# Multi100: Reanalysis of “Time horizon and cooperation in continuous time”

Maarten Speekenbrink (W9XD2)

2022-05-24

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# also need haven, afex
```

Here, we re-analyse data to assess evidence for the following claim:

> . . . in the short duration treatments, cooperation rates are significantly higher with a deterministic horizon than with a stochastic horizon (p 596)

## Design

The study has a design where two groups of 24 participants are assigned to each of 5 conditions. The conditions are: Short-Deterministic, Short-Stochastic, Long-Deterministic, Long-Stochastic, and Variable-Deterministic. The latter condition is not included in most analyses, where the experiment is treated as a 2 (Length: Long, Short) by 2 (Horizon: Deterministic, Stochastic) between-subjects design.

Within each group, each participant was paired with each other participant to play 23 continuous-time Prisoner’s Dilemma “supergames”. To assess the main claim, the main variable of interest is the cooperation rate, which is the proportion of time a player chooses the cooperative action within a supergame.

## Data

The supplementary materials provide several datasets. The “originals” folder contains 12 (tab delimited) excel sheets for each session. The “elaborations” folder contains processed data in `.dta` (STATA) format, and STATA scripts for the analysis. The folder “strategies” also contains processed data, and MATLAB files for analysing strategies.

Although it might have been best to use the data from the “originals” folder, due to time constraints, I have chosen to use the `data.dta` file in the “elaborations” folder, with additional use of the `subjects_data_ready.dta` file to obtain the identity of the experimental conditions.

```r
# read the subjects_data_ready.dta file which contains the conditions for each session
pdat <- haven::read_dta("replication_files/elaborations/subjects_data_ready.dta")
# create a table with condition identities for each session
pdat <- pdat %>%
  group_by(condition, session) %>%
  filter(row_number() == 1) %>%
  arrange(session) %>%
  select(condition, session)
# read the main data file to use
dat <- haven::read_dta("replication_files/elaborations/data.dta")
# show a quick overview of this data
head(dat)
```

```
## # A tibble: 6 x 72
##   session     treatment Period Subject Group Profit TotalProfit match  coop
##   <chr>           <dbl>  <dbl>   <dbl> <dbl>  <dbl>       <dbl> <dbl> <dbl>
## 1 101021_1136         1      1       1     1      0           0    -2     1
## 2 101021_1136         2      1       1     1      0           0    -1     0
## 3 101021_1136         3      1       1     1      0           0     0     0
## 4 101021_1136         4      1       1     1      0           0    NA    NA
## 5 101021_1136         5      1       1     1      0           0     1     0
## 6 101021_1136         5      2       1     1      0           0     2     0
## # ... with 63 more variables: othercoop <dbl>, coop_frequency <dbl>,
## #   num_switches <dbl>, payoff <dbl>, answer <dbl>, partner <dbl>,
## #   starttime1 <dbl>, starttime2 <dbl>, starttime3 <dbl>, starttime4 <dbl>,
## #   answer1 <dbl>, answer2 <dbl>, answer3 <dbl>, answer4 <dbl>, alert1 <dbl>,
## #   alert2 <dbl>, alert3 <dbl>, alert4 <dbl>, step <dbl>, sigma <dbl>,
## #   mu_rr <dbl>, mu_rl <dbl>, mu_ll <dbl>, c_r <dbl>, c_l <dbl>,
## #   num_periods <dbl>, expected_duration <dbl>, endowment <dbl>, ...
```

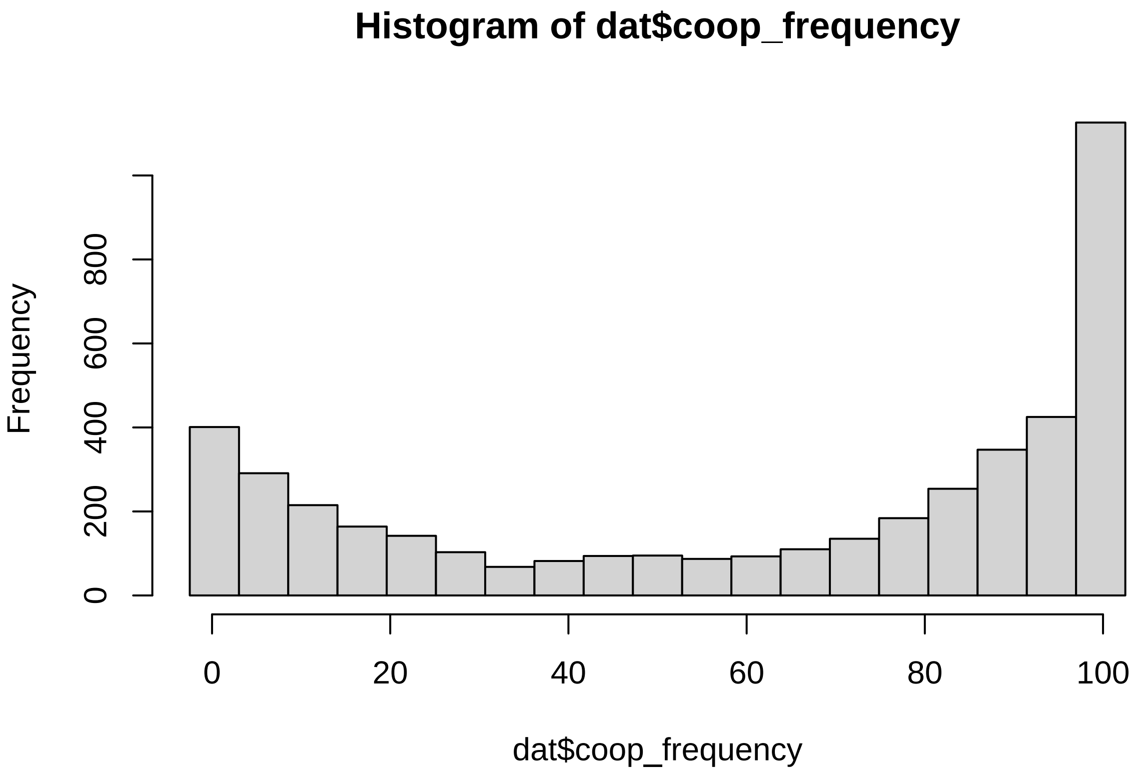
Unfortunately, there is no clear codebook to go along with the data. The names are usually descriptive enough (and labels are provided for many variables) to guess what the variables refer too.

I have done some minimal processing of the data to (1) remove the Variable-Deterministic condition and to create unique IDs for participants and their partners in the games.

```r
dat <- dat %>%
  filter(treatment == 5) %>%
  mutate(session = as.numeric(factor(session)))

# take condition from the table we created earlier
dat$condition <- pdat$condition[as.numeric(dat$session)]
# we can check correspondence by
# haven::read_dta("replication_files/elaborations/subjects_data_ready.dta") %>% group_by(condition) %>%
# dat %>% group_by(condition) %>% summarise(perf = mean(coop_frequency))

# create unique IDs for participants and partners
dat$u_participant <- interaction(dat$Subject, dat$session)
dat$u_partner <- interaction(dat$partner, dat$session)

# use only the first four conditions (not the Variable-Deterministic)
dat <- subset(dat, condition != 5)
```

Looking at the cooperation rates, we see and interesting distribution, with very low cooperation rates and very high cooperation rates more likely than medium cooperation rates:

```
hist(dat$coop_frequency, breaks=seq(-2.5,102.5, length=20))
```




Cooperation rates of exactly 0% and exactly 100% occur often, and excess observations on the bound of the scale will likely cause issues for linear models.

## Linear mixed-effects model

I deemed a linear mixed-effects model to be the most suitable for the data, with fixed effects for condition. As rates of cooperation likely depend on both players in a game, and because we have multiple observations for both players, it makes sense to include (crossed) random effects for both. The fixed effects are slopes for `duration` (short -1, long: 1), `horizon` (deterministic: 1, stochastic: -1), and their interaction. The model includes random intercepts for participants and for partners. The model can be written as:

$$\begin{aligned}
Y_{ij} &= \beta_0 + \gamma_i + \gamma_j + \beta_d \texttt{duration}_i + \beta_h \texttt{horizon}_i + \beta_{dh}(\texttt{duration} \times \texttt{horizon})_i + \epsilon_{ij} \\
\gamma_i &\sim \textbf{Normal}(0, \sigma_{\text{player}}) \\
\gamma_j &\sim \textbf{Normal}(0, \sigma_{\text{partner}}) \\
\epsilon_{ij} &\sim \textbf{Normal}(0, \sigma_\epsilon)
\end{aligned}$$

where $Y_{ij}$ is the cooperation rate in a game where participant $i$ plays with partner $j$ (both unique identifiers for participants over sessions), and $\gamma_i$, $\gamma_j$, and $\epsilon_{ij}$ are independently Normally distributed.

As cooperation rate is a percentage, there will likely be issues with the assumption of Normal distributed residuals. Although we could choose to use a *generalized* linear mixed-effects model instead, the appropriate distribution and link function for this data is not obvious. Instead, I chose to use a Box-Cox transformation with a (maximum likelihood) estimated value for the transformation parameter $\lambda$.

I believe this model strikes a reasonable balance between simplicity and allowing for straightforward conclusions, as well as respecting the somewhat complicated design of the study. The main claim to be tested is based on a comparison within the short-duration conditions. I will use contrast analysis (with the `emmeans` package) to test for the effect of `horizon` within the short- length conditions, in order to assess this main claim.

We start by fitting the model for the untransformed cooperation rates. Note that I add a small constant (.01) to all cooperation rates (percentages) to allow the later Box-Cox transformation, which requires strictly positive values.

```r
# create contrast codes for duration (called length here) and horizon
# note: condition is coded as follows:
# 1     long-deterministic
# 2        long-stochastic
# 3    short-deterministic
# 4       short-stochastic
dat$length <- 1
dat$length[dat$condition %in% c(3,4)] <- -1
dat$horizon <- 1
dat$horizon[dat$condition %in% c(2,4)] <- -1

# estimate the model, using afex and Kenward-Roger approximation to the degrees of freedom
mod0 <- afex::mixed(coop_frequency ~ horizon*length + (1|u_participant) + (1|u_partner), data=dat, metho
```

```
## Contrasts set to contr.sum for the following variables: u_participant, u_partner
```

```r
# we find a significant effect of length and duration*length interaction
mod0
```

```
## Mixed Model Anova Table (Type 3 tests, KR-method)
##
## Model: coop_frequency ~ horizon * length + (1 | u_participant) + (1 |
## Model:     u_partner)
## Data: dat
##           Effect        df       F p.value
## 1        horizon 1, 304.59    2.40    .122
## 2         length 1, 304.59 7.33 **    .007
## 3 horizon:length 1, 304.59  3.93 *    .048
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

```r
summary(mod0)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: coop_frequency ~ horizon * length + (1 | u_participant) + (1 |
##     u_partner)
##    Data: data
##
## REML criterion at convergence: 43685.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7393 -0.7352  0.2366  0.7138  2.8793
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  u_participant (Intercept)  246.2   15.69
##  u_partner     (Intercept)  172.3   13.13
##  Residual                  1000.4   31.63
## Number of obs: 4416, groups:  u_participant, 192; u_partner, 192
##
```

```
## Fixed effects:
##                 Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)       61.995      1.551 300.010  39.967  < 2e-16 ***
## horizon            2.403      1.551 300.010   1.549  0.12237
## length             4.199      1.551 300.010   2.707  0.00718 **
## horizon:length    -3.077      1.551 300.010  -1.984  0.04821 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) horizn length
## horizon     0.000
## length      0.000  0.000
## horzn:lngth 0.000  0.000  0.000
```

```
# get the estimated marginal means
emm0 <- emmeans::emmeans(mod0, ~ horizon*length, lmerTest.limit = 5000, pbkrtest.limit = 5000)
emm0
```

```
##  horizon length emmean  SE  df lower.CL upper.CL
##       -1     -1   52.3 3.1 305     46.2     58.4
##        1     -1   63.3 3.1 305     57.2     69.4
##       -1      1   66.9 3.1 305     60.8     73.0
##        1      1   65.5 3.1 305     59.4     71.6
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

```
# use contrast analysis to compare horizon within short-duration conditions
emmeans::contrast(emm0, method=list(test=c(-1,1,0,0)))
```

```
##  contrast estimate   SE  df t.ratio p.value
##  test           11 4.39 305   2.498  0.0130
##
## Degrees-of-freedom method: kenward-roger
```

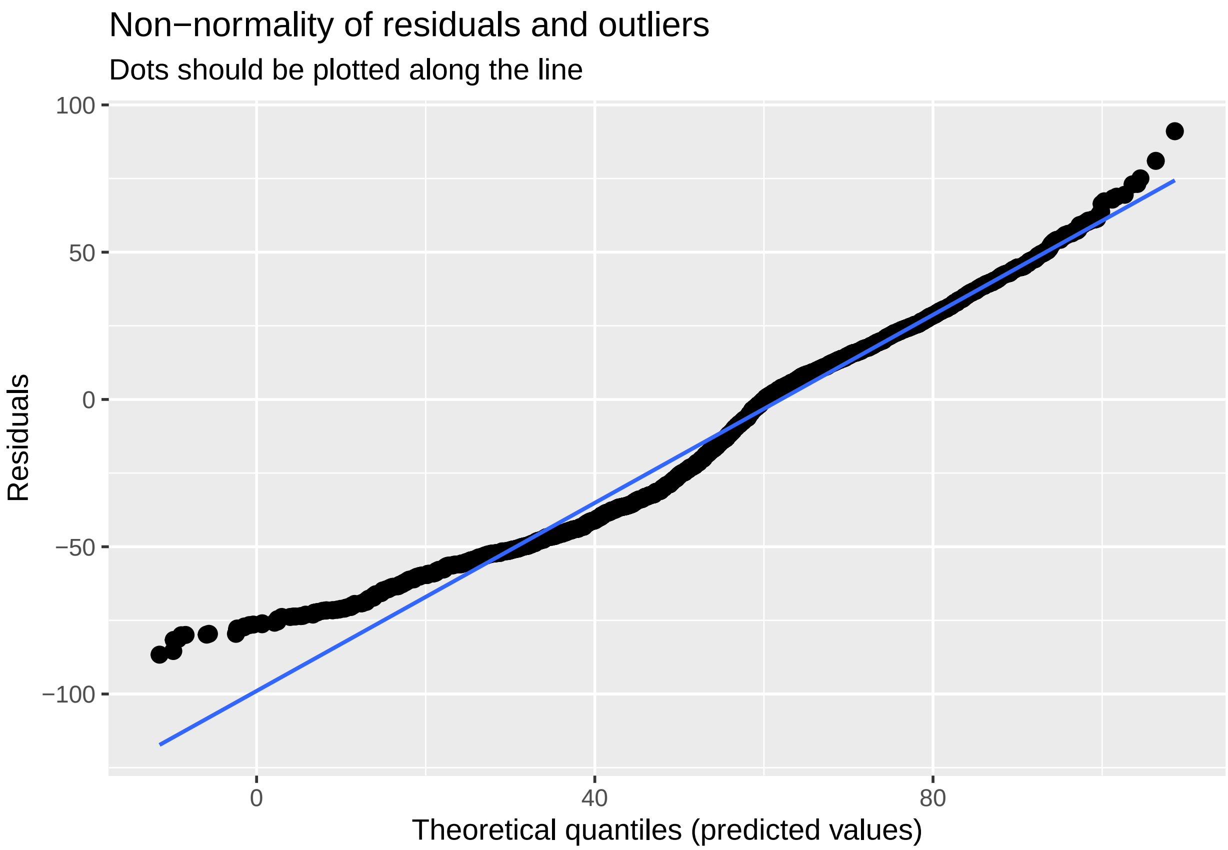
This model shows a significant effect of duration, $F(1, 304.59) = 7.33$, $p = .007$, and a significant interaction between duration and horizon, $F(1, 304.59) = 3.93$, $p = .048$. The contrast between the short-stochastic and short-deterministic condition is also significant, $\Delta M = 10.96$, 95% CI $[2.33, 19.59]$, $t(304.59) = 2.50$, $p = .013$.

The residuals of the model show clear deviations from Normality

```
diag_plots_0 <- sjPlot::plot_model(mod0$full_model, 'diag')
diag_plots_0
```

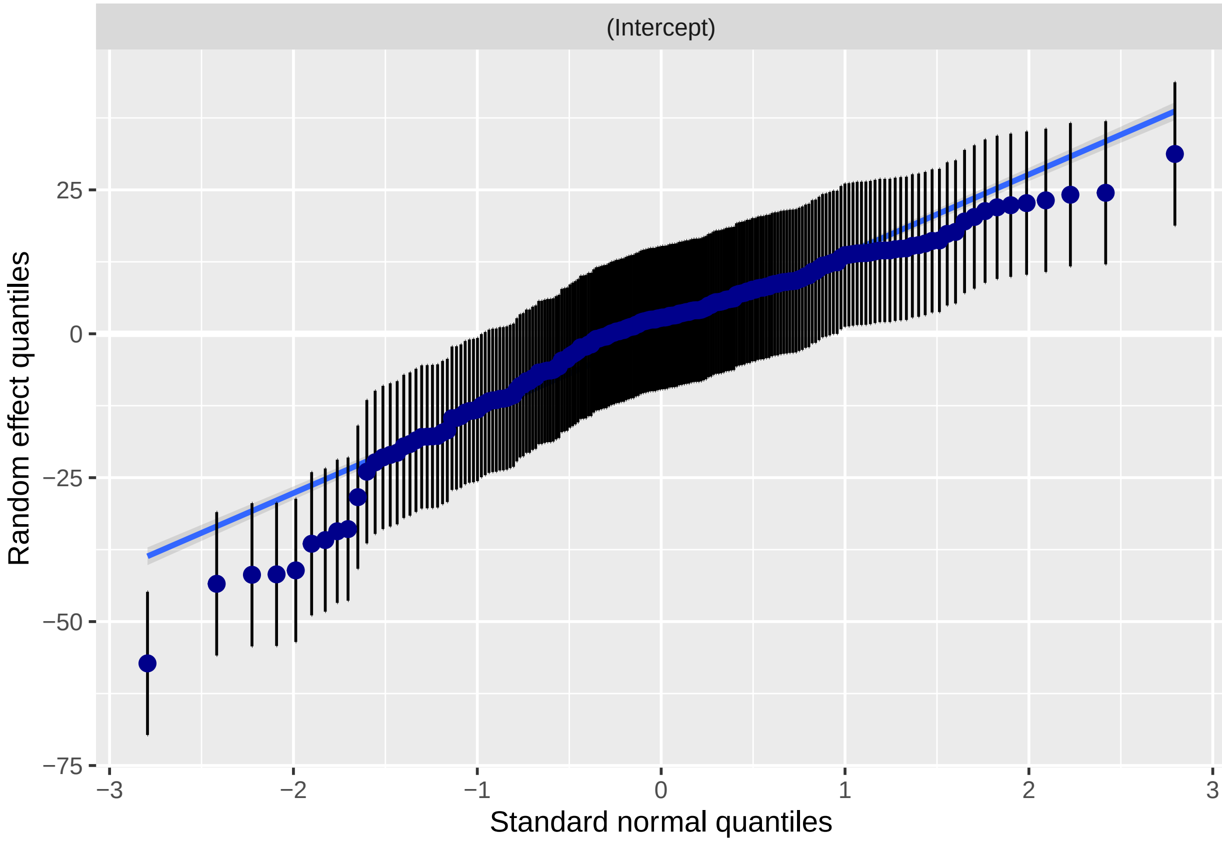
```
## [[1]]
```

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



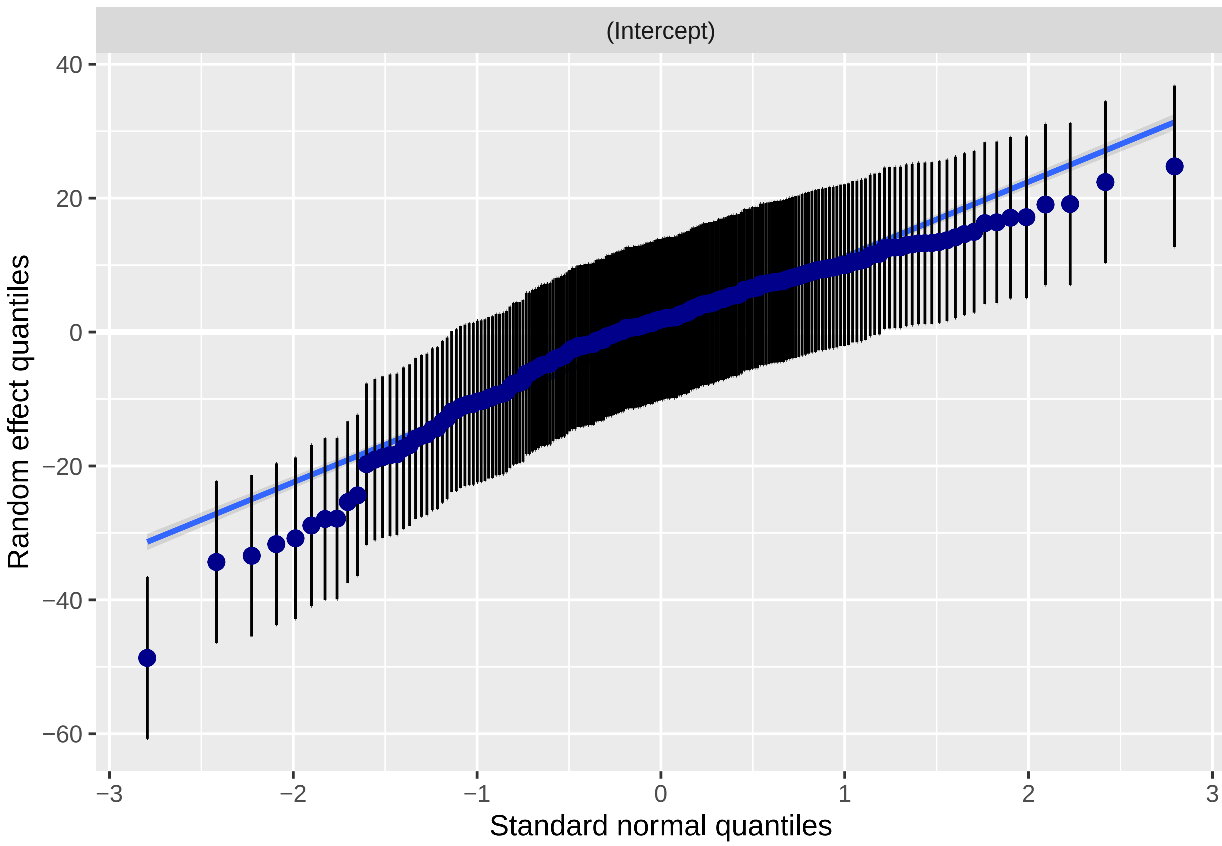

```
##
## [[2]]
## [[2]]$u_participant
```

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

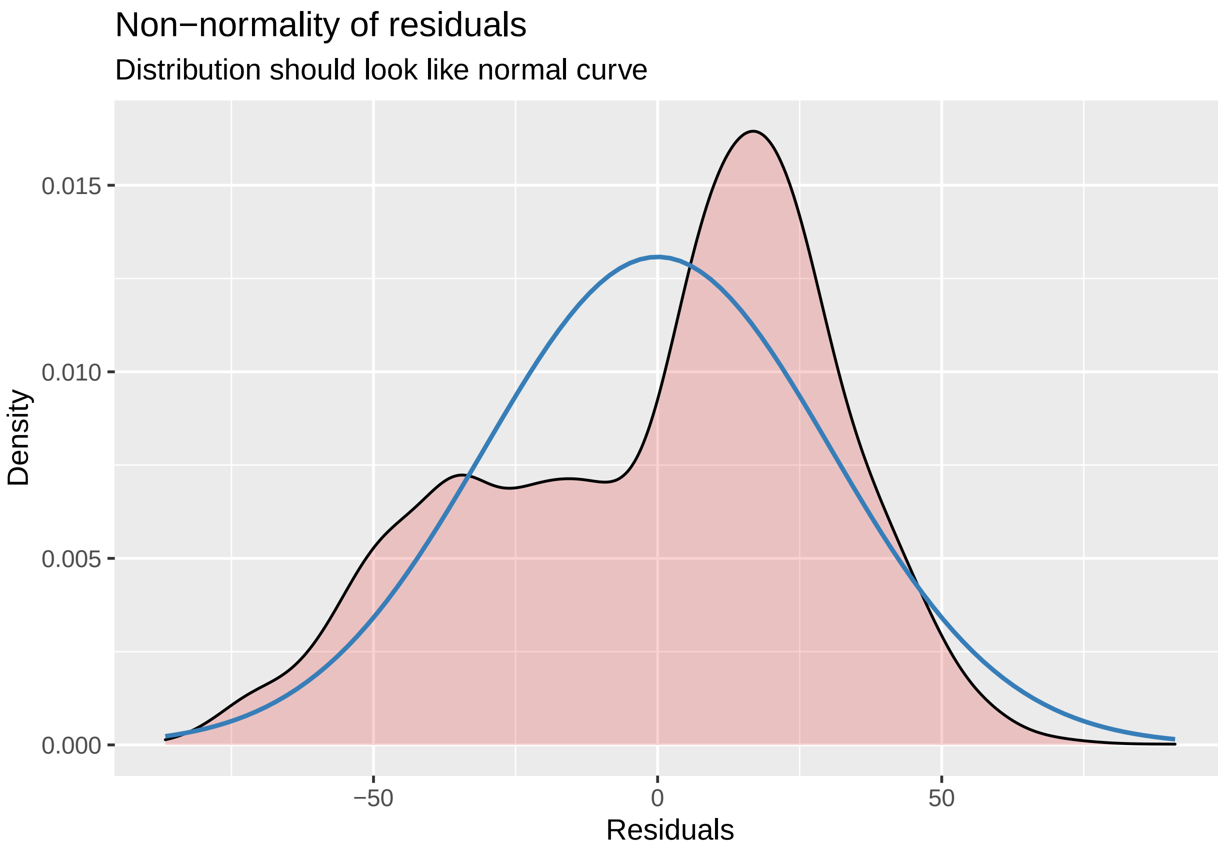



```
##
## [[2]]$u_partner
```

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

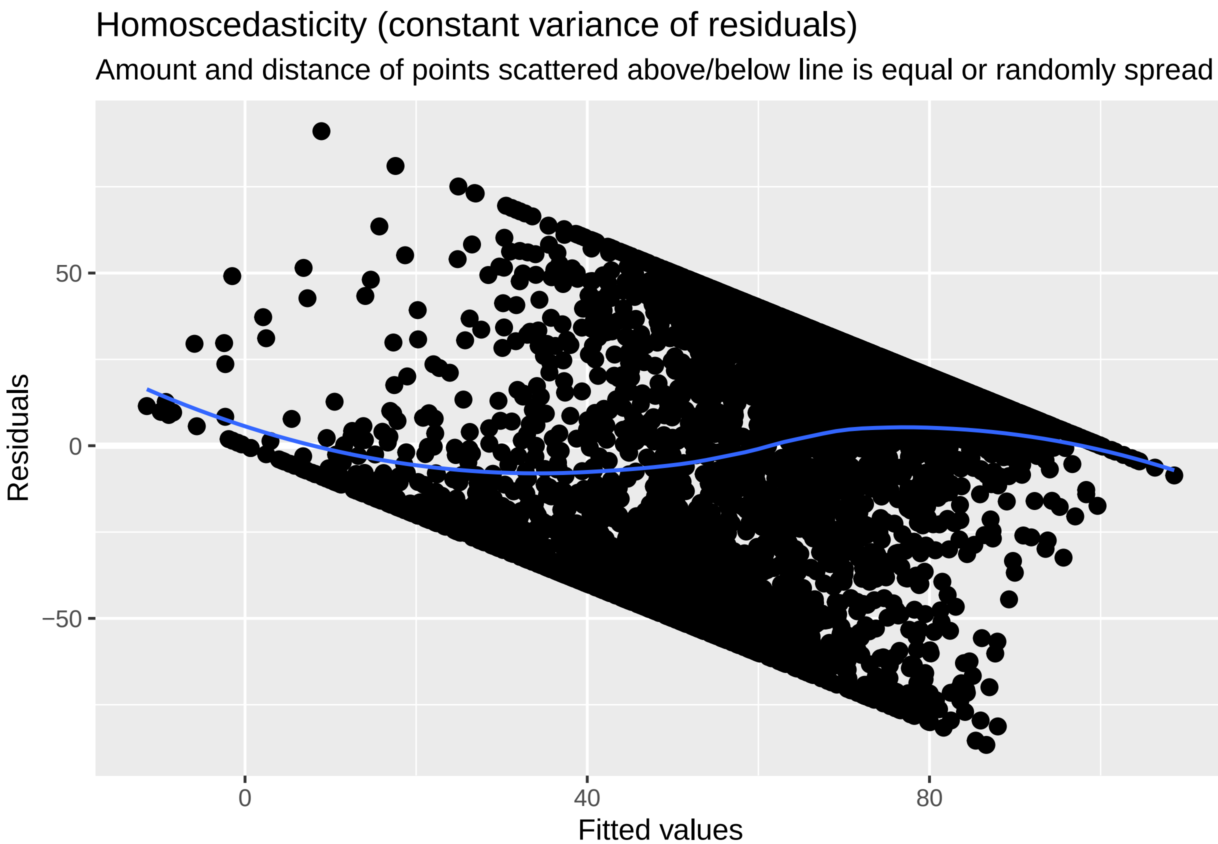

(Intercept)
40
20
0
−20
−40
−60
Random effect quantiles
−3
−2
−1
0
1
2
3
Standard normal quantiles

```
##
##
## [[3]]
```




```
##
## [[4]]
```

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




## Zero-one-inflated Beta mixed model

A main issue for the linear model above is the large number of observations on the bounds of the scale (0 and 100%).

```
# conditional upon cooperation > 0 and < 1, the average cooperation rates in
# the conditions are:
dat$coop_prop <- dat$coop_frequency/100
dat %>%
  group_by(condition) %>%
  filter(coop_prop > 0, coop_prop < 1) %>%
  summarize(mean_coop = mean(coop_prop))
```

```
## # A tibble: 4 x 2
##                  condition mean_coop
##                  <dbl+lbl>     <dbl>
## 1 1 [ long-deterministic]     0.642
## 2 2 [long-stochastic]         0.537
## 3 3 [short-deterministic]     0.616
## 4 4 [short-stochastic]        0.368
```

```
# the proportions of extreme cooperation rates (0 or 1 exactly) are:
dat %>%
  group_by(condition) %>%
  mutate(zoi = coop_prop %in% c(0,1)) %>%
  summarize(mean_zoi = mean(zoi))
```

```
## # A tibble: 4 x 2
##                  condition mean_zoi
##                  <dbl+lbl>    <dbl>
## 1 1 [ long-deterministic]    0.181
## 2 2 [long-stochastic]        0.380
## 3 3 [short-deterministic]    0.116
## 4 4 [short-stochastic]       0.466
```

```
# the proportions of cooperation rates of 1 within the extreme cooperation rates are:
dat %>%
  group_by(condition) %>%
  filter(coop_prop %in% c(0,1)) %>%
  summarize(mean_coi = mean(coop_prop))
```

```
## # A tibble: 4 x 2
##                  condition mean_coi
##                  <dbl+lbl>    <dbl>
## 1 1 [ long-deterministic]    0.715
## 2 2 [long-stochastic]        0.883
## 3 3 [short-deterministic]    0.758
## 4 4 [short-stochastic]       0.701
```

The short-stochastic condition has (1) the lowest non-extreme cooperation rate, (2) the highest proportion of extreme cooperation rates, and within these extreme cooperation rates, (3) the lowest proportion of 100% cooperation. Whilst (1) and (3) are consistent with the hypothesis, (2) indicates that there is potentially also a higher rate of %100 cooperation in this condition:

```
dat %>%
  group_by(condition, coop_prop) %>%
  summarize(n = n()) %>%
  mutate(prop = n/sum(n)) %>%
  filter(coop_prop == 1)
```

```
## `summarise()` has grouped output by 'condition'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   condition [4]
##                  condition coop_prop     n   prop
##                  <dbl+lbl>     <dbl> <int>  <dbl>
## 1 1 [ long-deterministic]          1   143 0.130
## 2 2 [long-stochastic]              1   371 0.336
## 3 3 [short-deterministic]          1    97 0.0879
## 4 4 [short-stochastic]             1   361 0.327
```

Whilst the proportion of %100 cooperation is highest in the long-stochastic condition, the short-stochastic condition is a close second, and both are far removed from the deterministic conditions.

There seem to be clear disparities between the conditions on the scores on the bounds of the scale. A monotone transformation will not resolve this issue. We therefore turn to a model which explicitly deals with these boundaries (or extreme) observations. This model is a zero-one-inflated Beta regression model. This model effectively consists of a three submodels: (1) a logistic regression model is used to analyse whether an observation is extreme (proportion of 0 or 1) or not, (2) a logistic regression model is used to analyse the proportion of full cooperation (proportion of 1) within these extreme proportions, and (3) the non-extreme proportions are modelled with a Beta regression model, using a logistic link function on the mean of the Beta distribution, and a log link on the dispersion of the Beta distribution.

We used a Bayesian implementation through the `brms` package in `R`, which allows inclusion of crossed random effects. for all four components: mean and dispersion of the Beta distribution, the logit of extreme events, and the logit of complete cooperation within extreme events. For convenience, we use dummy coding to code for the "fixed" effects of condition on these components, where the reference is the short-stochastic condition.

Let $\alpha_{ij}$ and $\beta_{ij}$ denote the parameters of the Beta distribution of the non-extreme cooperation rate for participant $i$ playing with partner $j$, $\pi_{ij}$ the probability of an extreme rate (of 1 or 0), and $\gamma_{ij}$ the conditional probability of 100% cooperation given an extreme cooperation rate.

$$p(c_{ij}) = \begin{cases} \pi_{ij}\gamma_{ij} & c_{ij} = 1 \\ \pi_{ij}(1-\gamma_{ij}) & c_{ij} = 0 \\ (1-\pi_{ij})\text{Beta}(\alpha_{ij}, \beta_{ij}) & 0 < c_{ij} < 1 \end{cases}$$

Letting $X_{1,i}$, $X_{2,i}$, and $X_{3,i}$ denote the dummy variables encoding condition, we model the parameters above as:

$$\begin{aligned}
\text{logit}\left(\frac{\alpha_{ij}}{\alpha_{ij}+\beta_{ij}}\right) &= b_{1,0} + b_{1,1}X_{1,i} + b_{1,2}X_{2,i} + b_{1,3}X_{3,i} + u_{1,i} + u_{1,j} \\
\log\left(\alpha_{ij}+\beta_{ij}\right) &= b_{2,0} + b_{2,1}X_{1,i} + b_{2,2}X_{2,i} + b_{2,3}X_{3,i} \\
\text{logit}(\pi_{ij}) &= b_{3,0} + b_{3,1}X_{1,i} + b_{3,2}X_{2,i} + b_{3,3}X_{3,i} + u_{3,i} + u_{3,j} \\
\text{logit}(\gamma_{ij}) &= b_{4,0} + b_{4,1}X_{1,i} + b_{4,2}X_{2,i} + b_{4,3}X_{3,i} + u_{4,i} + u_{4,j} \\
u_{1i} &\sim \text{Normal}(0, \sigma_{11}) \\
u_{1j} &\sim \text{Normal}(0, \sigma_{12}) \\
u_{3i} &\sim \text{Normal}(0, \sigma_{31}) \\
u_{3j} &\sim \text{Normal}(0, \sigma_{32}) \\
u_{4i} &\sim \text{Normal}(0, \sigma_{41}) \\
u_{4j} &\sim \text{Normal}(0, \sigma_{42})
\end{aligned}$$

Dummy regressors to compare the Short-Stochastic condition to the other conditions were computed as follows:

```r
dat$dum1 <- 0
dat$dum2 <- 0
dat$dum3 <- 0
dat$dum1[dat$condition == 1] <- 1
dat$dum2[dat$condition == 2] <- 1
dat$dum3[dat$condition == 3] <- 1
```

The model was defined and estimated as follows (sampling 2000 values for each of 4 chains after a burnin period of 8000 iterations):

```r
library(brms)
```

```
## Loading required package: Rcpp

## Loading 'brms' package (version 2.17.0). Useful instructions
## can be found by typing help('brms'). A more detailed introduction
## to the package is available through vignette('brms_overview').

##
## Attaching package: 'brms'

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

```r
set.seed(20220520)
zoibeta_mod <- bf(coop_prop ~ dum1 + dum2 + dum3 + (1|u_participant) + (1|u_partner),
```

```
          phi ~ dum1 + dum2 + dum3,
          zoi ~ dum1 + dum2 + dum3 + (1|u_participant) + (1|u_partner),
          coi ~ dum1 + dum2 + dum3 + (1|u_participant) + (1|u_partner),
          family=zero_one_inflated_beta())

zoibeta_fit <- brm(
  formula = zoibeta_mod,
  data = dat,
  cores = 4,
  file = "brm-zoibeta",
  iter = 10000,
  warmup = 8000,
  sample_prior = "yes"
)
```

Note that the default weakly informative priors of the `brms` package were used for all parameters. These priors are flat improper priors for the regression coefficients, half Student t-distributions (with df=3 and scale 2.5, truncated from below at 0) for the standard deviation of the random effects and for the intercept of the Beta distributions, and logistic distributions with mean 0 and scale 1 for the intercept of the zero-one inflations:

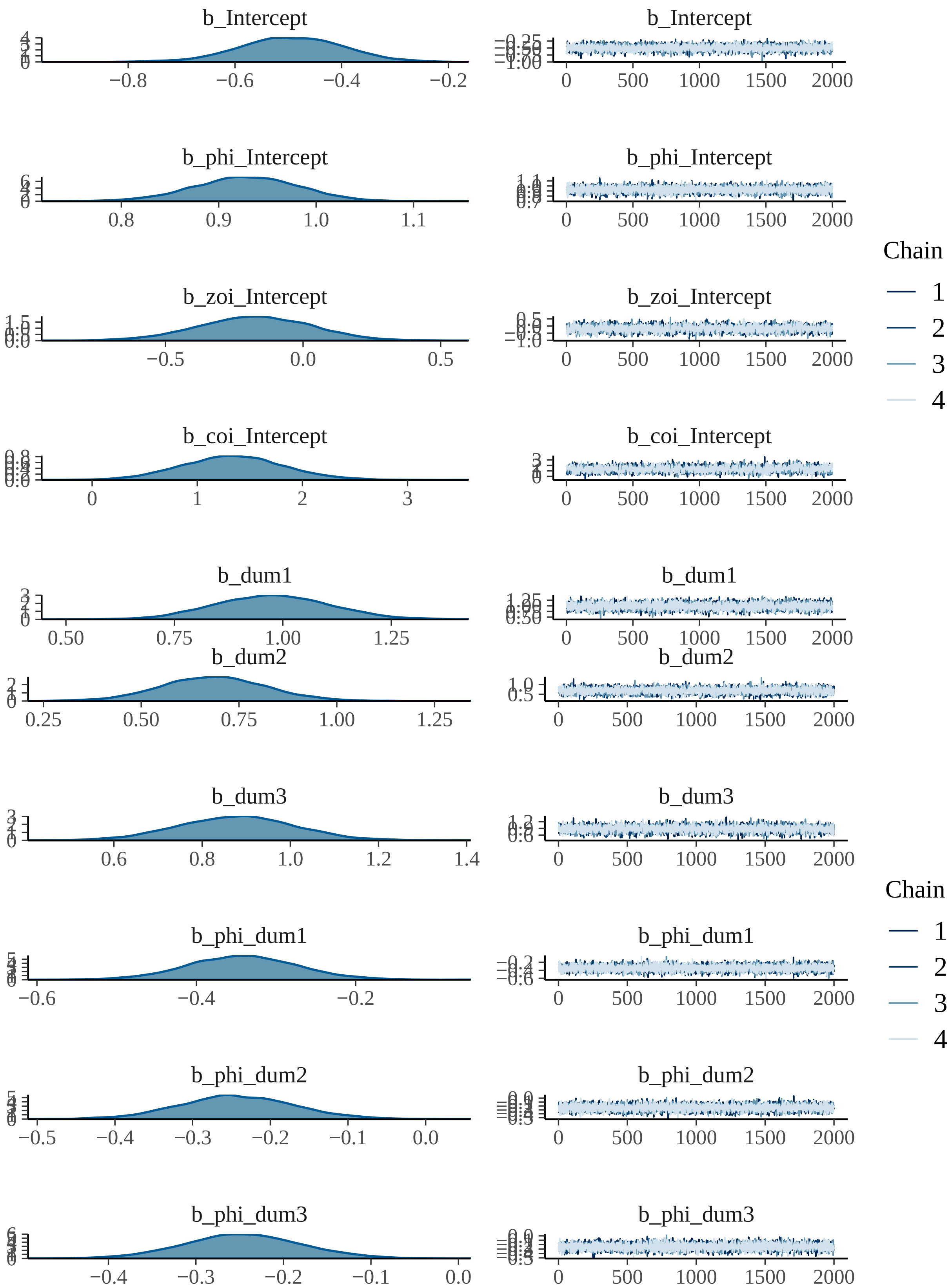
```
prior_summary(zoibeta_fit)
```

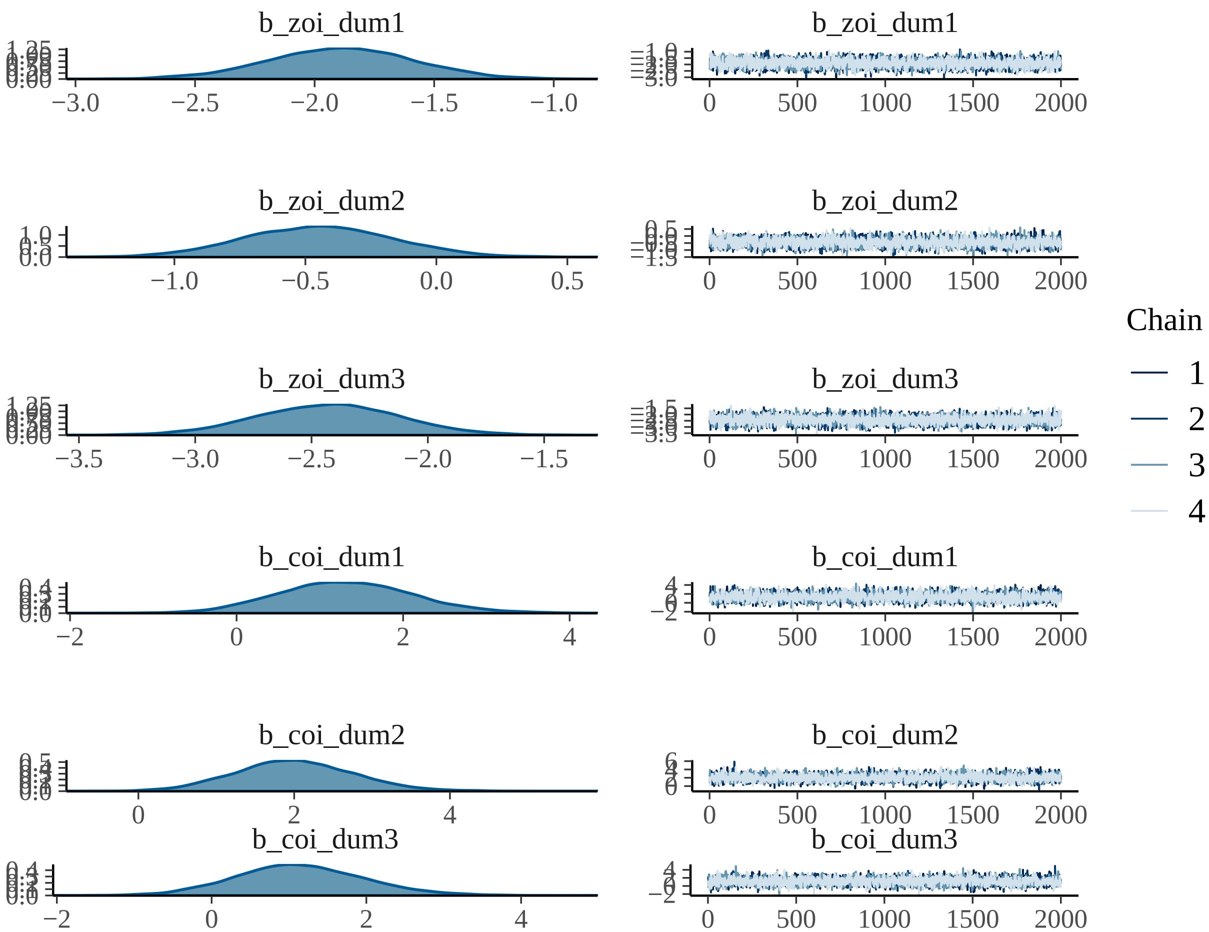
```
##                  prior     class      coef         group resp dpar nlpar lb ub
##                 (flat)         b
##                 (flat)         b      dum1
##                 (flat)         b      dum2
##                 (flat)         b      dum3
##                 (flat)         b                              coi
##                 (flat)         b      dum1                    coi
##                 (flat)         b      dum2                    coi
##                 (flat)         b      dum3                    coi
##                 (flat)         b                              phi
##                 (flat)         b      dum1                    phi
##                 (flat)         b      dum2                    phi
##                 (flat)         b      dum3                    phi
##                 (flat)         b                              zoi
##                 (flat)         b      dum1                    zoi
##                 (flat)         b      dum2                    zoi
##                 (flat)         b      dum3                    zoi
##  student_t(3, 0, 2.5) Intercept
##        logistic(0, 1) Intercept                               coi
##  student_t(3, 0, 2.5) Intercept                               phi
##        logistic(0, 1) Intercept                               zoi
##  student_t(3, 0, 2.5)        sd                                          0
##  student_t(3, 0, 2.5)        sd                               coi        0
##  student_t(3, 0, 2.5)        sd                               zoi        0
##  student_t(3, 0, 2.5)        sd           u_participant                  0
##  student_t(3, 0, 2.5)        sd Intercept u_participant                  0
##  student_t(3, 0, 2.5)        sd           u_participant       coi        0
##  student_t(3, 0, 2.5)        sd Intercept u_participant       coi        0
##  student_t(3, 0, 2.5)        sd           u_participant       zoi        0
##  student_t(3, 0, 2.5)        sd Intercept u_participant       zoi        0
##  student_t(3, 0, 2.5)        sd               u_partner                  0
##  student_t(3, 0, 2.5)        sd Intercept     u_partner                  0
```

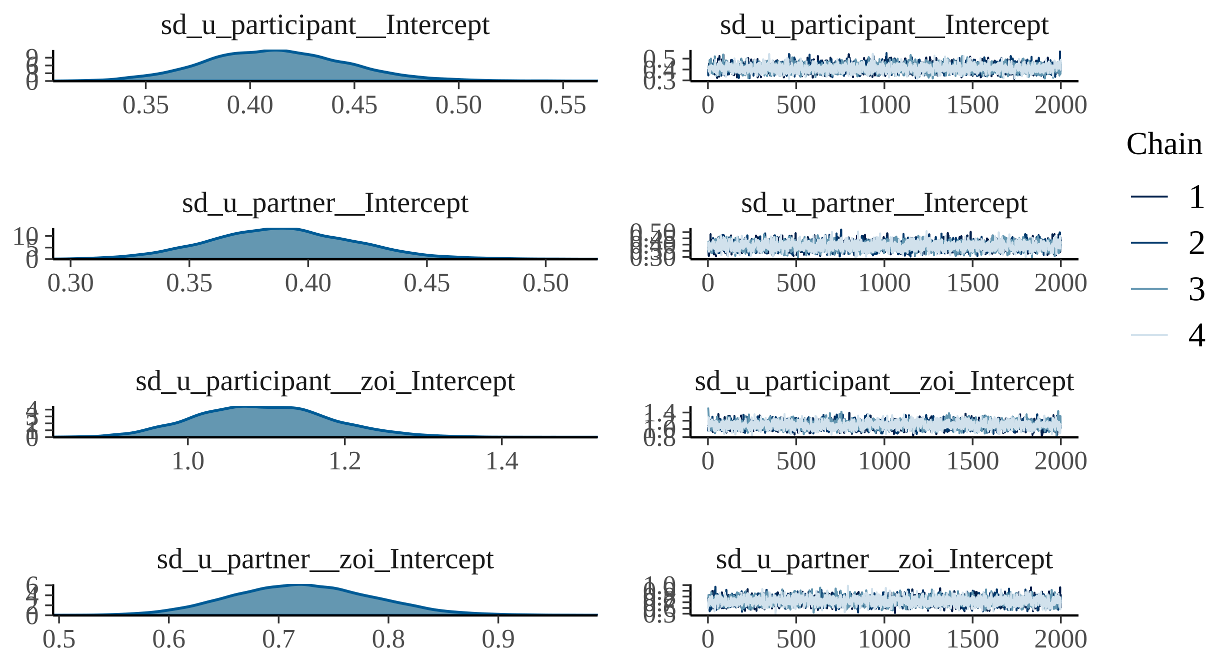
```
##  student_t(3, 0, 2.5)        sd                u_partner       coi        0
##  student_t(3, 0, 2.5)        sd Intercept      u_partner       coi        0
##  student_t(3, 0, 2.5)        sd                u_partner       zoi        0
##  student_t(3, 0, 2.5)        sd Intercept      u_partner       zoi        0
##        source
##       default
##  (vectorized)
##  (vectorized)
##  (vectorized)
##       default
##  (vectorized)
##  (vectorized)
##  (vectorized)
##       default
##  (vectorized)
##  (vectorized)
##  (vectorized)
##       default
##  (vectorized)
##  (vectorized)
##  (vectorized)
##       default
##       default
##       default
##       default
##       default
##       default
##       default
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
##  (vectorized)
```

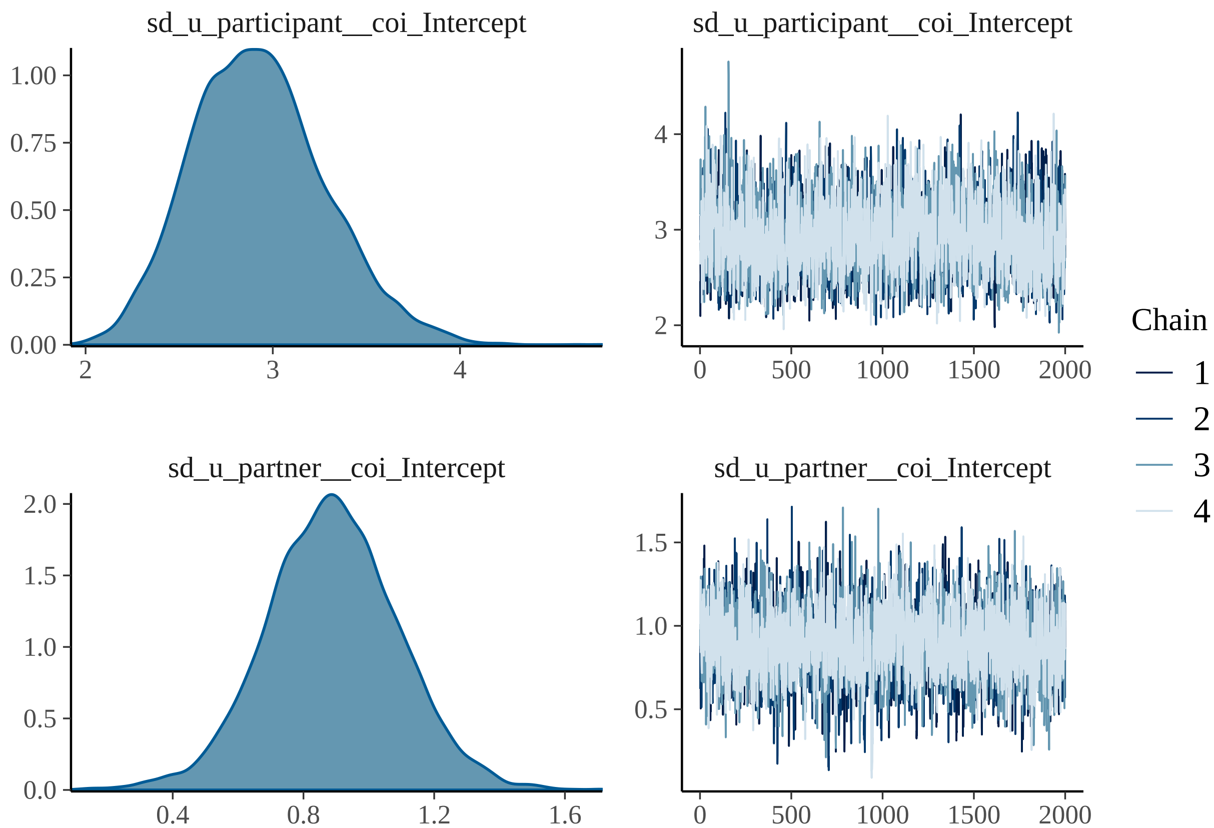
The posterior parameter distributions and traceplots are:

```
plot(zoibeta_fit, ask = FALSE)
```


b_Intercept
b_Intercept
b_phi_Intercept
b_phi_Intercept
b_zoi_Intercept
b_zoi_Intercept
b_coi_Intercept
b_coi_Intercept
b_dum1
b_dum1
b_dum2
b_dum2
b_dum3
b_dum3
b_phi_dum1
b_phi_dum1
b_phi_dum2
b_phi_dum2
b_phi_dum3
b_phi_dum3
Chain
1
2
3
4
Chain
1
2
3
4


b_zoi_dum1
b_zoi_dum1
b_zoi_dum2
b_zoi_dum2
b_zoi_dum3
b_zoi_dum3
b_coi_dum1
b_coi_dum1
b_coi_dum2
b_coi_dum2
b_coi_dum3
b_coi_dum3
Chain
1
2
3
4


sd_u_participant__Intercept
sd_u_participant__Intercept
sd_u_partner__Intercept
sd_u_partner__Intercept
sd_u_participant__zoi_Intercept
sd_u_participant__zoi_Intercept
sd_u_partner__zoi_Intercept
sd_u_partner__zoi_Intercept
Chain
1
2
3
4




There is no clear indication of convergence issues (e.g. RHat values are all very close to 1).

```
summary(zoibeta_fit)
```

```
##  Family: zero_one_inflated_beta
##   Links: mu = logit; phi = log; zoi = logit; coi = logit
## Formula: coop_prop ~ dum1 + dum2 + dum3 + (1 | u_participant) + (1 | u_partner)
##          phi ~ dum1 + dum2 + dum3
##          zoi ~ dum1 + dum2 + dum3 + (1 | u_participant) + (1 | u_partner)
##          coi ~ dum1 + dum2 + dum3 + (1 | u_participant) + (1 | u_partner)
##    Data: dat (Number of observations: 4416)
##   Draws: 4 chains, each with iter = 10000; warmup = 8000; thin = 1;
##          total post-warmup draws = 8000
##
## Group-Level Effects:
## ~u_participant (Number of levels: 192)
##                   Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)         0.41      0.03     0.35     0.48 1.00     3304     5609
## sd(zoi_Intercept)     1.10      0.09     0.94     1.27 1.00     2793     4720
## sd(coi_Intercept)     2.93      0.36     2.29     3.68 1.00     2756     4800
##
## ~u_partner (Number of levels: 192)
##                   Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)         0.39      0.03     0.33     0.45 1.00     3824     5968
## sd(zoi_Intercept)     0.72      0.07     0.60     0.85 1.00     3733     5575
## sd(coi_Intercept)     0.89      0.20     0.49     1.29 1.00     2316     3772
##
## Population-Level Effects:
##                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
```

```
## Intercept        -0.49      0.10     -0.68     -0.31 1.00     4941     5500
## phi_Intercept     0.93      0.05      0.82      1.03 1.00     9326     6408
## zoi_Intercept    -0.18      0.20     -0.58      0.22 1.00     3128     4488
## coi_Intercept     1.34      0.49      0.40      2.31 1.00     2513     4277
## dum1              0.98      0.13      0.73      1.24 1.00     4944     5710
## dum2              0.69      0.13      0.43      0.95 1.00     5401     5734
## dum3              0.88      0.13      0.62      1.14 1.00     5264     5842
## phi_dum1         -0.34      0.07     -0.48     -0.20 1.00     9437     6689
## phi_dum2         -0.24      0.07     -0.39     -0.10 1.00     9469     6903
## phi_dum3         -0.25      0.07     -0.38     -0.11 1.00     9411     7042
## zoi_dum1         -1.89      0.31     -2.50     -1.31 1.00     3812     4886
## zoi_dum2         -0.45      0.28     -1.00      0.11 1.00     3363     4727
## zoi_dum3         -2.43      0.31     -3.04     -1.83 1.00     4231     5274
## coi_dum1          1.32      0.81     -0.21      2.98 1.00     3224     4779
## coi_dum2          2.01      0.77      0.56      3.56 1.00     2976     4521
## coi_dum3          1.14      0.84     -0.42      2.85 1.00     3385     5007
##
## 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).
```

With regards to differences between the Short-Deterministic and Short-Stochastic condition (the effects of `dum3`), we see that the non-extreme cooperation rates are higher in the former condition (`dum3` is negative), with less variability (`phi_dum3` is negative). The probability of an extreme cooperation rate is also lower in the Short-Deterministic condition (`zoi_dum3` is negative), but there is no evidence that conditional probability of 100% cooperation differs between the conditions (the 95% credible interval for `coi_dum3` includes 0).

The main hypothesis to be tested can be written as follows:

$$H_0 : \pi_{\text{SD}}\gamma_{\text{SD}} + (1 - \pi_{\text{SD}})\frac{\alpha_{\text{SD}}}{\alpha_{\text{SD}} + \beta_{\text{SD}}} = \pi_{\text{SS}}\gamma_{\text{SS}} + (1 - \pi_{\text{SS}})\frac{\alpha_{\text{SS}}}{\alpha_{\text{SS}} + \beta_{\text{SS}}}$$

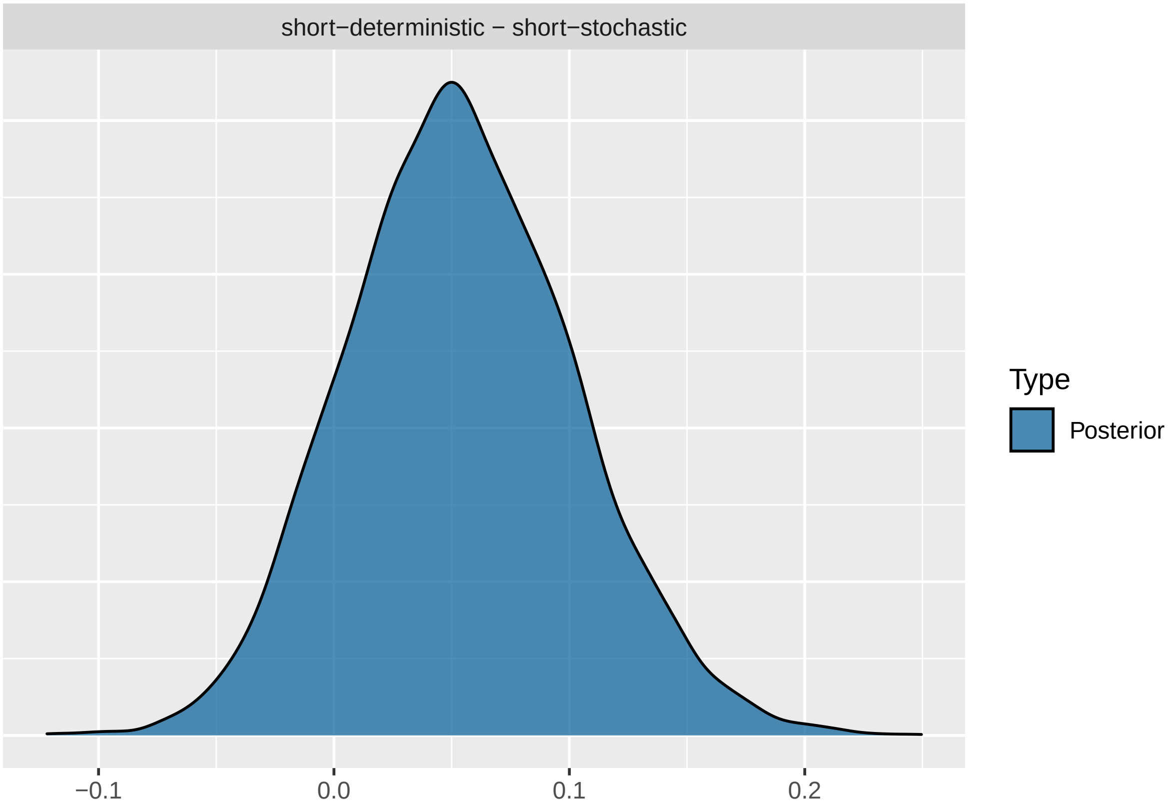
and performed using the following code:

```
main_h <- c("short-deterministic - short-stochastic" = "plogis(zoi_Intercept + zoi_dum3)*plogis(coi_Inte
hypothesis(zoibeta_fit, main_h, seed = 20220524)
```

```
## Hypothesis Tests for class b:
##                 Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 short-determinist...stic     0.05      0.05    -0.04     0.15         NA
##   Post.Prob Star
## 1        NA
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

This shows that the 95% credible interval contains 0, and hence the hypothesis of equality between the conditions is not rejected. The posterior distribution of the difference is shown in the plot below:

```
plot(hypothesis(zoibeta_fit, main_h, seed = 20220524))
```




Whilst the non-extreme cooperation rates are higher in the Short-Deterministic condition:

```
main_h2 <- c("short-deterministic - short-stochastic" = "plogis(Intercept + dum3) = plogis(Intercept)")
hypothesis(zoibeta_fit, main_h2, seed = 20220524)
```

```
## Hypothesis Tests for class b:
##                Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 short-determinist...stic     0.22      0.03     0.15     0.28         NA
##   Post.Prob Star
## 1        NA    *
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

because the probability of an extreme cooperation rate is higher in the Short-Stochastic condition

```
main_h3 <- c("short-deterministic - short-stochastic" = "plogis(zoi_Intercept + zoi_dum3) = plogis(zoi_I
hypothesis(zoibeta_fit, main_h3, seed = 20220524)
```

```
## Hypothesis Tests for class b:
##                Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 short-determinist...stic    -0.39      0.05    -0.49    -0.28         NA
##   Post.Prob Star
## 1        NA    *
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
```

```
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

with no clear difference in the conditional probability of 100% cooperation

```r
main_h4 <- c("short-deterministic - short-stochastic" = "plogis(coi_Intercept + coi_dum3) = plogis(coi_I
hypothesis(zoibeta_fit, main_h4, seed = 20220524)
```

```
## Hypothesis Tests for class b:
##                 Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 short-determinist...stic     0.13       0.1    -0.06     0.33         NA
##   Post.Prob Star
## 1        NA
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

it appears the decrease in non-extreme cooperation rates is countered by the increase in extreme (100%) cooperation rates:

```r
main_h5 <- c("short-deterministic - short-stochastic" = "plogis(zoi_Intercept + zoi_dum3)*plogis(coi_Int
hypothesis(zoibeta_fit, main_h5, seed = 20220524)
```

```
## Hypothesis Tests for class b:
##                 Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 short-determinist...stic    -0.29      0.06     -0.4    -0.19         NA
##   Post.Prob Star
## 1        NA    *
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

Note that, to check the robustness of the results, the analysis was repeated with a model where the flat improper priors for the intercepts and regression coefficients were replaced by Normal distributions with a standard deviation of 5. The results were qualitatively similar, and no evidence was found for the main claim.

## Conclusion

Using a zero-one-inflated Beta regression model with crossed random effects for participants and partners, we found no evidence for the claim that cooperation rates are higher in the Short-Deterministic as compared to the Short-Stochastic condition.

## Summary of analysis and results

> Please report the most important steps of the analysis to the level of detail that you would provide in a methods/analysis section of a typical research article. Include any preprocessing steps that you conducted on the dataset. Describe the exact statistical hypothesis you tested and explain the reason for choosing the statistical procedure you applied. Finally, please report the result of your statistical test(s).

The main claim to be tested is that, within the short-duration conditions, cooperation rates are higher in the deterministic as compared to stochastic horizon condition. As the direction of this difference was not based on an a priori hypothesis, I chose to focus on a less specified hypothesis, namely that cooperation rates differ between the short-stochastic and short-deterministic condition. In the study, each of 24 participants within a session was paired with each of the remaining 23 participants to play 23 Prisoner's dilemma supergames. As cooperation rates likely depend on both players, a model with crossed random effects for participants and their partners seems suitable. The main variable of interest is the cooperation rate, determined as the proportion of time within a supergame that a participant chose the cooperation action. This variable is bounded between 0 and 1, but has many observations on these bounds, which poses problems for linear models. The data was therefore analysed with a zero-one-inflated Beta mixed-effects regression model, including crossed random-effects for the non-extreme cooperation rates, the probability of extreme cooperation rates and the conditional probability of 100% cooperation. The model was estimated in a Bayesian framework, using the brms package for R and using default uninformative and weakly informative prior distributions. Parameters of the model were combined to obtain posterior estimates of overall cooperation rate in the conditions, and the posterior distribution of the difference between the short-deterministic and short-stochastic condition of these overall cooperation rates was used to assess the main claim. The 95% credible interval of this posterior distribution is $[-0.04, 0.15]$ and includes 0, so the main claim is not supported.

```
sessionInfo()
```

```
## R version 4.2.0 (2022-04-22)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04.4 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/atlas/libblas.so.3.10.3
## LAPACK: /usr/lib/x86_64-linux-gnu/atlas/liblapack.so.3.10.3
##
## locale:
##  [1] LC_CTYPE=en_GB.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_GB.UTF-8        LC_COLLATE=en_GB.UTF-8
##  [5] LC_MONETARY=en_GB.UTF-8    LC_MESSAGES=en_GB.UTF-8
##  [7] LC_PAPER=en_GB.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_GB.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] brms_2.17.0  Rcpp_1.0.8.3 dplyr_1.0.9
##
## loaded via a namespace (and not attached):
##   [1] minqa_1.2.4          colorspace_2.0-3     ellipsis_0.3.2
##   [4] tinylabels_0.2.3     ggridges_0.5.3       sjlabelled_1.2.0
##   [7] estimability_1.3     markdown_1.1         parameters_0.17.0
##  [10] base64enc_0.1-3      rstudioapi_0.13      rstan_2.21.5
##  [13] glmmTMB_1.1.3        farver_2.1.0         DT_0.23
##  [16] fansi_1.0.3          mvtnorm_1.1-3        diffobj_0.3.5
##  [19] codetools_0.2-18     bridgesampling_1.1-2 splines_4.2.0
##  [22] papaja_0.1.0.9999    knitr_1.39           shinythemes_1.2.0
##  [25] sjmisc_2.8.9         bayesplot_1.9.0      afex_1.1-1
##  [28] nloptr_2.0.1         ggeffects_1.1.2      pbkrtest_0.5.1
##  [31] broom_0.8.0          effectsize_0.6.0.1   shiny_1.7.1
##  [34] readr_2.1.2          compiler_4.2.0       sjstats_0.18.1
##  [37] emmeans_1.7.3        backports_1.4.1      assertthat_0.2.1
##  [40] Matrix_1.4-1         fastmap_1.1.0        cli_3.3.0
##  [43] later_1.3.0          prettyunits_1.1.1    htmltools_0.5.2
```

```
##  [46] tools_4.2.0          igraph_1.3.1         lmerTest_3.1-3
##  [49] coda_0.19-4          gtable_0.3.0         glue_1.6.2
##  [52] reshape2_1.4.4       posterior_1.2.1      carData_3.0-5
##  [55] vctrs_0.4.1          sjPlot_2.8.10        nlme_3.1-157
##  [58] crosstalk_1.2.0      insight_0.17.1       tensorA_0.36.2
##  [61] xfun_0.31            stringr_1.4.0        ps_1.7.0
##  [64] lme4_1.1-29          miniUI_0.1.1.1       mime_0.12
##  [67] lifecycle_1.0.1      gtools_3.9.2         MASS_7.3-57
##  [70] zoo_1.8-10           scales_1.2.0         colourpicker_1.1.1
##  [73] hms_1.1.1            promises_1.2.0.1     Brobdingnag_1.2-7
##  [76] parallel_4.2.0       inline_0.3.19        shinystan_2.6.0
##  [79] TMB_1.8.1            RColorBrewer_1.1-3   yaml_2.3.5
##  [82] gridExtra_2.3        ggplot2_3.3.6        StanHeaders_2.21.0-7
##  [85] loo_2.5.1            stringi_1.7.6        highr_0.9
##  [88] bayestestR_0.12.1    dygraphs_1.1.1.6     checkmate_2.1.0
##  [91] pkgbuild_1.3.1       boot_1.3-28          rlang_1.0.2
##  [94] pkgconfig_2.0.3      matrixStats_0.62.0   distributional_0.3.0
##  [97] evaluate_0.15        lattice_0.20-45      purrr_0.3.4
## [100] rstantools_2.2.0     htmlwidgets_1.5.4    labeling_0.4.2
## [103] processx_3.5.3       tidyselect_1.1.2     plyr_1.8.7
## [106] magrittr_2.0.3       R6_2.5.1             generics_0.1.2
## [109] DBI_1.1.2            pillar_1.7.0         haven_2.5.0
## [112] mgcv_1.8-40          xts_0.12.1           datawizard_0.4.1
## [115] abind_1.4-5          tibble_3.1.7         performance_0.9.0
## [118] modelr_0.1.8         crayon_1.5.1         car_3.0-13
## [121] utf8_1.2.2           tzdb_0.3.0           rmarkdown_2.14
## [124] grid_4.2.0           callr_3.7.0          forcats_0.5.1
## [127] threejs_0.3.3        digest_0.6.29        xtable_1.8-4
## [130] tidyr_1.2.0          httpuv_1.6.5         numDeriv_2016.8-1.1
## [133] RcppParallel_5.1.5   stats4_4.2.0         munsell_0.5.0
## [136] shinyjs_2.1.0
```