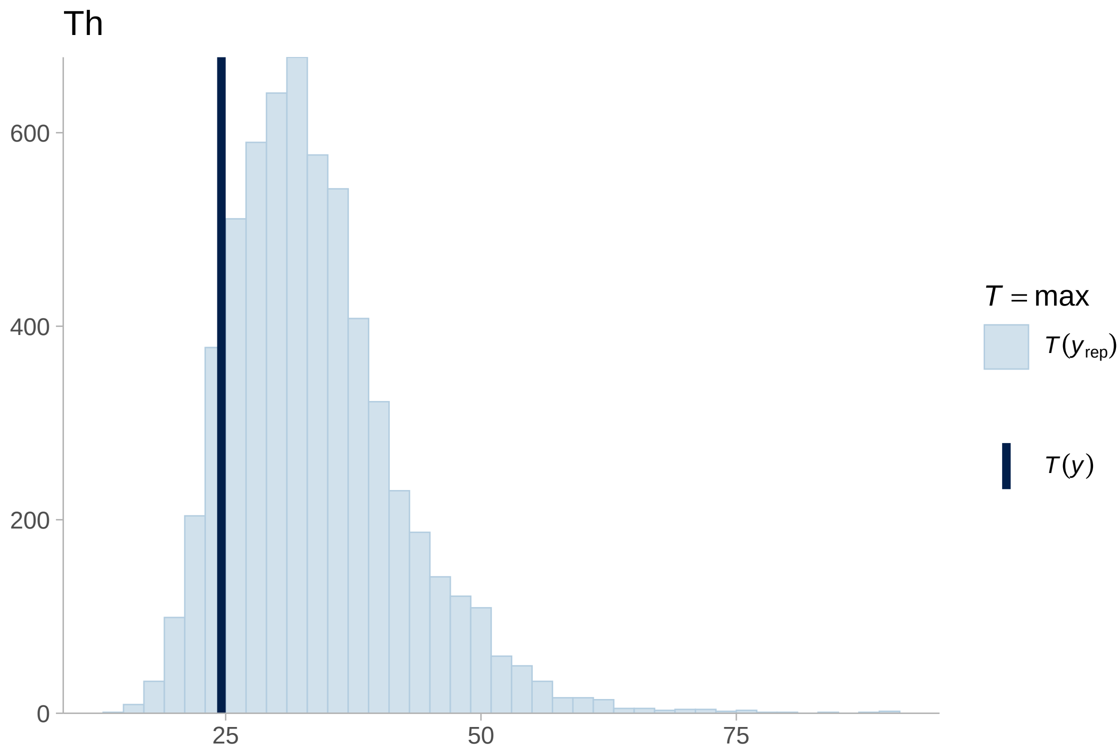

In
600
400
200
0
10
20
T = max
T(yrep)
T(y)

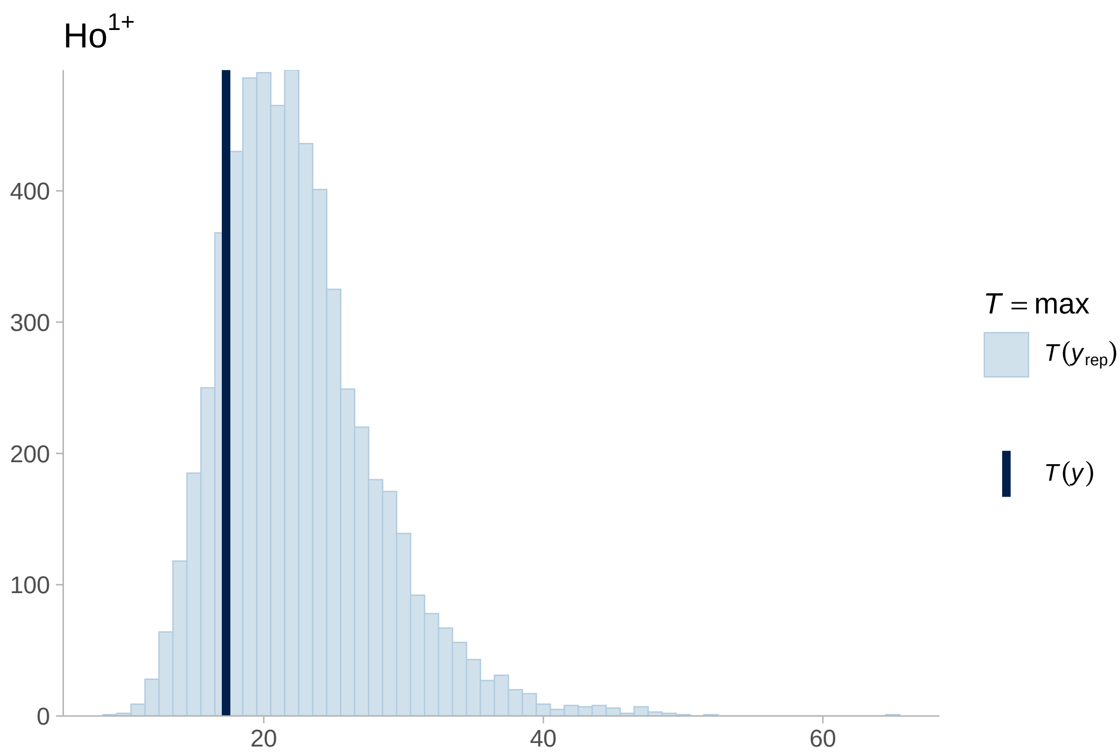

Sc
600
400
200
0
5
10
15
20
T = max
T(y_rep)
T(y)


Y
400
300
200
100
0
5
10
15
20
T = max
T(y_rep)
T(y)


Be
300
200
100
0
10
20
30
40
50
T = max
T(y_rep)
T(y)


Co
750
500
250
0
10
20
T = max
T(y_rep)
T(y)





This check looks pretty reasonable with regard to the model's ability to replicate the max for all methods.

Posterior inferences

Next, the posterior inferences from the selenium model.

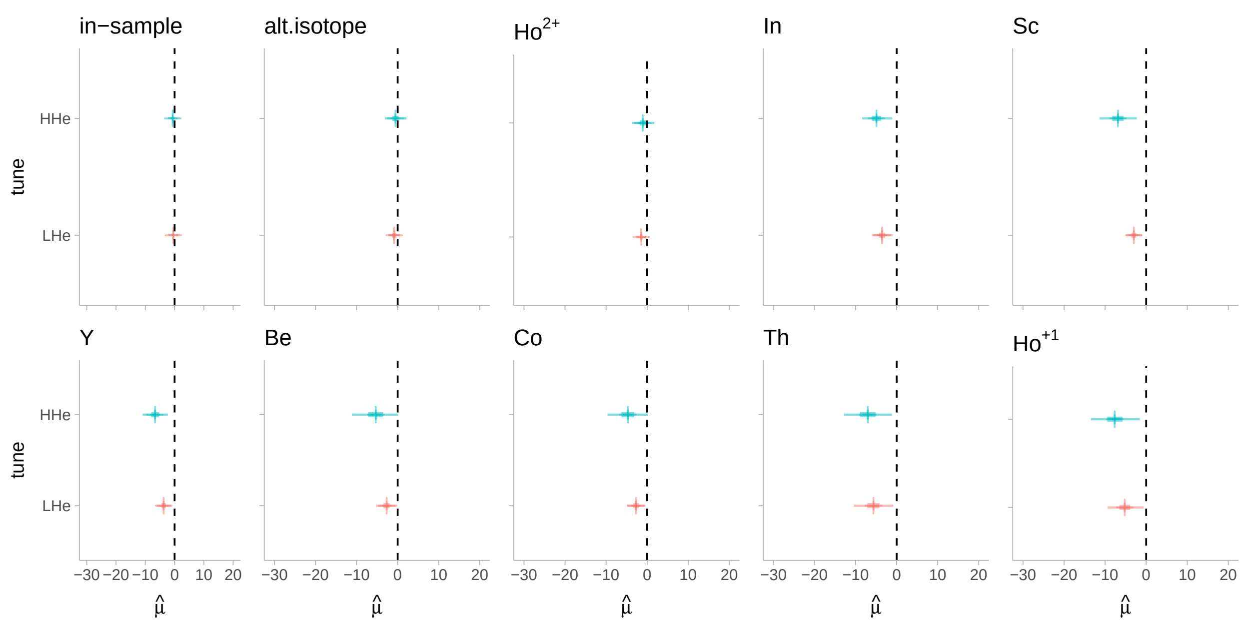
Conditional means

The conditional means estimated for the selenium portion of the experiment are below.

μ First, the estimated conditional means for the μ component of the model.

Tune Estimate means conditional on method and tune while marginalizing over matrix and day.

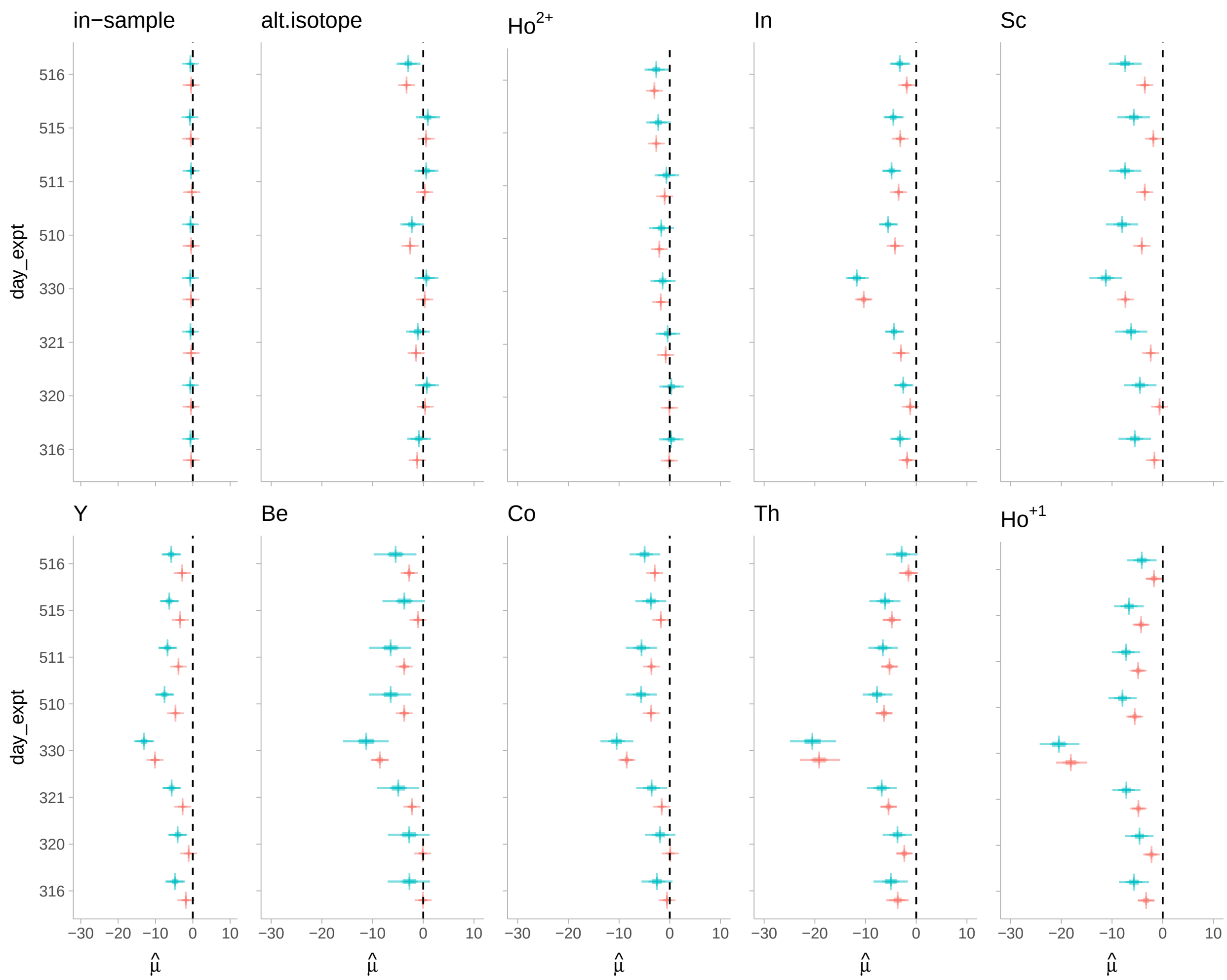
```
load("full-analysis-files/df_mv_se.rda")
load("full-analysis-files/mod3_Se_mv.rda")
fitted_method_tune_se <- df_mv_se %>%
  add_fitted_draws(mod3,
                   dpar = FALSE,
                   re_formula = NA,
                   cores = 1)
save(fitted_method_tune_se, file = "full-analysis-files/fitted_method_tune_se.rda")
```



Day Estimate means conditional on method and day after marginalizing over matrix.

```
load("full-analysis-files/df_mv_se.rda")
load("full-analysis-files/mod3_Se_mv.rda")
fitted_method_day_se <- df_mv_se %>%
  add_fitted_draws(mod3,
                   dpar = FALSE,
                   re_formula = ~ (1 | day_expt),
```

```
                    cores = 1)
save(fitted_method_day_se, file = "full-analysis-files/fitted_method_day_se.rda")
```



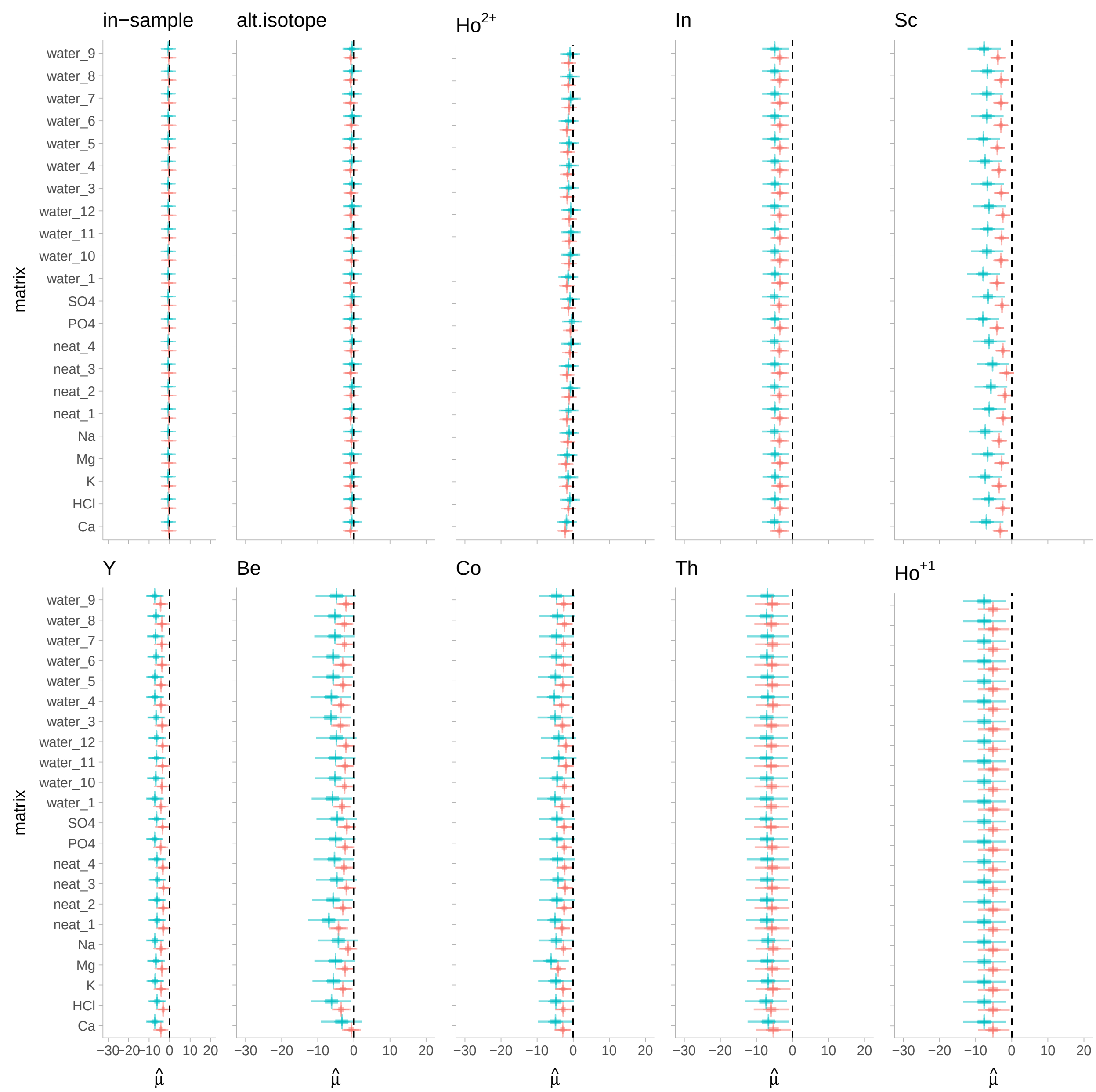
This figure illustrates the consistent over-corrections (i.e, negative bias) expected for all of the +1 methods across all days for both tune settings, with some exceptions here and there (e.g., *Sc*, HHe, 3/20). For those methods, 3/30 also stands out with even more extreme over-corrections. By comparison, the means for the +2 methods are relatively consistent across the days of the experiment, particularly for the in-sample method. Also note that, relative to the arsenic estimates, the +2 methods tend to have fewer estimated under-corrections (i.e., positive bias) estimated across days, and the alternative isotope and Ho^{+2} methods actually are estimated to have more over-corrections. Again, the in-sample method is generally estimated to be unbiased on all days for both tune settings.

The tune effects, overall, look negligible within days for all of the +2 methods. There may be some differences due to tune setting for the +1 methods, but those differences would be fairly hard to ascertain with much certainty.

Matrix Estimate means conditional on method and matrix after marginalizing over day.

```r
load("full-analysis-files/df_mv_se.rda")
load("full-analysis-files/mod3_Se_mv.rda")

fitted_method_matrix_se <- df_mv_se %>%
  add_fitted_draws(mod3,
                   dpar = FALSE,
                   re_formula = ~ (1 | matrix),
                   cores = 1)
save(fitted_method_matrix_se, file = "full-analysis-files/fitted_method_matrix_se.rda")
```



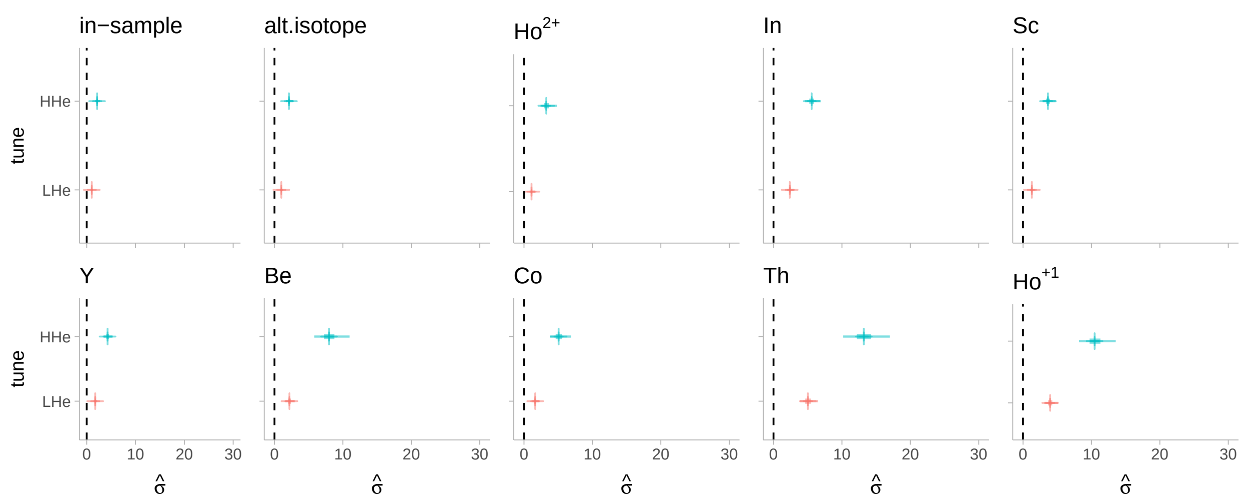
The estimates vary little by matrix for all of the methods. Again, the general pattern of over-correction for the +1 methods and minimal bias for the +2 methods is clear. Again, the tune effects look to be uncertain to minimal.

σ Next, the conditional means for the σ component of the model.

```
load("full-analysis-files/df_mv_se.rda")
load("full-analysis-files/mod3_Se_mv.rda")

fitted_sigma_tune_se <- df_mv_se %>%
  add_fitted_draws(mod3,
                   dpar = "sigma",
                   re_formula = NA,
                   cores = 1)
save(fitted_sigma_tune_se, file = "full-analysis-files/fitted_sigma_tune_se.rda")
```

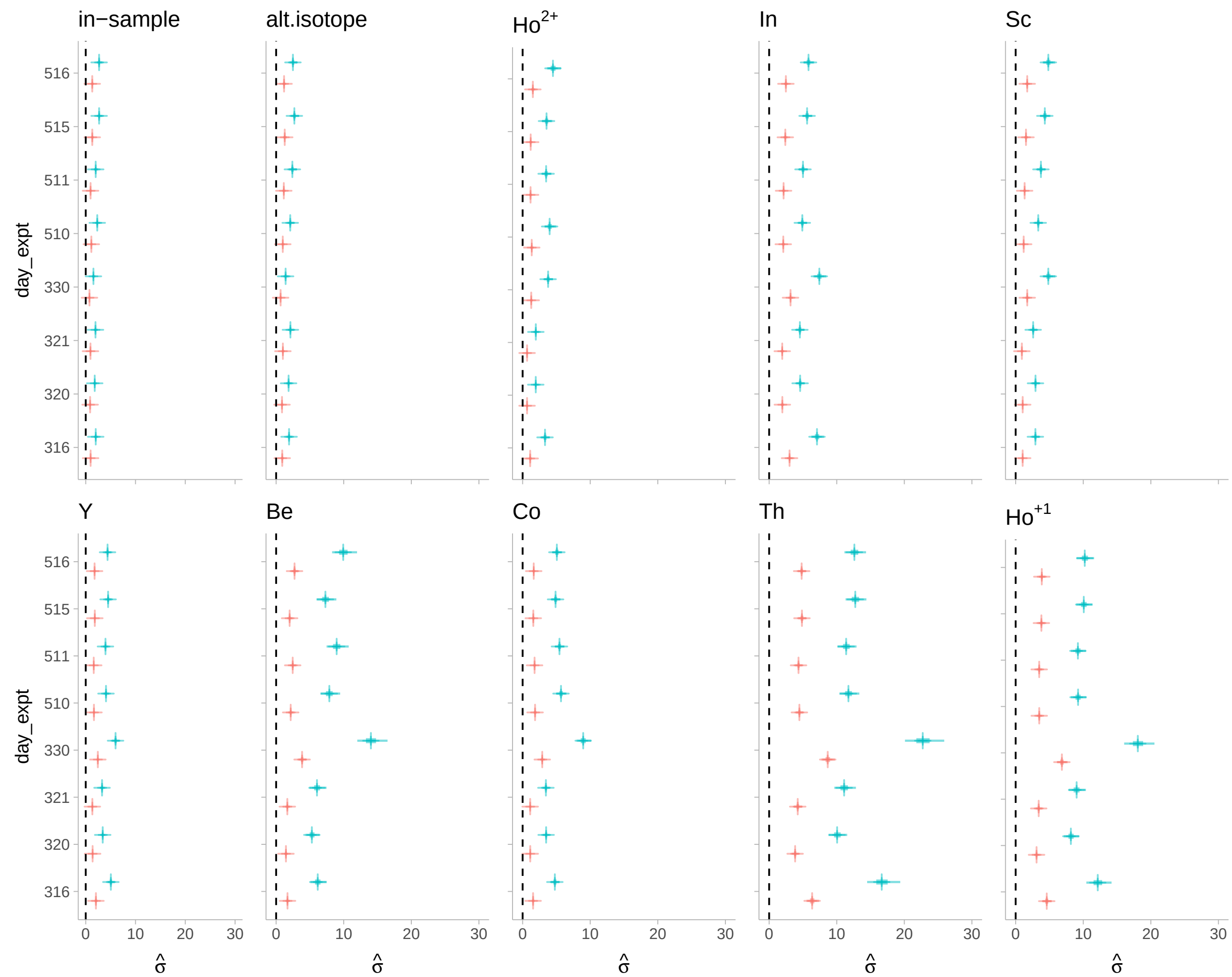
Tune The expected standard deviations by day after marginalizing over matrix are below.




```
load("full-analysis-files/df_mv_se.rda")
load("full-analysis-files/mod3_Se_mv.rda")

fitted_sigma_day_se <- df_mv_se %>%
  add_fitted_draws(mod3,
                   dpar = "sigma",
                   re_formula = sigma ~ (1 | day_expt),
                   cores = 1)
save(fitted_sigma_day_se, file = "full-analysis-files/fitted_sigma_day_se.rda")
```

Day The expected standard deviations by day after marginalizing over matrix are below.

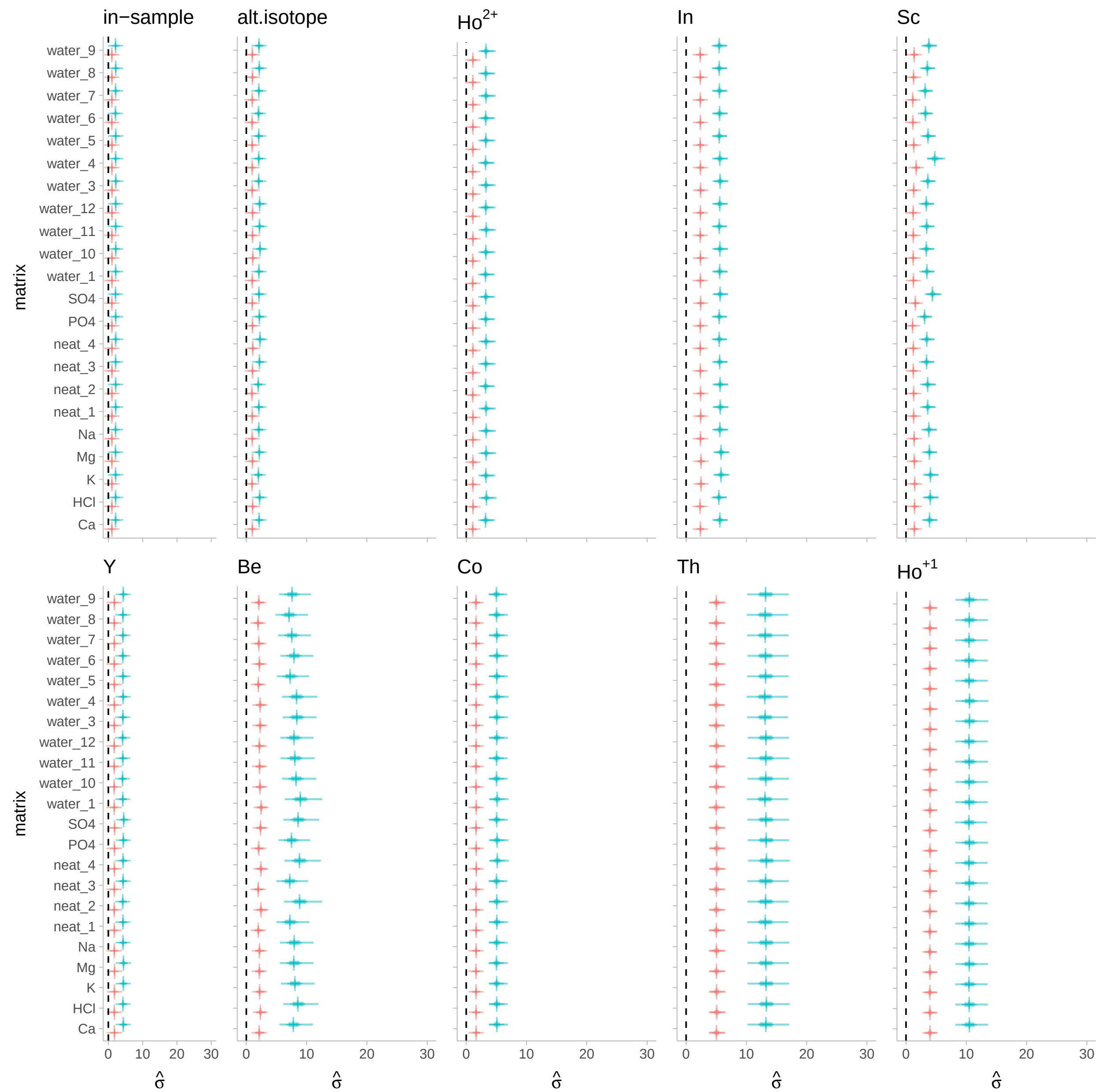


Standard deviation is estimated to vary considerably from day to day for some of the +1 methods. Much of that variability, however, is attributed to the large deviation on 3/30. Interestingly, and in contrast to the arsenic data, that 3/30 effect isn't indicated for In, Sc and Y. Standard deviation didn't clearly vary by day for the +2 methods either. This figure does suggest some clear differences in standard deviation due to tune setting, particularly for the +1 methods; and possibly for the Ho^{2+} method.

```
load("full-analysis-files/df_mv_se.rda")
load("full-analysis-files/mod3_Se_mv.rda")

fitted_sigma_matrix_se <- df_mv_se %>%
  add_fitted_draws(mod3,
                   dpar = "sigma",
                   re_formula = sigma ~ (1 | matrix),
                   cores = 1)
save(fitted_sigma_matrix_se, file = "full-analysis-files/fitted_sigma_matrix_se.rda")
```

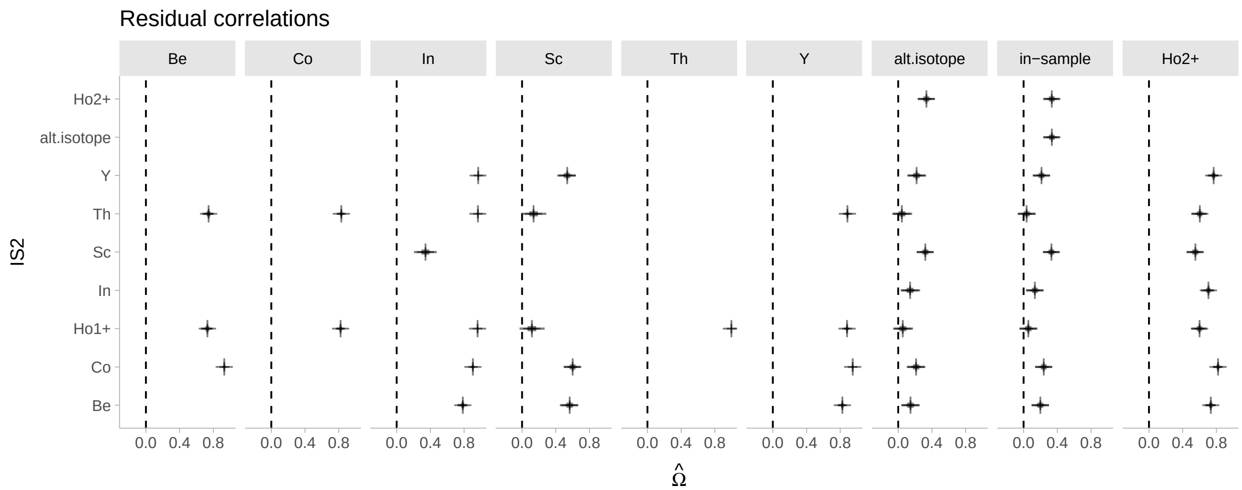
Matrix The expected standard deviations by matrix while marginalizing over day.



Overall, excepting perhaps the *Sc* and *Be* methods, matrix to matrix variability in standard deviation was estimated to be fairly negligible. The tune effects on standard deviation are clear in this figure, which depended on the method.

Residual Correlations

Next, the estimated residual correlations among IS methods for the selenium model.

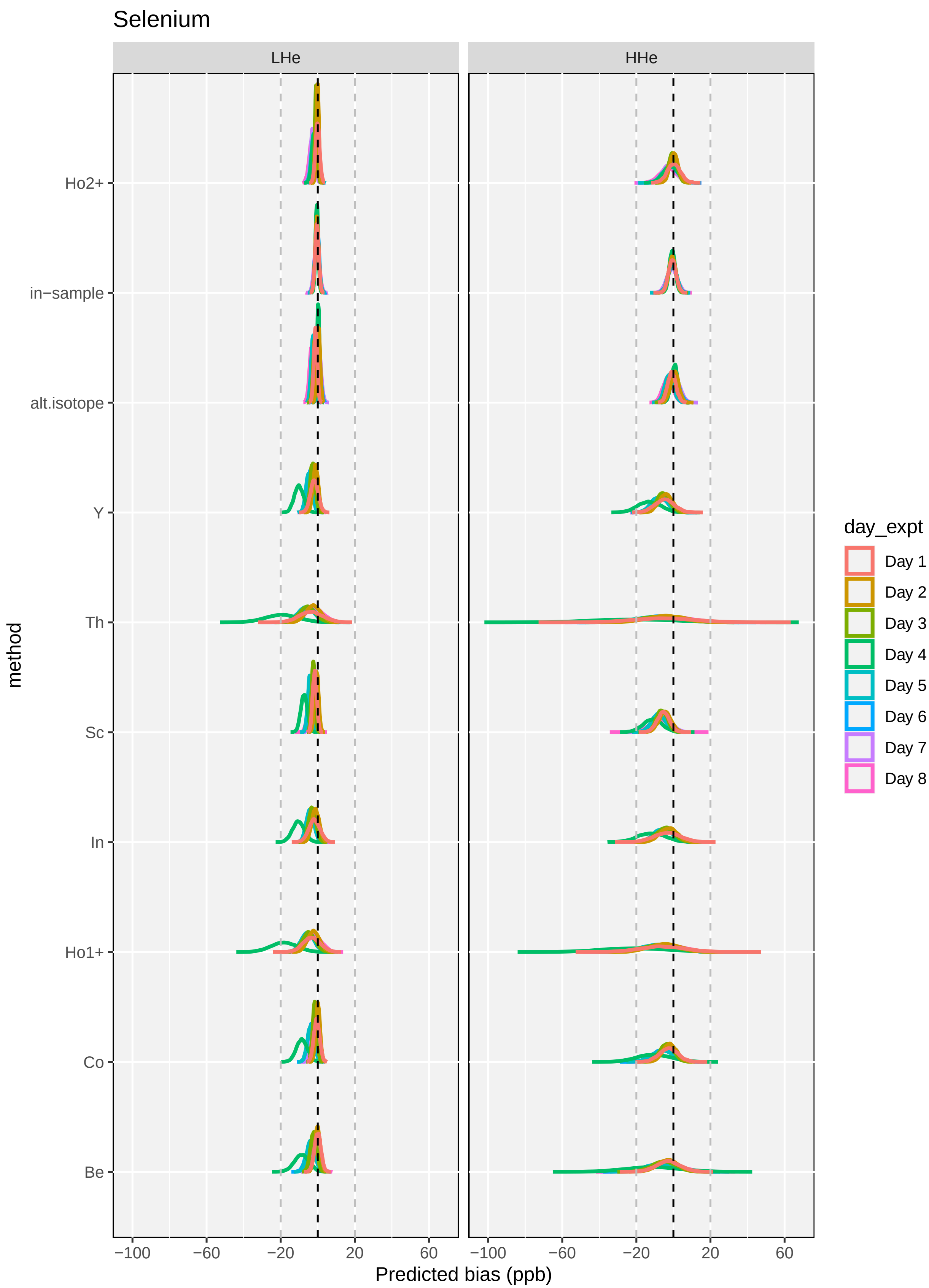


The patterns in the residual correlations are very similar to the patterns with arsenic above. The correlations are generally stronger between the +1 methods. However, it is notable that the correlations between the in-sample and alt.isotope methods and the others are considerably smaller for the selenium model compared to the one for arsenic, although the models are the same structurally.

Predictions

Finally, the predictions for the selenium model.

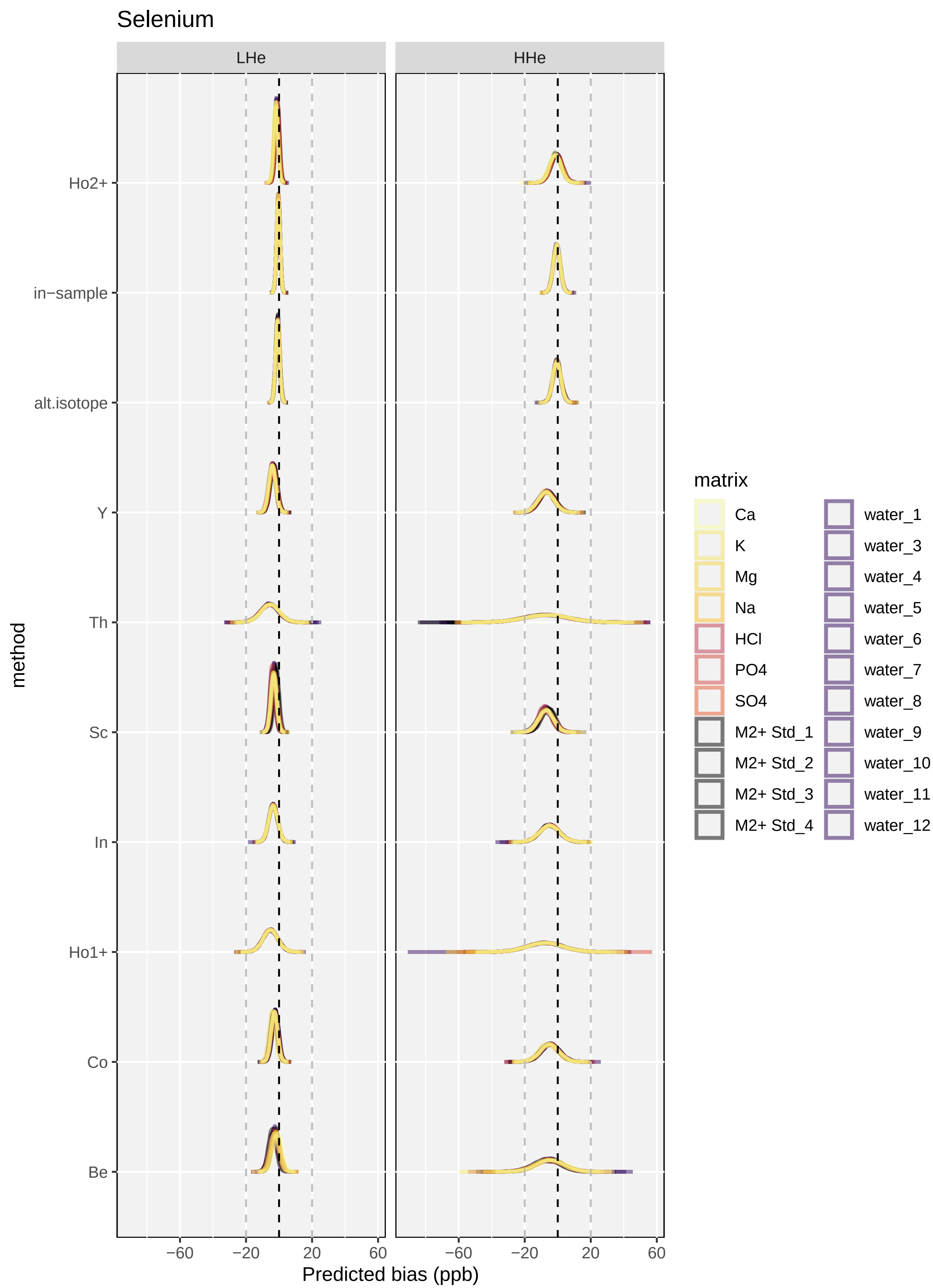
Day Predictions to the observed days for a new matrix.



Selenium
LHe
HHe
Ho2+
in−sample
alt.isotope
Y
Th
Sc
In
Ho1+
Co
Be
method
−100
−60
−20
20
60
Predicted bias (ppb)
day_expt
Day 1
Day 2
Day 3
Day 4
Day 5
Day 6
Day 7
Day 8

Clearly the in-sample method is predicted to have the least bias, on average, and the least variation across and within days. There is no indication of important day to day variability for that method. The alt.isotope and Ho^{+2} methods are also predicted to be reasonably unbiased and consistent, but some day to day variation in bias is apparent, if small. Across all of the +2 methods, bias was predicted with far greater precision for LHe tune compared to the HHe tune.

Matrix The predictions to the observed matrices while marginalizing the day effects are below.

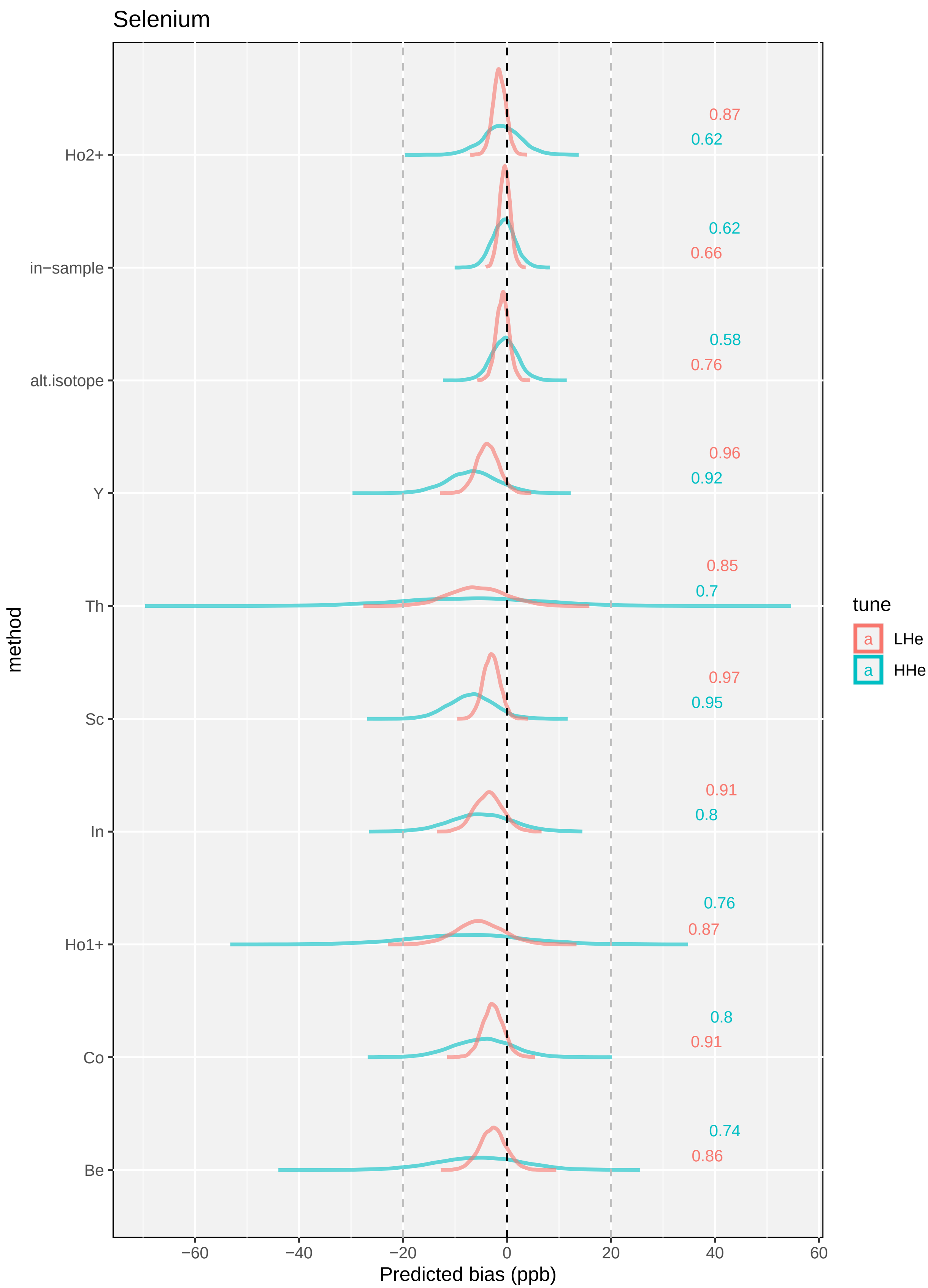

Selenium
LHe
HHe
method
Ho2+
in−sample
alt.isotope
Y
Th
Sc
In
Ho1+
Co
Be
−60
−20
20
60
Predicted bias (ppb)
matrix
Ca
K
Mg
Na
HCl
PO4
SO4
M2+ Std_1
M2+ Std_2
M2+ Std_3
M2+ Std_4
water_1
water_3
water_4
water_5
water_6
water_7
water_8
water_9
water_10
water_11
water_12

Matrix to matrix variability is predicted to be largely negligible for most methods. This is a bit of a contrast to the arsenic predictions, which suggested a more variation across matrices, particularly in the 250ppm matrices vs. others.

New day and matrix Finally, the predictions to a new matrix and new day from the selenium model.

The probability of over-correction for each case can be calculated from the posterior predictive distribution.

```
## # A tibble: 20 x 4
## # Groups:   .category [10]
##    .category method      tune  p_over
##    <fct>     <fct>       <fct>  <dbl>
##  1 Std       in-sample   LHe     0.66
##  2 Std       in-sample   HHe     0.62
##  3 Alt       alt.isotope LHe     0.76
##  4 Alt       alt.isotope HHe     0.58
##  5 Ho2       Ho2+        LHe     0.87
##  6 Ho2       Ho2+        HHe     0.62
##  7 In        In          LHe     0.91
##  8 In        In          HHe     0.8
##  9 Sc        Sc          LHe     0.97
## 10 Sc        Sc          HHe     0.95
## 11 Y         Y           LHe     0.96
## 12 Y         Y           HHe     0.92
## 13 Be        Be          LHe     0.86
## 14 Be        Be          HHe     0.74
## 15 Co        Co          LHe     0.91
## 16 Co        Co          HHe     0.8
## 17 Th        Th          LHe     0.85
## 18 Th        Th          HHe     0.7
## 19 Ho1       Ho1+        LHe     0.87
## 20 Ho1       Ho1+        HHe     0.76
```


Selenium
0.87
0.62
0.62
0.66
0.58
0.76
0.96
0.92
0.85
0.7
0.97
0.95
0.91
0.8
0.76
0.87
0.8
0.91
0.74
0.86
Ho2+
in−sample
alt.isotope
Y
Th
Sc
In
Ho1+
Co
Be
method
−60
−40
−20
0
20
40
60
Predicted bias (ppb)
tune
LHe
HHe

This figure again emphasizes differences in precision between the predictions for the +2 and +1 methods as well as the two tune settings. Compared to the model for arsenic, this one for selenium allocates more probability towards over-corrections for the +2 internal standard methods; whereas for the arsenic model, all +2 methods were predicted with most of the probability in the region of under-correction. The predictions for the +1 methods are largely in the direction of over-correction.

Overall, the in-sample method predictions compare favorably compared to most other methods and tune settings. The expected bias is nearer zero and the uncertainty in the predictions are considerably smaller, even for the HHe tune setting.

References

Bürkner, Paul-Christian. 2017. "Brms: An R Package for Bayesian Multilevel Models Using Stan." Journal Article. *2017* 80 (1): 28. https://doi.org/10.18637/jss.v080.i01.

Gabry, Jonah, Daniel Simpson, Aki Vehtari, Michael Betancourt, and Andrew Gelman. 2019. "Visualization in Bayesian Workflow." Journal Article. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 182 (2): 389–402. https://doi.org/10.1111/rssa.12378.

Stan Development Team. 2018a. "RStan: The R Interface to Stan." Journal Article. http://mc-stan.org.

———. 2018b. "Stan Modeling Language Users Guide and Reference Manual, Version 2.18.0." Journal Article. http://mc-stan.org.

———. 2018c. "The Stan Core Library, Version 2.18.0." Journal Article. http://mc-stan.org.

Vehtari, Aki, Andrew Gelman, and Jonah Gabry. 2017. "Practical Bayesian Model Evaluation Using Leave-One-Out Cross-Validation and Waic." Journal Article. *Statistics and Computing* 27 (5): 1413–32. https://doi.org/10.1007/s11222-016-9696-4.

Session Info

```r
sessionInfo()
```

```
## R version 4.0.5 (2021-03-31)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 18363)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] brms_2.16.1          Rcpp_1.0.7           ggrepel_0.9.1
```

```
##  [4] ggdist_3.0.0         tidybayes_3.0.1      bayesplot_1.8.1
##  [7] loo_2.4.1            rstan_2.21.2         StanHeaders_2.21.0-7
## [10] forcats_0.5.1        stringr_1.4.0        dplyr_1.0.7
## [13] purrr_0.3.4          readr_2.1.0          tidyr_1.1.4
## [16] tibble_3.1.6         tidyverse_1.3.1      readxl_1.3.1
## [19] gridExtra_2.3        ggExtra_0.9          ggplot2_3.3.5
##
## loaded via a namespace (and not attached):
##   [1] minqa_1.2.4          colorspace_2.0-2     ellipsis_0.3.2
##   [4] ggridges_0.5.3       rsconnect_0.8.25     markdown_1.1
##   [7] base64enc_0.1-3      fs_1.5.0             rstudioapi_0.13
##  [10] farver_2.1.0         DT_0.20              svUnit_1.0.6
##  [13] fansi_0.5.0          mvtnorm_1.1-3        lubridate_1.8.0
##  [16] xml2_1.3.2           splines_4.0.5        bridgesampling_1.1-2
##  [19] codetools_0.2-18     knitr_1.36           shinythemes_1.2.0
##  [22] projpred_2.0.2       jsonlite_1.7.2       nloptr_1.2.2.3
##  [25] broom_0.7.10         dbplyr_2.1.1         shiny_1.7.1
##  [28] compiler_4.0.5       httr_1.4.2           backports_1.3.0
##  [31] assertthat_0.2.1     Matrix_1.3-4         fastmap_1.1.0
##  [34] cli_3.1.0            later_1.3.0          htmltools_0.5.2
##  [37] prettyunits_1.1.1    tools_4.0.5          igraph_1.2.8
##  [40] coda_0.19-4          gtable_0.3.0         glue_1.5.0
##  [43] reshape2_1.4.4       posterior_1.1.0      V8_3.6.0
##  [46] cellranger_1.1.0     vctrs_0.3.8          nlme_3.1-153
##  [49] crosstalk_1.2.0      tensorA_0.36.2       xfun_0.28
##  [52] ps_1.6.0             lme4_1.1-27.1        rvest_1.0.2
##  [55] mime_0.12            miniUI_0.1.1.1       lifecycle_1.0.1
##  [58] gtools_3.9.2         zoo_1.8-9            MASS_7.3-54
##  [61] scales_1.1.1         colourpicker_1.1.1   hms_1.1.1
##  [64] promises_1.2.0.1     Brobdingnag_1.2-6    parallel_4.0.5
##  [67] inline_0.3.19        shinystan_2.5.0      gamm4_0.2-6
##  [70] yaml_2.2.1           curl_4.3.2           stringi_1.7.5
##  [73] dygraphs_1.1.1.6     checkmate_2.0.0      boot_1.3-28
##  [76] pkgbuild_1.2.0       rlang_0.4.12         pkgconfig_2.0.3
##  [79] matrixStats_0.61.0   distributional_0.2.2 evaluate_0.14
##  [82] lattice_0.20-45      labeling_0.4.2       htmlwidgets_1.5.4
##  [85] rstantools_2.1.1     processx_3.5.2       tidyselect_1.1.1
##  [88] plyr_1.8.6           magrittr_2.0.1       R6_2.5.1
##  [91] generics_0.1.1       DBI_1.1.1            mgcv_1.8-38
##  [94] pillar_1.6.4         haven_2.4.3          withr_2.4.2
##  [97] xts_0.12.1           abind_1.4-5          modelr_0.1.8
## [100] crayon_1.4.2         arrayhelpers_1.1-0   utf8_1.2.2
## [103] tzdb_0.2.0           rmarkdown_2.11       grid_4.0.5
## [106] callr_3.7.0          threejs_0.3.3        reprex_2.0.1
## [109] digest_0.6.28        xtable_1.8-4         httpuv_1.6.3
## [112] RcppParallel_5.1.4   stats4_4.0.5         munsell_0.5.0
## [115] shinyjs_2.0.0
```