Analyses pilot 2

Matteo Lisi

2024-12-18

Table of contents

Libraries

```
library(lme4)
library(kableExtra)
library(tidyverse)
library(patchwork)
library(sjPlot)
```

Data

The dataset contain responses from a total of 101 participants

```
d <- read_csv("ECFullPilot2.csv")
nrow(d)
```

```
[1] 100
```

Demographic summary

Table 1: Demographic Information of Participants

Demographic	Value
Avearge age (Std.)	36.68 (11.28)
Age Range	18 to 67
N. males (Gender = 1)	45
N. female (Gender = 2)	51
N. non binary (Gender = 3)	0

Exclusions

Table 2: Each line indicates how many participants failed each criterion (applied sequentially).

Exclusion Criterion	Number Excluded	Fraction Excluded	Percentage Excluded
Status == 0	0	0.00	0
Finished == 1	2	0.02	2
Q_RecaptchaScore >= 0.5	0	0.00	0
failed attention check	0	0.00	0
Total	2	0.02	2

Use of external resources

AS shown below, this was generally low, and there seems to be not much difference between conditions.

Table 3: Percentage and number of responses (pooled over participants) reporting use of external resources split by condition.

condition	%	N
baseline	4.86	7
feedback	8.00	12
feedback + justification	2.00	3
justification	2.08	3

Familiarity

A large fraction of participants reported familiarity with some of the CRT problems:

Table 4: Percentage and number of people reporting familiarity split by CRT item.

CRT	%	N
1	41.84	41
2	34.69	34
3	11.22	11
4	34.69	34
5	33.67	33
6	9.18	9

Data transformation

We transform the data into a long format:

```r
d_crt_long <- d %>%
  mutate(participant = row_number()) %>% # Add participant ID
  # ID + cols matching 'crt#_i' or 'crt#_r'
  select(participant, justification, feedback, matches('^crt[1-6]_[ir]$')) %>%
  pivot_longer(
    cols = matches('^crt[1-6]_[ir]$'),
    names_to = 'item',
    values_to = 'response'
  ) %>%
  extract( # Extract 'problem' number and 'condition' from 'item'
    item,
    into = c('crt', 'problem', 'condition'),
    regex = '(crt)([1-6])_([ir])'
  ) %>%
  mutate(
    problem = as.integer(problem),
    condition = recode(condition, 'i' = 'intuitive', 'r' = 'reflexive')
  ) %>%
  select(participant, problem, condition, response, justification, feedback)

# Pivot familiarity data (one value per participant-problem)
d_fam_long <- d %>%
  mutate(participant = row_number()) %>%
  pivot_longer(
    cols = matches("^familiarity_[1-6]$"),
    names_to = "fam_item",
    values_to = "familiarity"
  ) %>%
  mutate(problem = as.integer(str_extract(fam_item, "[1-6]"))) %>%
  select(participant, problem, familiarity)

# Pivot ext_resources_use data (one value per participant-problem)
d_ext_long <- d %>%
  mutate(participant = row_number()) %>%
  pivot_longer(
    cols = matches("^ext_resources_use_[1-6]$"),
    names_to = "ext_item",
    values_to = "ext_resources_use"
  ) %>%
  mutate(problem = as.integer(str_extract(ext_item, "[1-6]"))) %>%
  select(participant, problem, ext_resources_use)

# Pivot dot_crt responses (one value per participant-problem)
d_dot_long <- d %>%
  mutate(participant = row_number()) %>%
  pivot_longer(
    cols = matches("^dot_crt[1-6]$"),
    names_to = "dot_item",
    values_to = "dot_response"
  ) %>%
  mutate(problem = as.integer(str_extract(dot_item, "[1-6]"))) %>%
```

```r
  select(participant, problem, dot_response)

# correct answers to dot problems (double checked on Qualtrics)
dot_correct <- c(3,2,4,2,1, 3)
d_dot_long <- d_dot_long %>%
  mutate(dot_accuracy = ifelse(dot_response==dot_correct[d_dot_long$problem], 1, 0))

# Join everything together
d_long <- d_crt_long %>%
  left_join(d_fam_long, by = c("participant","problem")) %>%
  left_join(d_ext_long, by = c("participant","problem")) %>%
  left_join(d_dot_long, by = c("participant","problem")) %>%
  # Now each (participant, problem) has two rows (for each phase),
  # and familiarity and ext_resources_use are duplicated for both.
  select(participant, response, justification, feedback, problem,
         condition, problem, familiarity, ext_resources_use, dot_accuracy)

# check output
str(d_long)
```

```
tibble [1,176 x 9] (S3: tbl_df/tbl/data.frame)
 $ participant      : int [1:1176] 1 1 1 1 1 1 1 1 1 1 ...
 $ response         : num [1:1176] 1 1 2 1 1 3 1 1 1 1 ...
 $ justification    : chr [1:1176] "J" "J" "J" "J" ...
 $ feedback         : chr [1:1176] "F" "F" "F" "F" ...
 $ problem          : int [1:1176] 1 2 3 4 5 6 1 2 3 4 ...
 $ condition        : chr [1:1176] "intuitive" "intuitive" "intuitive" "intuitive" ...
 $ familiarity      : num [1:1176] 1 1 0 0 1 0 1 1 0 0 ...
 $ ext_resources_use: num [1:1176] 0 0 0 0 0 0 0 0 0 0 ...
 $ dot_accuracy     : num [1:1176] 0 1 0 1 1 1 0 1 0 1 ...
```

Next we add response accuracy (the response option “1” was always the correct one)

```r
# change any -99 to NA
d_long$response <- ifelse(d_long$response==-99, NA, d_long$response)

# compute accuracy
d_long$accuracy <- ifelse(!is.na(d_long$response),
                          ifelse(d_long$response==1,1,0),
                          d_long$response)
```

Dot accuracy (concurrent memory task in the intuitive phase 1)

We can check the accuracy of responses to the dot task in the intuitive phase

```r
mean(d_long$dot_accuracy)
```

```
[1] 0.8911565
```

```r
with(d_long, tapply(dot_accuracy, problem, mean))
```

```
        1         2         3         4         5         6 
0.9183673 0.8775510 0.8061224 0.9591837 0.9591837 0.8265306 
```

Prepare data for modelling

In order to enter the data in the models specified as in the pre-registration, we need some additional steps. We first transform the data such that the responses in the two phases are on distinct columns:

```
d_all <- d_long %>%
  pivot_wider(
    names_from = condition,
    values_from = c(response, accuracy),
    names_sep = "_")
str(d_all)
```

```
tibble [588 x 11] (S3: tbl_df/tbl/data.frame)
 $ participant       : int [1:588] 1 1 1 1 1 1 2 2 2 2 ...
 $ justification     : chr [1:588] "J" "J" "J" "J" ...
 $ feedback          : chr [1:588] "F" "F" "F" "F" ...
 $ problem           : int [1:588] 1 2 3 4 5 6 1 2 3 4 ...
 $ familiarity       : num [1:588] 1 1 0 0 1 0 0 0 0 1 ...
 $ ext_resources_use : num [1:588] 0 0 0 0 0 0 0 0 0 0 ...
 $ dot_accuracy      : num [1:588] 0 1 0 1 1 1 1 1 1 1 ...
 $ response_intuitive: num [1:588] 1 1 2 1 1 3 2 1 1 2 ...
 $ response_reflexive: num [1:588] 1 1 1 1 1 2 1 1 1 1 ...
 $ accuracy_intuitive: num [1:588] 1 1 0 1 1 0 0 1 1 0 ...
 $ accuracy_reflexive: num [1:588] 1 1 1 1 1 0 1 1 1 1 ...
```

Apply exclusions

This is a good time to also exclude responses to problems in which participants declared consulting external resources (this will remove both phase 1 and phase 2 responses).

```
d_all <- d_all[d_all$ext_resources_use==0,]
```

We can also apply the exclusion criteria based on the accuracy of the dot task

```
d_all <- d_all[d_all$dot_accuracy==1,]
```

After excluding responses based on use of external resources & failed concurrent memory (dot) task, exclusion based on familiarity will exclude a further 27.74 % of total responses)

If instead we excluded responses with self-reported familiarity AND a correct answer in either intuitive or reflexive phase we would exclude 23.35 % of responses. Alternatively, if we excluded responses with self-reported familiarity AND correct answer in BOTH intuitive or reflexive phase, we would have 16 % responses excluded.

Although it exclude a substantial fraction of data, we proceed applying the exclusion criteria as in the preregistration draft (i.e. by excluding all responses with self-reported familiarity)

```
d_all <- d_all[d_all$familiarity==0,]
```

Compute dependent variables for specific hypotheses

For Hypothesis 1 (*"less than half of correct answers will arise from error correction during deliberation"*) we restrict analyses to all correct responses, and examine the proportion of correct responses made in phase 2 out of all correct responses:

```
dat_H1 <- d_all %>%
  filter(accuracy_intuitive == 1 | accuracy_reflexive == 1) %>% # keep only correct, either at phase 1 o
  filter(!is.na(accuracy_intuitive)) %>% # excluded missing responses in phase 1
  mutate(correct_phase2 = ifelse(accuracy_intuitive == 0, accuracy_reflexive, 0)) %>%
  select(correct_phase2, participant, problem, justification, feedback)
str(dat_H1)
```

```
tibble [204 x 5] (S3: tbl_df/tbl/data.frame)
 $ correct_phase2: num [1:204] 0 1 0 0 0 1 0 0 0 0 ...
 $ participant   : int [1:204] 1 2 2 2 2 3 3 3 3 3 ...
 $ problem       : int [1:204] 4 1 2 3 5 1 2 4 5 6 ...
 $ justification : chr [1:204] "J" "NJ" "NJ" "NJ" ...
 $ feedback      : chr [1:204] "F" "F" "F" "F" ...
```

Note that from the dataset for H1 I am excluding missing phase 1 responses. This is because we cannot know whether participant would have responded correctly or not in phase 1 for those problems.

The remaining hypotheses concern the probability intuitive incorrect answers (errors in phase 1) are corrected in the reflexive phase (phase 2). These hypotheses are not relevant for the pilot since we have only 1 condition. We can still use this data to estimate the probability of correction in the baseline condition.

```
# The other hypotheses
dat_Hother <- d_all %>%
  filter(accuracy_intuitive == 0) %>% # keep only intuitive errors
  mutate(corrected = accuracy_reflexive) %>%
  select(corrected, participant, problem, justification, feedback)
```

Data summaries

Accuracy

Proportion of correct responses split by item and phase:

Table 5: Mean accuracy by CRT problem and phase (here 'intuitive' is phase 1 and 'reflexive' is phase 2)

justification	feedback	problem	intuitive	reflexive
J	F	1	0.13	0.60
J	F	2	0.33	0.80
J	F	3	0.00	0.21
J	F	4	0.44	0.81
J	F	5	0.28	0.50
J	F	6	0.06	0.38
J	NF	1	0.25	0.75
J	NF	2	0.46	0.92
J	NF	3	0.11	0.61
J	NF	4	0.33	0.67
J	NF	5	0.18	0.27
J	NF	6	0.44	0.33
NJ	F	1	0.00	0.92
NJ	F	2	0.46	0.62
NJ	F	3	0.08	0.31
NJ	F	4	0.23	0.69
NJ	F	5	0.21	0.43
NJ	F	6	0.17	0.61
NJ	NF	1	0.33	0.67
NJ	NF	2	0.40	0.60
NJ	NF	3	0.00	0.50
NJ	NF	4	0.45	0.73
NJ	NF	5	0.08	0.54
NJ	NF	6	0.29	0.21

Missing responses

The number of missing responses is much lower compared to the first pilot

Table 6: Proportion and number of missing responses in phase 1 (intuitive)

problem	prop_missing	N_missing
1	0.03	3
2	0.02	2
3	0.07	7
4	0.03	3
5	0.06	6
6	0.13	13

Responses correct in phase 1 and turned into errors in phase 2

Using the data transformed as in the object `d_all` above we can easily check how frequently participants made a correct response in phase 1, and changed it into an error in phase 2. The table below shows how many times this occurred for each CRT problem:

```
correct2error_table <- d_all %>%
  filter(accuracy_intuitive == 1 | accuracy_reflexive == 1) %>%
  select(accuracy_intuitive,accuracy_reflexive, problem) %>%
  filter(accuracy_reflexive==0) %>%
  group_by(problem) %>%
  summarise(N = sum(accuracy_intuitive))

print(correct2error_table)
```

```
# A tibble: 6 x 2
  problem     N
    <int> <dbl>
1       1     1
2       2     2
3       3     1
4       4     1
5       5     1
6       6     7
```

Overall across all problems this occurred 13 times.

Confidence calibration

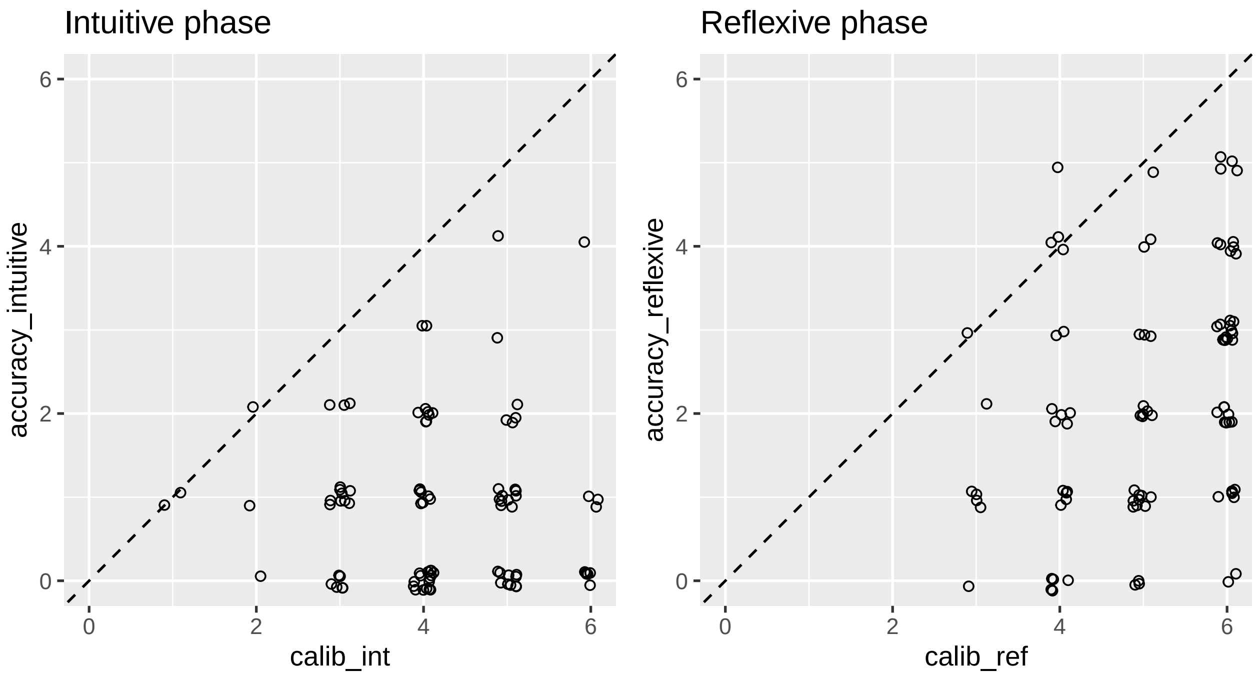

Figure 1: Number of correct answer (vertical coordinates) plotted as a function of the answer to the calibration responses (e.g. the participant's estimates of their own accuracy; horizontal coordinates). Some jitter added for visibility. Both plots demonstrates substantial overconfidence, since in both cases most points lie below the identity line.

Analyses

Recode justification and feedback as dummy variables

```
dat_H1$justification <- ifelse(dat_H1$justification=="J", 1, 0)
dat_H1$feedback <- ifelse(dat_H1$feedback=="F", 1, 0)

dat_Hother$justification <- ifelse(dat_Hother$justification=="J", 1, 0)
dat_Hother$feedback <- ifelse(dat_Hother$feedback=="F", 1, 0)
```

H1

For the pilot we use `glm` instead of `glmer` since there is not geopolitcal region to group random effects as pre-registered.

```
mH1_p <- glm(correct_phase2 ~ feedback * justification, family = binomial("logit"), data=dat_H1)
summary(mH1_p)
```

```

Call:
glm(formula = correct_phase2 ~ feedback * justification, family = binomial("logit"),
    data = dat_H1)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             0.46262    0.30961   1.494    0.135
feedback                0.29115    0.43333   0.672    0.502
justification          -0.25498    0.40691  -0.627    0.531
feedback:justification -0.02878    0.58201  -0.049    0.961

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 272.34  on 203  degrees of freedom
Residual deviance: 270.47  on 200  degrees of freedom
AIC: 278.47

Number of Fisher Scoring iterations: 4
```

The model indicate that out of all correct responses, the proportion of correct responses made in the reflective phase is actually greater than 50% in the baseline condition (although not significant), thus contrary to the hypothesis. It can be computed by transforming the intercept parameter from log-odds to probability:

```
exp(coef(mH1_p)['(Intercept)'])/(1+exp(coef(mH1_p)['(Intercept)']))
```

```
(Intercept)
  0.6136364
```

```
H1_CI <- confint(mH1_p)
exp(H1_CI[1,])/(1+exp(H1_CI[1,]))
```

```
    2.5 %    97.5 %
0.4663315 0.7480624
```

We can use `glmer` for the robustness check with random intercepts grouped by CRT problem:

```
mH1_mm <- glmer(correct_phase2 ~ feedback * justification + (1|problem), family = binomial("logit"), da
summary(mH1_mm)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: correct_phase2 ~ feedback * justification + (1 | problem)
   Data: dat_H1

     AIC      BIC   logLik deviance df.resid
   272.2    288.8   -131.1    262.2      199

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.6375 -1.0272  0.4930  0.8037  1.1708

Random effects:
 Groups  Name        Variance Std.Dev.
 problem (Intercept) 0.4534   0.6734
Number of obs: 204, groups:  problem, 6

Fixed effects:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             0.48005    0.42589   1.127    0.260
feedback                0.43268    0.45475   0.951    0.341
justification          -0.25477    0.42884  -0.594    0.552
feedback:justification  0.01928    0.61022   0.032    0.975

Correlation of Fixed Effects:
            (Intr) fedbck jstfct
feedback    -0.538
justificatn -0.577  0.541
fdbck:jstfc  0.408 -0.736 -0.707
```

Probability of correction

```
mHx_p <- glm(corrected ~ feedback * justification, family = binomial("logit"), data=dat_Hother)
summary(mHx_p)
```

```
Call:
glm(formula = corrected ~ feedback * justification, family = binomial("logit"),
    data = dat_Hother)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             -0.1382     0.2632  -0.525    0.600
feedback                 0.1680     0.3592   0.468    0.640
justification            0.2366     0.3675   0.644    0.520
feedback:justification  -0.5384     0.4998  -1.077    0.281

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 360.05  on 259  degrees of freedom
Residual deviance: 358.64  on 256  degrees of freedom
AIC: 366.64

Number of Fisher Scoring iterations: 3
```

The probability of correction (i.e. that an error in the intuitive phase is corrected in the deliberative phase) in the baseline condition is

```
beta <- coef(mHx_p)
exp(beta[1]) / (1 + exp(beta[1]))
```

```
(Intercept)
  0.4655172
```

It increases to over 50% in both feedback and justification condition:

```
exp(beta[1] + beta[2:3]) / (1 + exp(beta[1] + beta[2:3]))
```

```
     feedback justification
    0.5074627     0.5245902
```

It does not increase more in the combined feedback + justification condition:

```
exp(sum(beta)) / (1 + exp(sum(beta)))
```

```
[1] 0.4324324
```

Additional model with random intercept for CRT problem:

```
mHx_mm <- glmer(corrected ~ feedback * justification + (1|problem) + (1|participant), family = binomial
summary(mHx_mm)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula:
corrected ~ feedback * justification + (1 | problem) + (1 | participant)
   Data: dat_Hother
```

```
     AIC      BIC   logLik deviance df.resid 
   356.3    377.6   -172.1    344.3      254 

Scaled residuals: 
    Min      1Q  Median      3Q     Max 
-1.4170 -0.7453 -0.4483  0.7676  2.0205 

Random effects:
 Groups      Name        Variance Std.Dev.
 participant (Intercept) 1.0082   1.0041  
 problem     (Intercept) 0.5332   0.7302  
Number of obs: 260, groups:  participant, 87; problem, 6

Fixed effects:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             0.12043    0.49583   0.243    0.808
feedback                0.01402    0.53124   0.026    0.979
justification           0.04522    0.53712   0.084    0.933
feedback:justification -0.36619    0.73415  -0.499    0.618

Correlation of Fixed Effects:
            (Intr) fedbck jstfct
feedback    -0.587              
justificatn -0.578  0.535       
fdbck:jstfc  0.423 -0.721 -0.732
```