CCM_final

KeFang

2023-05-07

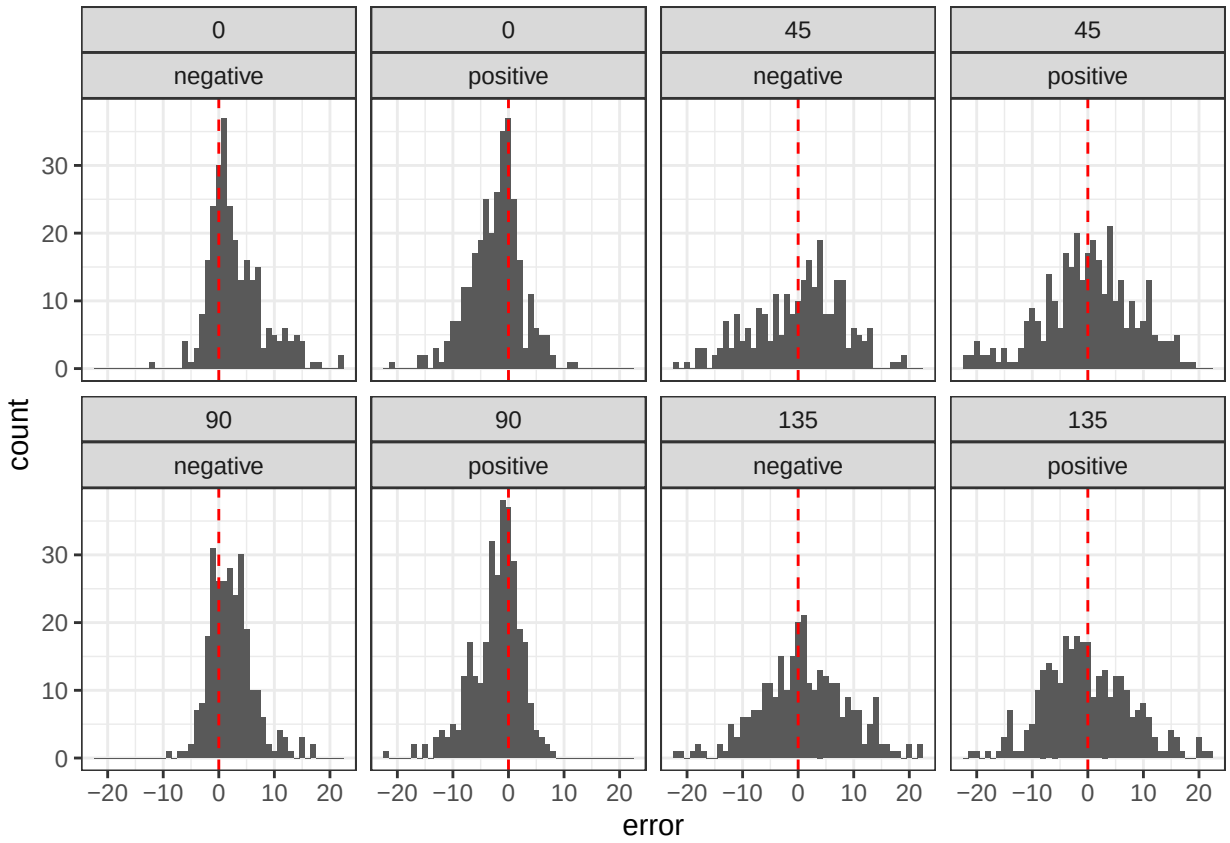
```
# Install required packages
library(tidyverse)
```

```
# import data
ccm_data_clean <- read_csv("ccm_data_clean.csv")
```

```
## Rows: 2276 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (18): subjID, run, trial, phase, oriRef, oriJitt, oriFinal, oriRespInit,...
## lgl  (1): outoftime
##
## 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.
```

```
ccm_data_clean2 <- ccm_data_clean %>%
  select(-run) %>%
  rename(prior = oriRef2,
         stimulus = oriFinal2,
         response = oriRespFinal2,
         stimulus_centered = oriFinal3,
         response_centered = oriRespFinal3) %>%
  select(subjID, trial, prior, stimulus, stimulus_centered, response, response_centered, error) %>%
  mutate(stimulus_pos_neg = ifelse(stimulus_centered >= 0, "positive", "negative"),
         stimulus_value = abs(stimulus_centered)) %>%
  mutate(error_pos_neg = ifelse(error >= 0, "positive", "negative"),
         error_value = abs(error))
```

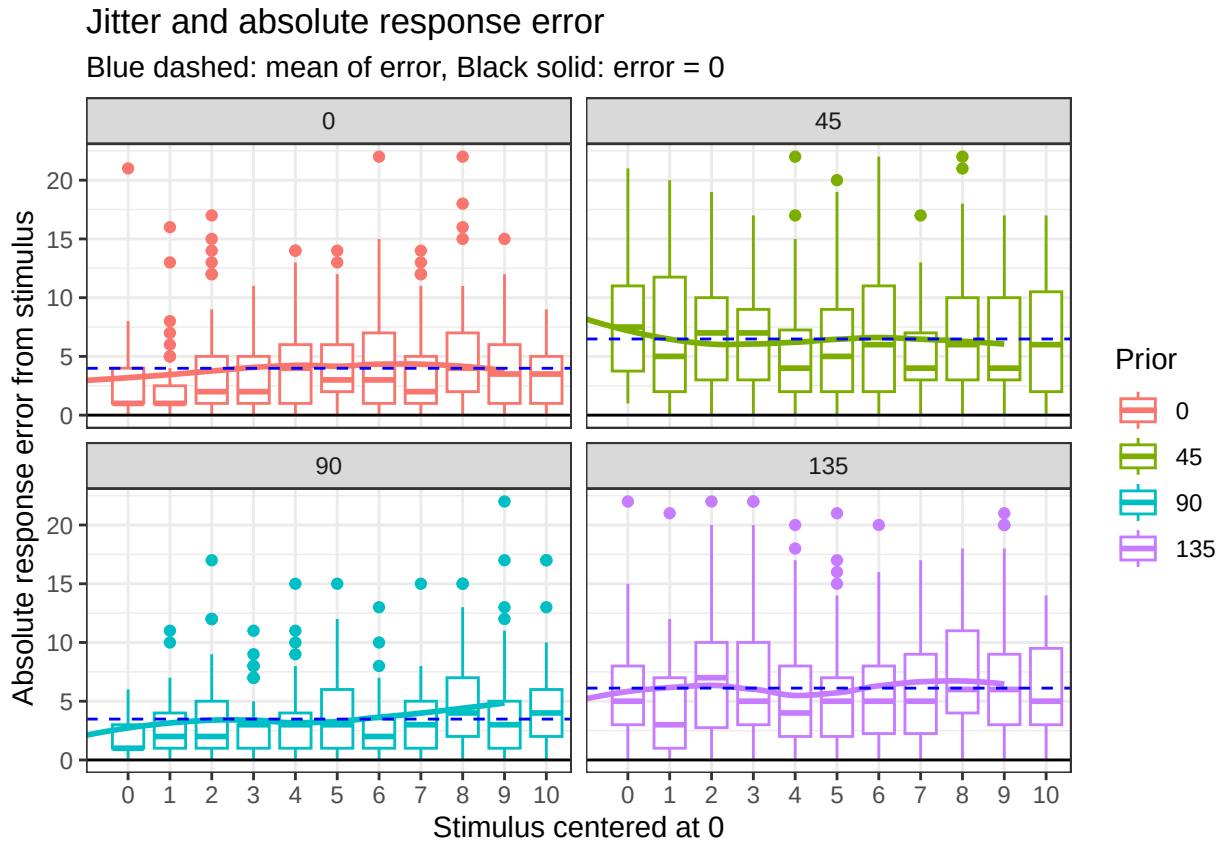
```
ggplot(ccm_data_clean2, aes(x = error)) +
  geom_histogram(binwidth = 1) +
  geom_vline(xintercept = 0, linetype = "dashed", color = 'red') +
  facet_wrap(~as.factor(prior) + stimulus_pos_neg, nrow = 2) +
  theme_bw()
```




```
# Calculate mean error for each group
mean_error <- ccm_data_clean2 %>%
  group_by(prior) %>%
  summarise(mean_error = mean(abs(error)))

ggplot(ccm_data_clean2, aes(color = as.factor(prior))) +
  geom_boxplot(aes(x = as.factor(abs(stimulus_centered)), y = abs(error))) +
  geom_smooth(aes(x = as.numeric(abs(stimulus_centered)), y = as.numeric(abs(error))), se= F) +
  geom_hline(yintercept = 0, linetype = "solid") +
  geom_hline(aes(yintercept = mean_error, group = as.factor(prior)), data = mean_error, linetype = "dash
  facet_wrap(~as.factor(prior))+
  labs(title = "Jitter and absolute response error",
       subtitle = 'Blue dashed: mean of error, Black solid: error = 0',
       x = "Stimulus centered at 0",
       y = "Absolute response error from stimulus",
       color = "Prior") +
  theme_bw()
```

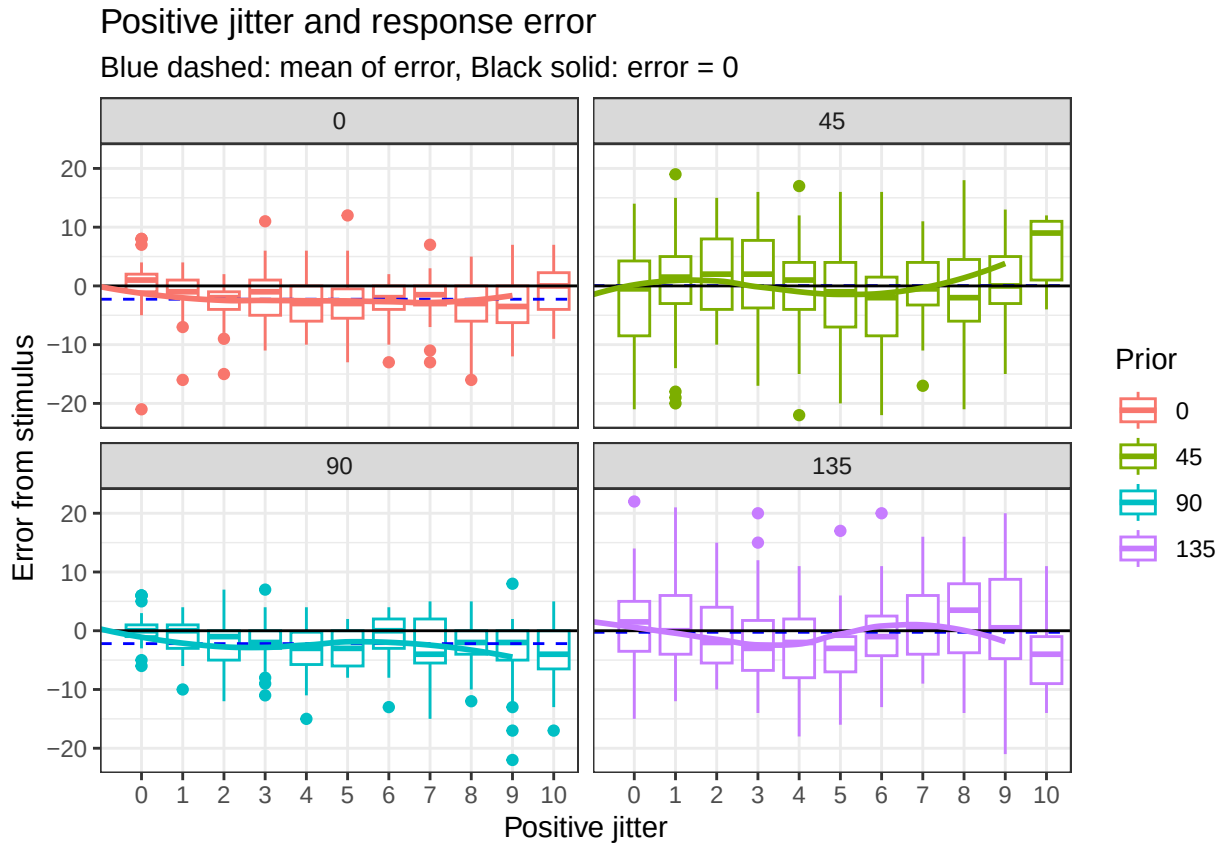
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
# Calculate mean error for each group
pos_mean_error <- ccm_data_clean2 %>%
  filter(stimulus_pos_neg == "positive") %>%
  group_by(prior) %>%
  summarise(pos_mean_error = mean(error))

# Create the ggplot
ggplot(ccm_data_clean2 %>% filter(stimulus_pos_neg == "positive"),
       aes(x = as.factor(stimulus_centered), y = error, color = as.factor(prior))) +
  geom_boxplot() +
  geom_hline(aes(yintercept = pos_mean_error, group = as.factor(prior)), data = pos_mean_error, linetyp
  geom_hline(yintercept = 0, linetype = "solid", color = "black") +
  geom_smooth(aes(x = as.numeric(abs(stimulus_centered)), y = as.numeric(error)), se = F) +
  facet_wrap(~as.factor(prior)) +
  labs(title = "Positive jitter and response error",
       subtitle = 'Blue dashed: mean of error, Black solid: error = 0',
       x = "Positive jitter",
       y = "Error from stimulus",
       color = "Prior") +
  theme_bw()
```

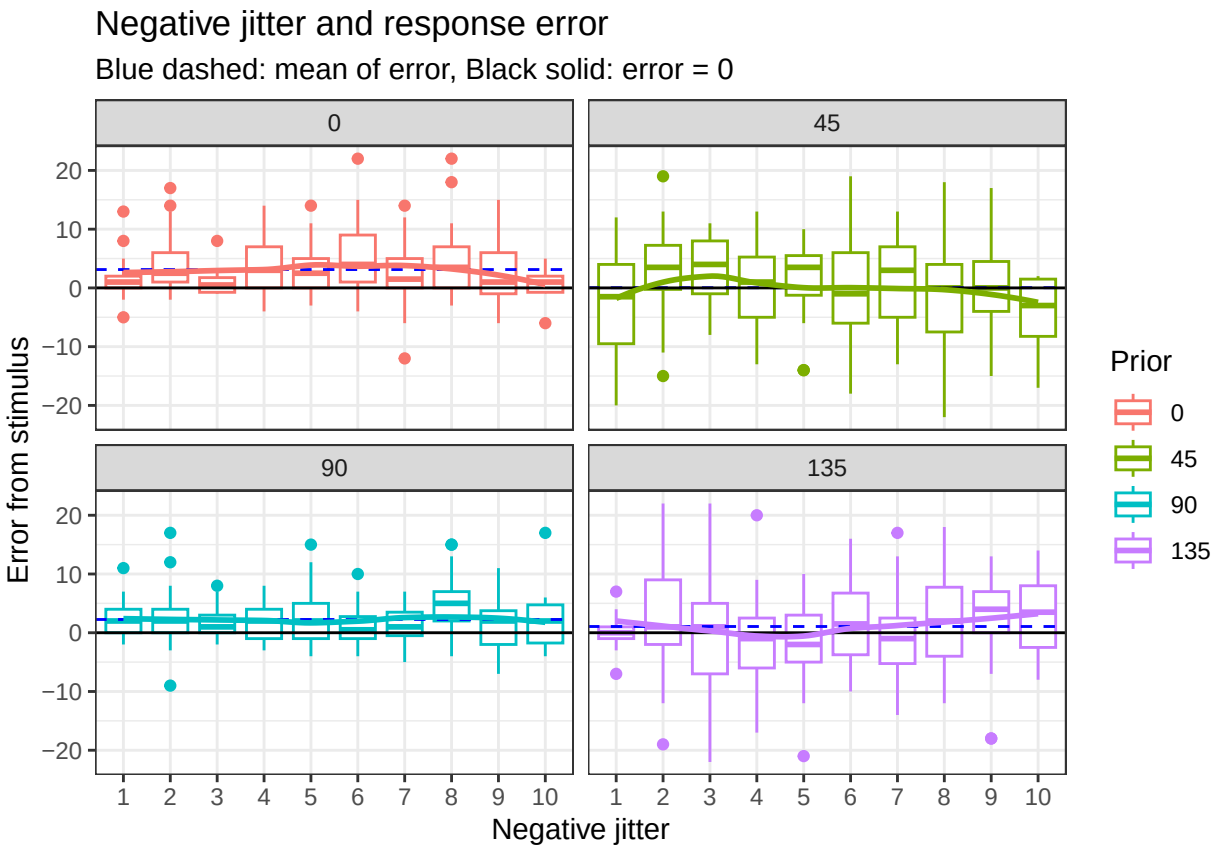
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
# Calculate mean error for each group
neg_mean_error <- ccm_data_clean2 %>%
  filter(stimulus_pos_neg == "negative") %>%
  group_by(prior) %>%
  summarise(neg_mean_error = mean(error))

# Create the ggplot
ggplot(ccm_data_clean2 %>% filter(stimulus_pos_neg == "negative"),
       aes(x = as.factor(abs(stimulus_centered)), y = error, color = as.factor(prior))) +
  geom_boxplot() +
  geom_hline(aes(yintercept = neg_mean_error, group = as.factor(prior)), data = neg_mean_error, linetyp
  geom_hline(yintercept = 0, linetype = "solid", color = "black") +
  geom_smooth(aes(x = as.numeric(abs(stimulus_centered)), y = as.numeric(error)), se = F) +
  facet_wrap(~as.factor(prior)) +
  labs(title = "Negative jitter and response error",
       subtitle = 'Blue dashed: mean of error, Black solid: error = 0',
       x = "Negative jitter",
       y = "Error from stimulus",
       color = "Prior") +
  theme_bw()
```

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


Negative jitter and response error
Blue dashed: mean of error, Black solid: error = 0
0
45
90
135
Error from stimulus
20
10
0
−10
−20
1
2
3
4
5
6
7
8
9
10
Negative jitter
Prior
0
45
90
135