Supplementary Material 4: Manipulation Checks (not preregistered)

Angel V. Jimenez

26 June 2019

```
# DATASET
setwd("C:/files/Angel/tablets/experiment")
d<-read.csv("tablets_exp.csv")
# Gender as factor
d$GENDER <- ifelse((d$GENDER==1),"Male",
                   ifelse((d$GENDER==2),"female","other"))
d$GENDER <- as.factor(d$GENDER)
```

```
# PACKAGES
library(tidyverse)
```

```
## -- Attaching packages --------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.1.0       v purrr   0.3.2
## v tibble  2.1.1       v dplyr   0.8.0.1
## v tidyr   0.8.3       v stringr 1.4.0
## v readr   1.3.1       v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following object is masked from 'package:tidyr':
##
##     expand
```

```
library(sjstats)
library(arm)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
##
## arm (Version 1.10-1, built: 2018-4-12)
```

```
## Working directory is C:/Users/aj419/OneDrive - University of Exeter/2019/THESIS/CHAPTER 4 (tablets)/a
```

```r
library(coefplot)
```

```
##
## Attaching package: 'coefplot'
```

```
## The following objects are masked from 'package:arm':
##
##     coefplot, coefplot.default, invlogit
```

```r
library(brms)
```

```
## Loading required package: Rcpp
```

```
## Loading 'brms' package (version 2.8.0). 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:lme4':
##
##     ngrps
```

```r
library(janitor)
```

```r
# MANIPULATION CHECKS (PRESTIGE - ORDINAL)
# Descriptive statistics
# Median, minimum and  maximum in prestige for the three sources
d %>%
  group_by(SOURCE) %>%
  summarise(medianptg = median(PRESTIGE),  minpgt=min(PRESTIGE), maxpgt=max(PRESTIGE))
```

```
## # A tibble: 3 x 4
##   SOURCE   medianptg minpgt maxpgt
##   <fct>        <dbl>  <dbl>  <dbl>
## 1 CLEANER          2      1      4
## 2 EDUCATOR         4      1      5
## 3 PILOT            3      1      5
```

```r
# Medians: Educator = 4, Pilot = 3, Cleaner = 2
# Min: Educator = 1, Pilot = 1, Cleaner = 1
# Max: Educator = 5, Pilot = 5, Cleaner = 4
```

```r
# Frequences of each rating for the Educator (prestige)
library(janitor)
d %>%
  filter(SOURCE=="EDUCATOR") %>%
  tabyl(PRESTIGE, sort = TRUE)
```

```
##  PRESTIGE  n   percent
##         1  2 0.0156250
##         2  8 0.0625000
##         3 47 0.3671875
##         4 64 0.5000000
##         5  7 0.0546875
```

```r
# Frequencies of each rating for the Pilot
library(janitor)
```

```r
d %>%
  filter(SOURCE=="PILOT") %>%
  tabyl(PRESTIGE, sort = TRUE)
```

```
##  PRESTIGE  n   percent
##         1  1 0.0078125
##         2  9 0.0703125
##         3 73 0.5703125
##         4 41 0.3203125
##         5  4 0.0312500
```

```r
# Frequencies of each rating for the Cleaner (prestige)
library(janitor)
d %>%
  filter(SOURCE=="CLEANER") %>%
  tabyl(PRESTIGE, sort = TRUE)
```

```
##  PRESTIGE  n   percent
##         1 34 0.2656250
##         2 51 0.3984375
##         3 39 0.3046875
##         4  4 0.0312500
```

```r
# Source model for prestige (Bayesian ordinal model with participant as random effect)
d<-within(d, SOURCE <- relevel(SOURCE, ref = "CLEANER"))
ptg.ordinal<-brm(PRESTIGE~SOURCE+(1|PARTICIPANT), control = list(max_treedepth = 15), family=cumulative
```

```
## Compiling the C++ model
```

```
## Start sampling
```

```
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.001 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 9.7 seconds (Warm-up)
## Chain 1:                7.97 seconds (Sampling)
## Chain 1:                17.67 seconds (Total)
## Chain 1:
```

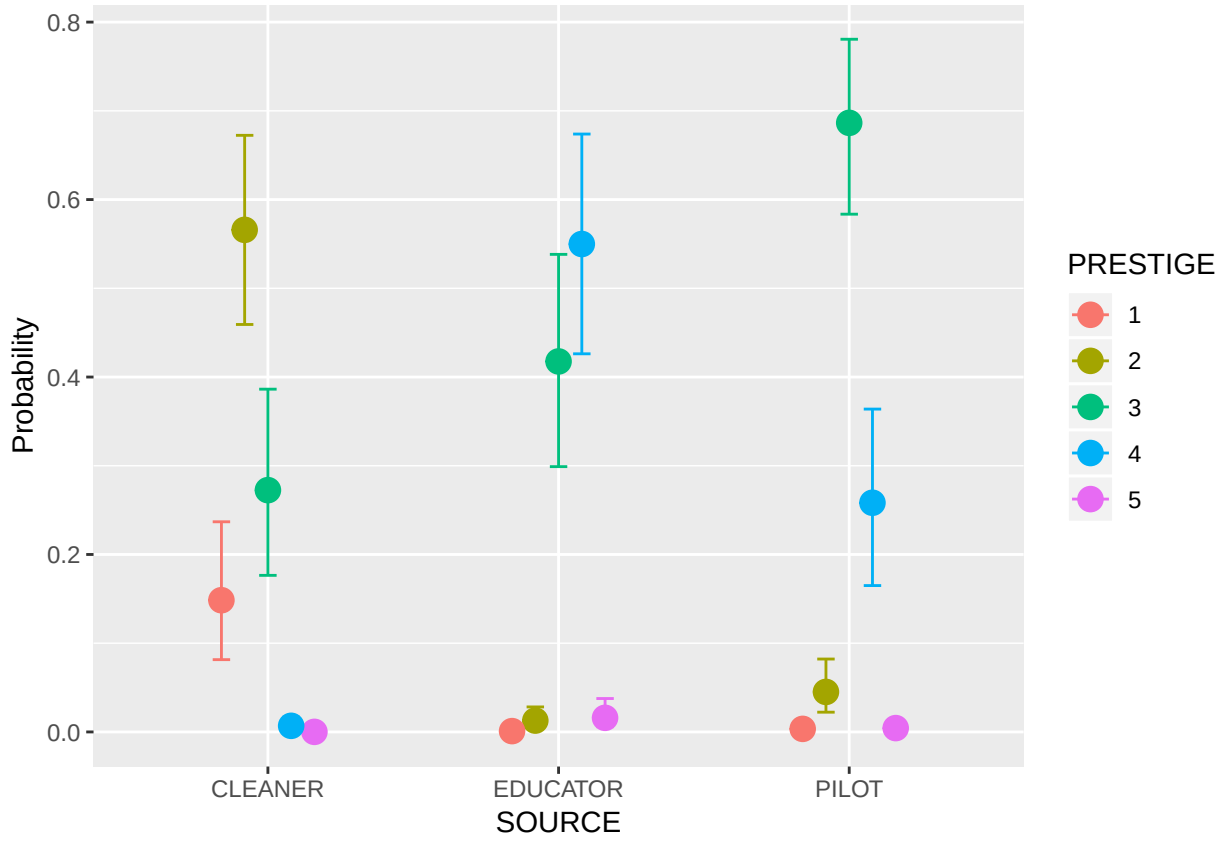
```
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0.001 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 8.029 seconds (Warm-up)
## Chain 2:                7.092 seconds (Sampling)
## Chain 2:                15.121 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0.001 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 8.024 seconds (Warm-up)
## Chain 3:                7.186 seconds (Sampling)
## Chain 3:                15.21 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
```

```
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 7.592 seconds (Warm-up)
## Chain 4:                7.227 seconds (Sampling)
## Chain 4:                14.819 seconds (Total)
## Chain 4:
```

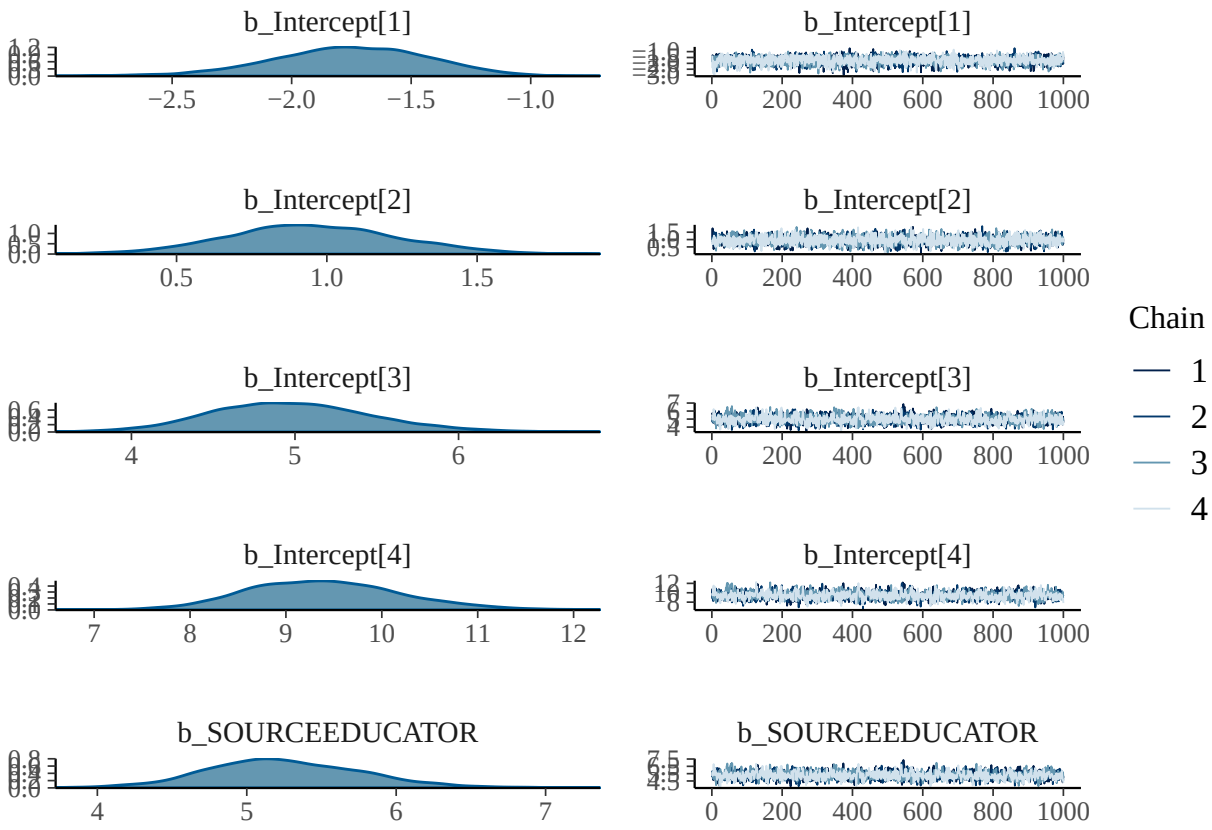
```
summary(ptg.ordinal, prob=0.89)
```

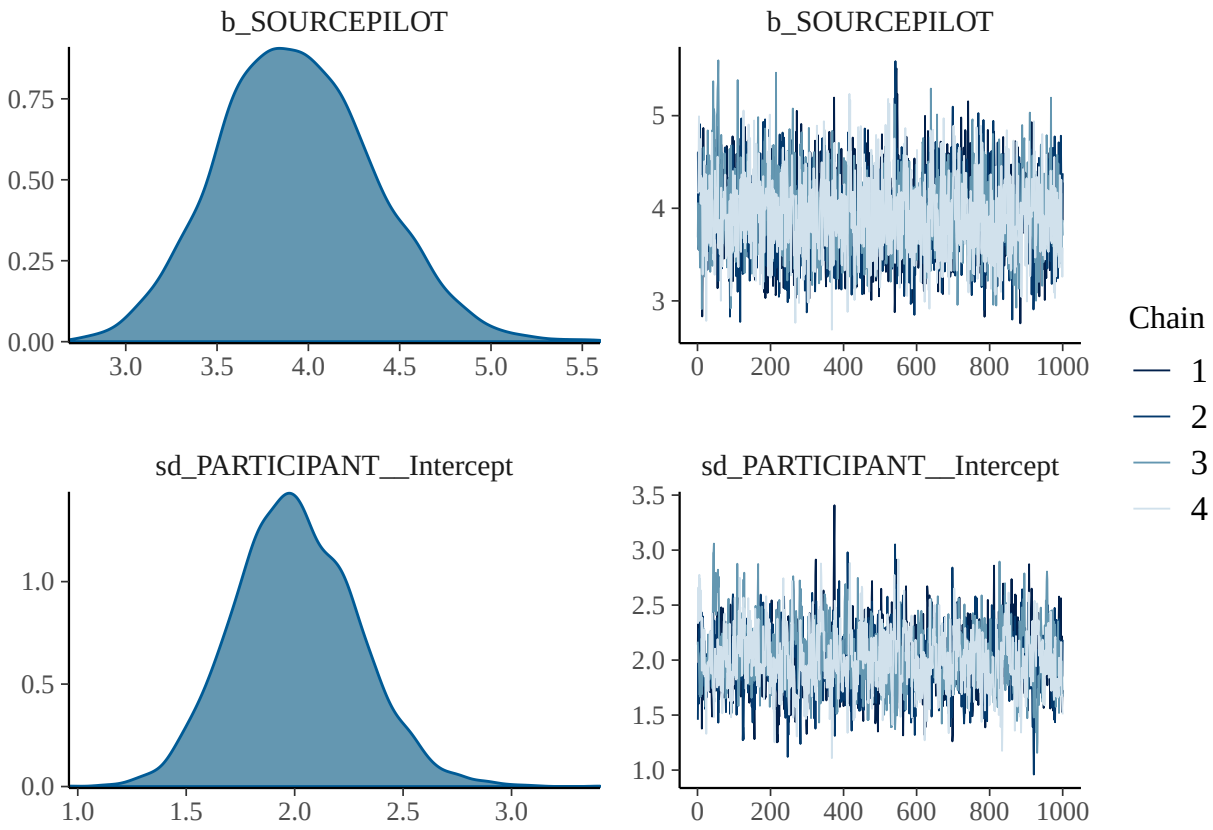
```
##  Family: cumulative
##   Links: mu = logit; disc = identity
## Formula: PRESTIGE ~ SOURCE + (1 | PARTICIPANT)
##    Data: d (Number of observations: 384)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup samples = 4000
##
## Group-Level Effects:
## ~PARTICIPANT (Number of levels: 192)
##               Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## sd(Intercept)     2.01      0.28     1.57     2.47        829 1.00
##
## Population-Level Effects:
##                Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## Intercept[1]      -1.76      0.32    -2.29    -1.26       2887 1.00
## Intercept[2]       0.95      0.28     0.51     1.42       3627 1.00
## Intercept[3]       4.98      0.49     4.23     5.81       1749 1.00
## Intercept[4]       9.40      0.81     8.18    10.75       1458 1.00
## SOURCEEDUCATOR     5.26      0.52     4.48     6.11       1533 1.00
## SOURCEPILOT        3.94      0.43     3.28     4.64       1927 1.00
##
## Samples were drawn using sampling(NUTS). For each parameter, Eff.Sample
## is a crude measure of effective sample size, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
marginal_effects(ptg.ordinal, categorical=TRUE)
```

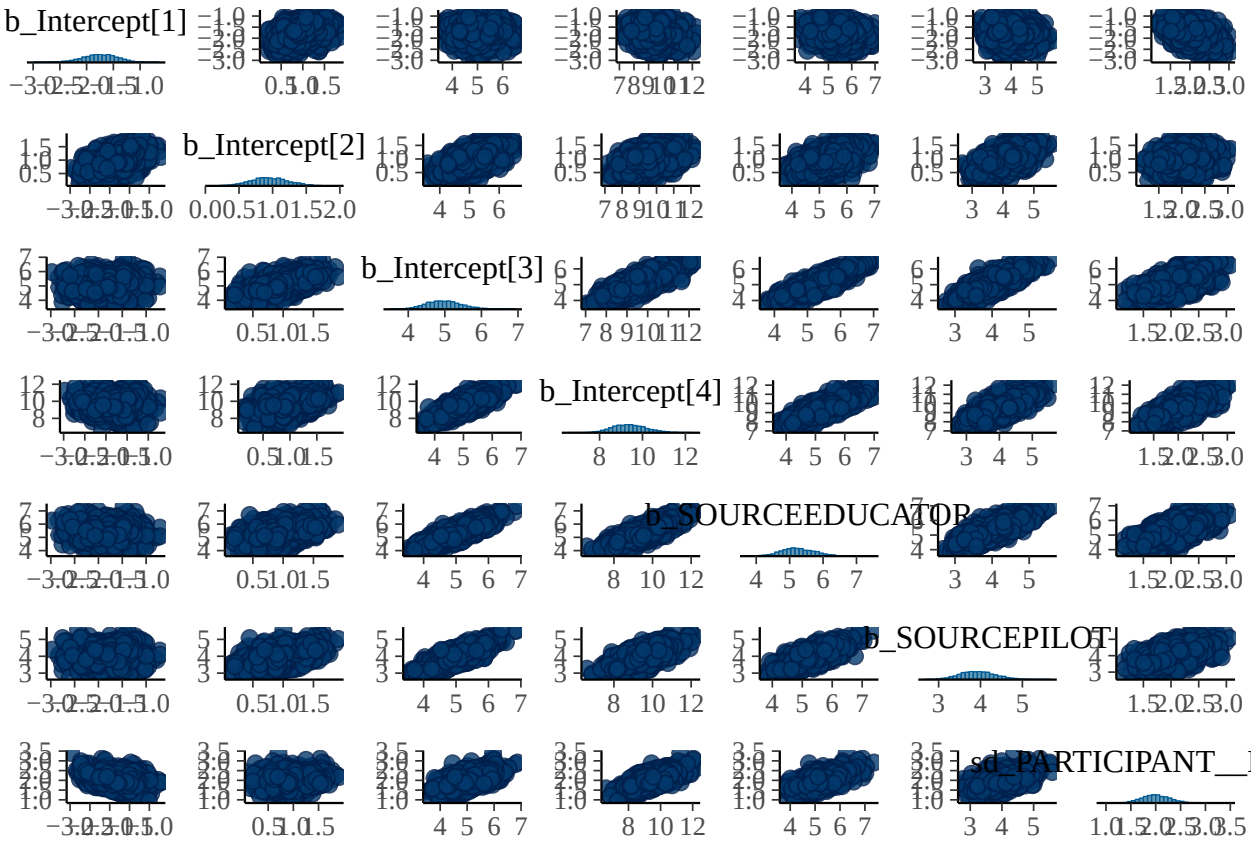



```
plot(ptg.ordinal)
```


b_Intercept[1]
b_Intercept[1]
b_Intercept[2]
b_Intercept[2]
b_Intercept[3]
b_Intercept[3]
b_Intercept[4]
b_Intercept[4]
b_SOURCEEDUCATOR
b_SOURCEEDUCATOR
Chain
1
2
3
4




```
pairs(ptg.ordinal)
```




```
d<-within(d, SOURCE <- relevel(SOURCE, ref = "PILOT"))
ptg.ordinal<-brm(PRESTIGE~SOURCE+(1|PARTICIPANT), control = list(max_treedepth = 15), family=cumulative
```

```
## Compiling the C++ model
## recompiling to avoid crashing R session
## Start sampling
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.001 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
```

```
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 9.104 seconds (Warm-up)
## Chain 1:                8.123 seconds (Sampling)
## Chain 1:                17.227 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 7.591 seconds (Warm-up)
## Chain 2:                7.039 seconds (Sampling)
## Chain 2:                14.63 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 7.814 seconds (Warm-up)
## Chain 3:                8.608 seconds (Sampling)
```

```
## Chain 3:                16.422 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 9.239 seconds (Warm-up)
## Chain 4:                8.693 seconds (Sampling)
## Chain 4:                17.932 seconds (Total)
## Chain 4:
```

```
summary(ptg.ordinal, prob=0.89)
```

```
##  Family: cumulative
##   Links: mu = logit; disc = identity
## Formula: PRESTIGE ~ SOURCE + (1 | PARTICIPANT)
##    Data: d (Number of observations: 384)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup samples = 4000
##
## Group-Level Effects:
## ~PARTICIPANT (Number of levels: 192)
##               Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## sd(Intercept)     2.01      0.29     1.57     2.48        722 1.00
##
## Population-Level Effects:
##                Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## Intercept[1]      -5.71      0.54    -6.62    -4.88       1400 1.00
## Intercept[2]      -3.00      0.37    -3.59    -2.45       2315 1.00
## Intercept[3]       1.04      0.28     0.60     1.50       3287 1.00
## Intercept[4]       5.46      0.59     4.56     6.44       1709 1.00
## SOURCECLEANER     -3.95      0.42    -4.65    -3.30       1984 1.00
## SOURCEEDUCATOR     1.31      0.32     0.82     1.83       3699 1.00
##
## Samples were drawn using sampling(NUTS). For each parameter, Eff.Sample
## is a crude measure of effective sample size, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

MANIPULATION CHECKS FOR RELEVANCE

Descriptive statistics

```
d %>%
  group_by(SOURCE) %>%
  summarise(medianrel = median(RELEVANCE),  minrel=min(RELEVANCE), maxrel=max(RELEVANCE))
```

```
## # A tibble: 3 x 4
##   SOURCE   medianrel minrel maxrel
##   <fct>        <dbl>  <dbl>  <dbl>
## 1 PILOT            1     -3      3
## 2 CLEANER          0     -3      3
## 3 EDUCATOR         2     -2      3
```

```
# Median: Educator = 2, Pilot=1, Cleaner = 0
# Minimum: Educator = -2, Pilot = -3, Cleaner = -3
# Maximum: Eudcator = 3, Pilot = 3, Cleaner = 3
```

```
# Frequencies of the different ratings of relevance for the Educator
library(janitor)
d %>%
  filter(SOURCE=="EDUCATOR") %>%
  tabyl(RELEVANCE, sort = TRUE)
```

```
##  RELEVANCE  n   percent
##         -2  3 0.0234375
##         -1  3 0.0234375
##          0  7 0.0546875
##          1 12 0.0937500
##          2 45 0.3515625
##          3 58 0.4531250
```

```
# Frequencies of the different ratings of relevance for the Pilot
library(janitor)
d %>%
  filter(SOURCE=="PILOT") %>%
  tabyl(RELEVANCE, sort = TRUE)
```

```
##  RELEVANCE  n   percent
##         -3  3 0.0234375
##         -2 10 0.0781250
##         -1 15 0.1171875
##          0 23 0.1796875
##          1 42 0.3281250
##          2 24 0.1875000
##          3 11 0.0859375
```

```
# Frequencies of the different ratings of relevance for the Cleaner
library(janitor)
d %>%
  filter(SOURCE=="CLEANER") %>%
  tabyl(RELEVANCE, sort = TRUE)
```

```
##  RELEVANCE  n   percent
```

```
##         -3  1 0.0078125
##         -2 14 0.1093750
##         -1 23 0.1796875
##          0 29 0.2265625
##          1 39 0.3046875
##          2 10 0.0781250
##          3 12 0.0937500
```

```
# Source model for relevance (Bayesian ordinal model with participant as random effect)
d<-within(d, SOURCE <- relevel(SOURCE, ref = "EDUCATOR"))
# Transform Relevance into positive ratings from 1 to 7
d<-d%>%
  mutate(RELEVANCEPOS = RELEVANCE+4)
head(d)
```

```
##   X OBSERVATION PARTICIPANT PRETEST PRE_FAMILIAR AGE GENDER NATIONALITY
## 1 1           1           1      -2            1  47  other           1
## 2 2           2           1      -2            1  47  other           1
## 3 1           3           2      -1            3  59  other           1
## 4 2           4           2      -1            3  59  other           1
## 5 1           5           3      -2            3  22  other           1
## 6 2           6           3      -2            3  22  other           1
##   ENGLISH POSTTEST GENERATION RECALL CHAIN CONDITION CONDITION.2  ORDER
## 1       1       -2         F1      4   CH1        C1           A  FIRST
## 2       1       -2         F1      5   CH1        C1           A SECOND
## 3       1       -1         F2      3   CH1        C1           A  FIRST
## 4       1       -1         F2      4   CH1        C1           A SECOND
## 5       1       -1         F3      1   CH1        C1           A  FIRST
## 6       1       -1         F3      3   CH1        C1           A SECOND
##     SOURCE        VIEW PRESTIGE RELEVANCE T_RECALL T_SOURCE RELEVANCEPOS
## 1  CLEANER  PROTABLETS        3         3  185.360   49.780            7
## 2 EDUCATOR ANTITABLETS        4         3  200.689   31.139            7
## 3  CLEANER  PROTABLETS        2        -1  300.108   81.496            3
## 4 EDUCATOR ANTITABLETS        3         3  300.010   56.995            7
## 5  CLEANER  PROTABLETS        2        -1  100.889   19.610            3
## 6 EDUCATOR ANTITABLETS        3         2   46.682   26.031            6
```

```
rel.ordinal<-brm(RELEVANCEPOS~SOURCE+(1|PARTICIPANT), control = list(max_treedepth = 15), family=cumula
```

```
## Compiling the C++ model
```

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

```
## Start sampling
```

```
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
```

```
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 11.474 seconds (Warm-up)
## Chain 1:                7.225 seconds (Sampling)
## Chain 1:                18.699 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0.001 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 10.636 seconds (Warm-up)
## Chain 2:                7.173 seconds (Sampling)
## Chain 2:                17.809 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0.001 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
```

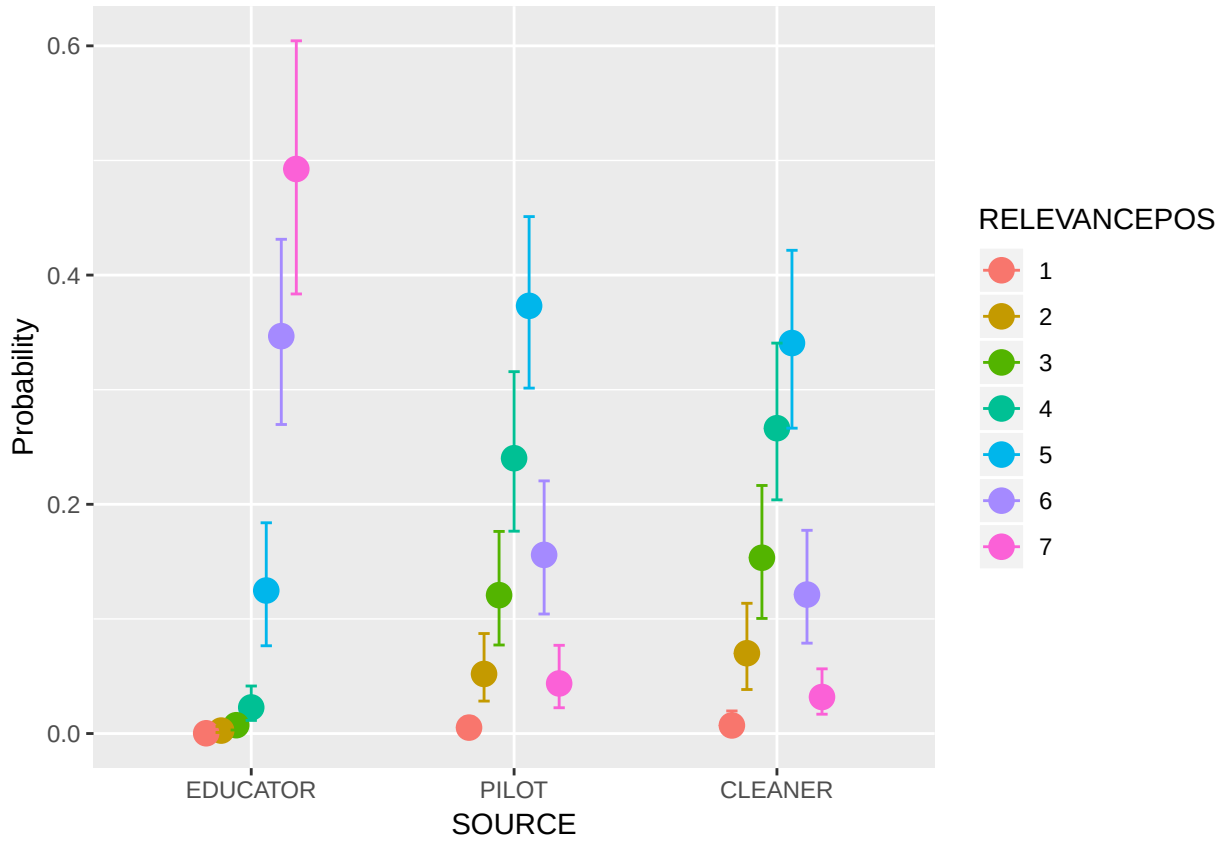
```
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 10.84 seconds (Warm-up)
## Chain 3:                7.129 seconds (Sampling)
## Chain 3:                17.969 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 11.526 seconds (Warm-up)
## Chain 4:                6.902 seconds (Sampling)
## Chain 4:                18.428 seconds (Total)
## Chain 4:
```

```
summary(rel.ordinal, prob=0.89)
```

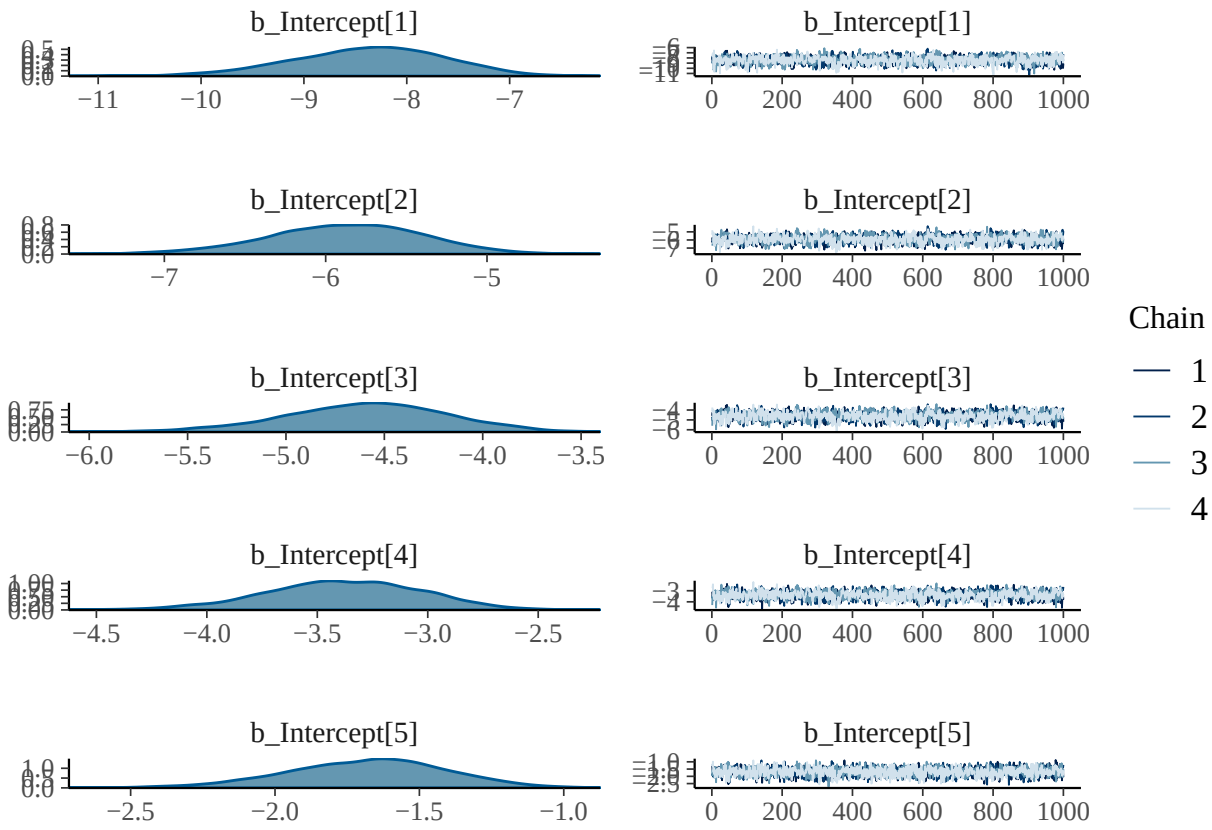
```
##  Family: cumulative
##   Links: mu = logit; disc = identity
## Formula: RELEVANCEPOS ~ SOURCE + (1 | PARTICIPANT)
##    Data: d (Number of observations: 384)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup samples = 4000
##
## Group-Level Effects:
## ~PARTICIPANT (Number of levels: 192)
##               Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## sd(Intercept)     1.40      0.24     1.03     1.77        639 1.01
##
## Population-Level Effects:
##               Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## Intercept[1]     -8.38      0.74    -9.61    -7.25       1097 1.00
## Intercept[2]     -5.88      0.48    -6.67    -5.14        904 1.00
## Intercept[3]     -4.59      0.41    -5.28    -3.95        921 1.00
## Intercept[4]     -3.38      0.35    -3.96    -2.84       1065 1.00
```

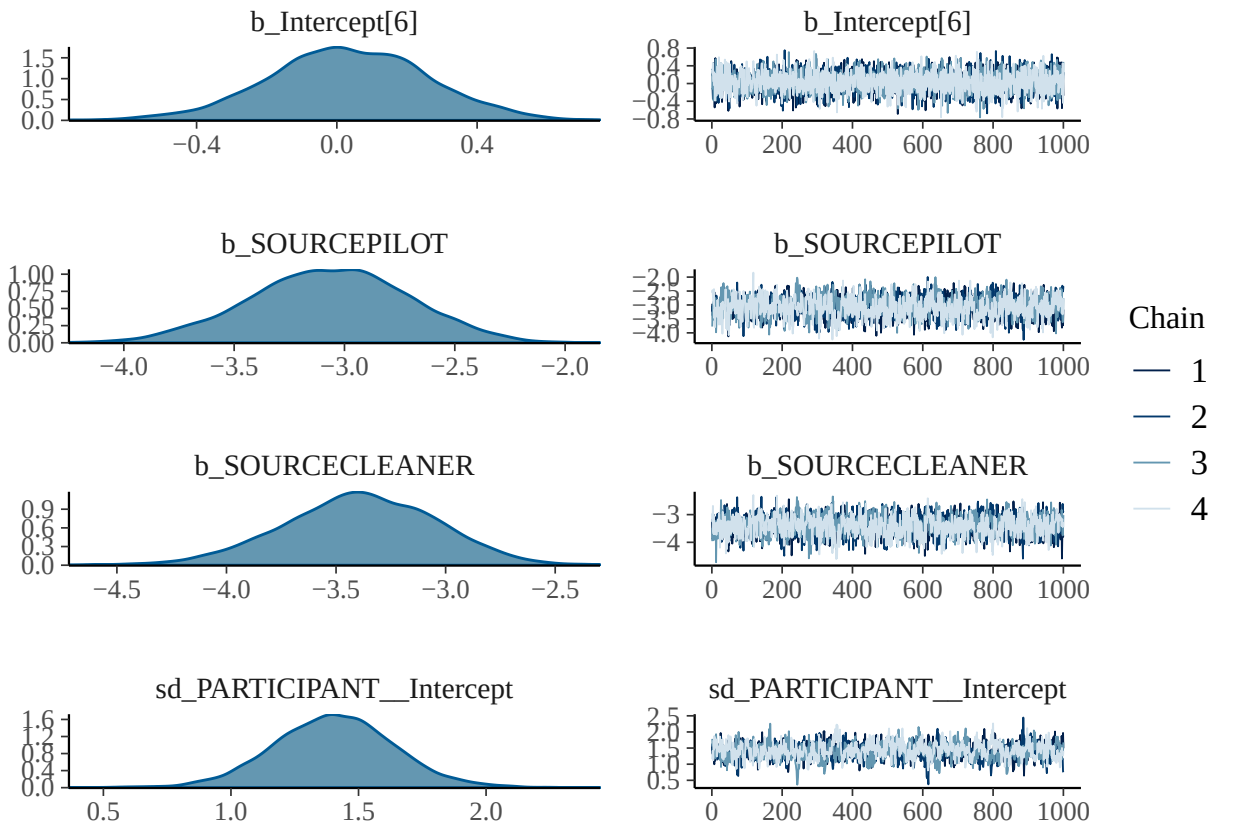
```
## Intercept[5]     -1.68      0.27    -2.14    -1.26       1652 1.00
## Intercept[6]      0.03      0.23    -0.33     0.40       4801 1.00
## SOURCEPILOT      -3.07      0.36    -3.67    -2.50       1026 1.00
## SOURCECLEANER    -3.39      0.35    -3.96    -2.85       1171 1.00
##
## Samples were drawn using sampling(NUTS). For each parameter, Eff.Sample
## is a crude measure of effective sample size, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
marginal_effects(rel.ordinal, categorical=TRUE)
```

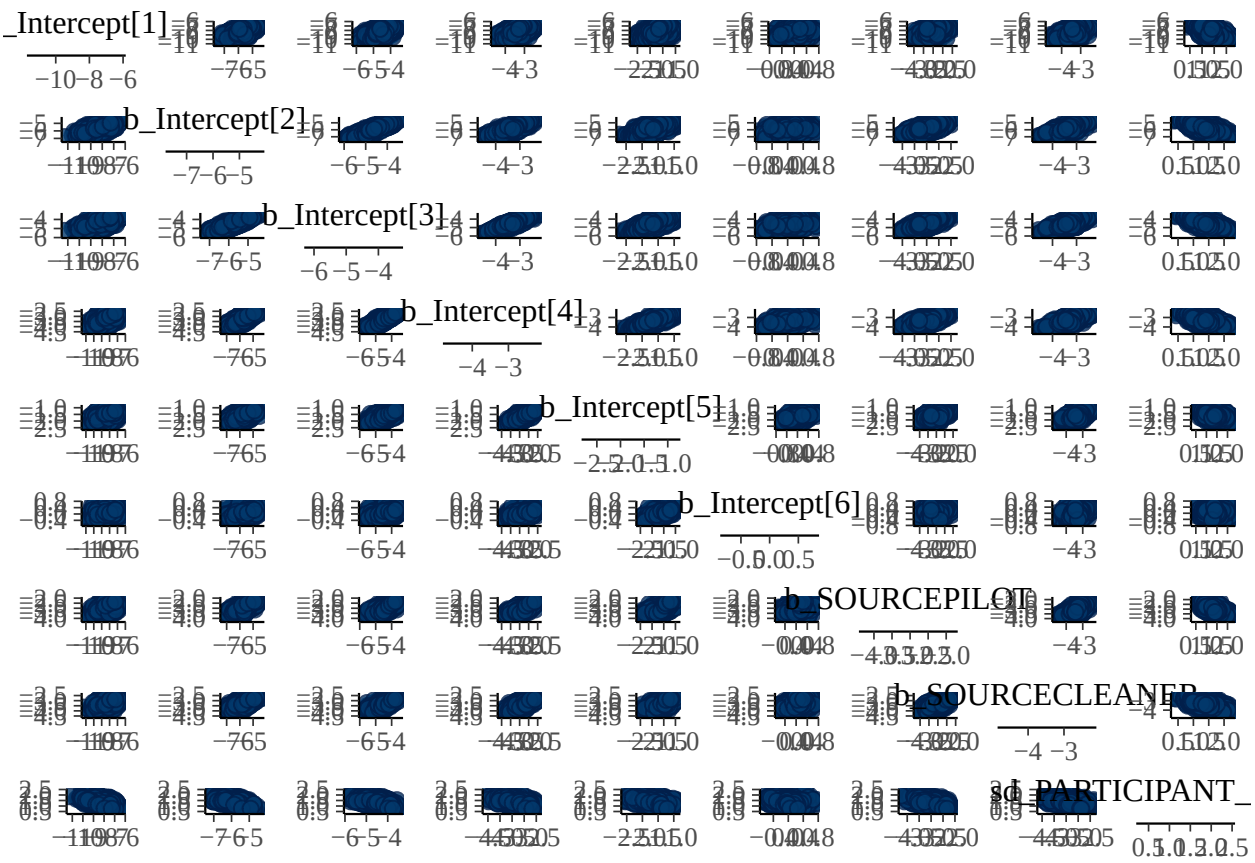



```
plot(rel.ordinal)
```


b_Intercept[1]
b_Intercept[1]
b_Intercept[2]
b_Intercept[2]
b_Intercept[3]
b_Intercept[3]
b_Intercept[4]
b_Intercept[4]
b_Intercept[5]
b_Intercept[5]
Chain
1
2
3
4




```
pairs(rel.ordinal)
```




```
d<-within(d, SOURCE <- relevel(SOURCE, ref = "CLEANER"))
rel.ordinal<-brm(RELEVANCEPOS~SOURCE+(1|PARTICIPANT), control = list(max_treedepth = 15), family=cumula
```

```
## Compiling the C++ model
## recompiling to avoid crashing R session
## Start sampling
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
```

```
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 11.06 seconds (Warm-up)
## Chain 1:                11.205 seconds (Sampling)
## Chain 1:                22.265 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 11.599 seconds (Warm-up)
## Chain 2:                7.293 seconds (Sampling)
## Chain 2:                18.892 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 10.26 seconds (Warm-up)
## Chain 3:                6.938 seconds (Sampling)
```

```
## Chain 3:                17.198 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL '1b726ff50c235e73d05586071a94906d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 11.292 seconds (Warm-up)
## Chain 4:                9.916 seconds (Sampling)
## Chain 4:                21.208 seconds (Total)
## Chain 4:
```

```
summary(rel.ordinal, prob=0.89)
```

```
##  Family: cumulative
##   Links: mu = logit; disc = identity
## Formula: RELEVANCEPOS ~ SOURCE + (1 | PARTICIPANT)
##    Data: d (Number of observations: 384)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup samples = 4000
##
## Group-Level Effects:
## ~PARTICIPANT (Number of levels: 192)
##               Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## sd(Intercept)     1.39      0.23     1.02     1.75        581 1.00
##
## Population-Level Effects:
##                Estimate Est.Error l-89% CI u-89% CI Eff.Sample Rhat
## Intercept[1]      -4.97      0.61    -5.97    -4.06       1845 1.00
## Intercept[2]      -2.47      0.29    -2.94    -2.03       1704 1.00
## Intercept[3]      -1.19      0.23    -1.56    -0.84       2251 1.00
## Intercept[4]       0.02      0.21    -0.31     0.36       3028 1.00
## Intercept[5]       1.71      0.24     1.34     2.10       1665 1.00
## Intercept[6]       3.42      0.32     2.94     3.94       1192 1.00
## SOURCEEDUCATOR     3.39      0.35     2.85     3.96       1071 1.00
## SOURCEPILOT        0.33      0.24    -0.06     0.72       3606 1.00
##
## Samples were drawn using sampling(NUTS). For each parameter, Eff.Sample
## is a crude measure of effective sample size, and Rhat is the potential
```

```
## scale reduction factor on split chains (at convergence, Rhat = 1).
```