

# E4-analysis-calibration

Rachel Ryskin

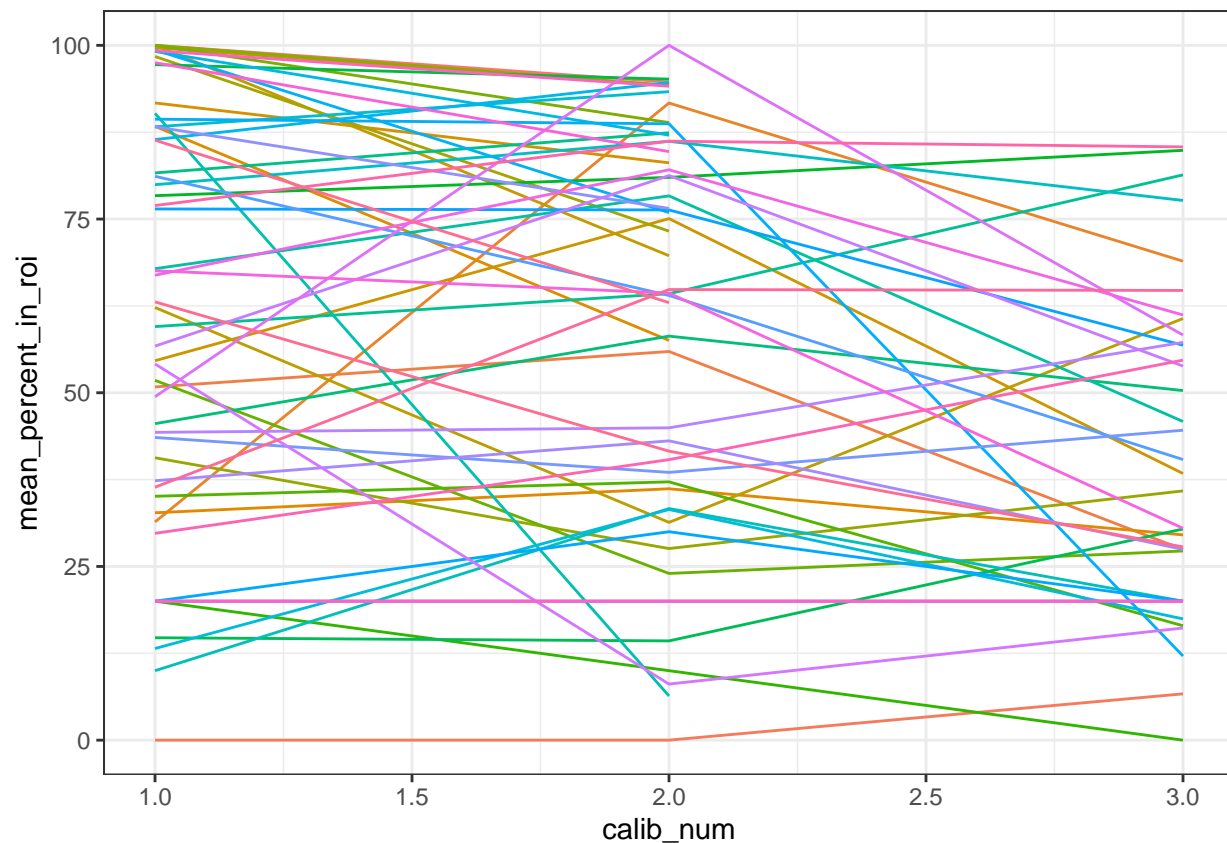
```
data.files <- list.files('data/run-2', full.names = TRUE)
data.tables <- lapply(data.files, function(file){
  data.table <- fromJSON(file)
  return(data.table)
})
all.data <- bind_rows(data.tables)
```

```
all.data.calib = all.data %>%
  filter(trial_type == "webgazer-validate") %>%
  dplyr::select(subject, trial_index, percent_in_roi, average_offset) %>%
  tidyr::unnest(percent_in_roi)
```

```
summary.data.calib = all.data.calib %>%
  group_by(subject, trial_index) %>%
  summarize(mean_percent_in_roi = mean(percent_in_roi)) %>%
  group_by(subject) %>%
  mutate(calib_num = row_number())
```

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

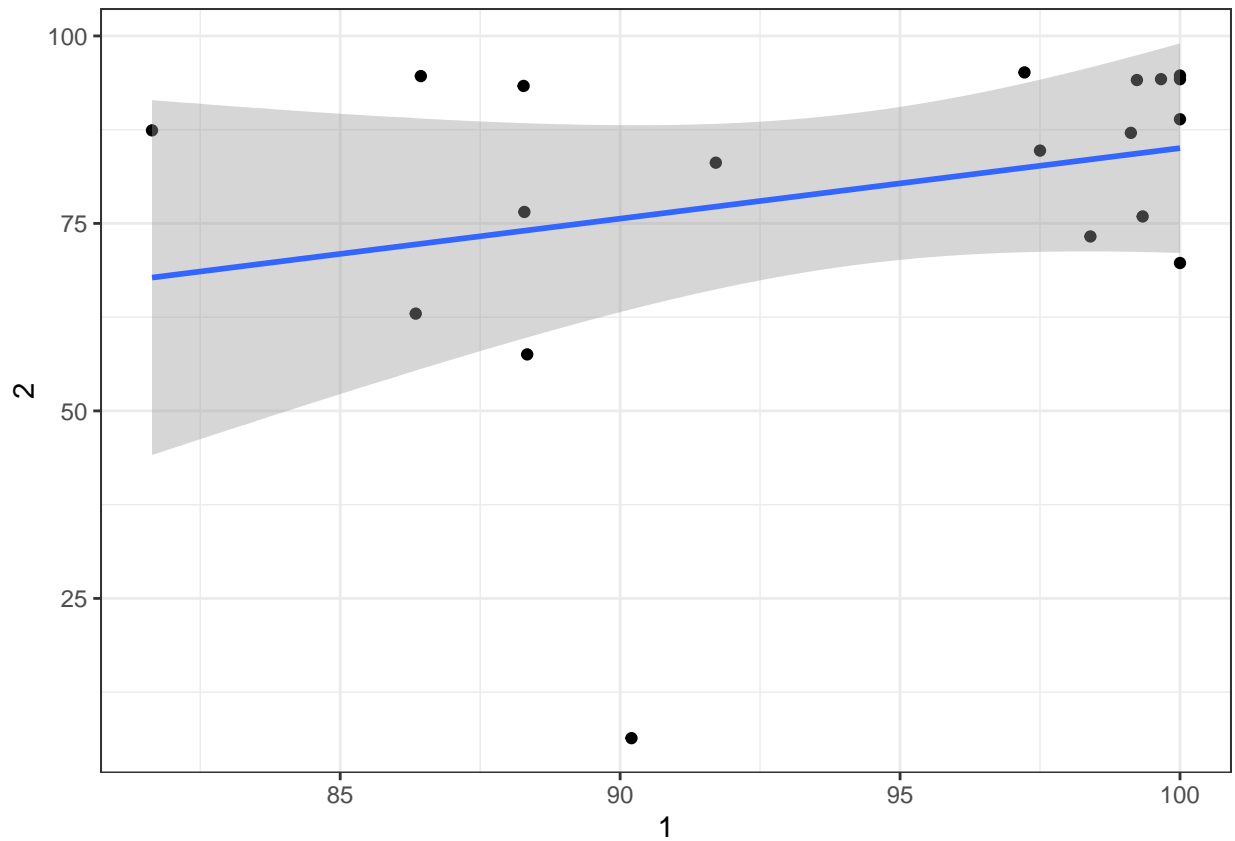
```
ggplot(summary.data.calib)+
  geom_line(aes(x = calib_num, y = mean_percent_in_roi, color=subject))+
  theme_bw()+
  theme(legend.position = "none")
```



```
summary.data.calib.wide = summary.data.calib %>%
  select(-trial_index) %>%
  pivot_wider(id_cols = subject, names_from=calib_num, values_from = mean_percent_in_roi )

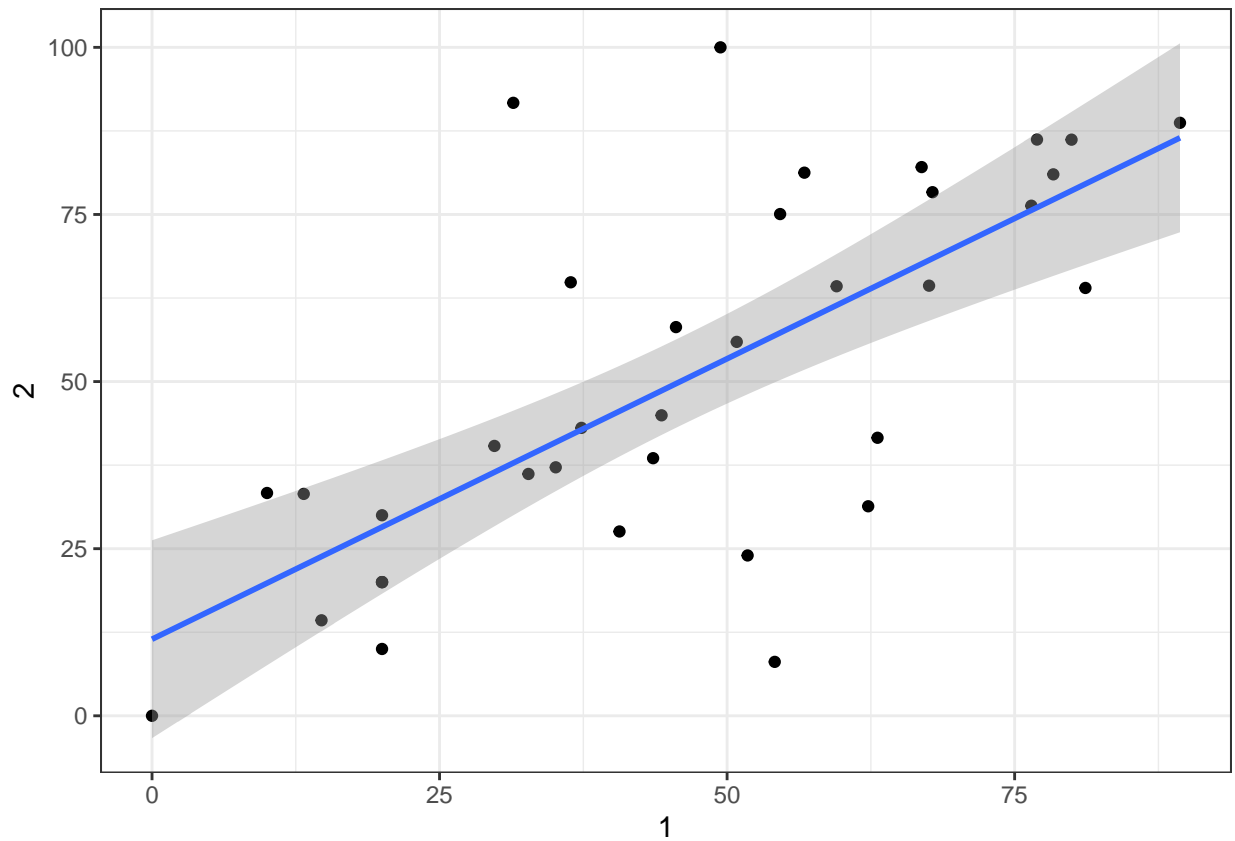
# correlation between initial and halfway calibration
ggplot(summary.data.calib.wide %>% filter(is.na(`3`)), aes(x = `1`, y = `2`))+
  geom_point()+
  geom_smooth(method = 'lm')+
  theme_bw()+
  theme(legend.position = "none")
```

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



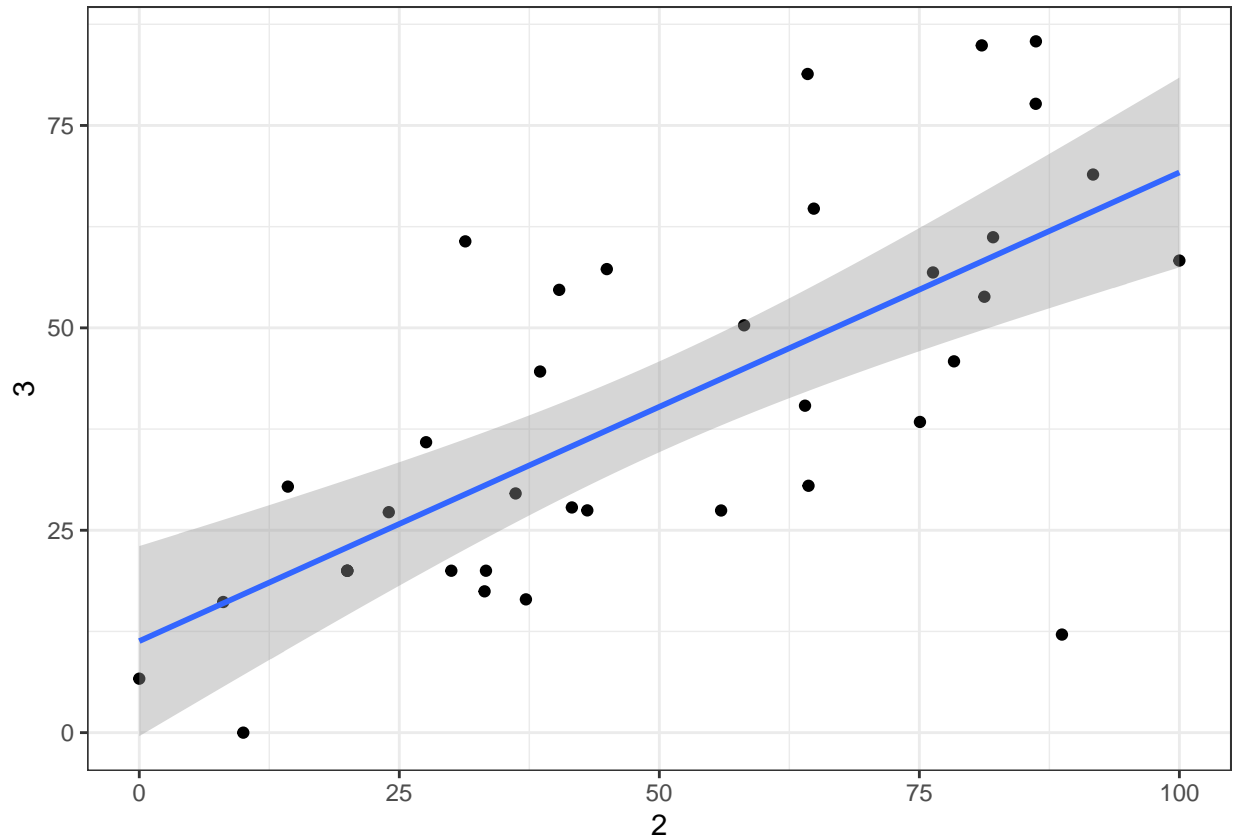
```
# correlation between 2 successive calibration attempts
ggplot(summary.data.calib.wide %>% filter(!is.na(`3`)), aes(x = `1`, y = `2`))+
  geom_point()+
  geom_smooth(method = 'lm')+
  theme_bw()+
  theme(legend.position = "none")
```

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



```
# correlation between second attempt calibration and halfway
ggplot(summary.data.calib.wide %>% filter(!is.na(`3`)), aes(x = `2`, y = `3`))+
  geom_point()+
  geom_smooth(method = 'lm')+
  theme_bw()+
  theme(legend.position = "none")
```

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



```
calib.by.subject = summary.data.calib %>%
  group_by(subject) %>%
  summarize(mean_percent_in_roi = mean(mean_percent_in_roi))
```

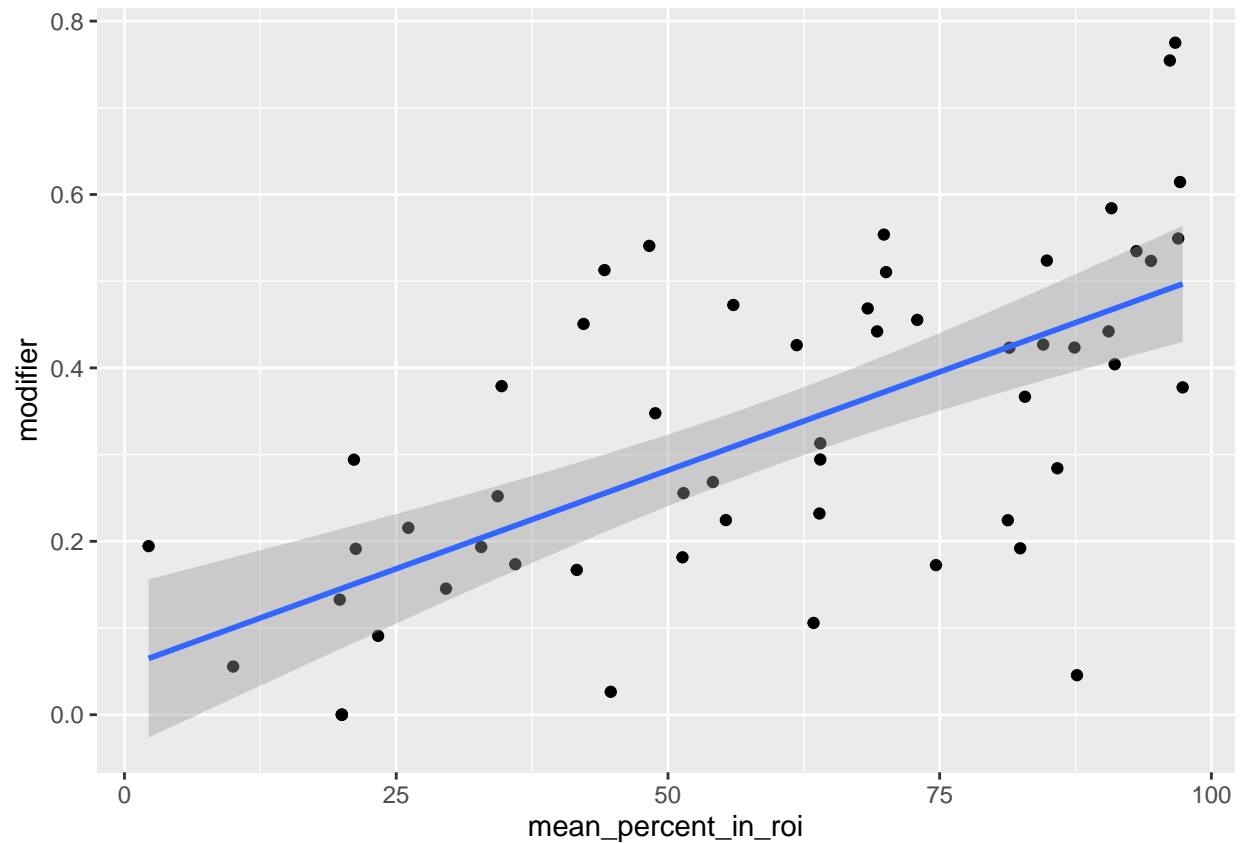
```
eyetracking.effects.by.subject = read_csv( "output/E4_eye-tracking_data.csv") %>%
  rename("condition" = compatibility) %>%
  filter(time.window == "post-instrument-onset", condition != "filler" ) %>%
  group_by(condition, time.window, subject) %>%
  summarize(M = mean(prop.fixations.animal)) %>%
  pivot_wider(names_from = condition, values_from = M) %>%
  mutate(bias_effect = modifier - instrument) %>%
  left_join(calib.by.subject, by = "subject")
```

```
## Rows: 14667 Columns: 7
## -- Column specification -----
## Delimiter: ","
## chr (4): subject, sound, compatibility, time.window
## dbl (3): trialID, prop.fixations.animal, prop.fixations.instrument
##
## 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.
## 'summarise()' has grouped output by 'condition', 'time.window'. You can override using the '.groups'
```

```
ggplot(eyetracking.effects.by.subject, aes(x = mean_percent_in_roi, y = modifier))+
  geom_point()+
```

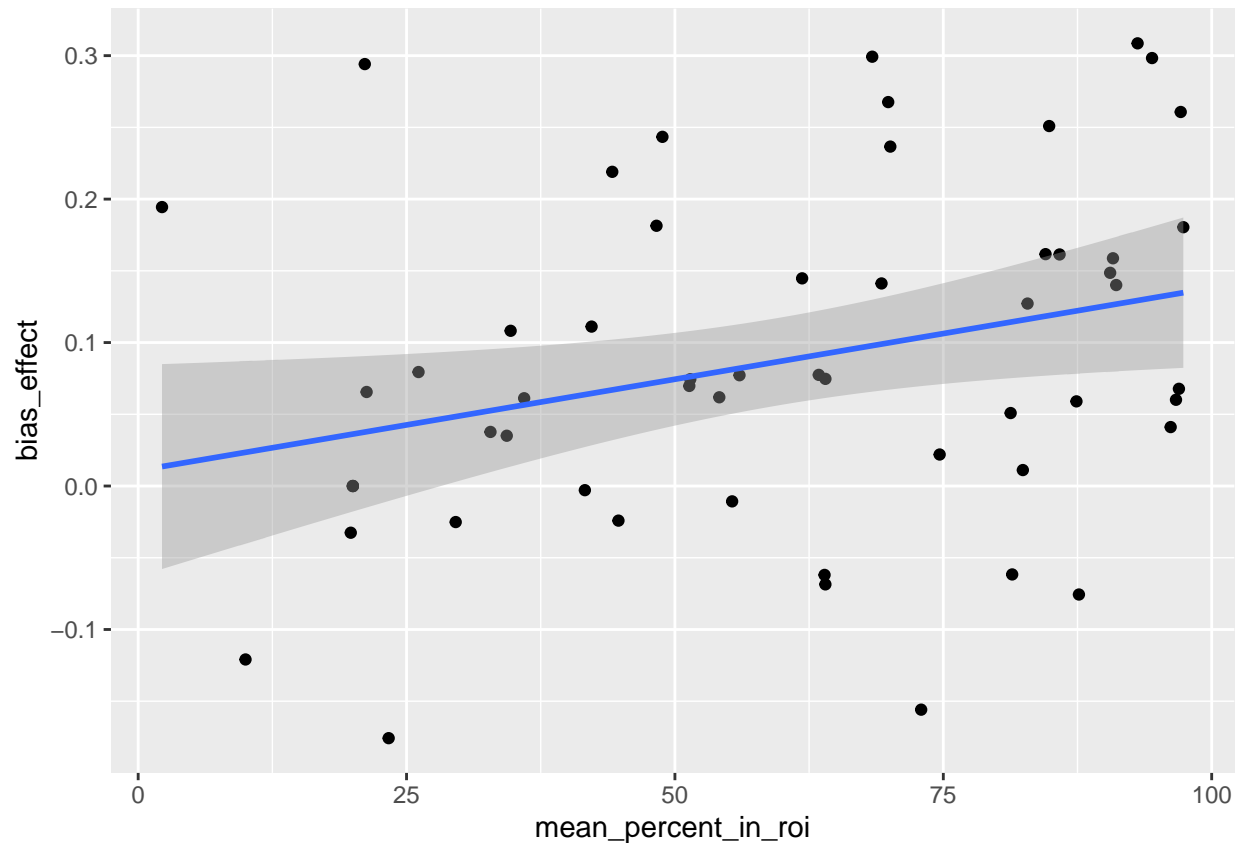
```
geom_smooth(method = "lm")
```

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



```
ggplot(eyetracking.effects.by.subj, aes(x = mean_percent_in_roi, y = bias_effect))+  
  geom_point()+  
  geom_smooth(method = "lm")
```

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



```
eyetracking.window.3 = read_csv( "output/E4_eye-tracking_data.csv") %>%
  filter(time.window == "post-instrument-onset", compatibility != "filler" ) %>%
  mutate(condition = factor(compatibility, levels = c('instrument', 'equibiased', 'modifier'))) %>%
  left_join(calib.by.subj, by = "subject")
```

```
## Rows: 14667 Columns: 7
## -- Column specification -----
## Delimiter: ","
## chr (4): subject, sound, compatibility, time.window
## dbl (3): trialID, prop.fixations.animal, prop.fixations.instrument
##
## 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.
```

```
# Add orthogonal contrasts to model
contrasts(eyetracking.window.3$condition) <- cbind(c(-2/3, 1/3, 1/3), c(0, -1/2, 1/2))
```

```
model.time.window.3 <- lmer(prop.fixations.animal ~ condition + (1 | subject) + (1 | trialID), data=eyetracking.window.3,
  control = lmerControl(optimizer = "bobyqa",
    optCtrl = list(maxfun = 2e6)))
summary(model.time.window.3)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
```

```
## Formula: prop.fixations.animal ~ condition + (1 | subject) + (1 | trialID)
## Data: eyetracking.window.3
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+06))
##
## REML criterion at convergence: 1136.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.4899 -0.6730 -0.1992  0.5645  3.4436
##
## Random effects:
## Groups Name Variance Std.Dev.
## trialID (Intercept) 0.002869 0.05356
## subject (Intercept) 0.026268 0.16208
## Residual 0.078085 0.27944
## Number of obs: 3056, groups: trialID, 108; subject, 56
##
## Fixed effects:
## Estimate Std. Error df t value Pr(>|t|)
## (Intercept) 0.28906 0.02284 60.60274 12.656 < 2e-16 ***
## condition1 0.07929 0.01533 100.99421 5.173 1.17e-06 ***
## condition2 0.02584 0.01771 101.12543 1.459 0.148
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) cndtn1
## condition1 0.000
## condition2 0.000 0.000
```

```
eyetracking.window.3.good.calib = eyetracking.window.3 %>%
  filter(mean_percent_in_roi >= 50)
```

```
# Add orthogonal contrasts to model
contrasts(eyetracking.window.3.good.calib$condition) <- cbind(c(-2/3, 1/3, 1/3), c(0, -1/2, 1/2))
```

```
model.time.window.3.good.calib <- lmer(prop.fixations.animal ~ condition + (1 | subject) + (1 | trialID),
  control = lmerControl(optimizer = "bobyqa",
    optCtrl = list(maxfun = 2e6)))
summary(model.time.window.3.good.calib)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: prop.fixations.animal ~ condition + (1 | subject) + (1 | trialID)
## Data: eyetracking.window.3.good.calib
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+06))
##
## REML criterion at convergence: 937.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.4355 -0.7687 -0.1473  0.7058  3.1387
##
```



```
## Random effects:
##   Groups   Name                Variance Std.Dev.
## trialID  (Intercept) 0.004803 0.0693
## subject  (Intercept) 0.021622 0.1470
## Residual                    0.086382 0.2939
## Number of obs: 1943, groups: trialID, 108; subject, 35
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   0.35476    0.02660   38.36333  13.335 5.55e-16 ***
## condition1    0.09785    0.02003  100.65106   4.884 3.92e-06 ***
## condition2    0.02462    0.02314  100.70311   1.064 0.29
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) cndtn1
## condition1 0.000
## condition2 0.000 0.000
```

```
eyetracking.window.3.great.calib = eyetracking.window.3 %>%
  filter(mean_percent_in_roi >= 75)
```

```
# Add orthogonal contrasts to model
contrasts(eyetracking.window.3.great.calib$condition) <- cbind(c(-2/3, 1/3, 1/3), c(0, -1/2, 1/2))
```

```
model.time.window.3.great.calib <- lmer(prop.fixations.animal ~ condition + (1 | subject) + (1 | trialID),
  control = lmerControl(optimizer = "bobyqa",
    optCtrl = list(maxfun = 2e6)))
summary(model.time.window.3.great.calib)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: prop.fixations.animal ~ condition + (1 | subject) + (1 | trialID)
## Data: eyetracking.window.3.great.calib
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+06))
##
## REML criterion at convergence: 601
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.5514 -0.7828 -0.1282  0.7340  3.0258
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
## trialID  (Intercept) 0.009071 0.09524
## subject  (Intercept) 0.026568 0.16300
## Residual                    0.089664 0.29944
## Number of obs: 1080, groups: trialID, 108; subject, 19
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   0.38954    0.03960   19.94470   9.837 4.28e-09 ***
```

```

## condition1    0.10716    0.02745 105.45809    3.904 0.000167 ***
## condition2    0.04780    0.03170 105.45809    1.508 0.134492
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) cndtn1
## condition1 0.000
## condition2 0.000 0.000

```