

Data Analysis Assignment

5637058

General Guidelines

There are two tasks. Your answers should have *two separate sections for each task*, one immediately after the other.

In the first section, write out your answers using complete sentences, as you would for the results section of a paper. Include descriptive statistics in the text, tables or figures, as appropriate. Tables and figures should be of publication quality (i.e., fully labelled, etc.). Integrate inferential statistics into your description of the results. Your answers might be quite short. **Given the validity of the statistical analysis, the first section will play the main role for your mark.**

The second section should include the complete R code that you used and its output. Add comments (using a #) to explain what the code does. The code should show all of the commands that you used; enough to replicate exactly what you did (I will be copying and pasting code to run checks, so make sure that works). You can include additional figures in the second section that you used to explore the data, which you do not wish to include in the first section. I will use the second section to help identify the source of any mistakes. For practical reports and papers you would only submit the first section; thus the first section should stand alone without requiring the reader to refer to the second section.

Libraries

```
library(tidyverse)
library(afex)
library(emmeans)
library(BayesFactor)
library(bayestestR)
library(insight)
library(rstanarm)
```

TASK 2 STARTS ON PAGE 33

Task 1

Decision making is fundamentally noisy, and experimental participants will often make different choices between the same two risky alternative even during the same experimental session. An interesting question is why choices are inconsistent: is it because participants' decisions are just inherently noisy? Or perhaps is it because participants' deterministic preferences are dynamically changing throughout the experiment, and so they only appear to be stochastic?

To test whether participants' preferences are dynamically changing it is possible to manipulate the amount of time (i.e., number of intervening trials) between repeated choices. If participants' preferences are dynamically changing, we would expect that for shorter intervals between repeated choices that participants would make more consistent decisions compared to the consistency for longer intervals.

The file `consistency.csv` contains simulated data about choice consistency, where participants were split at random to repeat choices at different intervals (variable `Interval`), where the interval is the number of intervening trials. Participants (variable `id`) completed a long series of choices between two alternatives with the repetitions of various alternative pairs hidden within this series. The variable `Consistency` gives the proportion of repeated choices for which a participant made the same choice across the two repetitions.

Please analyse the data using both frequentist and Bayesian linear regression models. The main research question is whether choice consistency decreases with longer intervals. For this task only, please implement your Bayesian linear regression using a variety of R-based implementations and priors, including AIC and BIC, and compare your results for robustness across these implementations and priors.

Present the results as you would in a paper, using APA format. In other words, describe your statistical model and results and describe the conclusions with regards to the research questions. In the first section of your answer for this task, include only one figure (which may contain multiple panels), with an appropriate figure caption.

Part 1: Results Description

An experiment was conducted to investigate whether a person's decision would change when presented with two alternatives, with a varying interval size (i.e., number of intervening trials) between them. The core research question is to explore whether choice consistency (i.e., participant choosing the same option) decreases as interval size increases.

The relationship between interval size and the proportion of choice consistency ($M = .849$, $SD = .154$) across both alternatives was analysed using both frequentist and Bayesian linear regression models. Interval size was not a significant predictor of choice consistency with the Bayes factor suggesting there is no evidence for this alternate model ($\beta = -.00408$, $p = .0529$, $BF_{10} = 1.020$; Figure 1). Hence, the overall linear model was not significant ($F(1, 139) = 3.81$, $p = .0529$) where only 0.0197% of the variance in the proportion of choice consistency can be explained by the size of the interval (adjusted $R^2 = .01969$).

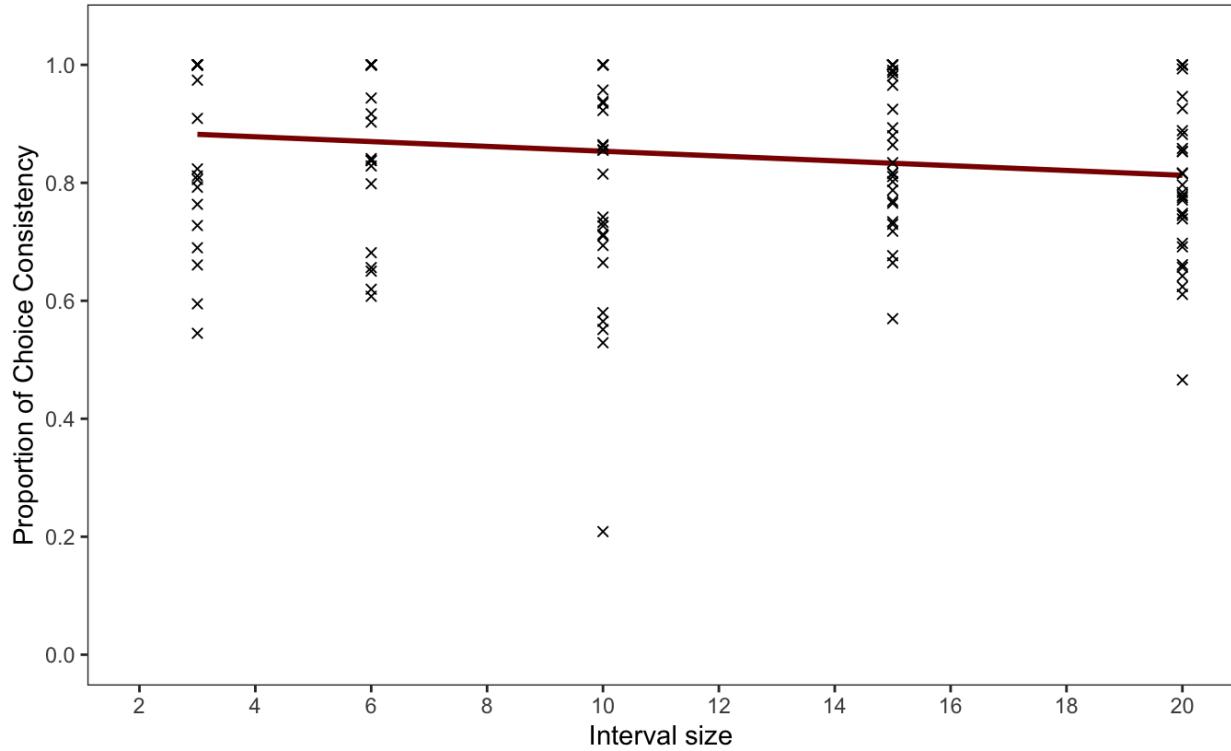


Figure 1. Regression model predicting Choice Consistency from Interval Size (dark red line). Every data point within the entire dataset is shown as a cross.

While the Bayes factor (from the `BayesFactor` library) above was based on a `medium` prior width (i.e., `rscaleCont = $\frac{\sqrt{2}}{4}$`), the results were slightly different from the `medium` prior with both `wide` (`rscaleCont = 0.5`, $BF_{10} = .793$) and `ultrawide` (`rscaleCont = $\frac{\sqrt{2}}{2}$` , $BF_{10} = .592$) priors suggesting anecdotal evidence for the null model. Nevertheless, the Bayes factors provided either no evidence or at most anecdotal evidence for the null model. This interpretation difference should not be a cause for concern as all three Bayes factors are relatively close to one another (.592 to 1.020). Overall, this regression model suggests that there is no significant relationship nor anymore than anecdotal evidence for the relationship between choice consistency made and interval size.

Moreover, HDI, ETI and ROPE plots were analysed for each `BayesFactor` prior. For all priors, visual observations of these diagrams align with the evidence showing almost no deviation from zero with the ROPE showing 100% of the posterior distribution fell within the set region indicating no effect.

Additionally, BF_{AIC} and BF_{BIC} scores were calculated for this model in comparison with the null model.

The BF_{AIC} was 2.477 suggesting anecdotal evidence for the alternate model while the BF_{BIC} was 0.567 suggesting anecdotal evidence for the null model. Nevertheless, neither provides sufficient evidence to support either the null or alternate model, coinciding with previous findings.

Finally, using the `rstanarm` library, this Bayesian linear model was fit to the data using Markov Chain Monte Carlo (MCMC) sampling to estimate the posterior distributions of the model parameters (e.g., the intercept and slope for Interval) using the data and six specified priors (0.25, 0.5, 0.75, 1, 5, 10). Visual comparison of the prior and posterior distributions highlights the posterior distribution remained at zero favouring the null model over the alternate model. Ultimately, these findings further evidence that choice consistency does not decrease with longer intervals because there is no statistical evidence supporting a relationship between choice consistency and interval size. However, there is some anecdotal evidence for the null model but this is not sufficiently evidenced to be a valid explanation.

Part 2: R Code

```

# Loading data
con_data <- read_csv("consistency.csv")

## New names:
## Rows: 141 Columns: 4
## -- Column specification
## -----
## (4): ...1, id, Interval, Consistency
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * ` ` -> `...1`

con_data %>% str()

## #> #> spc_tbl_ [141 x 4] (S3: spc_tbl_df/tbl_df/tbl/data.frame)
## #> $ ...1      : num [1:141] 1 2 3 4 5 6 7 8 9 10 ...
## #> $ id        : num [1:141] 1 2 3 4 5 6 7 8 9 10 ...
## #> $ Interval   : num [1:141] 3 3 3 3 3 3 3 3 3 3 ...
## #> $ Consistency: num [1:141] 0.806 0.824 0.545 1 1 ...
## #> - attr(*, "spec")=
## #>   .. cols(
## #>     .. ...1 = col_double(),
## #>     .. id = col_double(),
## #>     .. Interval = col_double(),
## #>     .. Consistency = col_double()
## #>   .. )
## #> - attr(*, "problems")=<externalptr>

con_data_tidy <- con_data %>%
  select(-c(...1))

con_data_tidy %>% head(10)

## #> #> # A tibble: 10 x 3
## #>   id   Interval Consistency
## #>   <dbl>    <dbl>       <dbl>
## #> 1 1       3       0.806
## #> 2 2       3       0.824
## #> 3 3       3       0.545
## #> 4 4       3       1
## #> 5 5       3       1
## #> 6 6       3       0.974
## #> 7 7       3       0.909
## #> 8 8       3       1
## #> 9 9       3       0.595
## #> 10 10     3       1

con_data_tidy %>%
  summarise(
    mean_interval = mean(Consistency),
    sd_interval = sd(Consistency)
  )

## #> #> # A tibble: 1 x 2
## #>   mean_interval sd_interval

```

```
##          <dbl>      <dbl>
## 1        0.849     0.154
```

Frequentist Analysis

```
# Linear model based on the entire dataset
rc_freq_lm <- lm(
  formula = Consistency ~ Interval,
  data = con_data_tidy
)

rc_freq_lm %>% summary()

##
## Call:
## lm(formula = Consistency ~ Interval, data = con_data_tidy)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.644494 -0.10338  0.03096  0.13009  0.18722
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.894397   0.026760 33.423   <2e-16 ***
## Interval    -0.004081   0.002090 -1.952   0.0529 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.152 on 139 degrees of freedom
## Multiple R-squared:  0.02669, Adjusted R-squared:  0.01969
## F-statistic: 3.811 on 1 and 139 DF, p-value: 0.05292

rc_freq_lm %>% confint()  # confidence intervals

##           2.5 %      97.5 %
## (Intercept) 0.841486830 9.473065e-01
## Interval    -0.008214065 5.210462e-05

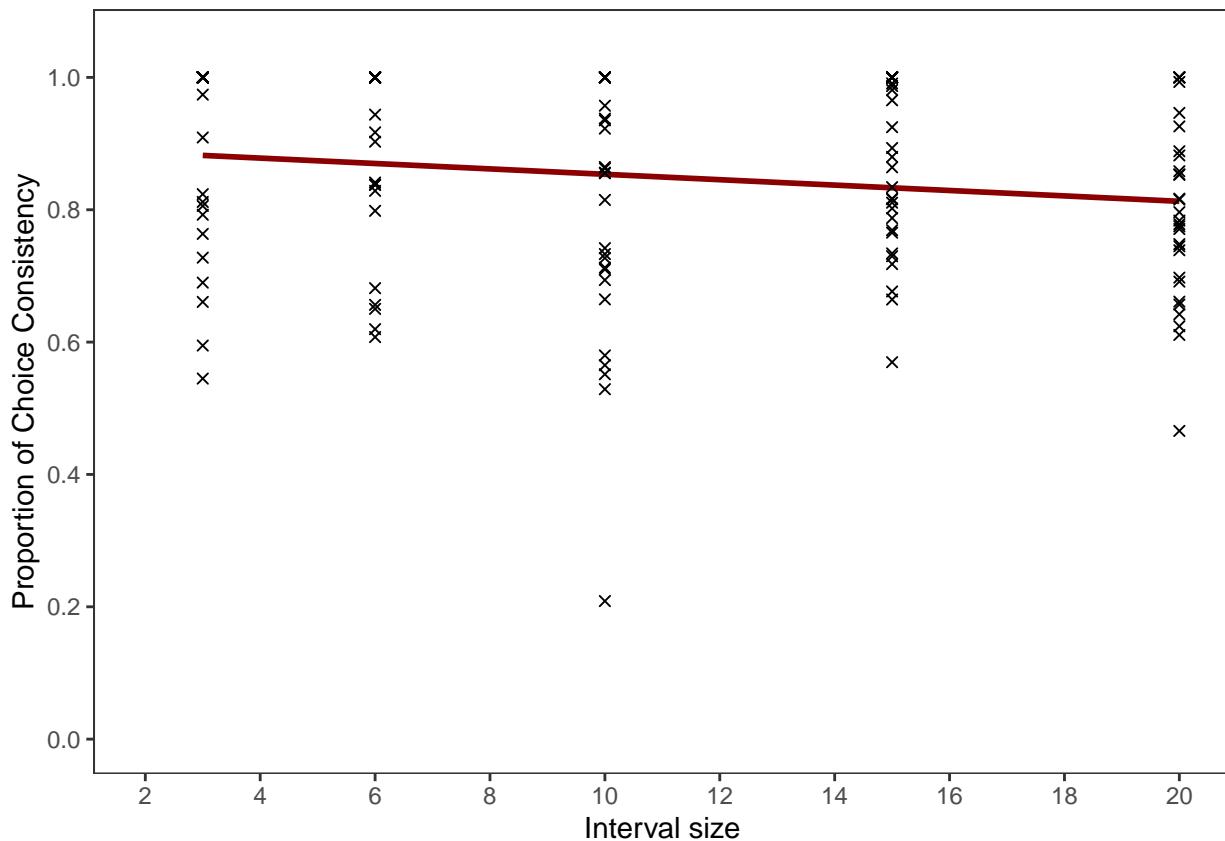
ggplot(con_data_tidy, aes(x = Interval, y = Consistency)) +
  geom_smooth(method = lm, se = FALSE, color = "darkred") +
  geom_point(pch = 4) +
  scale_x_continuous(
    limits = c(2, 20),
    breaks = seq(0, 20, by = 2)
  ) +
  scale_y_continuous(
    limits = c(0, 1.05),
    breaks = seq(0, 1, by = 0.2)
  ) +
  labs(
    x = "Interval size",
    y = "Proportion of Choice Consistency"
  ) +
  theme_bw() +
  theme(
```

```

    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
)

## `geom_smooth()` using formula = 'y ~ x'

```



Bayesian Analysis

```
prior_widths = prior_widths = c('medium', 'wide', 'ultrawide')

for (prior in prior_widths) {

  print("-----")
  print(paste("PRIOR: ", prior))

  # Bayesian linear regression with different prior widths
  con_bf_lm <- lmBF(
    formula = Consistency ~ Interval,
    data = con_data_tidy,
    rscaleCont = prior    # prior widths for continuous scales
  )

  print(con_bf_lm %>% summary())
  print("-----")

  # Highlight interested parameters (i.e., intercept and slope)
  con_bf_lm_samples <- get_parameters(
    con_bf_lm,
    effects = 'fixed'
  )

  con_bf_lm_parameters <- names(con_bf_lm_samples)[!grepl("g_*", names(con_bf_lm_samples))]

  # Highest density interval (HDI) of the posterior
  con_bf_lm_hdi <- hdi(con_bf_lm_samples[con_bf_lm_parameters])
  con_bf_lm_hdi_plot <- plot(con_bf_lm_hdi, con_bf_lm) + ggtitle(paste("HDI for", prior, "prior"))

  print(con_bf_lm_hdi)
  print(con_bf_lm_hdi_plot)
  print("-----")

  # Equal tailed interval (ETI) of the posterior
  con_bf_lm_eti <- eti(con_bf_lm_samples[con_bf_lm_parameters])

  con_bf_lm_eti_plot <- plot(con_bf_lm_eti, con_bf_lm) + ggtitle(paste("ETI for", prior, "prior"))

  print(con_bf_lm_eti)
  print(con_bf_lm_eti_plot)
  print("-----")

  # Identify the posterior probability within the region of practical equivalence (ROPE)
  con_bf_lm_rope <- rope(
    con_bf_lm_samples[con_bf_lm_parameters],
    range = rope_range(con_bf_lm)
  )

  con_bf_lm_rope_plot <- plot(
    con_bf_lm_rope,
    range = rope_range(con_bf_lm),
    con_bf_lm
```

```

) + ggtitle(paste("ROPE for", prior, "prior"))

print(con_bf_lm_rope)
print(con_bf_lm_rope_plot)
print("-----")

}

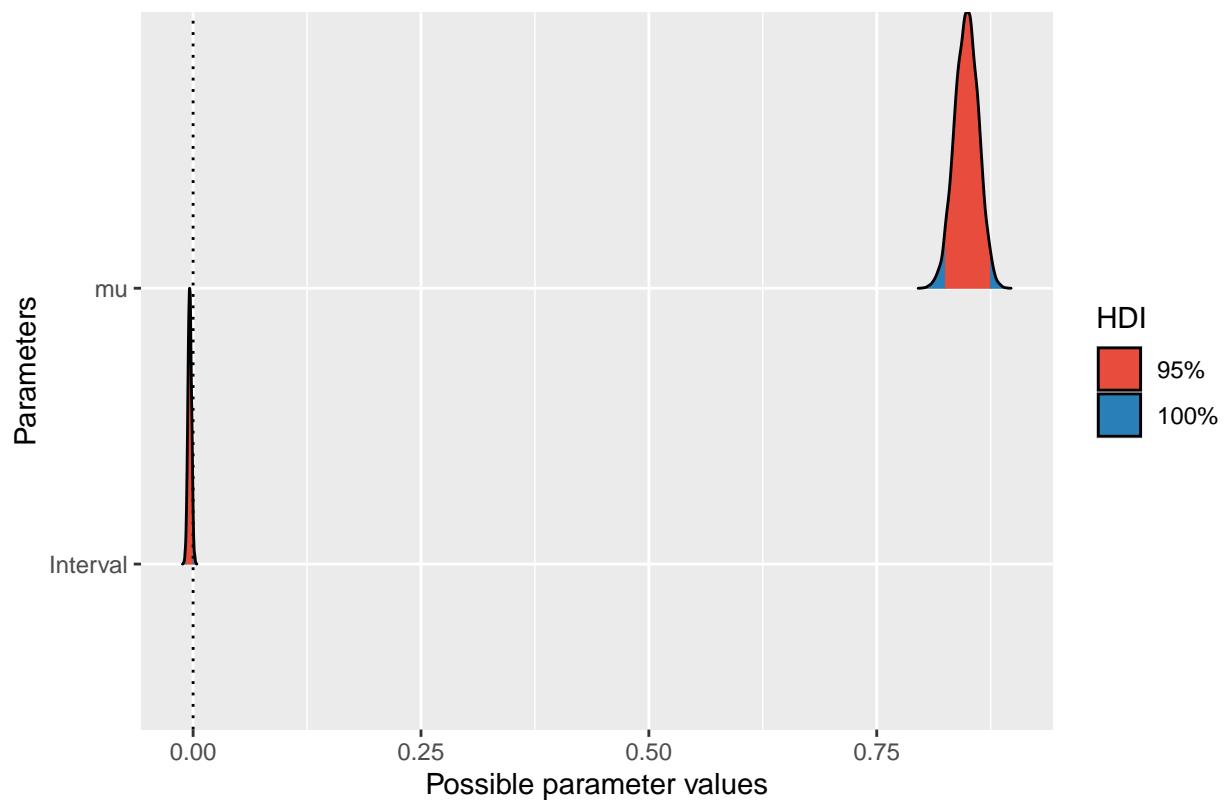
## [1] -----
## [1] "PRIOR: medium"

## Warning: data coerced from tibble to data frame

## Bayes factor analysis
## -----
## [1] Interval : 1.020306 ±0%
##
## Against denominator:
##   Intercept only
## ---
## Bayes factor type: BFlinearModel, JZS
##
## NULL
## [1] -----
## Highest Density Interval
##
## Parameter |      95% HDI
## -----
## mu        | [ 0.83,  0.87]
## Interval  | [-0.01,  0.00]

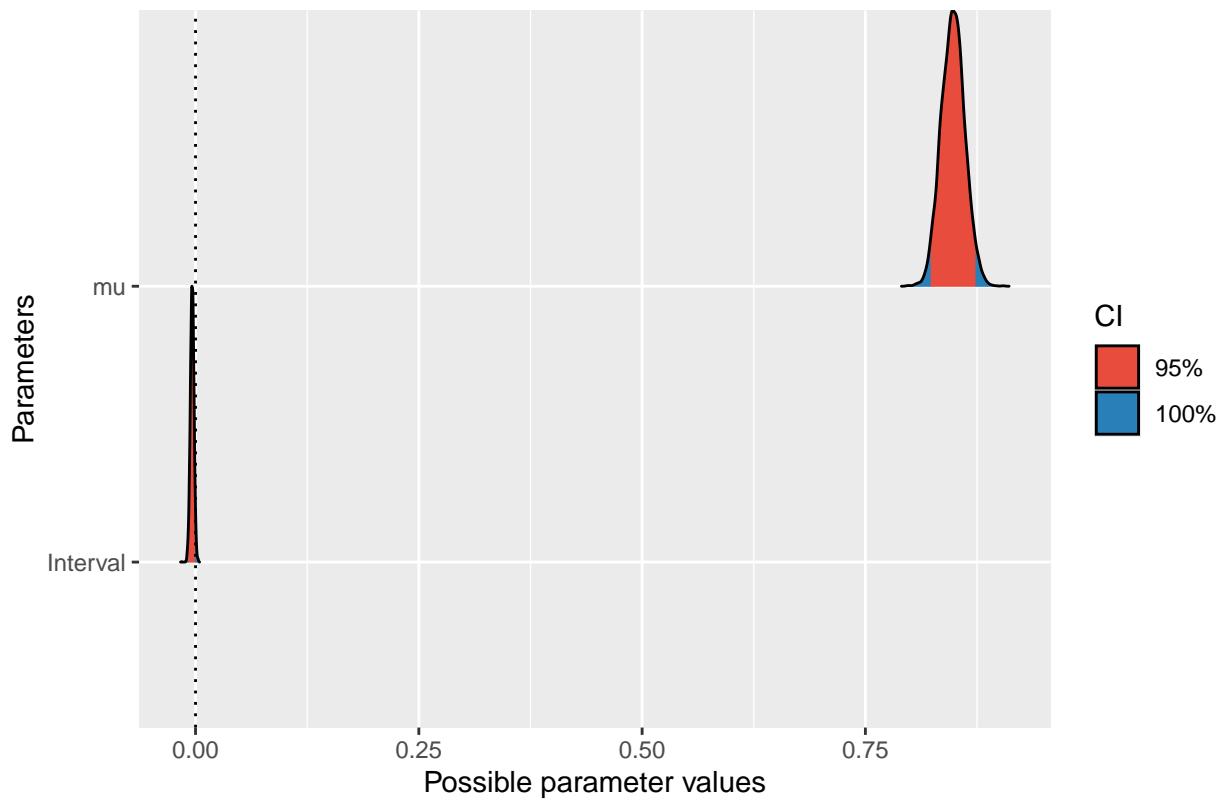
```

HDI for medium prior



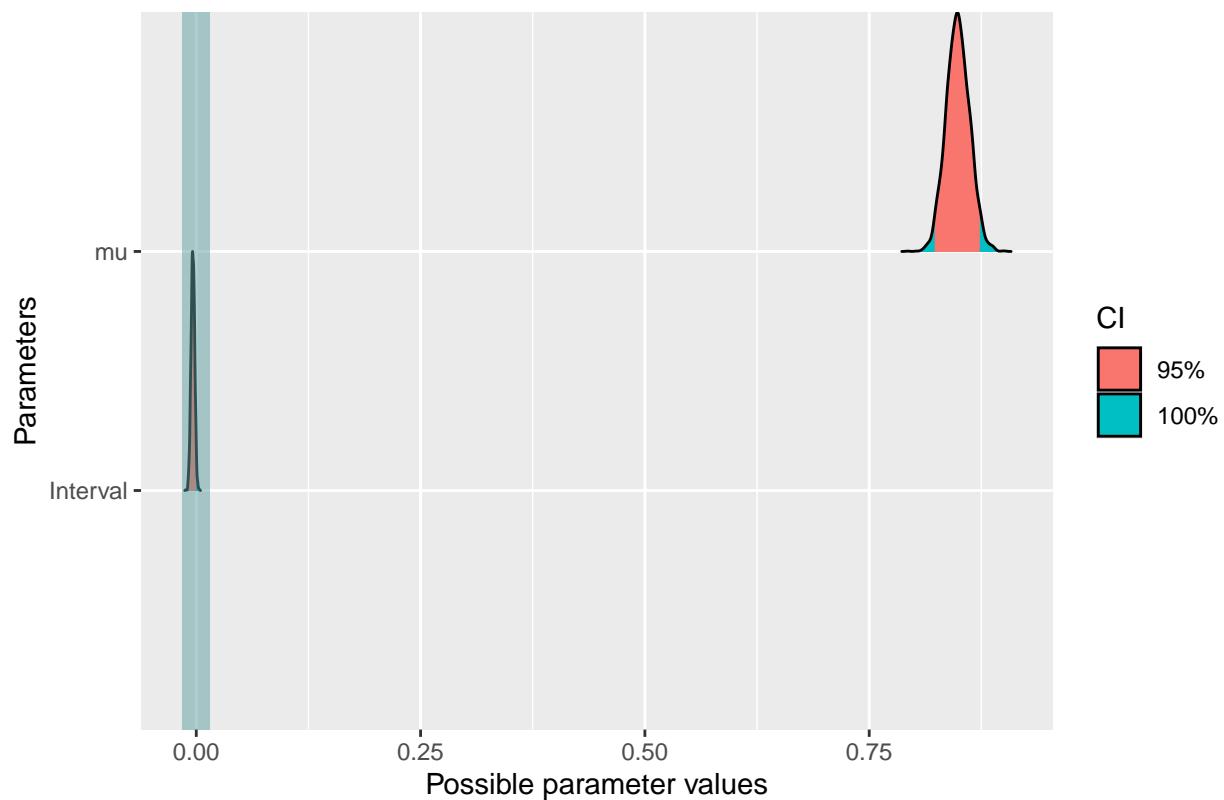
```
## [1] "-----"  
## Equal-Tailed Interval  
##  
## Parameter |      95% ETI  
## -----  
## mu        | [ 0.82, 0.87]  
## Interval   | [-0.01, 0.00]
```

ETI for medium prior



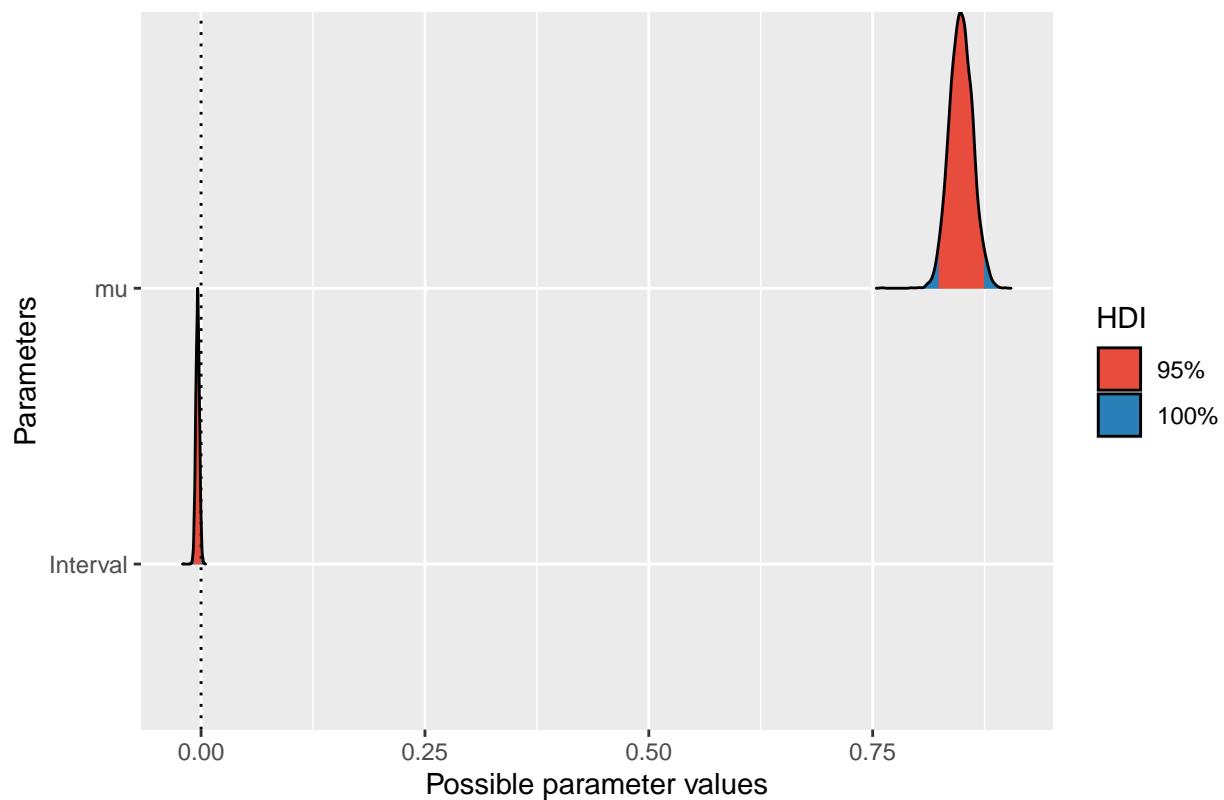
```
## [1] "-----"  
## # Proportion of samples inside the ROPE [-0.02, 0.02]:  
##  
## Parameter | inside ROPE  
## -----  
## mu        |      0.00 %  
## Interval   |    100.00 %
```

ROPE for medium prior



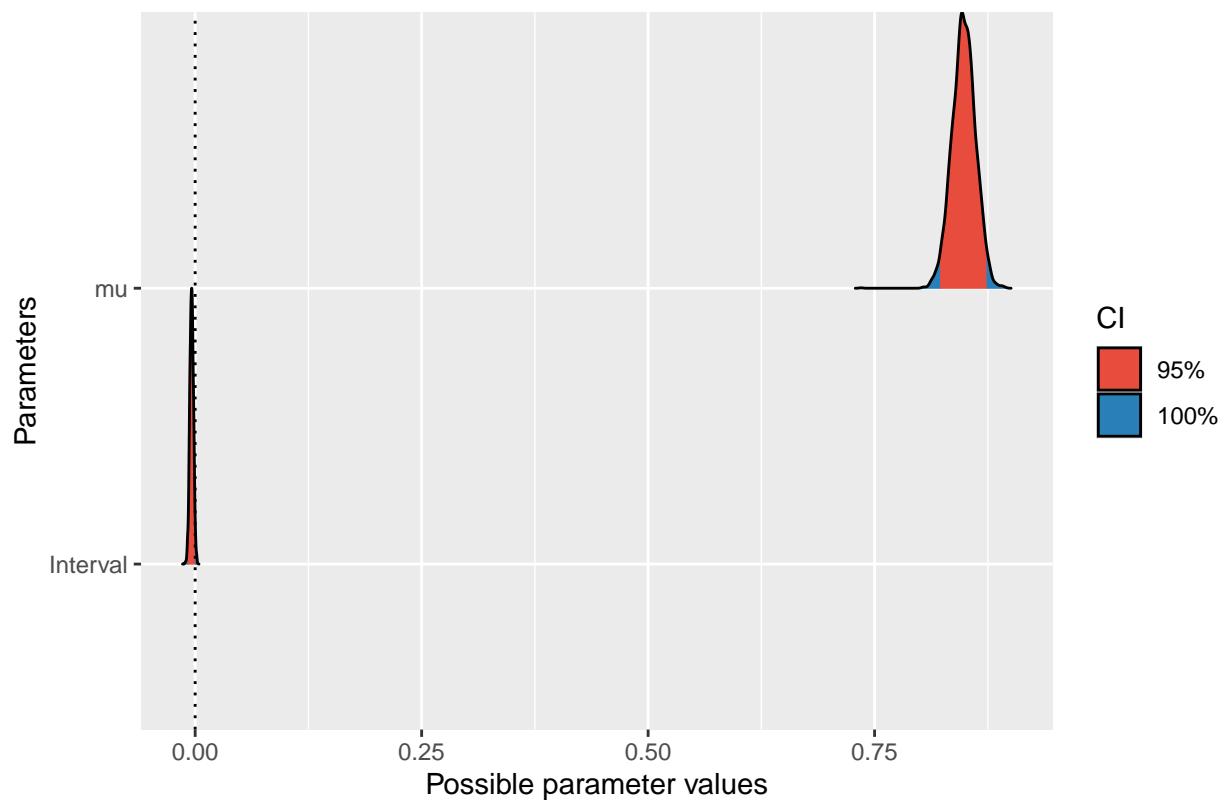
```
## [1] "-----"
## [1] "-----"
## [1] "PRIOR: wide"
## Warning: data coerced from tibble to data frame
## Bayes factor analysis
## -----
## [1] Interval : 0.7926663 ±0.01%
## 
## Against denominator:
##   Intercept only
## 
## Bayes factor type: BFlinearModel, JZS
## 
## NULL
## [1] "-----"
## Highest Density Interval
## 
## Parameter |      95% HDI
## -----
## mu        | [ 0.82, 0.87]
## Interval  | [-0.01, 0.00]
```

HDI for wide prior



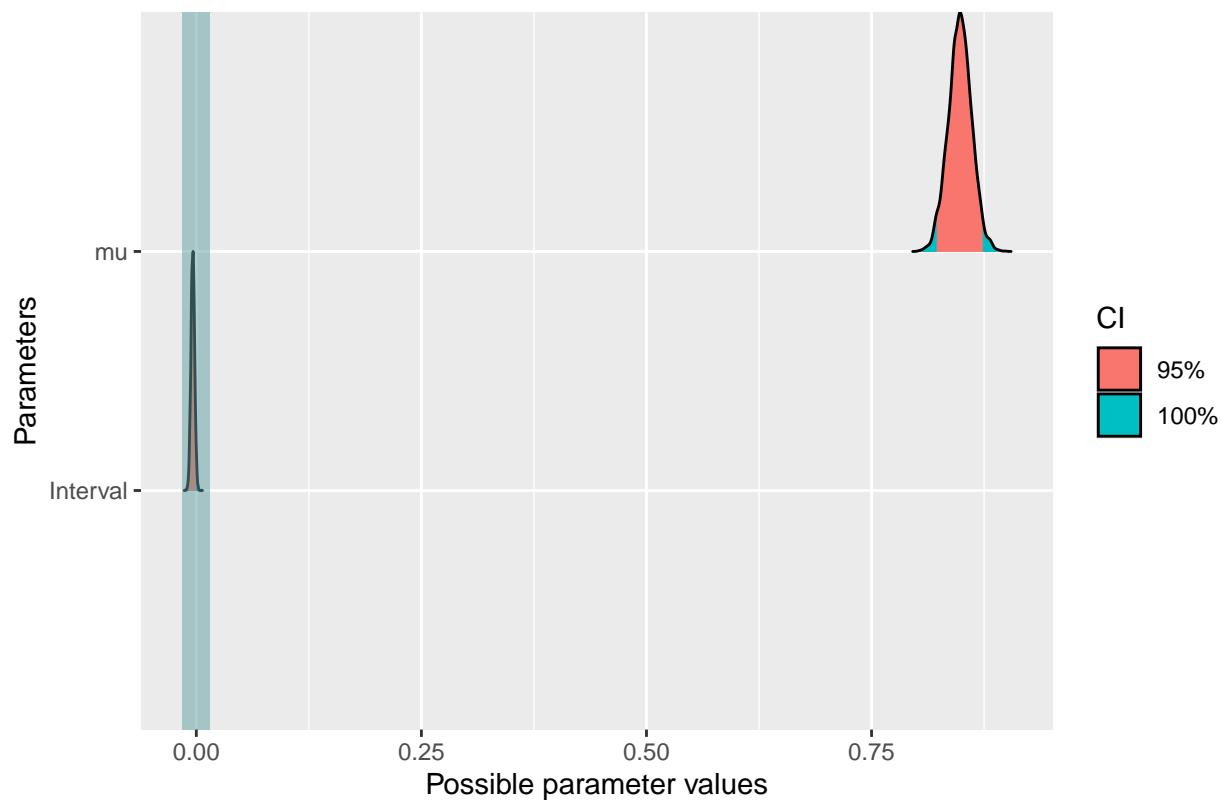
```
## [1] "-----"  
## Equal-Tailed Interval  
##  
## Parameter |      95% ETI  
## -----  
## mu        | [ 0.82, 0.87]  
## Interval   | [-0.01, 0.00]
```

ETI for wide prior



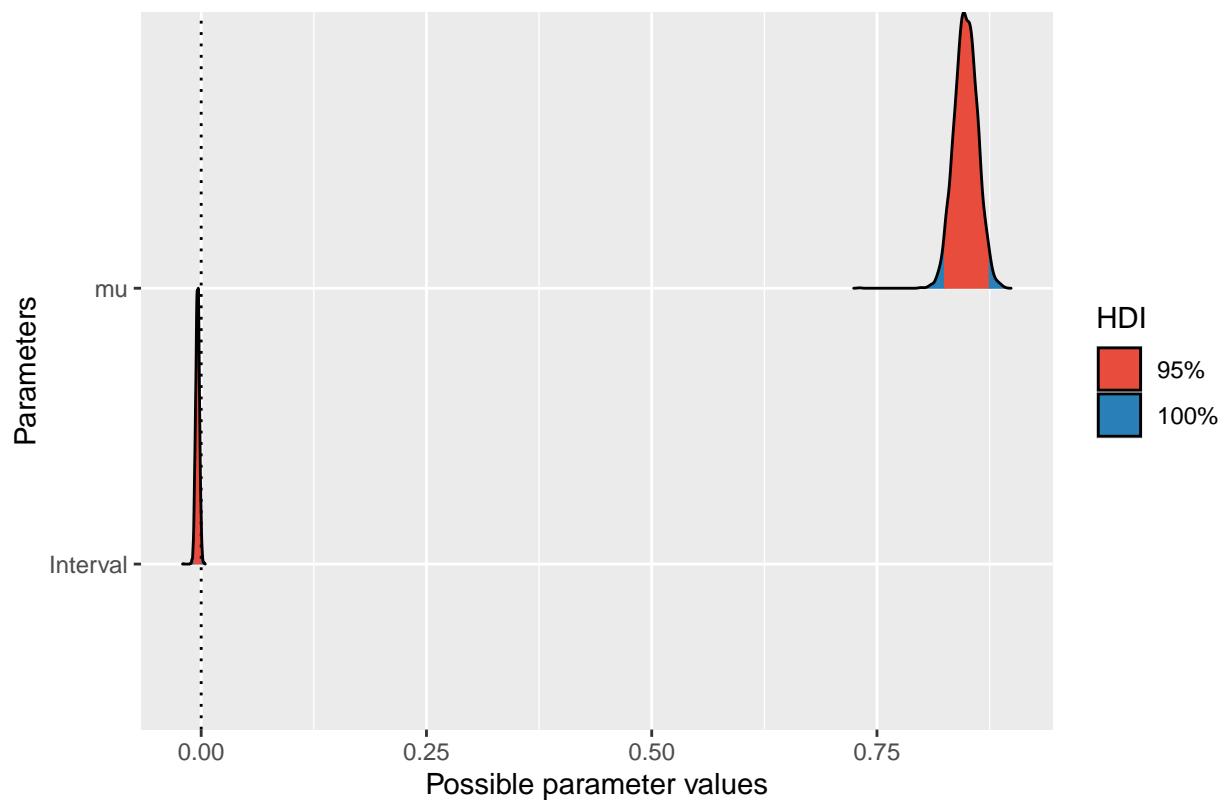
```
## [1] "-----"  
## # Proportion of samples inside the ROPE [-0.02, 0.02]:  
##  
## Parameter | inside ROPE  
## -----  
## mu        |      0.00 %  
## Interval  |    100.00 %
```

ROPE for wide prior



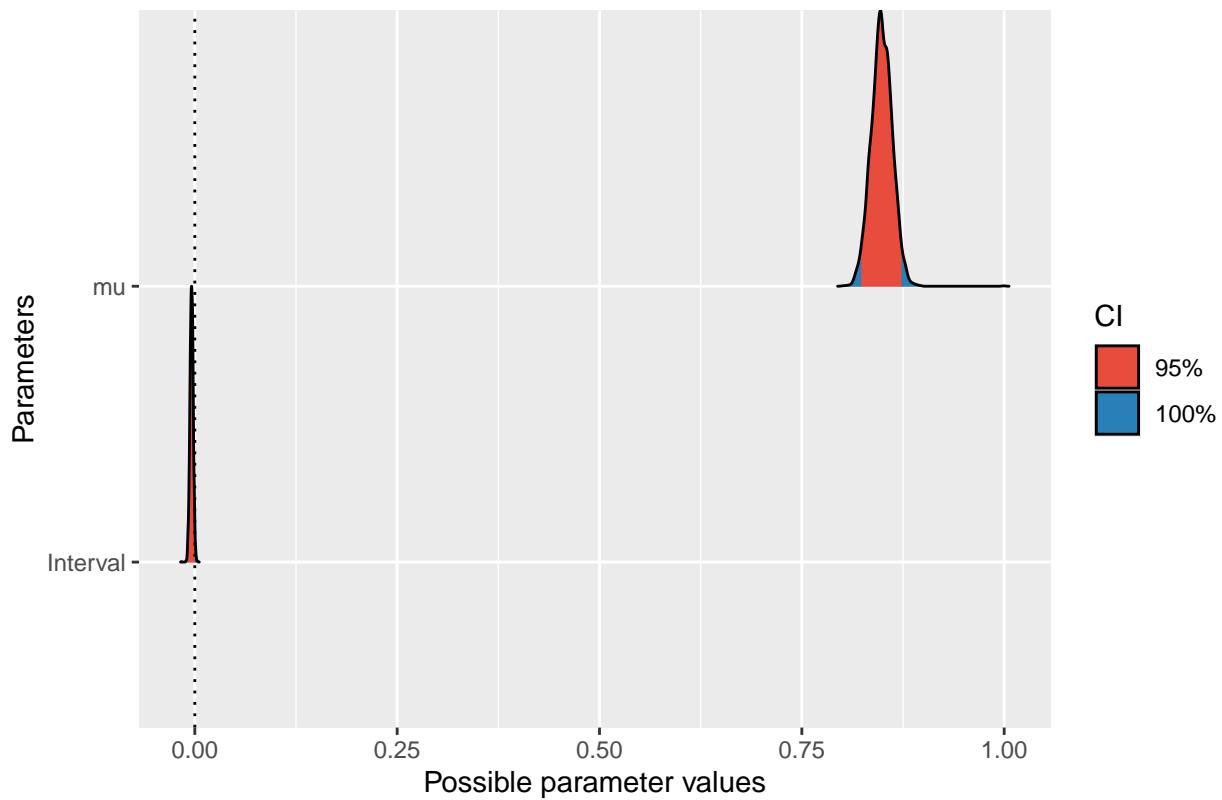
```
## [1] "-----"
## [1] "-----"
## [1] "PRIOR: ultrawide"
## Warning: data coerced from tibble to data frame
## Bayes factor analysis
## -----
## [1] Interval : 0.5924007 ±0.01%
## 
## Against denominator:
##   Intercept only
## 
## Bayes factor type: BFlinearModel, JZS
## 
## NULL
## [1] "-----"
## Highest Density Interval
## 
## Parameter |      95% HDI
## -----
## mu        | [ 0.82, 0.87]
## Interval  | [-0.01, 0.00]
```

HDI for ultrawide prior



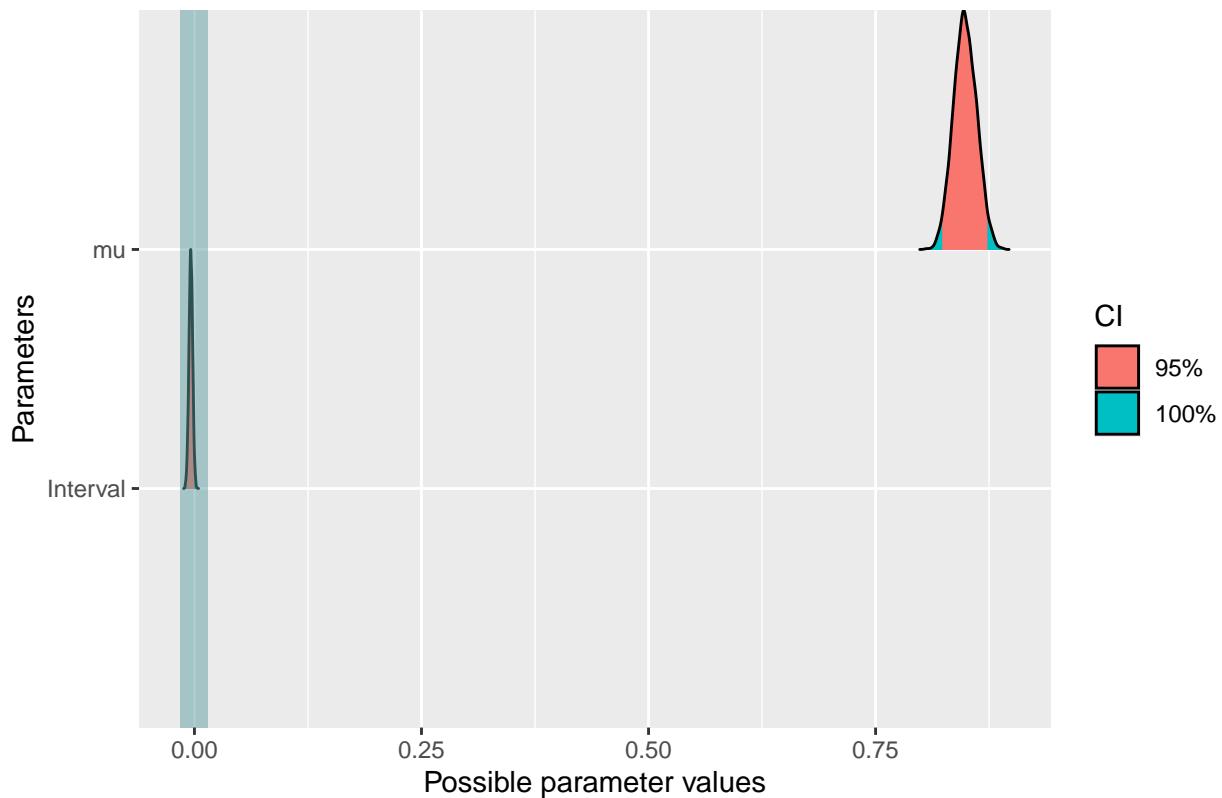
```
## [1] "-----"  
## Equal-Tailed Interval  
##  
## Parameter |      95% ETI  
## -----  
## mu        | [ 0.82, 0.87]  
## Interval   | [-0.01, 0.00]
```

ETI for ultrawide prior



```
## [1] "-----"  
## # Proportion of samples inside the ROPE [-0.02, 0.02]:  
##  
## Parameter | inside ROPE  
## -----  
## mu        |    0.00 %  
## Interval  | 100.00 %
```

ROPE for ultrawide prior



```
## [1] "-----"
# Calculate AIC & BIC Bayes factors

con_lm2 <- lm(
  formula = Consistency ~ Interval,
  data = con_data_tidy
)

con_lm_null <- lm(
  formula = Consistency ~ 1,
  data = con_data_tidy
)

con_aics <- AIC(con_lm2, con_lm_null)
con_bics <- BIC(con_lm2, con_lm_null)

# AIC/BIC converted to Bayes factors
ic_as_bf <- function(x) {exp(-x[1]/2) / exp(-x[2]/2)}

con_bics

##           df      BIC
## con_lm2     3 -118.2121
## con_lm_null  2 -119.3468
con_aics

##           df      AIC
```

```

## con_lm2      3 -127.0583
## con_lm_null  2 -125.2443
# Final scores
ic_as_bf(con_bics$BIC)

## [1] 0.5670297
ic_as_bf(con_aics$AIC)

## [1] 2.476971
prior_sd = list(0.25, 0.5, 0.75, 1, 5, 10)

for (sd in prior_sd) {

  print("-----NEXT PRIOR-----")
  print(cat("PRIOR WITH SD ", sd, "\n"))

  # Fits a Bayesian linear model to the data using Markov Chain Monte Carlo (MCMC) sampling via Stan to
  con_stan_lm <- stan_glm(
    Consistency ~ Interval,
    data = con_data_tidy,
    refresh = 0,      # progress bar
    prior = normal(scale = sd),    # prior width (i.e., weakly informative priors)
    diagnostic_file = file.path(tempdir(), "df.csv")
  )

  print(con_stan_lm)
  print("-----HDI PLOT-----")

  # Plot HDI
  con_stan_hdi <- hdi(con_stan_lm)
  con_stan_hdi_plot <- plot(hdi(con_stan_lm)) + ggtitle(paste("HDI for Prior for SD ", sd))

  print(con_stan_hdi)
  print(con_stan_hdi_plot)
  print("-----BAYES FACTOR-----")

  # Calculate Bayes factor for model parameters
  con_stan_parameters <- bayesfactor_parameters(con_stan_lm, null = 0)
  con_stan_prior_post_plot <- plot(con_stan_parameters)

  print(con_stan_parameters)
  print(con_stan_prior_post_plot)
  print("-----")
}

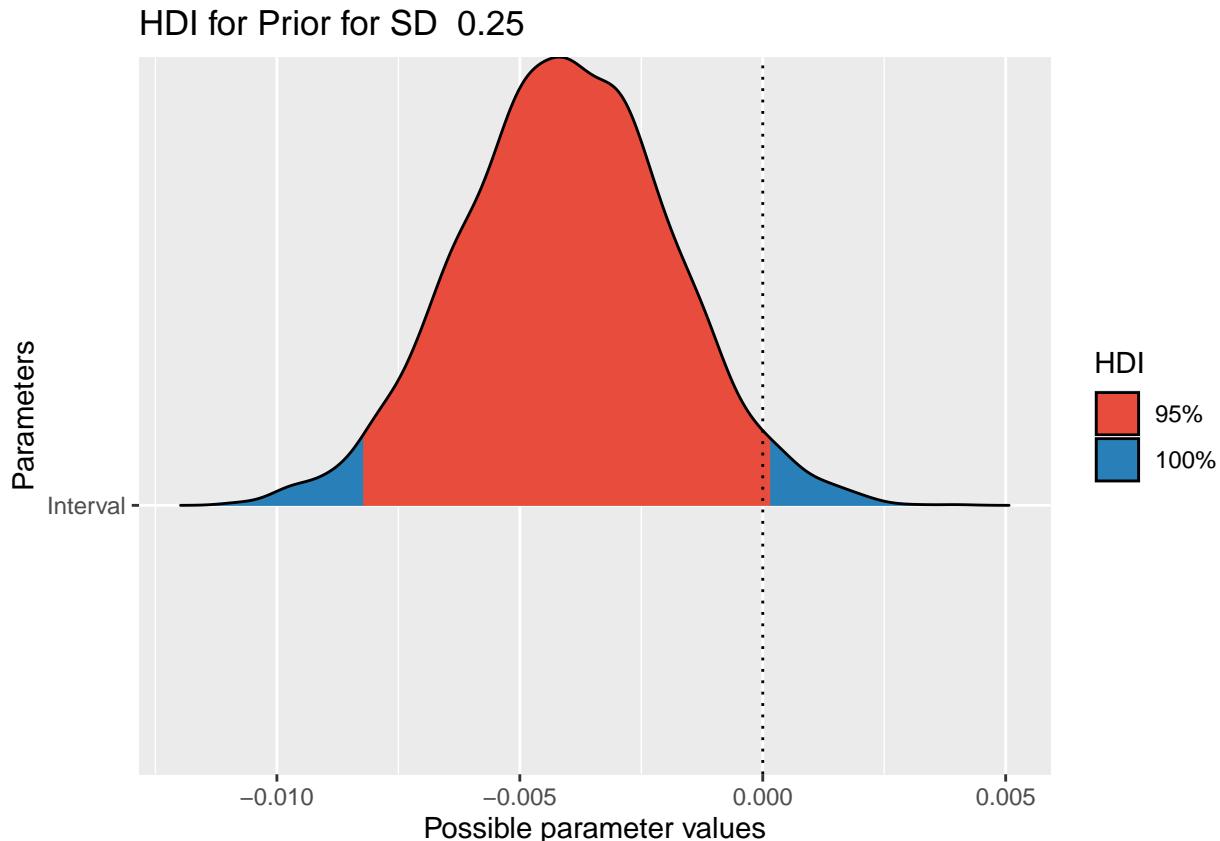
## [1] "-----NEXT PRIOR-----"
## PRIOR WITH SD  0.25
## NULL
## stan_glm
##   family:      gaussian [identity]
##   formula:     Consistency ~ Interval
##   observations: 141
##   predictors:   2

```

```

## -----
##           Median MAD_SD
## (Intercept) 0.9      0.0
## Interval    0.0      0.0
##
## Auxiliary parameter(s):
##           Median MAD_SD
## sigma 0.2      0.0
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----HDI PLOT-----"
## Highest Density Interval
##
## Parameter | 95% HDI
## -----
## (Intercept) | [ 0.84, 0.95]
## Interval    | [-0.01, 0.00]

```

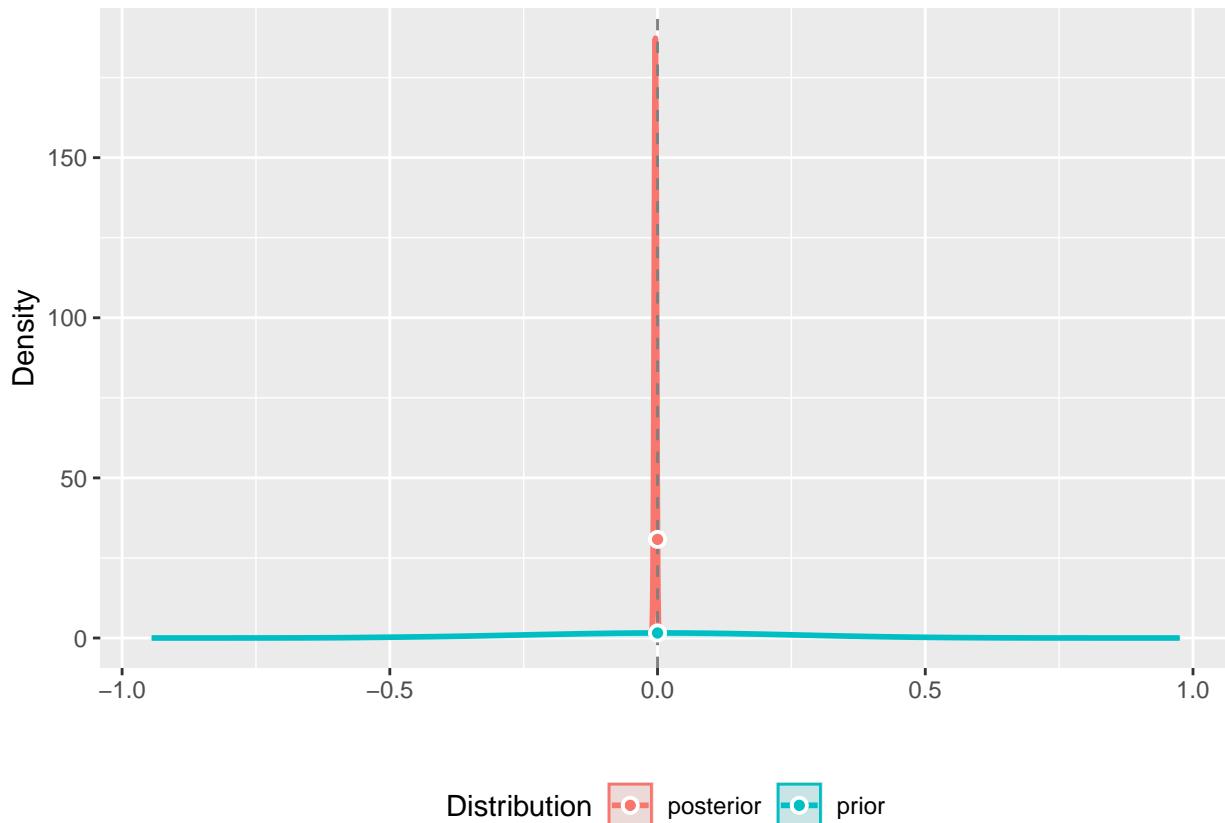


```

## [1] "-----BAYES FACTOR-----"
## Bayes Factor (Savage-Dickey density ratio)
##
## Parameter |     BF
## -----
## (Intercept) | 1.63e+32
## Interval    |  0.051

```

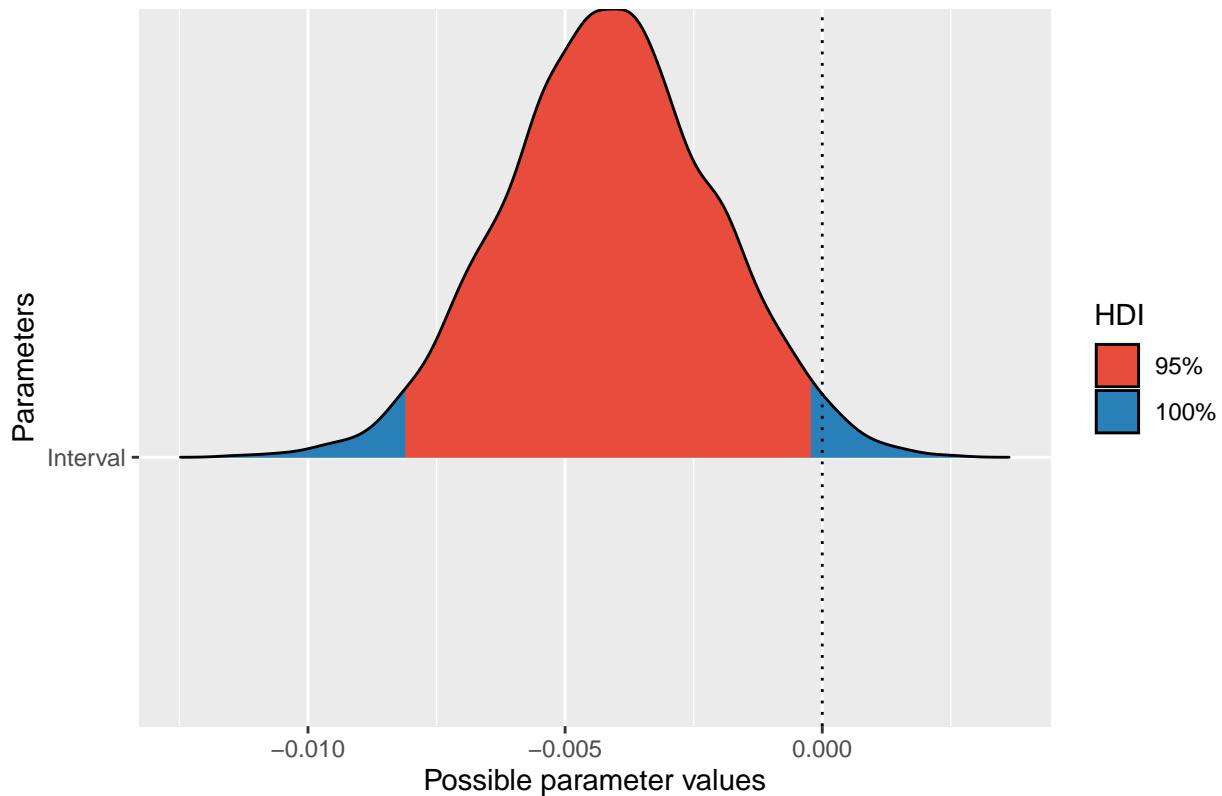
```
##  
## * Evidence Against The Null: 0
```



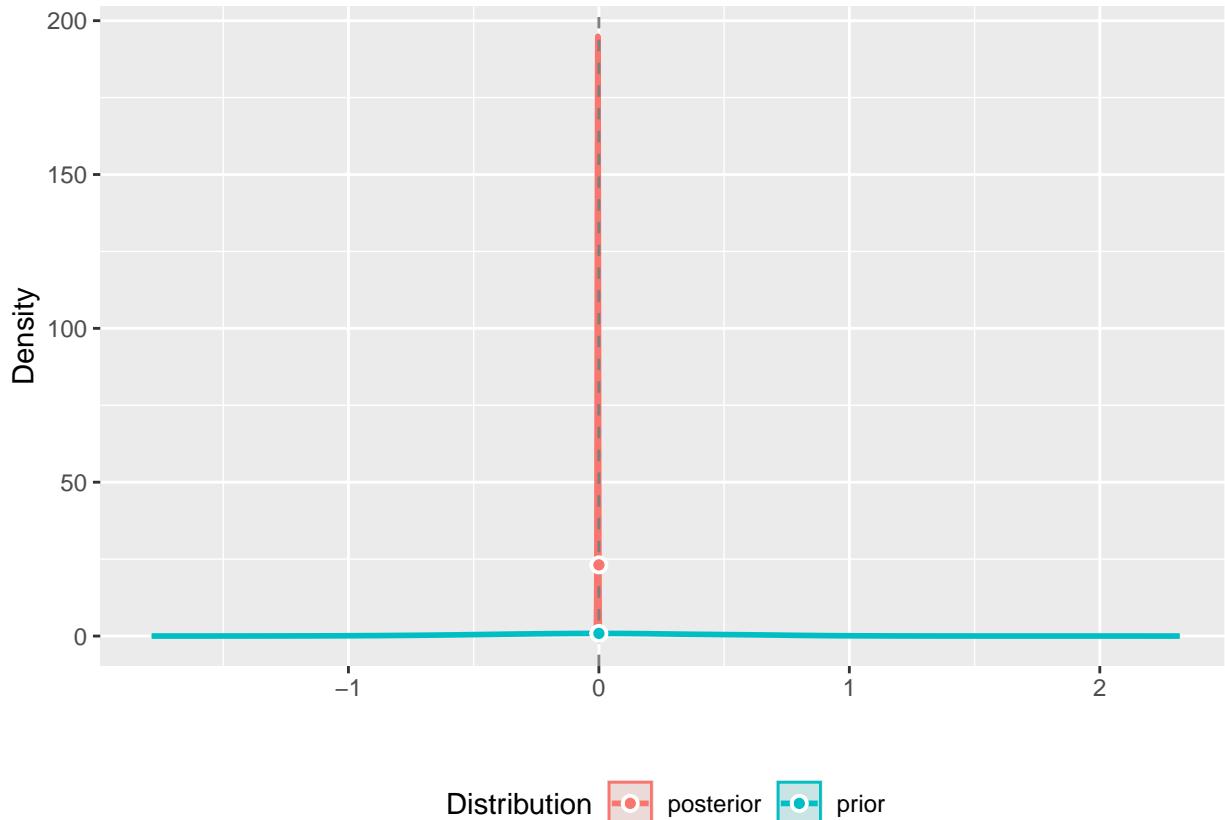
```
## [1] "-----"  
## [1] "-----NEXT PRIOR-----"  
## PRIOR WITH SD 0.5  
## NULL  
## stan_glm  
##   family: gaussian [identity]  
##   formula: Consistency ~ Interval  
##   observations: 141  
##   predictors: 2  
## -----  
##           Median MAD_SD  
## (Intercept) 0.9    0.0  
## Interval    0.0    0.0  
##  
## Auxiliary parameter(s):  
##   Median MAD_SD  
## sigma 0.2    0.0  
##  
## -----  
## * For help interpreting the printed output see ?print.stanreg  
## * For info on the priors used see ?prior_summary.stanreg  
## [1] "-----HDI PLOT-----"  
## Highest Density Interval  
##  
## Parameter | 95% HDI
```

```
## -----  
## (Intercept) | [ 0.85, 0.95]  
## Interval    | [-0.01, 0.00]
```

HDI for Prior for SD 0.5



```
## [1] "-----BAYES FACTOR-----"  
## Bayes Factor (Savage-Dickey density ratio)  
## -----  
## Parameter | BF  
## -----  
## (Intercept) | 1.50e+33  
## Interval    | 0.037  
## -----  
## * Evidence Against The Null: 0
```



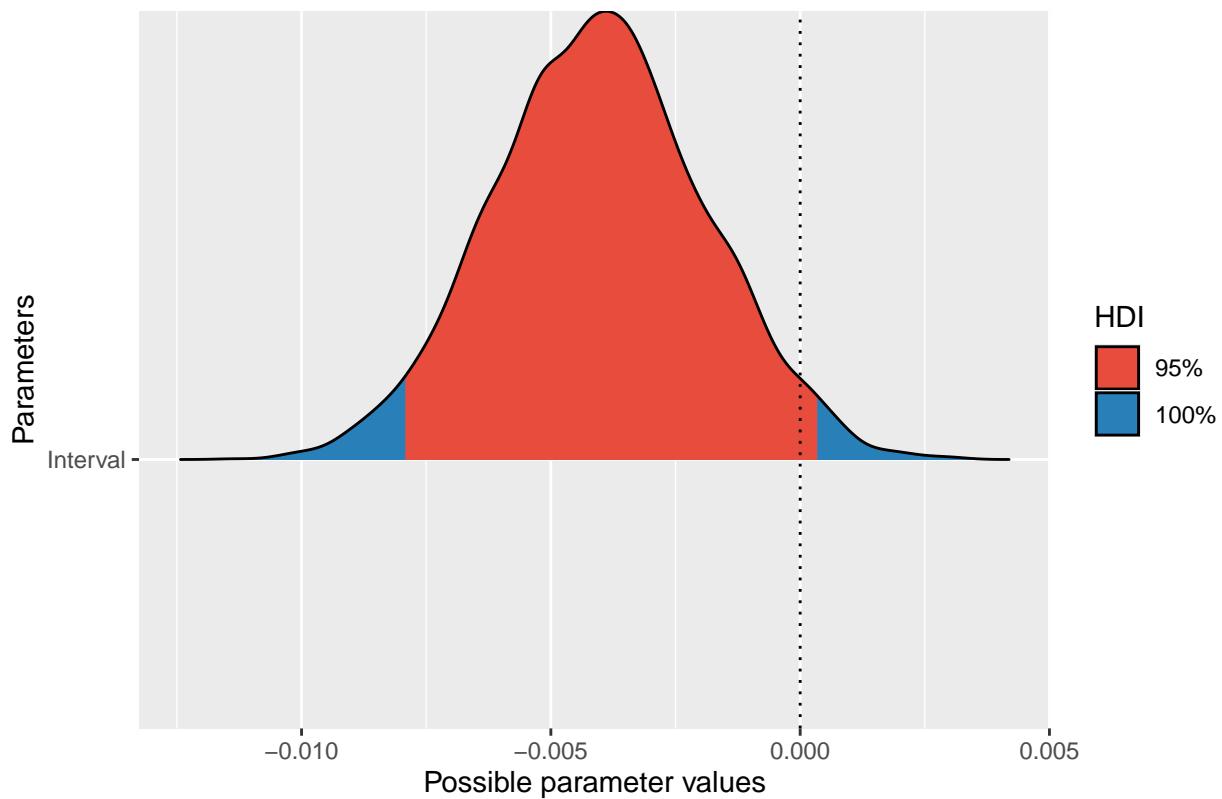
```

## [1] "-----"
## [1] "-----NEXT PRIOR-----"
## PRIOR WITH SD  0.75
## NULL
## stan_glm
##   family:      gaussian [identity]
##   formula:     Consistency ~ Interval
##   observations: 141
##   predictors:   2
## -----
##           Median MAD_SD
## (Intercept) 0.9    0.0
## Interval     0.0    0.0
##
## Auxiliary parameter(s):
##           Median MAD_SD
## sigma 0.2    0.0
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----HDI PLOT-----"
## Highest Density Interval
##
## Parameter | 95% HDI
## -----
## (Intercept) | [ 0.84, 0.95]

```

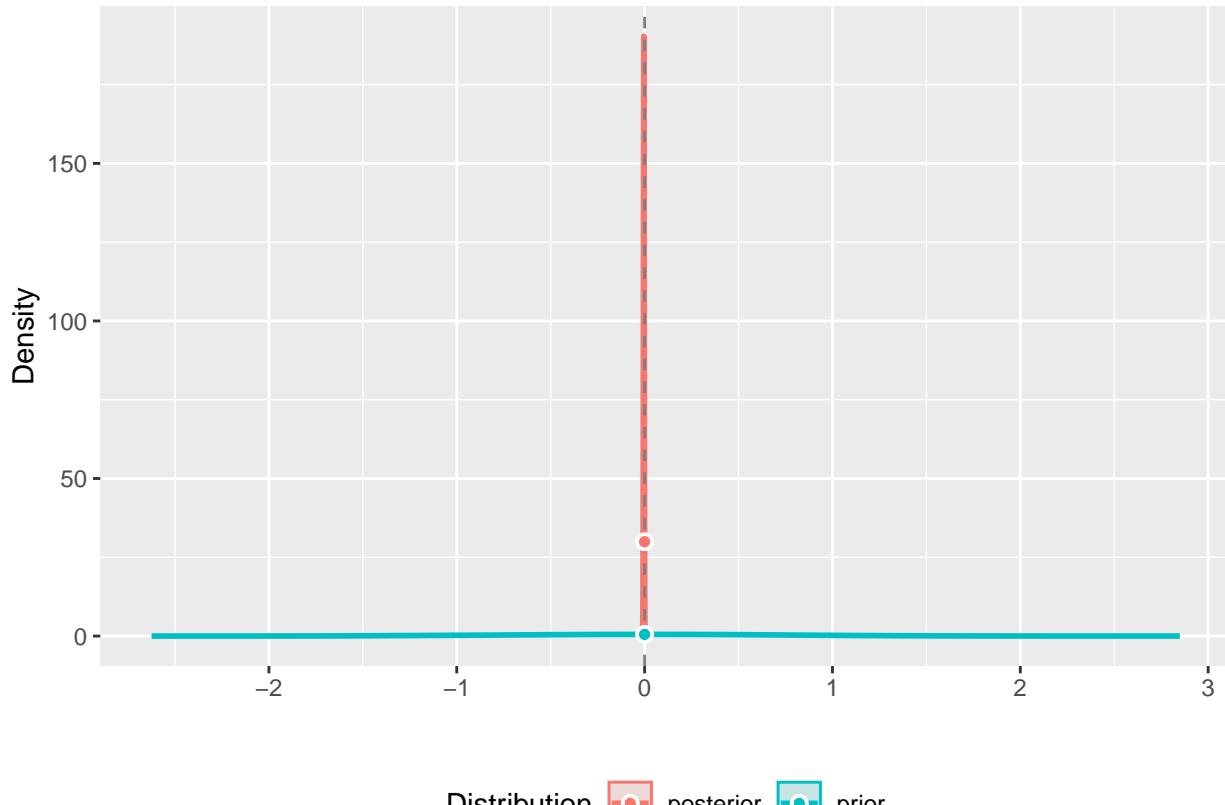
```
## Interval | [-0.01, 0.00]
```

HDI for Prior for SD 0.75



```
## [1] "-----BAYES FACTOR-----"
```

```
## Bayes Factor (Savage-Dickey density ratio)
##
## Parameter |      BF
## -----
## (Intercept) | 8.49e+39
## Interval    |  0.018
##
## * Evidence Against The Null: 0
```



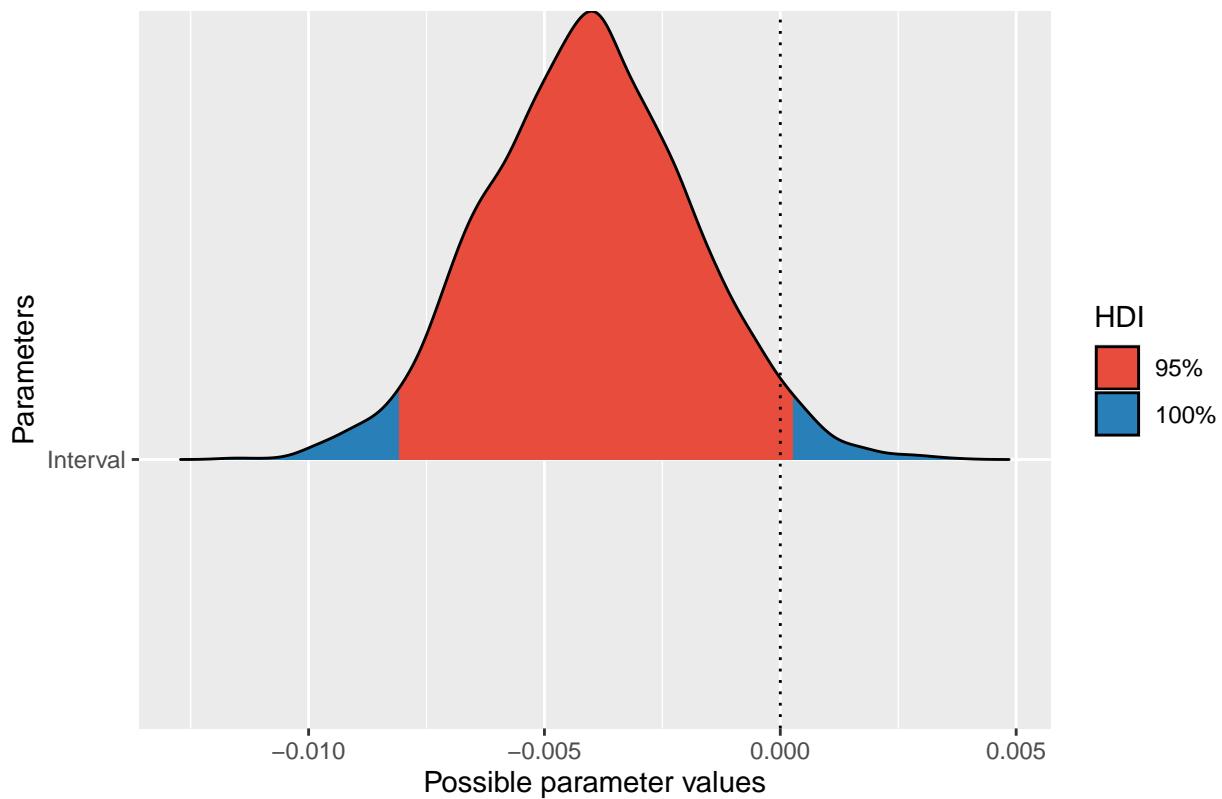
```

## [1] "-----"
## [1] "-----NEXT PRIOR-----"
## PRIOR WITH SD 1
## NULL
## stan_glm
##   family:      gaussian [identity]
##   formula:     Consistency ~ Interval
##   observations: 141
##   predictors:  2
## -----
##           Median MAD_SD
## (Intercept) 0.9    0.0
## Interval    0.0    0.0
##
## Auxiliary parameter(s):
##           Median MAD_SD
## sigma 0.2    0.0
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----HDI PLOT-----"
## Highest Density Interval
##
## Parameter | 95% HDI
## -----
## (Intercept) | [ 0.84, 0.95]

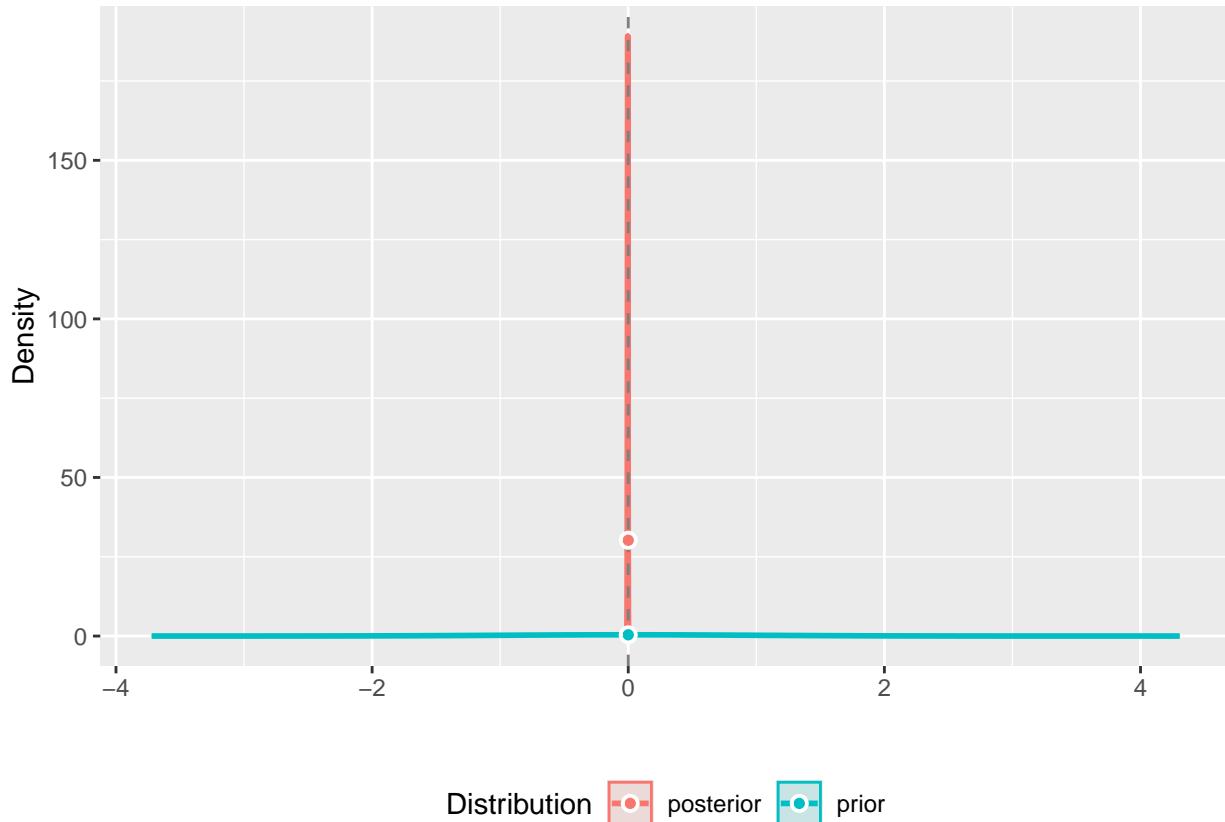
```

```
## Interval | [-0.01, 0.00]
```

HDI for Prior for SD 1



```
## [1] "-----BAYES FACTOR-----"  
## Bayes Factor (Savage-Dickey density ratio)  
##  
## Parameter | BF  
## -----  
## (Intercept) | 2.24e+31  
## Interval | 0.013  
##  
## * Evidence Against The Null: 0
```



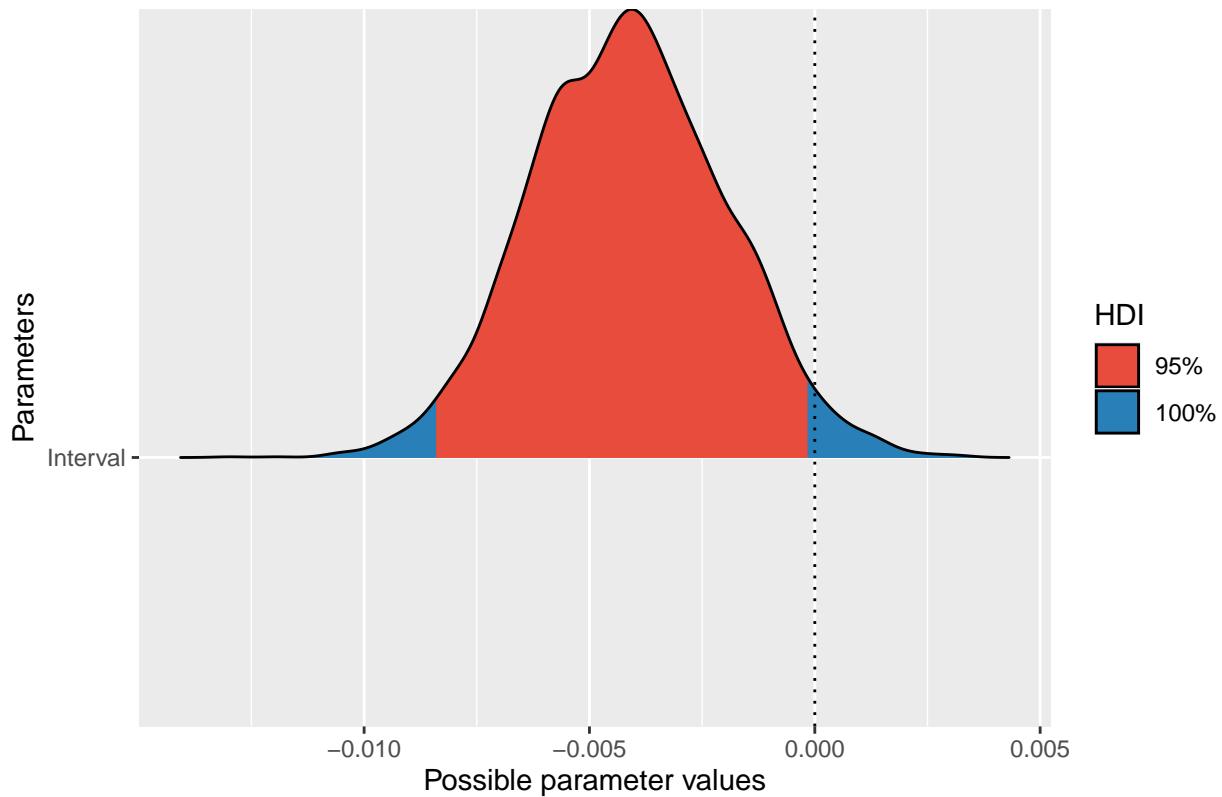
```

## [1] "-----"
## [1] "-----NEXT PRIOR-----"
## PRIOR WITH SD 5
## NULL
## stan_glm
##   family:      gaussian [identity]
##   formula:     Consistency ~ Interval
##   observations: 141
##   predictors:   2
## -----
##           Median MAD_SD
## (Intercept) 0.9    0.0
## Interval    0.0    0.0
##
## Auxiliary parameter(s):
##           Median MAD_SD
## sigma 0.2    0.0
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----HDI PLOT-----"
## Highest Density Interval
##
## Parameter |      95% HDI
## -----
## (Intercept) | [ 0.84,  0.95]

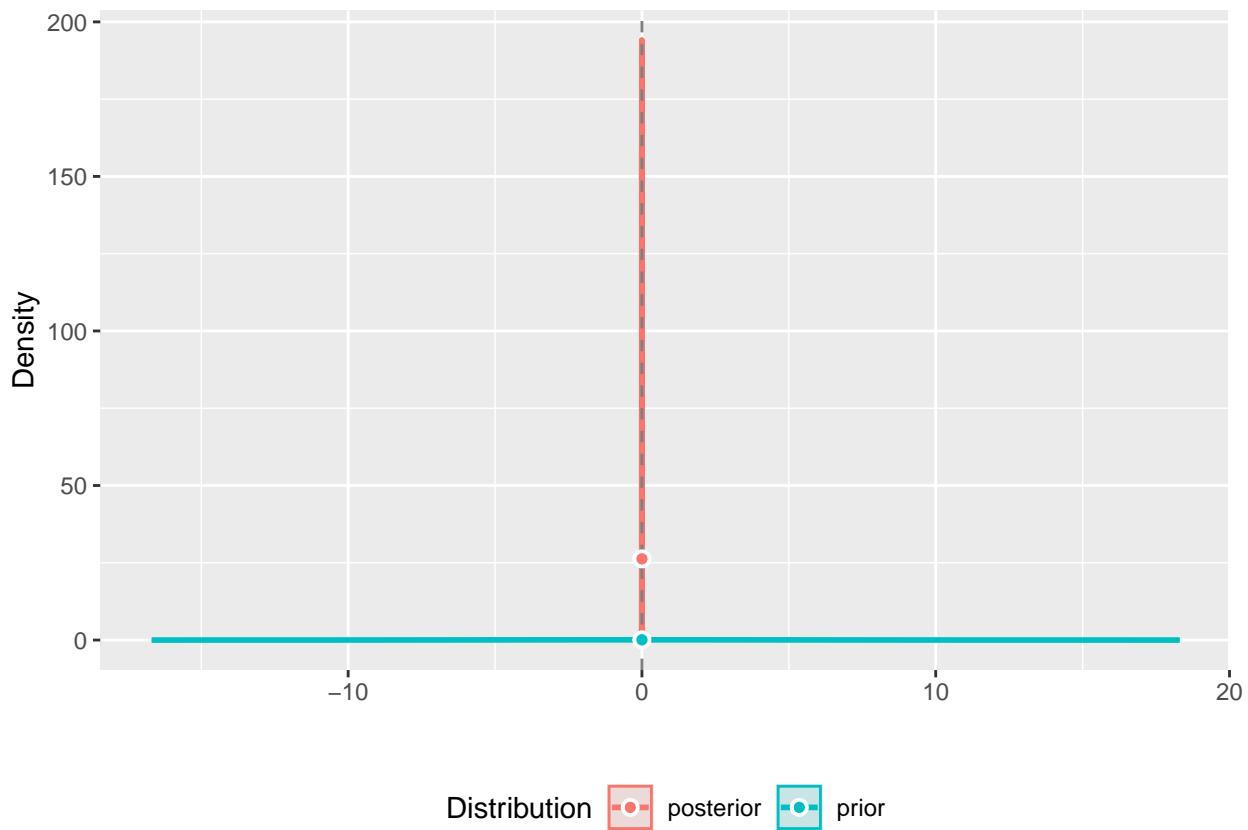
```

```
## Interval | [-0.01, 0.00]
```

HDI for Prior for SD 5



```
## [1] "-----BAYES FACTOR-----"  
## Bayes Factor (Savage-Dickey density ratio)  
##  
## Parameter | BF  
## -----  
## (Intercept) | 3.82e+32  
## Interval | 0.003  
##  
## * Evidence Against The Null: 0
```



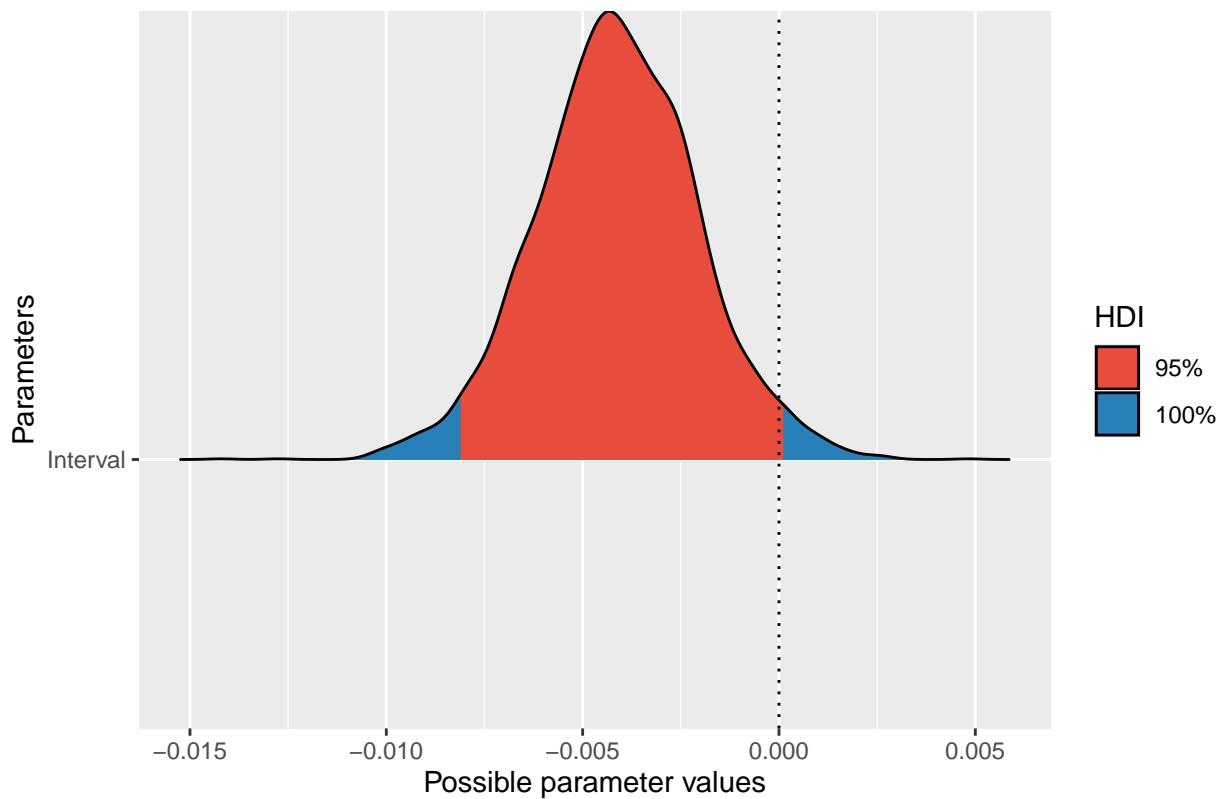
```

## [1] "-----"
## [1] "-----NEXT PRIOR-----"
## PRIOR WITH SD 10
## NULL
## stan_glm
##   family:      gaussian [identity]
##   formula:     Consistency ~ Interval
##   observations: 141
##   predictors:   2
## -----
##           Median MAD_SD
## (Intercept) 0.9    0.0
## Interval    0.0    0.0
##
## Auxiliary parameter(s):
##           Median MAD_SD
## sigma 0.2    0.0
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----HDI PLOT-----"
## Highest Density Interval
##
## Parameter | 95% HDI
## -----
## (Intercept) | [ 0.84, 0.95]

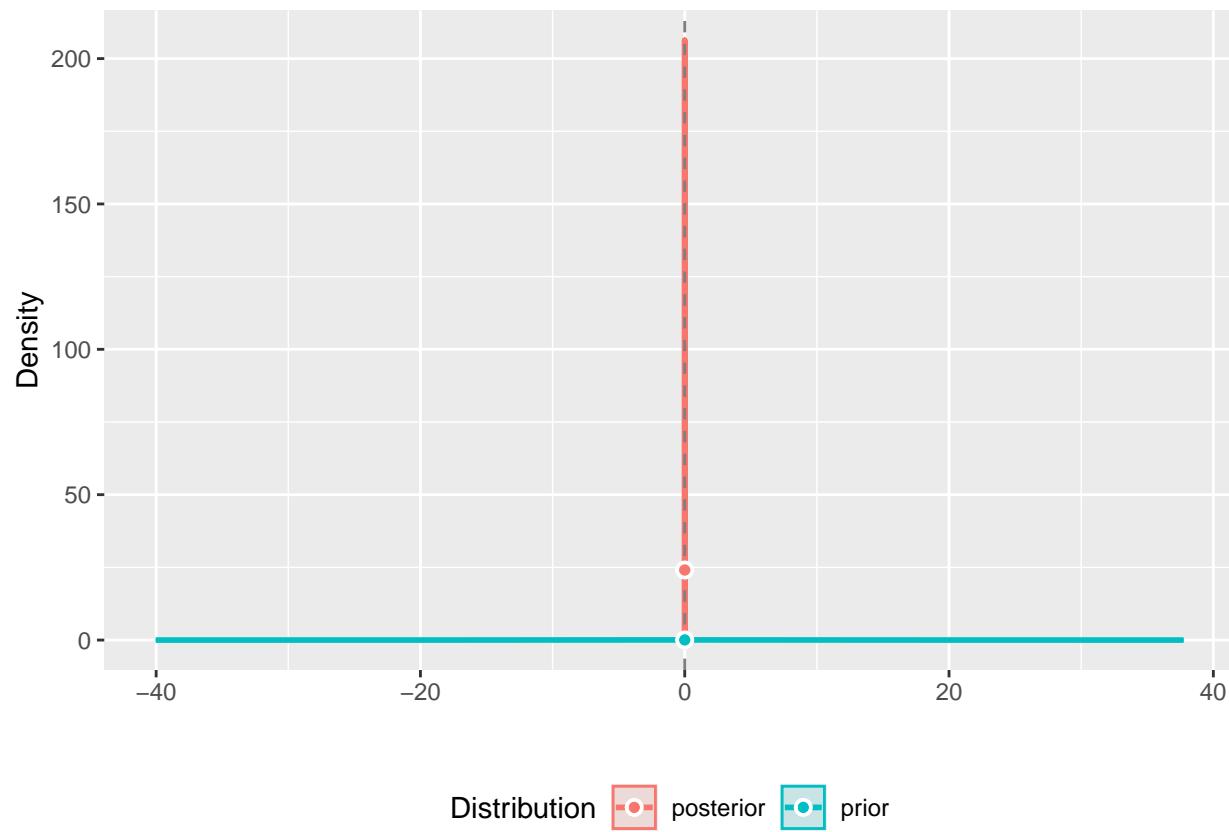
```

```
## Interval | [-0.01, 0.00]
```

HDI for Prior for SD 10



```
## [1] "-----BAYES FACTOR-----"  
## Bayes Factor (Savage-Dickey density ratio)  
##  
## Parameter | BF  
## -----  
## (Intercept) | 3.59e+53  
## Interval | 0.002  
##  
## * Evidence Against The Null: 0
```



```
## [1] "-----"
```

Task 2

How does the nature of decision making depend change with repeated experience? Certainly feedback will influence people's later decisions, but do decisions change even without feedback? One possibility is that people become more normative with repeated experience: that their choices correspond more to expected utility theory. That is, for Alternative A with a p_{A1} probability of outcome x_{A1} and a p_{A2} probability of outcome x_{A2} , people will (at least implicitly) convert this into an expected utility. If we assume a power law utility function with exponent 0.5, then $E(u_A) = p_{A1}(x_{A1})^{0.5} + p_{A2}(x_{A2})^{0.5}$. Then $E(u_A)$ is compared with the similarly calculated $E(u_B)$, and the alternative with the higher expected utility is chosen.

Another possibility is that choices become more heuristic with practice: that they alight on simple rules for making choices that allow for effective performance with little effort. For example, people may begin to rely more on simple features of the options that allow them to decide between them. One simple heuristic would be to avoid the option that has the lowest possible reward. So for Alternative A, this heuristic value would be $h_A = \min(x_{A1}, x_{A2})$, for Alternative B the heuristic value would be $h_B = \min(x_{B1}, x_{B2})$, and people would choose Alternative A if $h_A > h_B$. This heuristic will often correspond to expected utility, but will not always do so.

The file `repeated_choice.csv` contains experimental data from a repeated measures factorial experiment that investigated how choices changed with repeated experience with the choices. Participants (`id`) were presented with pairs of alternatives, each with different probabilities of two possible rewards, as above. There were 75 alternative pairs in the experiment, and participants were presented with all 75 pairs in each block. Over the course of several days, participants completed 6 blocks (`Block`) of the experiment. For each block we calculated as a dependent variable the proportion of trials that matched the predictions of expected utility (`ExpectedUtilityMatch`) and the proportion of trials that matched the predictions of the lowest value heuristic (`LowestValueMatch`).

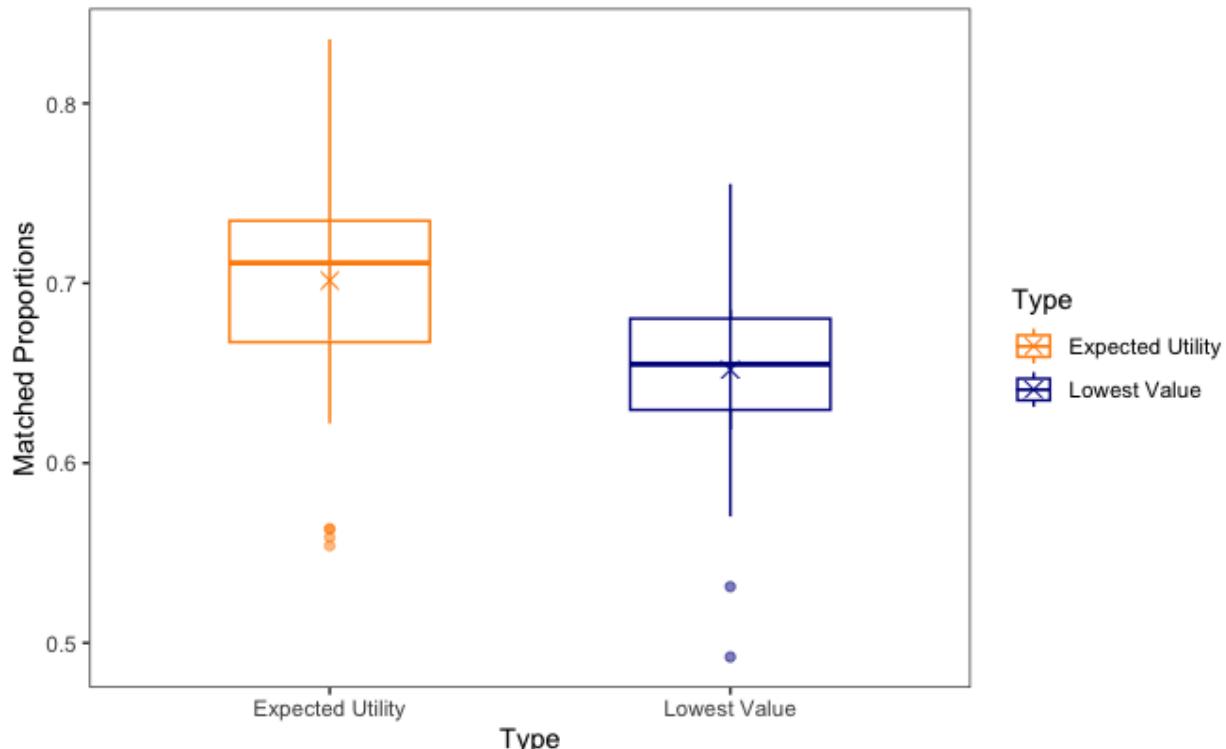
Please analyse the data using both a frequentist and Bayesian ANOVA (e.g., using `afex` and `BayesFactor`), using post-hoc tests where appropriate, and present the results as you would for a paper, using APA format. In other words, describe your statistical model and results and describe the conclusions with regard to the research question. Be careful not to draw unwarranted causal conclusions. In the first section for this task, include only one figure (which may contain multiple panels), with an appropriate figure caption.

Part 1: Results Description

This study aims to understand whether the nature of decision making changes with repeated experience. More specifically, this study focuses on whether decision making is more likely to follow the *expected utility theory* (EUT; $E(u_A)$ vs $E(u_B)$) or the *lowest value heuristic* (LVH; h_A vs h_B) when participants are not provided feedback about their prior decisions.

A mixed-methods ANOVA (following both frequentist and Bayesian (`BayesFactor` R library) approaches) was run with decision making Type (i.e., EUT vs LVH) as the within-subjects factor and Block (i.e., each time they completed the experiment) as the between-participants factor. The Bayesian analysis was run using a mixed effects model with a medium prior (`rscaleFixed = 0.5`) and summarised using a Bayes factor of inclusion for matching models. There was a significant effect of and decisive evidence for the effect of Type on the proportion of trials that matched the predictions depending on Type; $F(1, 27) = 4.68, p = .039, BF_{inclusion} = 1.06 \times 10^6$. However, there was no main effect of Block on proportion scores; $F(4, 107.9) = .94, p = .445, BF_{inclusion} = .006$, nor a Block by Type interaction $F(4.13, 111.47) = 1.91, p = .111, BF_{inclusion} = .072$. Here, both Bayes factors providing decisive evidence for there being no difference in proportions between EUT and LVH (i.e., the null hypothesis). These results were qualitatively the same for both wide (`rscaleFixed = $\frac{\sqrt{2}}{2}$`) and ultrawide (`rscaleFixed = 1`) prior widths.

Given the significant effect of Type, post hoc tests were conducted between EUT and LVH. Proportion of trials that matched the predictions were significantly higher for EUT ($M = .701$) than LVH ($M = .652$) with the Bayes factor providing significant and decisive evidence ($\Delta = .0496, p = .0395, BF_{10} = 161\,065$) for the effect of Type. This significant difference is visualised in Figure 1.



Figure

1. Boxplots showing the distribution of matched proportions by EUT (left in orange) and LVH (right in blue). The cross within each boxplot indicates mean ratings for each condition with 95% error bars.

Part 2: R Code

```
# Loading data
rc_data <- read.csv("repeated_choice.csv")

rc_data %>% str()

## 'data.frame':    168 obs. of  5 variables:
## $ X             : int  1 2 3 4 5 6 7 8 9 10 ...
## $ id            : int  13 13 13 13 20 20 20 20 22 22 ...
## $ Block         : int  1 2 3 4 1 2 3 4 1 2 ...
## $ ExpectedUtilityMatch: num  0.789 0.746 0.704 0.69 0.746 ...
## $ LowestValueMatch     : num  0.641 0.641 0.703 0.625 0.672 ...

# Tidy data
rc_data_tidy <- rc_data %>%
  pivot_longer(
    cols = c(ExpectedUtilityMatch, LowestValueMatch),
    names_to = 'Type',
    values_to = 'Matched_Proportions'
  ) %>%
  mutate(
    id = factor(id),
    Block = factor(Block),
    Type = factor(Type)
  )

rc_data_tidy %>% str()

## tibble [336 x 5] (S3:tbl_df/tbl/data.frame)
## $ X             : int [1:336] 1 1 2 2 3 3 4 4 5 5 ...
## $ id            : Factor w/ 28 levels "10","11","12",...: 4 4 4 4 4 4 4 4 8 8 ...
## $ Block         : Factor w/ 6 levels "1","2","3","4",...: 1 1 2 2 3 3 4 4 1 1 ...
## $ Type          : Factor w/ 2 levels "ExpectedUtilityMatch",...: 1 2 1 2 1 2 1 2 1 2 ...
## $ Matched_Proportions: num [1:336] 0.789 0.641 0.746 0.641 0.704 ...

# Overall
rc_data %>%
  summarise(
    eu_mean = mean(ExpectedUtilityMatch),
    eu_sd = sd(ExpectedUtilityMatch),
    lv_mean = mean(LowestValueMatch),
    lv_sd = sd(LowestValueMatch)
  )

##      eu_mean      eu_sd      lv_mean      lv_sd
## 1 0.7014588 0.08690832 0.7014588 0.07747532

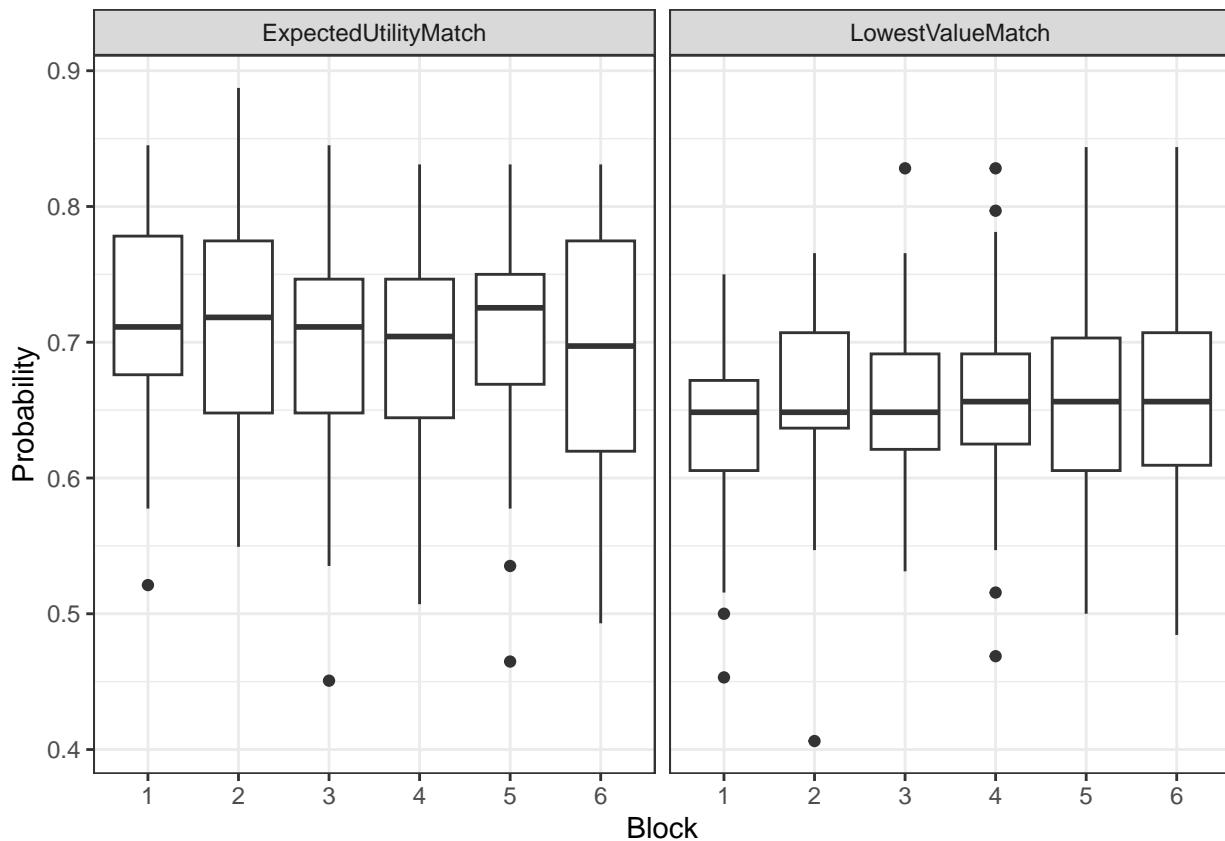
# Per block
rc_data %>%
  group_by(Block) %>%
  summarise(
    eu_mean = mean(ExpectedUtilityMatch),
    eu_sd = sd(ExpectedUtilityMatch),
    lv_mean = mean(LowestValueMatch),
    lv_sd = sd(LowestValueMatch)
```

```

)
## # A tibble: 6 x 5
##   Block eu_mean eu_sd lv_mean lv_sd
##   <int>   <dbl>   <dbl>   <dbl>   <dbl>
## 1     1    0.716  0.0768  0.636  0.0680
## 2     2    0.720  0.0833  0.651  0.0762
## 3     3    0.695  0.0890  0.653  0.0667
## 4     4    0.691  0.0867  0.657  0.0817
## 5     5    0.700  0.0888  0.660  0.0860
## 6     6    0.686  0.0977  0.655  0.0882

# Boxplot
ggplot(rc_data_tidy, aes(x = Block, y = Matched_Proportions)) +
  geom_boxplot() +
  facet_wrap(~ Type) +
  labs(y = 'Probability') +
  theme_bw()

```



```

rc_data_tidy %>%
  select(Matched_Proportions) %>%
  is.na() %>%
  colSums()

## Matched_Proportions
## 0
rc_data_tidy %>% head()

```

```

## # A tibble: 6 x 5
##       X id   Block Type      Matched_Proportions
##   <int> <fct> <fct> <fct>                <dbl>
## 1     1 13     1 ExpectedUtilityMatch    0.789
## 2     1 13     1 LowestValueMatch      0.641
## 3     2 13     2 ExpectedUtilityMatch    0.746
## 4     2 13     2 LowestValueMatch      0.641
## 5     3 13     3 ExpectedUtilityMatch    0.704
## 6     3 13     3 LowestValueMatch      0.703

# Column names
rc_data_tidy %>% colnames()

## [1] "X"                  "id"                 "Block"
## [4] "Type"               "Matched_Proportions"

# Levels of each interested variable
rc_data_tidy %>% select(-c('X')) %>% sapply(levels)

## $id
##  [1] "10" "11" "12" "13" "15" "17" "19" "20" "21" "22" "23" "24" "25" "26" "27"
## [16] "30" "31" "33" "34" "35" "36" "37" "38" "39" "40" "43" "44" "45"
##
## $Block
##  [1] "1" "2" "3" "4" "5" "6"
##
## $Type
##  [1] "ExpectedUtilityMatch" "LowestValueMatch"
##
## $Matched_Proportions
##  NULL

```

Frequentist Analysis

```
# Repeated measures ANOVA

rc_freq_a1 <- aov_ez(
  id = 'id', dv = 'Matched_Proportions', data = rc_data_tidy,
  within = c('Block', 'Type')
)

rc_freq_a1

## Anova Table (Type 3 tests)
##
## Response: Matched_Proportions
##          Effect      df   MSE      F ges p.value
## 1       Block 4.00, 107.90 0.00  0.94 .004   .445
## 2       Type     1, 27 0.04 4.68 * .085   .039
## 3 Block:Type 4.13, 111.47 0.00  1.91 .013   .111
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
##
## Sphericity correction method: GG
```

- There is a significant effect of Type on matched proportion scores

– $F(1, 27) = 4.68$, $p = .039$

```
# Conditional means
rc_freq_em1 <- emmeans(rc_freq_a1, 'Type')

rc_freq_em1
```

```
##  Type           emmean      SE df lower.CL upper.CL
##  ExpectedUtilityMatch 0.701 0.0146 27    0.671    0.731
##  LowestValueMatch     0.652 0.0116 27    0.628    0.676
## 
## Results are averaged over the levels of: Block
## Confidence level used: 0.95
```

```
# Contrasts
rc_freq_con <- list(
  eu_vs_lv = c(1, -1)
)
```

```
contrast(rc_freq_em1, rc_freq_con)
```

```
## contrast estimate      SE df t.ratio p.value
## eu_vs_lv  0.0496 0.0229 27   2.164  0.0395
## 
```

```
## Results are averaged over the levels of: Block
```

```
# OR
```

```
pairs(rc_freq_em1)
```

```
## contrast                  estimate      SE df t.ratio p.value
## ExpectedUtilityMatch - LowestValueMatch  0.0496 0.0229 27   2.164  0.0395
## 
## Results are averaged over the levels of: Block
```

- Probabilities are significantly greater for Expected Utility ($M = 0.701$) than Lowest Value ($M = 0.652$) ($\Delta = .0496$, $p = .0395$).

```

afex_plot(
  rc_freq_a1,
  x = 'Type',
  error = "within",
  mapping = c("color"),
  data_geom = ggplot2::geom_boxplot,
  data_alpha = 0.5,
  data_arg = list(width = 0.5),
  point_arg = list(size = 3, pch = 4),
  factor_levels = list(Type = c("Expected Utility", "Lowest Value"))
) + 

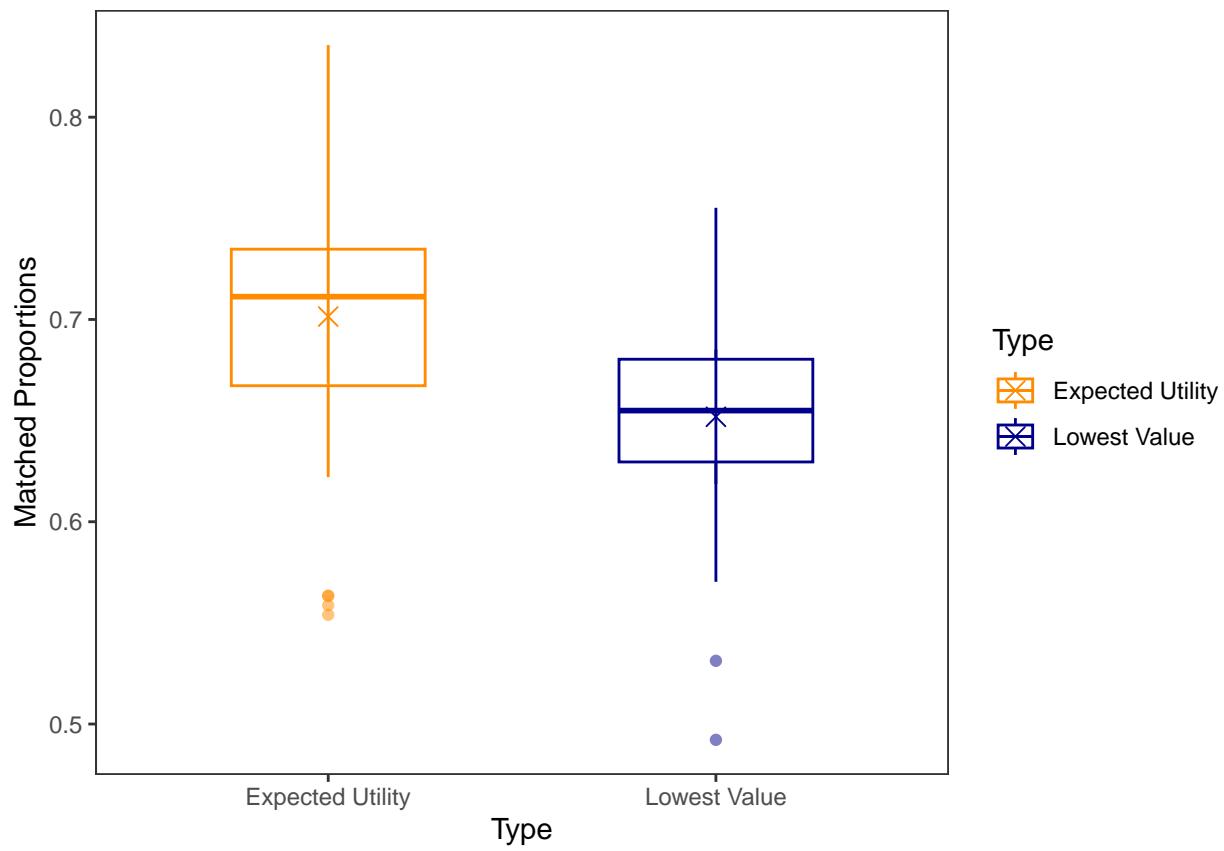
  scale_color_manual(
    values = c("Expected Utility" = "darkorange", "Lowest Value" = "darkblue"),
    name = "Type"
) + 

  labs(y = 'Matched Proportions') + 

  theme_bw() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
)

## Renaming/reordering factor levels of 'Type':
##   ExpectedUtilityMatch -> Expected Utility
##   LowestValueMatch -> Lowest Value

```



Bayesian Analysis

```
prior_widths = c('medium', 'wide', 'ultrawide')

for (prior in prior_widths) {

  # ANOVA
  rc_bf_anova <- anovaBF(
    Matched_Proportions ~ Block * Type + id,
    data = rc_data_tidy,
    whichRandom = "id",    # random factor (i.e., each participant contributes multiple observations)
    rscaleFixed = prior,   # prior beliefs for fixed effects (i.e., prior width)
    progress = FALSE      # progress bar
  )

  # Averages across all models
  rc_bf_anova_model_avg <- bf_inclusion(
    rc_bf_anova,
    match_models = TRUE    # only models that include the same set of variables (except for the one being
  )

  cat("\n")
  print(prior)
  print(summary(rc_bf_anova))
  print(rc_bf_anova_model_avg)
  cat("\n")

}

## Warning: data coerced from tibble to data frame
##
## [1] "medium"
## Bayes factor analysis
## -----
## [1] Block + id          : 0.006331883 ±0.47%
## [2] Type + id          : 982172.8    ±1.73%
## [3] Block + Type + id : 6439.735    ±1.8%
## [4] Block + Type + Block:Type + id : 459.3043    ±1.31%
##
## Against denominator:
##   Matched_Proportions ~ id
##   ---
## Bayes factor type: BFlinearModel, JZS
## 
## NULL
## Inclusion Bayes Factors (Model Averaged)
##
##          P(prior) P(posterior) Inclusion BF
## id        1.00      1.00
## Block     0.40      6.51e-03      0.007
## Type      0.40      1.00      9.82e+05
## Block:Type 0.20      4.64e-04      0.071
##
## * Compared among: matched models only
```

```

## *      Priors odds: uniform-equal
## Warning: data coerced from tibble to data frame
##
## [1] "wide"
## Bayes factor analysis
## -----
## [1] Block + id          : 0.001645439 ±0.47%
## [2] Type + id          : 876160.2    ±0.97%
## [3] Block + Type + id : 1501.907    ±1.28%
## [4] Block + Type + Block:Type + id : 39.94248   ±0.94%
##
## Against denominator:
##   Matched_Proportions ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
##
## NULL
## Inclusion Bayes Factors (Model Averaged)
##
##           P(prior) P(posterior) Inclusion BF
## id          1.00        1.00
## Block       0.40        1.71e-03     0.002
## Type        0.40        1.00        8.76e+05
## Block:Type  0.20        4.55e-05    0.027
##
## * Compared among: matched models only
## *      Priors odds: uniform-equal

## Warning: data coerced from tibble to data frame
##
## [1] "ultrawide"
## Bayes factor analysis
## -----
## [1] Block + id          : 0.0003743754 ±1.54%
## [2] Type + id          : 710483.8    ±0.85%
## [3] Block + Type + id : 284.5364    ±1.47%
## [4] Block + Type + Block:Type + id : 2.333571   ±2.19%
##
## Against denominator:
##   Matched_Proportions ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
##
## NULL
## Inclusion Bayes Factors (Model Averaged)
##
##           P(prior) P(posterior) Inclusion BF
## id          1.00        1.00
## Block       0.40        4.00e-04    4.00e-04
## Type        0.40        1.00        7.11e+05
## Block:Type  0.20        3.28e-06    0.008
##
## * Compared among: matched models only

```

```

## * Priors odds: uniform-equal

rc_bf_anova_med <- anovaBF(
  Matched_Proportions ~ Block * Type + id,
  data = rc_data_tidy,
  whichRandom = "id",
  rscaleFixed = 'medium',
  progress = FALSE
)

## Warning: data coerced from tibble to data frame
summary(rc_bf_anova_med)

## Bayes factor analysis
## -----
## [1] Block + id : 0.00629787 ±0.51%
## [2] Type + id : 968686.5 ±1.2%
## [3] Block + Type + id : 6444.056 ±1.78%
## [4] Block + Type + Block:Type + id : 462.3827 ±1.91%
##
## Against denominator:
##   Matched_Proportions ~ id
## ---
## Bayes factor type: BFlinearModel, JZS

• Results show how much more likely one model is compared to the baseline model (i.e., only accounting for individual differences)
• Model 1: Performs worse than the baseline model
• Model 2: Performs decisively better than the baseline mode
• Model 3: Strong support for including Block and Type together in the model but less support than Model 1 (i.e., Type alone)
• Model 4: This model is also better than the baseline but much weaker than Model 2 suggesting that Block:Type interaction is not as strong

model_all_parameters = rc_bf_anova_med[4]

# Extracts posterior samples for fixed effects
rf_bf_anova_samples <- get_parameters(
  model_all_parameters,
  effects = "fixed"
)

## Multiple `BFBayesFactor` models detected - posteriors are extracted from
##   the first numerator model.
##   See help("get_parameters", package = "insight").
# Only select fixed-effects parameters
rf_bf_parameters <- names(rf_bf_anova_samples)[!grepl("g_*", names(rf_bf_anova_samples))]

# Calculate HDI (i.e., credible intervals for fixed effects)
hdi(rf_bf_anova_samples[rf_bf_parameters])

## Highest Density Interval
## Parameter | 95% HDI

```

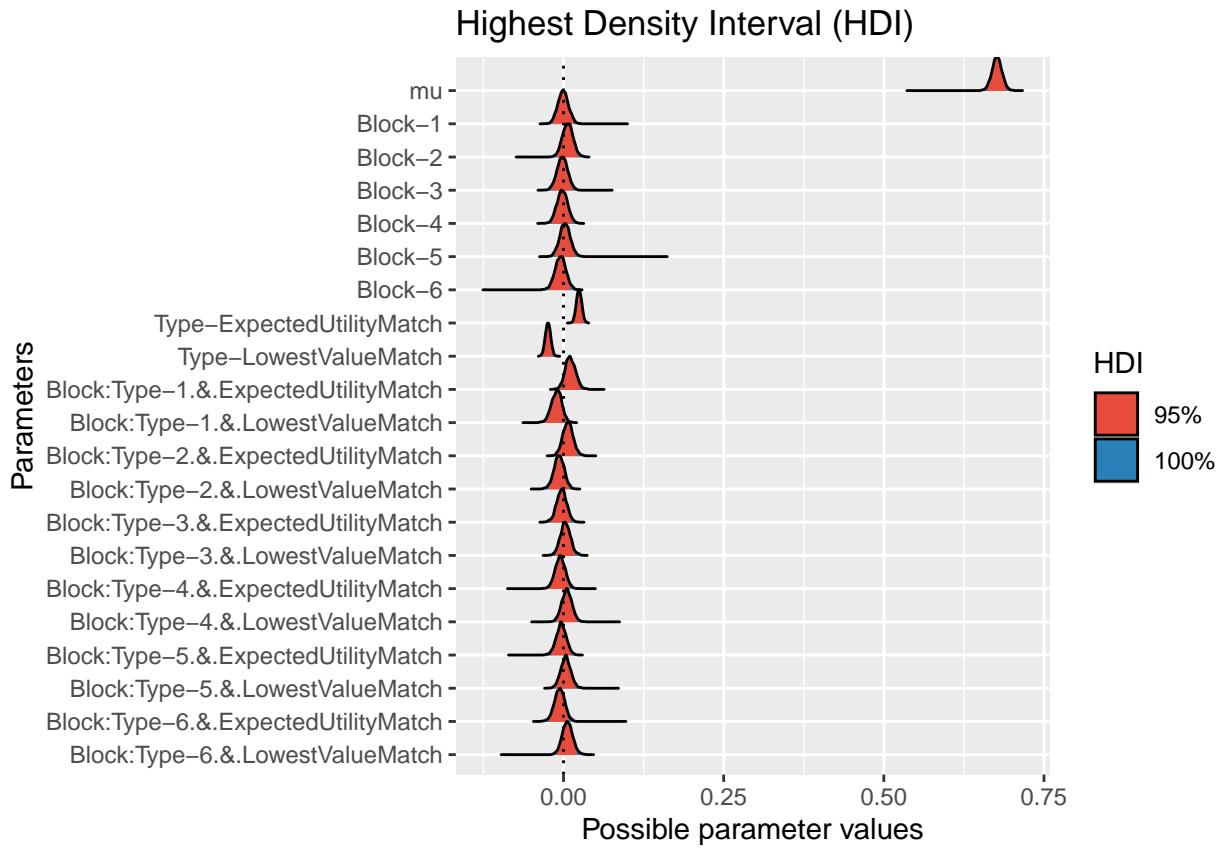
```

## -----
## mu | [ 0.66, 0.69]
## Block-1 | [-0.02, 0.02]
## Block-2 | [-0.01, 0.02]
## Block-3 | [-0.02, 0.01]
## Block-4 | [-0.02, 0.01]
## Block-5 | [-0.01, 0.02]
## Block-6 | [-0.02, 0.01]
## Type-ExpectedUtilityMatch | [ 0.02, 0.03]
## Type-LowestValueMatch | [-0.03, -0.02]
## Block>Type-1.&.ExpectedUtilityMatch | [ 0.00, 0.03]
## Block>Type-1.&.LowestValueMatch | [-0.03, 0.00]
## Block>Type-2.&.ExpectedUtilityMatch | [-0.01, 0.02]
## Block>Type-2.&.LowestValueMatch | [-0.02, 0.01]
## Block>Type-3.&.ExpectedUtilityMatch | [-0.02, 0.01]
## Block>Type-3.&.LowestValueMatch | [-0.01, 0.02]
## Block>Type-4.&.ExpectedUtilityMatch | [-0.02, 0.01]
## Block>Type-4.&.LowestValueMatch | [-0.01, 0.02]
## Block>Type-5.&.ExpectedUtilityMatch | [-0.02, 0.01]
## Block>Type-5.&.LowestValueMatch | [-0.01, 0.02]
## Block>Type-6.&.ExpectedUtilityMatch | [-0.02, 0.01]
## Block>Type-6.&.LowestValueMatch | [-0.01, 0.02]

# Plot HDI
plot(
  hdi(rf_bf_anova_samples[rf_bf_parameters]),
  data = model_all_parameters
)

## Multiple `BFBayesFactor` models detected - posteriors are extracted from
##   the first numerator model.
##   See help("get_parameters", package = "insight").

```



- HDI gives the range of values that have the highest probability of being true given the data
 - If a 95% HDI is [2.1, 4.5], it means there's a 95% probability that the true parameter value is within this range.

```
# Averaging across models
bf_inclusion(
  rc_bf_anova_med,
  match_models = TRUE
)

## Inclusion Bayes Factors (Model Averaged)
##
##          P(prior) P(posterior) Inclusion BF
## id        1.00       1.00
## Block     0.40      6.61e-03      0.007
## Type      0.40       1.00      9.69e+05
## Block:Type 0.20      4.74e-04      0.072
##
## * Compared among: matched models only
## *   Priors odds: uniform-equal

# Paired-samples t-test
ttestBF(
  x = rc_data_tidy$Matched_Proportions[rc_data_tidy$Type == 'ExpectedUtilityMatch'],
  y = rc_data_tidy$Matched_Proportions[rc_data_tidy$Type == 'LowestValueMatch'],
  rcale = 'medium'
)
```

```

## Bayes factor analysis
## -----
## [1] Alt., r=0.707 : 161064.6 ±0%
##
## Against denominator:
##   Null, mu1-mu2 = 0
## ---
## Bayes factor type: BFindepSample, JZS
prior_sd = list(0.25, 0.5, 0.75, 1, 5, 10)

for (sd in prior_sd) {

  print("-----")
  print(cat("PRIOR WITH SD ", sd))

  # Fits a Bayesian linear model to the data using Markov Chain Monte Carlo (MCMC) sampling via Stan
  rc_stan_anova <- stan_lmer(
    Matched_Proportions ~ Block * Type + (1|id),    # (1/id) accounts for individual-level variation
    data = rc_data_tidy,
    refresh = 0,
    prior = normal(scale = sd),
    diagnostic_file = file.path(tempdir(), "df.csv")
  )

  print(rc_stan_anova)
  print("-----")

  # Plot HDI
  rc_stan_hdi <- hdi(rc_stan_anova)
  rc_stan_hdi_plot <- plot(hdi(rc_stan_anova)) + ggtitle(paste("HDI for Prior for SD ", sd, "\n"))

  print(rc_stan_hdi)
  print(rc_stan_hdi_plot)
  print("-----")

  # Compare the posterior distribution to the prior distribution
  rc_stan_prior_post_plot2 <- posterior_vs_prior(
    rc_stan_anova,
    pars = "beta"
  )

  print(rc_stan_prior_post_plot2)
  print("-----")
}

## [1] "-----"
## PRIOR WITH SD 0.25NULL
## stan_lmer
##   family: gaussian [identity]
##   formula: Matched_Proportions ~ Block * Type + (1 | id)
##   observations: 336
##   -----
##   (Intercept)      Median MAD_SD
##                   0.7     0.0

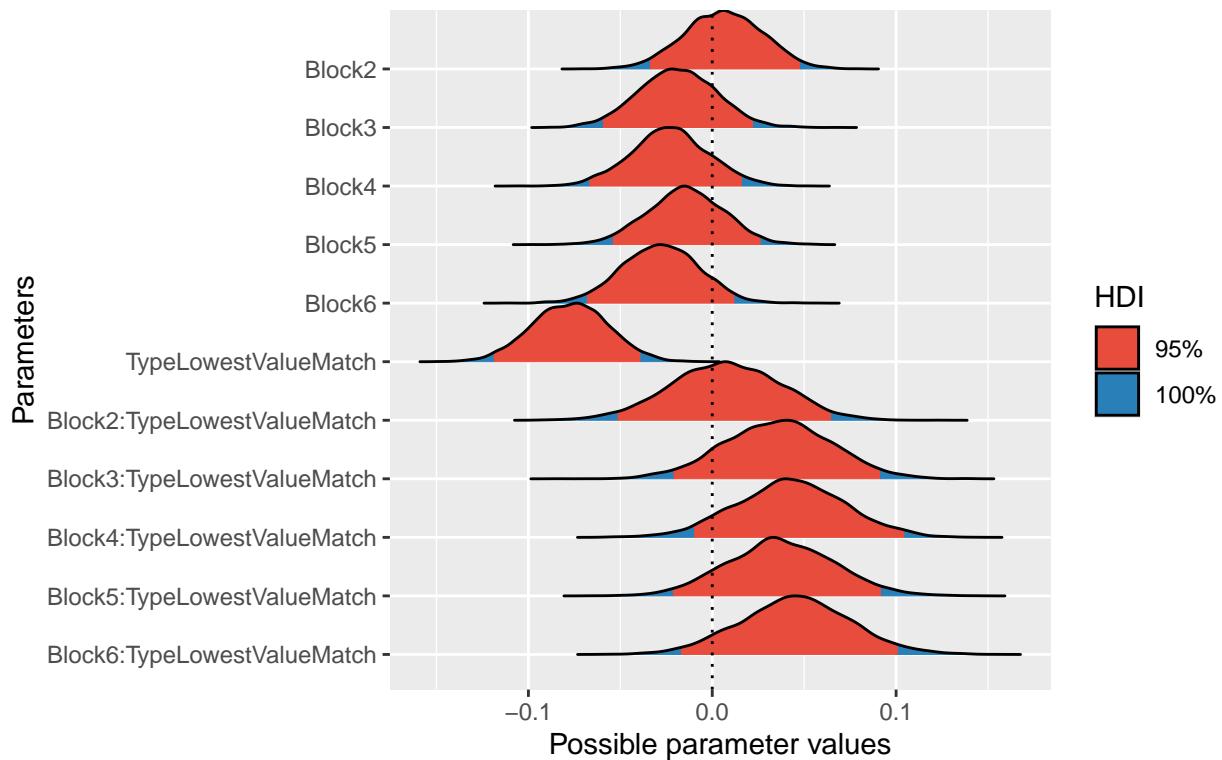
```

```

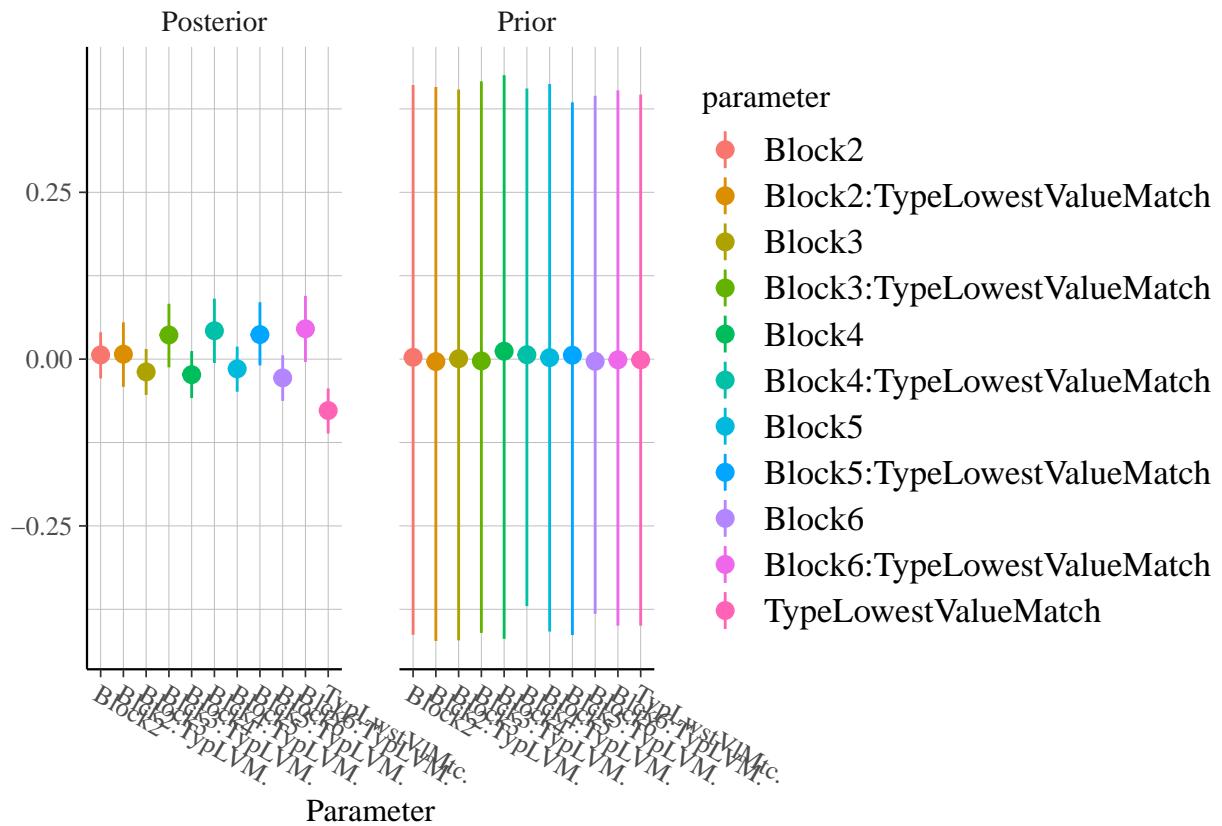
## Block2          0.0    0.0
## Block3          0.0    0.0
## Block4          0.0    0.0
## Block5          0.0    0.0
## Block6          0.0    0.0
## TypeLowestValueMatch -0.1  0.0
## Block2:TypeLowestValueMatch 0.0  0.0
## Block3:TypeLowestValueMatch 0.0  0.0
## Block4:TypeLowestValueMatch 0.0  0.0
## Block5:TypeLowestValueMatch 0.0  0.0
## Block6:TypeLowestValueMatch 0.0  0.0
##
## Auxiliary parameter(s):
##     Median MAD_SD
## sigma 0.1    0.0
##
## Error terms:
## Groups   Name      Std.Dev.
## id       (Intercept) 0.028
## Residual           0.079
## Num. levels: id 28
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----"
## Highest Density Interval
##
## Parameter          |      95% HDI
## -----
## (Intercept)        | [ 0.68,  0.74]
## Block2            | [-0.03,  0.05]
## Block3            | [-0.06,  0.02]
## Block4            | [-0.07,  0.02]
## Block5            | [-0.05,  0.03]
## Block6            | [-0.07,  0.01]
## TypeLowestValueMatch | [-0.12, -0.04]
## Block2:TypeLowestValueMatch | [-0.05,  0.06]
## Block3:TypeLowestValueMatch | [-0.02,  0.09]
## Block4:TypeLowestValueMatch | [-0.01,  0.10]
## Block5:TypeLowestValueMatch | [-0.02,  0.09]
## Block6:TypeLowestValueMatch | [-0.02,  0.10]

```

HDI for Prior for SD 0.25



```
## [1] "-----"
```

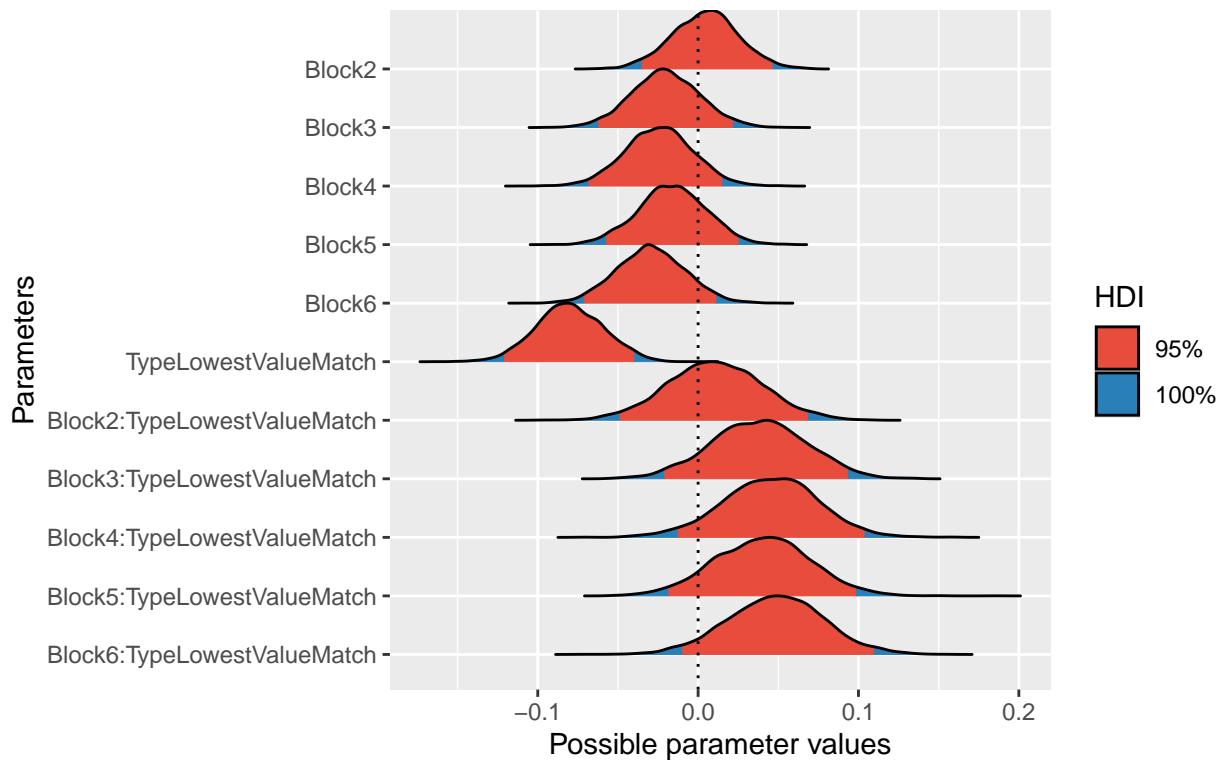


```

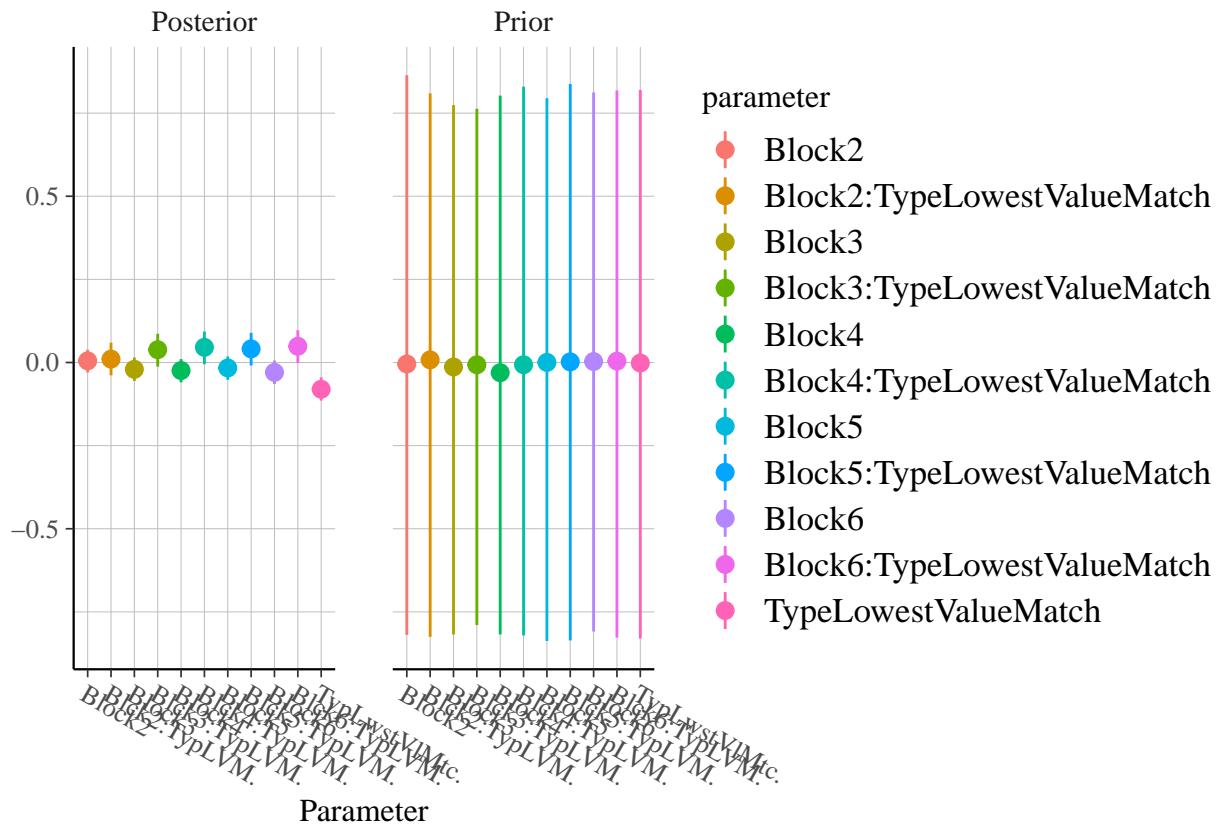
## [1] "-----"
## [1] -----
## PRIOR WITH SD 0.5NULL
## stan_lmer
##   family:      gaussian [identity]
##   formula:     Matched_Proportions ~ Block * Type + (1 | id)
##   observations: 336
## -----
##                               Median MAD_SD
## (Intercept)                  0.7    0.0
## Block2                      0.0    0.0
## Block3                      0.0    0.0
## Block4                      0.0    0.0
## Block5                      0.0    0.0
## Block6                      0.0    0.0
## TypeLowestValueMatch       -0.1    0.0
## Block2:TypeLowestValueMatch  0.0    0.0
## Block3:TypeLowestValueMatch  0.0    0.0
## Block4:TypeLowestValueMatch  0.0    0.0
## Block5:TypeLowestValueMatch  0.0    0.0
## Block6:TypeLowestValueMatch  0.0    0.0
##
## Auxiliary parameter(s):
##   Median MAD_SD
## sigma 0.1    0.0
##
## Error terms:
##   Groups   Name      Std.Dev.
##   id       (Intercept) 0.028
##   Residual           0.079
## Num. levels: id 28
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----"
## Highest Density Interval
##
## Parameter          |      95% HDI
## -----
## (Intercept)        | [ 0.69,  0.75]
## Block2            | [-0.04,  0.05]
## Block3            | [-0.06,  0.02]
## Block4            | [-0.07,  0.01]
## Block5            | [-0.06,  0.03]
## Block6            | [-0.07,  0.01]
## TypeLowestValueMatch | [-0.12, -0.04]
## Block2:TypeLowestValueMatch | [-0.05,  0.07]
## Block3:TypeLowestValueMatch | [-0.02,  0.09]
## Block4:TypeLowestValueMatch | [-0.01,  0.10]
## Block5:TypeLowestValueMatch | [-0.02,  0.10]
## Block6:TypeLowestValueMatch | [-0.01,  0.11]

```

HDI for Prior for SD 0.5



```
## [1] "-----"
```

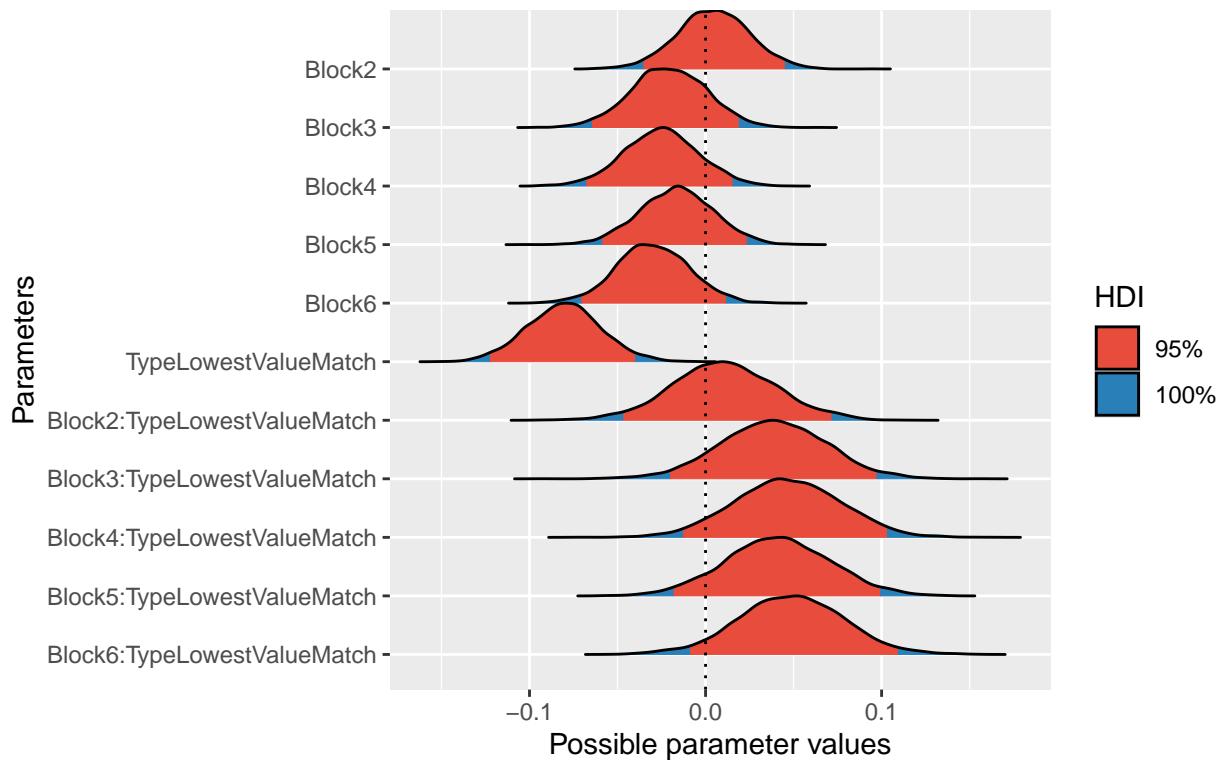


```

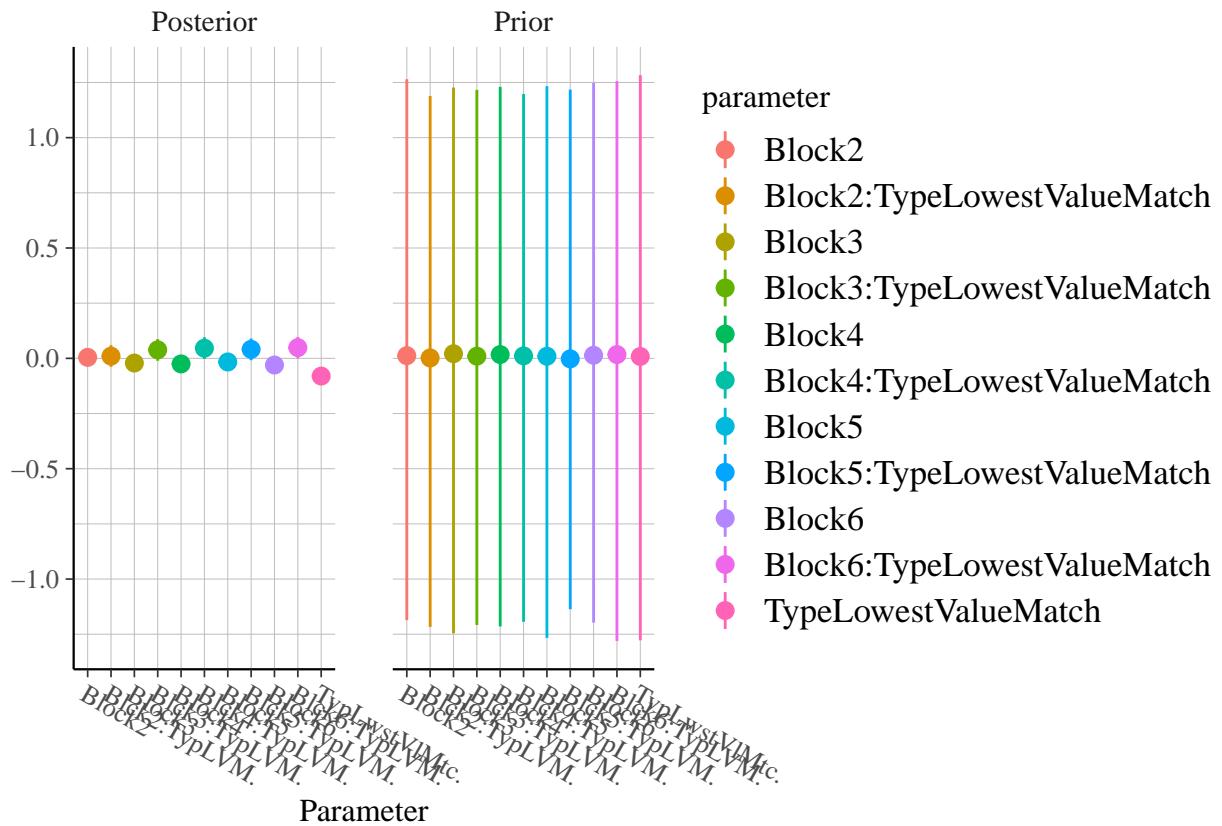
## [1] "-----"
## [1] -----
## PRIOR WITH SD  0.75NULL
## stan_lmer
##   family:      gaussian [identity]
##   formula:     Matched_Proportions ~ Block * Type + (1 | id)
##   observations: 336
## -----
##                               Median MAD_SD
## (Intercept)                  0.7    0.0
## Block2                      0.0    0.0
## Block3                      0.0    0.0
## Block4                      0.0    0.0
## Block5                      0.0    0.0
## Block6                      0.0    0.0
## TypeLowestValueMatch       -0.1    0.0
## Block2:TypeLowestValueMatch  0.0    0.0
## Block3:TypeLowestValueMatch  0.0    0.0
## Block4:TypeLowestValueMatch  0.0    0.0
## Block5:TypeLowestValueMatch  0.0    0.0
## Block6:TypeLowestValueMatch  0.0    0.0
##
## Auxiliary parameter(s):
##   Median MAD_SD
## sigma 0.1    0.0
##
## Error terms:
##   Groups   Name      Std.Dev.
##   id       (Intercept) 0.028
##   Residual           0.079
## Num. levels: id 28
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----"
## Highest Density Interval
##
## Parameter          |      95% HDI
## -----
## (Intercept)        | [ 0.69,  0.75]
## Block2            | [-0.04,  0.05]
## Block3            | [-0.06,  0.02]
## Block4            | [-0.07,  0.02]
## Block5            | [-0.06,  0.02]
## Block6            | [-0.07,  0.01]
## TypeLowestValueMatch | [-0.12, -0.04]
## Block2:TypeLowestValueMatch | [-0.05,  0.07]
## Block3:TypeLowestValueMatch | [-0.02,  0.10]
## Block4:TypeLowestValueMatch | [-0.01,  0.10]
## Block5:TypeLowestValueMatch | [-0.02,  0.10]
## Block6:TypeLowestValueMatch | [-0.01,  0.11]

```

HDI for Prior for SD 0.75



```
## [1] "-----"
```

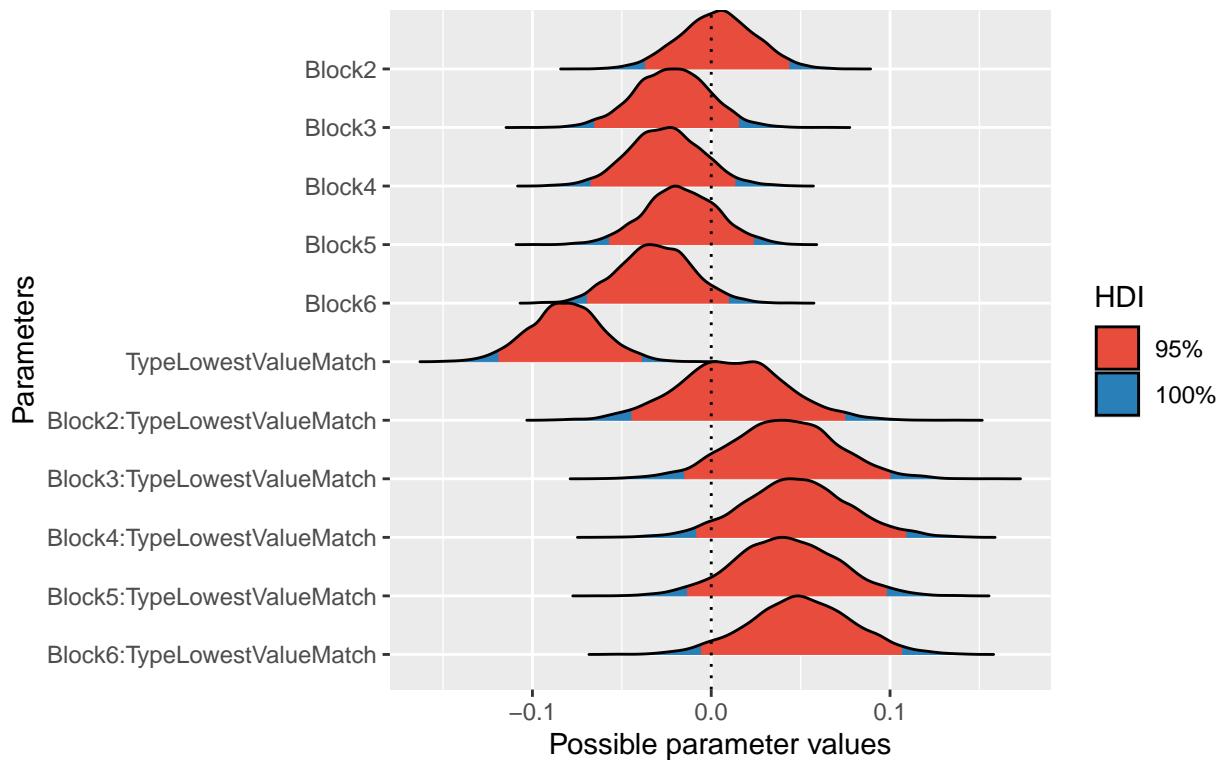


```

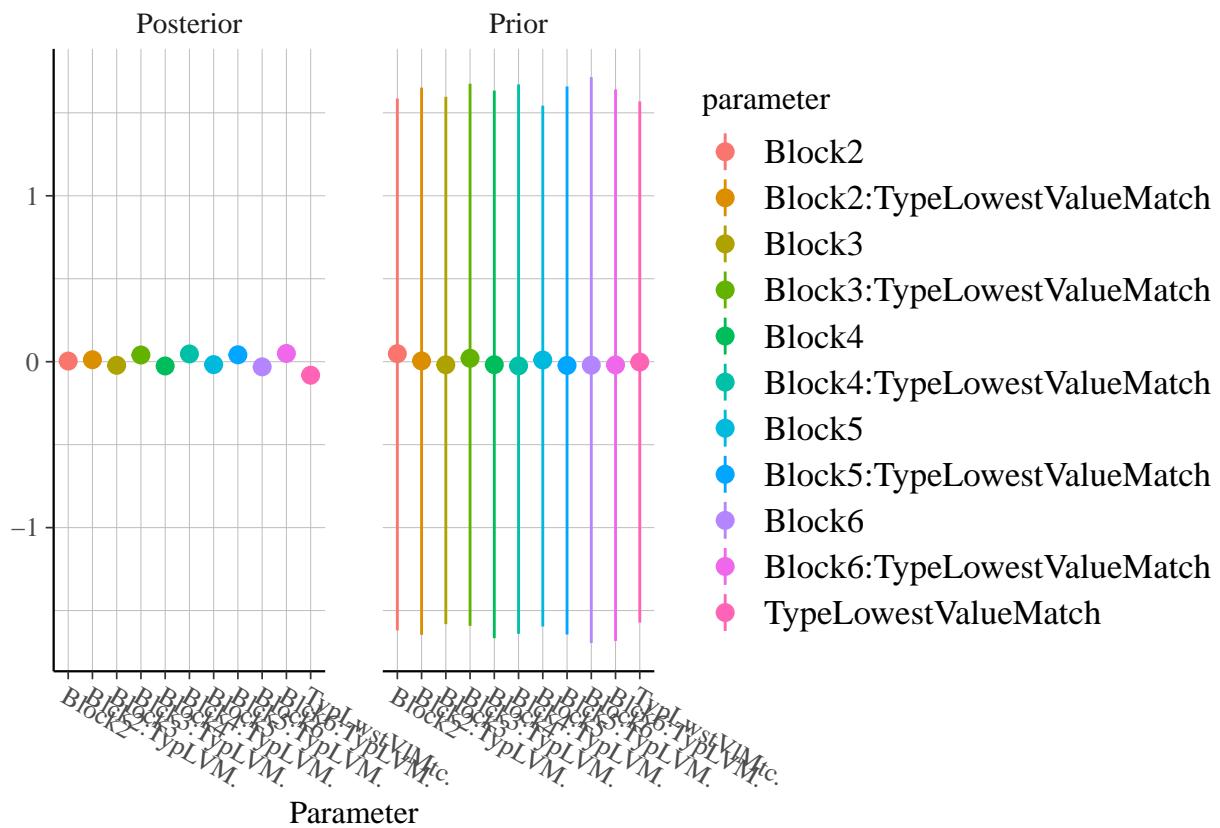
## [1] "-----"
## [1] -----
## PRIOR WITH SD 1NULL
## stan_lmer
##   family:      gaussian [identity]
##   formula:     Matched_Proportions ~ Block * Type + (1 | id)
##   observations: 336
## -----
##                               Median MAD_SD
## (Intercept)                  0.7    0.0
## Block2                      0.0    0.0
## Block3                      0.0    0.0
## Block4                      0.0    0.0
## Block5                      0.0    0.0
## Block6                      0.0    0.0
## TypeLowestValueMatch       -0.1    0.0
## Block2:TypeLowestValueMatch  0.0    0.0
## Block3:TypeLowestValueMatch  0.0    0.0
## Block4:TypeLowestValueMatch  0.0    0.0
## Block5:TypeLowestValueMatch  0.0    0.0
## Block6:TypeLowestValueMatch  0.0    0.0
##
## Auxiliary parameter(s):
##   Median MAD_SD
## sigma 0.1    0.0
##
## Error terms:
##   Groups   Name      Std.Dev.
##   id       (Intercept) 0.028
##   Residual           0.079
## Num. levels: id 28
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----"
## Highest Density Interval
##
## Parameter          |      95% HDI
## -----
## (Intercept)        | [ 0.69,  0.75]
## Block2            | [-0.04,  0.04]
## Block3            | [-0.07,  0.02]
## Block4            | [-0.07,  0.01]
## Block5            | [-0.06,  0.02]
## Block6            | [-0.07,  0.01]
## TypeLowestValueMatch | [-0.12, -0.04]
## Block2:TypeLowestValueMatch | [-0.04,  0.08]
## Block3:TypeLowestValueMatch | [-0.02,  0.10]
## Block4:TypeLowestValueMatch | [-0.01,  0.11]
## Block5:TypeLowestValueMatch | [-0.01,  0.10]
## Block6:TypeLowestValueMatch | [-0.01,  0.11]

```

HDI for Prior for SD 1



```
## [1] "-----"
```

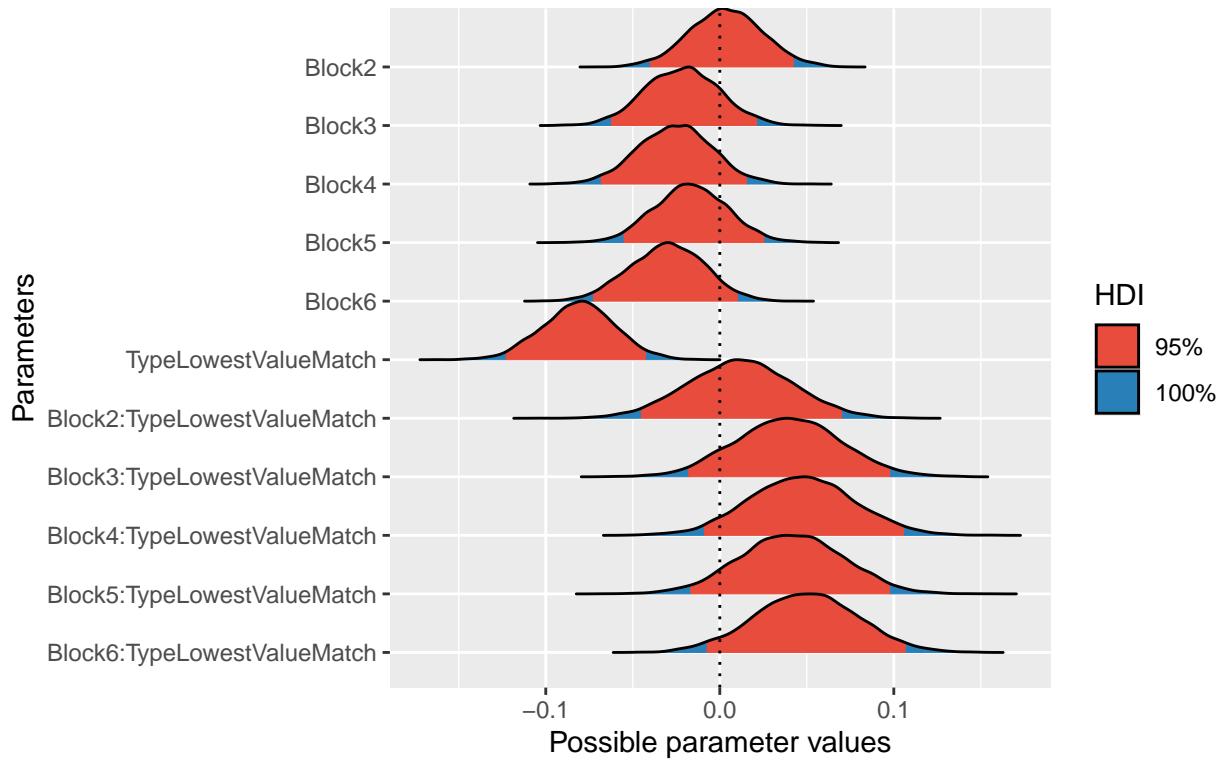


```

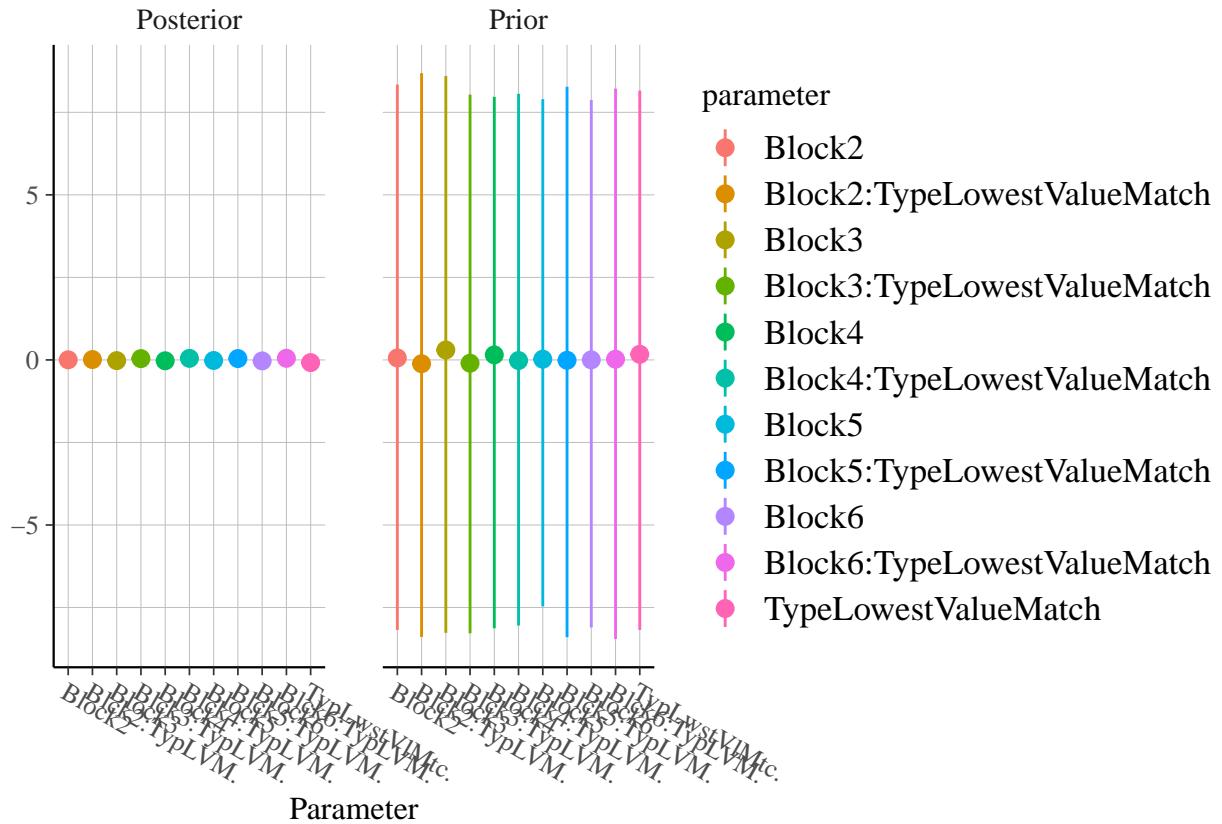
## [1] "-----"
## [1] -----
## PRIOR WITH SD 5NULL
## stan_lmer
##   family:      gaussian [identity]
##   formula:     Matched_Proportions ~ Block * Type + (1 | id)
##   observations: 336
## -----
##                               Median MAD_SD
## (Intercept)                  0.7    0.0
## Block2                      0.0    0.0
## Block3                      0.0    0.0
## Block4                      0.0    0.0
## Block5                      0.0    0.0
## Block6                      0.0    0.0
## TypeLowestValueMatch       -0.1    0.0
## Block2:TypeLowestValueMatch  0.0    0.0
## Block3:TypeLowestValueMatch  0.0    0.0
## Block4:TypeLowestValueMatch  0.0    0.0
## Block5:TypeLowestValueMatch  0.0    0.0
## Block6:TypeLowestValueMatch  0.0    0.0
##
## Auxiliary parameter(s):
##   Median MAD_SD
## sigma 0.1    0.0
##
## Error terms:
##   Groups   Name      Std.Dev.
##   id       (Intercept) 0.028
##   Residual           0.079
## Num. levels: id 28
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----"
## Highest Density Interval
##
## Parameter          |      95% HDI
## -----
## (Intercept)        | [ 0.68,  0.75]
## Block2            | [-0.04,  0.04]
## Block3            | [-0.06,  0.02]
## Block4            | [-0.07,  0.02]
## Block5            | [-0.06,  0.03]
## Block6            | [-0.07,  0.01]
## TypeLowestValueMatch | [-0.12, -0.04]
## Block2:TypeLowestValueMatch | [-0.05,  0.07]
## Block3:TypeLowestValueMatch | [-0.02,  0.10]
## Block4:TypeLowestValueMatch | [-0.01,  0.11]
## Block5:TypeLowestValueMatch | [-0.02,  0.10]
## Block6:TypeLowestValueMatch | [-0.01,  0.11]

```

HDI for Prior for SD 5



```
## [1] "-----"
```

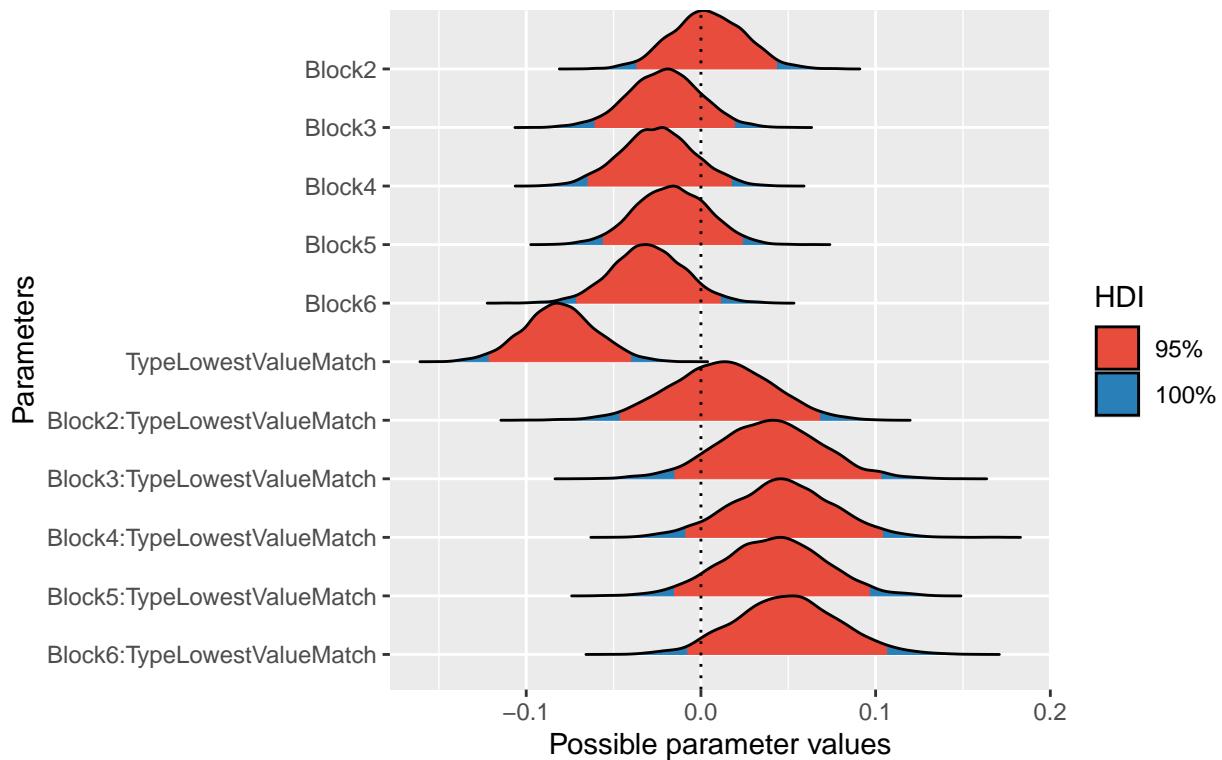


```

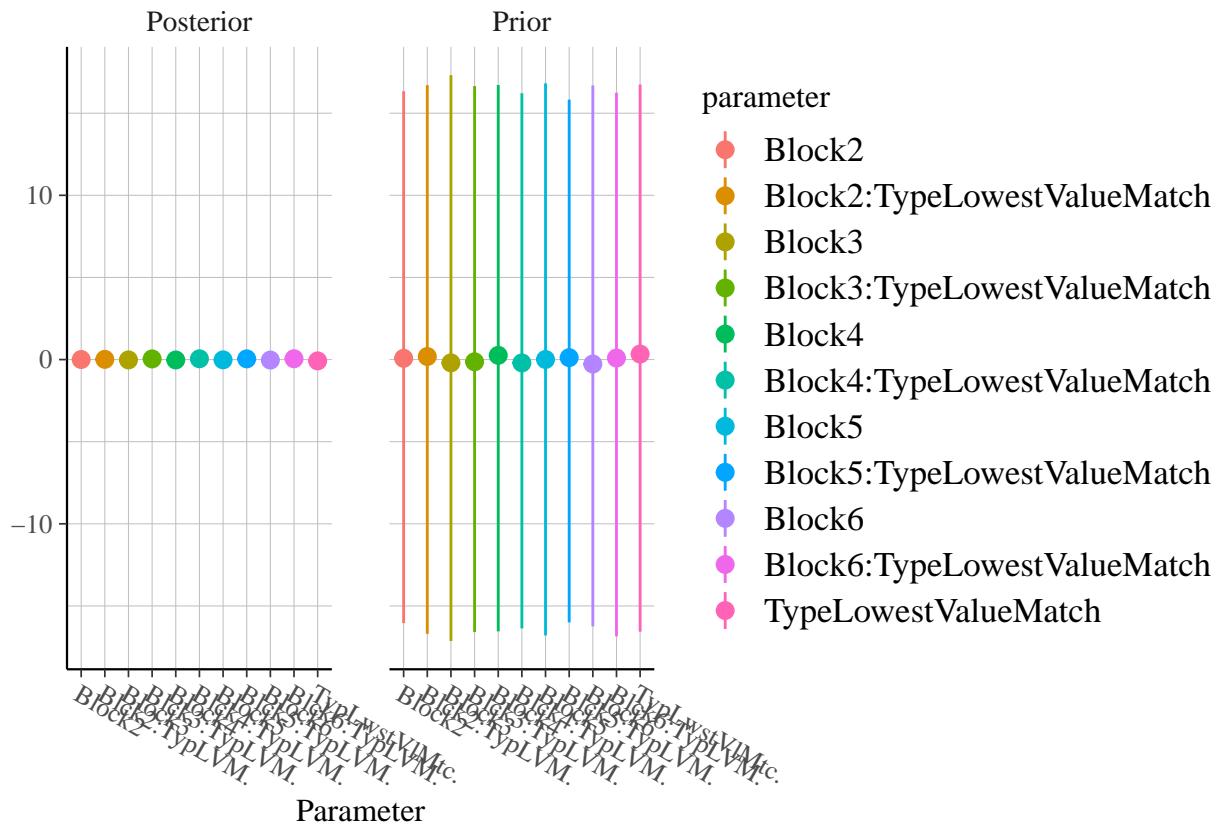
## [1] "-----"
## [1] -----
## PRIOR WITH SD 10NULL
## stan_lmer
##   family:      gaussian [identity]
##   formula:     Matched_Proportions ~ Block * Type + (1 | id)
##   observations: 336
## -----
##                               Median MAD_SD
## (Intercept)                  0.7    0.0
## Block2                      0.0    0.0
## Block3                      0.0    0.0
## Block4                      0.0    0.0
## Block5                      0.0    0.0
## Block6                      0.0    0.0
## TypeLowestValueMatch       -0.1    0.0
## Block2:TypeLowestValueMatch  0.0    0.0
## Block3:TypeLowestValueMatch  0.0    0.0
## Block4:TypeLowestValueMatch  0.0    0.0
## Block5:TypeLowestValueMatch  0.0    0.0
## Block6:TypeLowestValueMatch  0.0    0.0
##
## Auxiliary parameter(s):
##   Median MAD_SD
## sigma 0.1    0.0
##
## Error terms:
##   Groups   Name      Std.Dev.
##   id       (Intercept) 0.028
##   Residual           0.079
## Num. levels: id 28
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
## [1] "-----"
## Highest Density Interval
##
## Parameter          |      95% HDI
## -----
## (Intercept)        | [ 0.69,  0.75]
## Block2            | [-0.04,  0.04]
## Block3            | [-0.06,  0.02]
## Block4            | [-0.06,  0.02]
## Block5            | [-0.06,  0.02]
## Block6            | [-0.07,  0.01]
## TypeLowestValueMatch | [-0.12, -0.04]
## Block2:TypeLowestValueMatch | [-0.05,  0.07]
## Block3:TypeLowestValueMatch | [-0.02,  0.10]
## Block4:TypeLowestValueMatch | [-0.01,  0.10]
## Block5:TypeLowestValueMatch | [-0.02,  0.10]
## Block6:TypeLowestValueMatch | [-0.01,  0.11]

```

HDI for Prior for SD 10



```
## [1] "-----"
```



```
## [1] "-----"
```

- **Left Side (Posterior Distributions)**

- The posterior estimates of parameters after observing the data
- Points represent the mean or median estimate for each parameter
- Vertical bars indicate uncertainty (credible intervals)

- **Right Side (Prior Distributions)**

- The priors used before seeing any data
- These look much wider, indicating less certainty compared to the posterior.