```
> ########################################
> ####### OST-CONST-04_Analyses1 ########
> ########################################
>
> # Load packages
> library(ggplot2)
> library(magrittr)
> library("ggpubr")
> library(Matrix)
> library(languageR)
> library(dplyr)
> library(lsr)
> library(tidyr)
> library(reshape)
> library(lme4)
> library(emmeans)
> library(lmerTest)
> library(lsmeans)
> library(rcompanion)
>
>
> # How many participants are in each condition? (get overview using e.g. table())
>
>
> ################## Plots Master_ost:
> ################## mood, ios, excluded, ignored, NT1-4, perception of other player/group
> ### mood
> # Test
> # t-test if only comparing two groups (e.g. exclusion vs. inclusion)
> stats::t.test(master_ost$mood[master_ost$cyberball == 1],
+               master_ost$mood[master_ost$cyberball == 2])

        Welch Two Sample t-test
```

```
data:  master_ost$mood[master_ost$cyberball == 1] and master_ost$mood[master_ost$cyberball == 2]
t = 9.3133, df = 128.57, p-value = 4.411e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 2.497099 3.844319
sample estimates:
mean of x mean of y
 6.358209  3.187500

> # Anova to compare all four groups (e.g. main effect cyberball, main effect focus, interaction Cyberbal*focus)
> summary(stats::aov(mood ~ self_other_condition * ostracism_condition, data = master_ost))
                                       Df Sum Sq Mean Sq F value   Pr(>F)
self_other_condition                    1   17.9    17.9   4.759    0.031 *
ostracism_condition                     1  320.1   320.1  85.036 8.08e-16 ***
self_other_condition:ostracism_condition   1    2.2     2.2   0.596    0.441
Residuals                             127  478.0     3.8
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Effect size (Cohen's d): z.B.  cohensD(mood ~ ostracism_condition, data = master_ost)
> lsr::cohensD(mood ~ ostracism_condition, data = master_ost)
[1] 1.628279
>
> # Display mean & sd of groups (this function is very helpful for documenting results afterwards)
> dplyr::group_by(master_ost, cyberball, focus) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(mood, na.rm = TRUE),
+                    sd = stats::sd(mood, na.rm = TRUE))
`summarise()` regrouping output by 'cyberball' (override with `.groups` argument)
# A tibble: 4 x 5
# Groups:   cyberball [2]
  cyberball focus count  mean    sd
```

```
      <int> <int> <int> <dbl> <dbl>
1         1     1    35  6.49  1.72
2         1     2    32  6.22  2.17
3         2     1    29  3.62  2.21
4         2     2    35  2.83  1.67
>
> # Plot mood
> # Important info:
> # This plot cannot deal with NAs (need to be removed before, here not necessary)
>
> # Grouping variables (e.g. cyberball & focus) should not be numeric but factor
> # Is focus a factor variable?
> utils::str(master_ost$focus)
 int [1:131] 2 1 2 1 1 1 2 1 1 2 ...
> # Is self_other_condition a factor variable?
> utils::str(master_ost$self_other_condition)
 chr [1:131] "green" "yellow" "green" "yellow" "yellow" "yellow" "green" "yellow" "yellow" "green" "green"
"green" "green" "yellow" "yellow" "green" ...
> # Try to exchange focus with self_other_condition and see how the plot changes
> # You can also change self_other_condition to a factor variable and see how the plot changes
>
> Sum_mood <- rcompanion::groupwiseMean(mood ~ ostracism_condition + self_other_condition,
+                                       data   = master_ost,
+                                       conf   = 0.95,
+                                       digits = 3,
+                                       traditional = FALSE,
+                                       percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_mood, ggplot2::aes(x = ostracism_condition,
+                                        y = Mean,
+                                        color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
```

```
+                         width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("Mood value (1 = bad; 9 = good)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> ### ios1 (Relationship towards person group Green OR person group Yellow)
> # Test
> stats::t.test(master_ost$ios_outcome[master_ost$ios_condition == "group_yellow_own"],
+               master_ost$ios_outcome[master_ost$ios_condition == "group_green_other"])

        Welch Two Sample t-test

data:  master_ost$ios_outcome[master_ost$ios_condition == "group_yellow_own"] and
master_ost$ios_outcome[master_ost$ios_condition == "group_green_other"]
t = -3.1073, df = 128.16, p-value = 0.002326
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.5222592 -0.3378154
sample estimates:
mean of x mean of y
 3.562500  4.492537

>
> # ANOVA
> summary(stats::aov(ios_outcome ~ self_other_condition * ostracism_condition, data = master_ost))
                                        Df Sum Sq Mean Sq F value   Pr(>F)
self_other_condition                     1   28.3  28.313  10.781 0.001325 **
ostracism_condition                      1   30.4  30.361  11.560 0.000901 ***
self_other_condition:ostracism_condition  1   16.6  16.600   6.321 0.013183 *
Residuals                              127  333.5   2.626
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, ios_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(ios_outcome, na.rm = TRUE),
+                    sd = stats::sd(ios_outcome, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  ios_condition     count  mean    sd
  <chr>             <int> <dbl> <dbl>
1 group_green_other    67  4.49  1.82
2 group_yellow_own     64  3.56  1.60
>
> # Plot ios1
> Sum_ios_outcome ← rcompanion::groupwiseMean(ios_outcome ~ ostracism_condition + self_other_condition,
+                                             data   = master_ost,
+                                             conf   = 0.95,
+                                             digits = 3,
+                                             traditional = FALSE,
+                                             percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_ios_outcome, ggplot2::aes(x = ostracism_condition,
+                                               y = Mean,
+                                               color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("IOS_outcome value (1 = close; 7 = distant)") +
```

```
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ### ios2 (Relationship self and own group)
> # Test
> stats::t.test(master_ost$ios_group_ident[master_ost$ios_condition == "group_yellow_own"],
+               master_ost$ios_group_ident[master_ost$ios_condition == "group_green_other"])

        Welch Two Sample t-test

data:  master_ost$ios_group_ident[master_ost$ios_condition == "group_yellow_own"] and
master_ost$ios_group_ident[master_ost$ios_condition == "group_green_other"]
t = 1.9321, df = 125.93, p-value = 0.0556
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01486357  1.23921059
sample estimates:
mean of x mean of y
 4.671875  4.059701

>
> # ANOVA
> summary(stats::aov(ios_group_ident ~ self_other_condition * ostracism_condition, data = master_ost))
                                       Df Sum Sq Mean Sq F value Pr(>F)
self_other_condition                    1   12.3  12.267   3.782 0.0540 .
ostracism_condition                     1    0.4   0.392   0.121 0.7286
self_other_condition:ostracism_condition   1   15.6  15.567   4.800 0.0303 *
Residuals                             127  411.9   3.243
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Display mean & sd of groups
```

```
> dplyr::group_by(master_ost, ios_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(ios_group_ident, na.rm = TRUE),
+                    sd = stats::sd(ios_group_ident, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  ios_condition     count  mean    sd
  <chr>             <int> <dbl> <dbl>
1 group_green_other    67  4.06  1.99
2 group_yellow_own     64  4.67  1.62
>
> # Plot ios2
> Sum_ios_group_ident <- rcompanion::groupwiseMean(ios_group_ident ~ ostracism_condition + self_other_condition,
+                                                  data   = master_ost,
+                                                  conf   = 0.95,
+                                                  digits = 3,
+                                                  traditional = FALSE,
+                                                  percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_ios_group_ident, ggplot2::aes(x = ostracism_condition,
+                                                   y = Mean,
+                                                   color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 7) +
+   ggplot2::ylab("IOS_group_ident value (1 = close; 7 = distant)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
```

```
> ### How much do participants identify with their own group (positive experience and like being part)
> ## Positive experience
> # Test
> stats::t.test(master_ost$group_ident_experience[master_ost$ios_condition == "group_yellow_own"],
+               master_ost$group_ident_experience[master_ost$ios_condition == "group_green_other"])

        Welch Two Sample t-test

data:  master_ost$group_ident_experience[master_ost$ios_condition == "group_yellow_own"] and
master_ost$group_ident_experience[master_ost$ios_condition == "group_green_other"]
t = 0.32939, df = 127.55, p-value = 0.7424
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.4741021  0.6634677
sample estimates:
mean of x mean of y
 5.781250  5.686567

>
> # ANOVA
> summary(stats::aov(group_ident_experience ~ self_other_condition * ostracism_condition, data = master_ost))
                                       Df Sum Sq Mean Sq F value Pr(>F)
self_other_condition                    1    0.3   0.293   0.109 0.7413
ostracism_condition                     1    8.5   8.547   3.189 0.0765 .
self_other_condition:ostracism_condition   1    2.5   2.452   0.915 0.3406
Residuals                             127  340.4   2.680
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, ios_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(group_ident_experience, na.rm = TRUE),
```

```
+                    sd = stats::sd(group_ident_experience, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  ios_condition     count  mean    sd
  <chr>             <int> <dbl> <dbl>
1 group_green_other    67  5.69  1.77
2 group_yellow_own     64  5.78  1.52
>
> # Plot positive experience
> Sum_group_ident_experience <- rcompanion::groupwiseMean(group_ident_experience ~ ostracism_condition +
self_other_condition,
+                                                        data   = master_ost,
+                                                        conf   = 0.95,
+                                                        digits = 3,
+                                                        traditional = FALSE,
+                                                        percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_group_ident_experience, ggplot2::aes(x = ostracism_condition,
+                                                          y = Mean,
+                                                          color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("Positive_experience value (1 = disagree; 9 = agree)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ## Like being a member of Group Yellow
> # Test
```

```
> stats::t.test(master_ost$group_ident_like[master_ost$ios_condition == "group_yellow_own"],
+               master_ost$group_ident_like[master_ost$ios_condition == "group_green_other"])

        Welch Two Sample t-test

data:  master_ost$group_ident_like[master_ost$ios_condition == "group_yellow_own"] and
master_ost$group_ident_like[master_ost$ios_condition == "group_green_other"]
t = 0.12489, df = 124.83, p-value = 0.9008
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.5505563  0.6247168
sample estimates:
mean of x mean of y
 5.828125  5.791045

>
> # ANOVA
> summary(stats::aov(group_ident_like ~ self_other_condition * ostracism_condition, data = master_ost))
                                        Df Sum Sq Mean Sq F value Pr(>F)
self_other_condition                     1    0.0   0.045   0.015  0.901
ostracism_condition                      1    1.4   1.379   0.473  0.493
self_other_condition:ostracism_condition   1    4.8   4.796   1.646  0.202
Residuals                              127  370.0   2.913
>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, ios_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(group_ident_like, na.rm = TRUE),
+                    sd = stats::sd(group_ident_like, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  ios_condition     count  mean    sd
  <chr>             <int> <dbl> <dbl>
```

```
1 group_green_other    67  5.79  1.89
2 group_yellow_own     64  5.83  1.50
>
> # Plot liking
> Sum_group_ident_like ← rcompanion::groupwiseMean(group_ident_like ~ ostracism_condition +
self_other_condition,
+                                                 data   = master_ost,
+                                                 conf   = 0.95,
+                                                 digits = 3,
+                                                 traditional = FALSE,
+                                                 percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_group_ident_like, ggplot2::aes(x = ostracism_condition,
+                                                    y = Mean,
+                                                    color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("Like_beeing_member value (1 = disagree; 9 = agree)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ### Feeling excluded
> # Test
> stats::t.test(master_ost$excluded[master_ost$cyberball == 1],
+               master_ost$excluded[master_ost$cyberball == 2])

        Welch Two Sample t-test
```

```
data:  master_ost$excluded[master_ost$cyberball == 1] and master_ost$excluded[master_ost$cyberball == 2]
t = -17.057, df = 128.63, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -5.718184 -4.529484
sample estimates:
mean of x mean of y
 2.641791  7.765625

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(excluded, na.rm = TRUE),
+                    sd = stats::sd(excluded, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  2.64  1.71
2         2    64  7.77  1.73
>
> # Plot excluded
> Sum_excluded ← rcompanion::groupwiseMean(excluded ~ cyberball,
+                                          data   = master_ost,
+                                          conf   = 0.95,
+                                          digits = 3,
+                                          traditional = FALSE,
+                                          percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_excluded, ggplot2::aes(x = cyberball,
+                                            y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
```

```
+                                       ymax=Percentile.upper),
+                           width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("Exclusion value (1 = disagree; 9 = agree)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ### Feeling ignored
> # Test
> stats::t.test(master_ost$ignored[master_ost$cyberball == 1],
+               master_ost$ignored[master_ost$cyberball == 2])

        Welch Two Sample t-test

data:  master_ost$ignored[master_ost$cyberball == 1] and master_ost$ignored[master_ost$cyberball == 2]
t = -17.234, df = 123.5, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -5.672803 -4.503969
sample estimates:
mean of x mean of y
 2.552239  7.640625

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(ignored, na.rm = TRUE),
+                    sd = stats::sd(ignored, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
```

```
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  2.55  1.54
2         2    64  7.64  1.82
>
> # Plot excluded
> Sum_ignored ← rcompanion::groupwiseMean(ignored ~ cyberball,
+                                         data   = master_ost,
+                                         conf   = 0.95,
+                                         digits = 3,
+                                         traditional = FALSE,
+                                         percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_ignored, ggplot2::aes(x = cyberball,
+                                           y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("Ignoration value (1 = disagree; 9 = agree)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> ### Received tosses
> # Test
> stats::t.test(master_ost$received_tosses[master_ost$cyberball == 1],
+               master_ost$received_tosses[master_ost$cyberball == 2])

        Welch Two Sample t-test
```

```
data:  master_ost$received_tosses[master_ost$cyberball == 1] and master_ost$received_tosses[master_ost$cyberball
== 2]
t = 23.044, df = 87.003, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 22.90333 27.22727
sample estimates:
mean of x mean of y
  31.9403    6.8750

>
> # ANOVA
> summary(stats::aov(received_tosses ~ self_other_condition * ostracism_condition, data = master_ost))
                                        Df Sum Sq Mean Sq F value Pr(>F)
self_other_condition                     1    144     144   3.552 0.0618 .
ostracism_condition                      1  20426   20426 505.528 <2e-16 ***
self_other_condition:ostracism_condition   1     31      31   0.771 0.3816
Residuals                               127   5131      40
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(received_tosses, na.rm = TRUE),
+                    sd = stats::sd(received_tosses, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67 31.9   8.25
2         2    64  6.88  3.27
>
```

```
> # Plot received tosses
> Sum_received_tosses ← rcompanion::groupwiseMean(received_tosses ~ cyberball,
+                                                 data   = master_ost,
+                                                 conf   = 0.95,
+                                                 digits = 3,
+                                                 traditional = FALSE,
+                                                 percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_received_tosses, ggplot2::aes(x = cyberball,
+                                                   y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 60) +
+   ggplot2::ylab("Received_tosses value (1 = few; 60 = many)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> ### Need threat 1-4
> ## NT1
> # Test
> stats::t.test(master_ost$nt1_accepted[master_ost$cyberball == 1],
+               master_ost$nt1_accepted[master_ost$cyberball == 2])

        Welch Two Sample t-test

data:  master_ost$nt1_accepted[master_ost$cyberball == 1] and master_ost$nt1_accepted[master_ost$cyberball == 2]
t = 18.405, df = 126.41, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.104837 5.093857
```

```
sample estimates:
mean of x mean of y
 6.880597  2.281250

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(nt1_accepted, na.rm = TRUE),
+                    sd = stats::sd(nt1_accepted, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  6.88  1.56
2         2    64  2.28  1.29
>
> # Plot NT1
> Sum_nt1_accepted <- rcompanion::groupwiseMean(nt1_accepted ~ cyberball,
+                                               data   = master_ost,
+                                               conf   = 0.95,
+                                               digits = 3,
+                                               traditional = FALSE,
+                                               percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_nt1_accepted, ggplot2::aes(x = cyberball,
+                                                y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
```

```
+   ggplot2::ylab("NT1 value (1 = low; 9 = high)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ## NT2
> # Test
> stats::t.test(master_ost$nt2_valued[master_ost$cyberball == 1],
+               master_ost$nt2_valued[master_ost$cyberball == 2])

        Welch Two Sample t-test

data:  master_ost$nt2_valued[master_ost$cyberball == 1] and master_ost$nt2_valued[master_ost$cyberball == 2]
t = 13.493, df = 126.83, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 3.285207 4.414420
sample estimates:
mean of x mean of y
 6.537313  2.687500

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(nt2_valued, na.rm = TRUE),
+                    sd = stats::sd(nt2_valued, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  6.54  1.56
2         2    64  2.69  1.70
```

```
>
> # Plot NT2
> Sum_nt2_valued <- rcompanion::groupwiseMean(nt2_valued ~ cyberball,
+                                             data   = master_ost,
+                                             conf   = 0.95,
+                                             digits = 3,
+                                             traditional = FALSE,
+                                             percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_nt2_valued, ggplot2::aes(x = cyberball,
+                                              y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("NT2 value (1 = low; 9 = high)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ## NT3
> # Test
> stats::t.test(master_ost$nt3_powerfull[master_ost$cyberball == 1],
+               master_ost$nt3_powerfull[master_ost$cyberball == 2])

        Welch Two Sample t-test

data:  master_ost$nt3_powerfull[master_ost$cyberball == 1] and master_ost$nt3_powerfull[master_ost$cyberball ==
2]
t = 11.109, df = 128.4, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
```

```
95 percent confidence interval:
 2.695492 3.863744
sample estimates:
mean of x mean of y
 5.701493  2.421875

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(nt3_powerfull, na.rm = TRUE),
+                    sd = stats::sd(nt3_powerfull, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  5.70  1.67
2         2    64  2.42  1.71
>
> # Plot NT3
> Sum_nt3_powerfull <- rcompanion::groupwiseMean(nt3_powerfull ~ cyberball,
+                                                data   = master_ost,
+                                                conf   = 0.95,
+                                                digits = 3,
+                                                traditional = FALSE,
+                                                percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_nt3_powerfull, ggplot2::aes(x = cyberball,
+                                                 y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
```

```
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("NT3 value (1 = low; 9 = high)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> ## N4
> # Test
> stats::t.test(master_ost$nt4_recognized[master_ost$cyberball == 1],
+               master_ost$nt4_recognized[master_ost$cyberball == 2])

        Welch Two Sample t-test

data:  master_ost$nt4_recognized[master_ost$cyberball == 1] and master_ost$nt4_recognized[master_ost$cyberball ==
2]
t = 13.141, df = 123.46, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 3.415138 4.626373
sample estimates:
mean of x mean of y
 6.223881  2.203125

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(nt4_recognized, na.rm = TRUE),
+                    sd = stats::sd(nt4_recognized, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
```

```
1         1    67  6.22  1.97
2         2    64  2.20  1.51
>
> # Plot NT4
> Sum_nt4_recognized <- rcompanion::groupwiseMean(nt4_recognized ~ cyberball,
+                                                 data   = master_ost,
+                                                 conf   = 0.95,
+                                                 digits = 3,
+                                                 traditional = FALSE,
+                                                 percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_nt4_recognized, ggplot2::aes(x = cyberball,
+                                                  y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("NT4 value (1 = low; 9 = high)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
>
> ### How is Player 3 from Group Green perceived?
> ## Gender P3
> # Test
> stats::t.test(master_ost$gender_green[master_ost$cyberball == 1],
+               master_ost$gender_green[master_ost$cyberball == 2])

        Welch Two Sample t-test
```

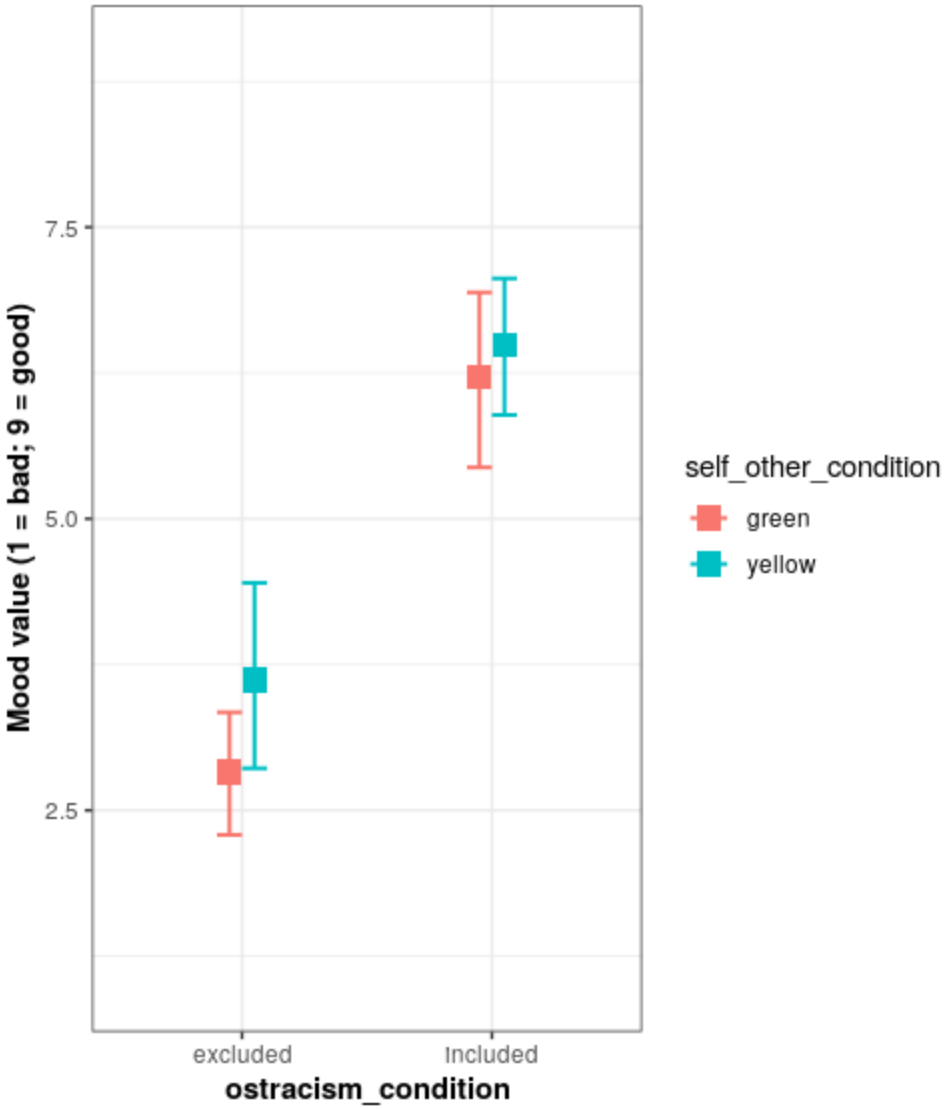
```
data:  master_ost$gender_green[master_ost$cyberball == 1] and master_ost$gender_green[master_ost$cyberball == 2]
t = -0.66663, df = 120.44, p-value = 0.5063
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.303671  0.150686
sample estimates:
mean of x mean of y
 1.298507  1.375000

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(gender_green, na.rm = TRUE),
+                    sd = stats::sd(gender_green, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  1.30 0.578
2         2    64  1.38 0.724
>
> # Plot gender P3
> Sum_gender_green <- rcompanion::groupwiseMean(gender_green ~ cyberball,
+                                               data   = master_ost,
+                                               conf   = 0.95,
+                                               digits = 3,
+                                               traditional = FALSE,
+                                               percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_gender_green, ggplot2::aes(x = cyberball,
+                                                y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
```

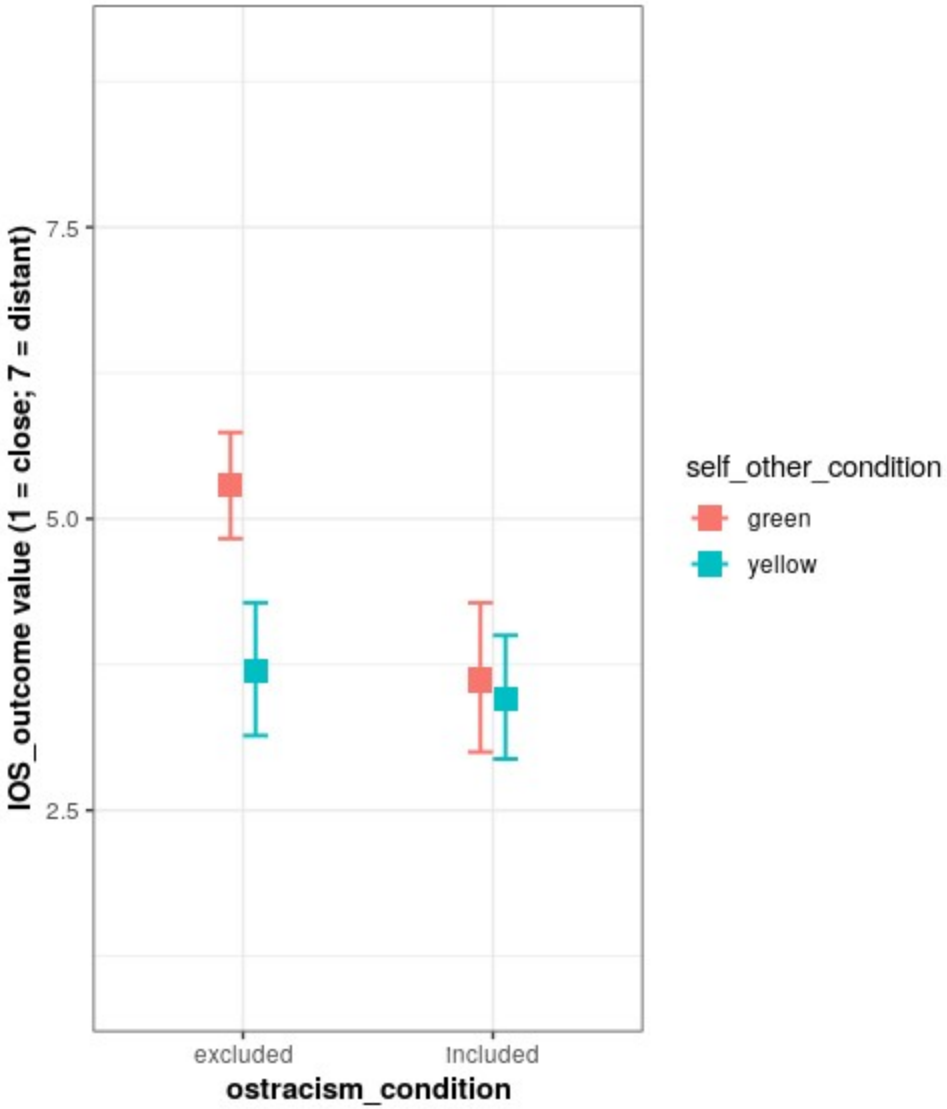
```
+                                       ymax=Percentile.upper),
+                           width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 3) +
+   ggplot2::ylab("Gender_green value (1 = male; 2 = female)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> ## Age P3
> # Test
> stats::t.test(master_ost$age_green[master_ost$cyberball == 1],
+               master_ost$age_green[master_ost$cyberball == 2])

        Welch Two Sample t-test

data:  master_ost$age_green[master_ost$cyberball == 1] and master_ost$age_green[master_ost$cyberball == 2]
t = 2.6526, df = 128.77, p-value = 0.008993
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1127095 0.7744174
sample estimates:
mean of x mean of y
 2.537313  2.093750

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(age_green, na.rm = TRUE),
+                    sd = stats::sd(age_green, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
```

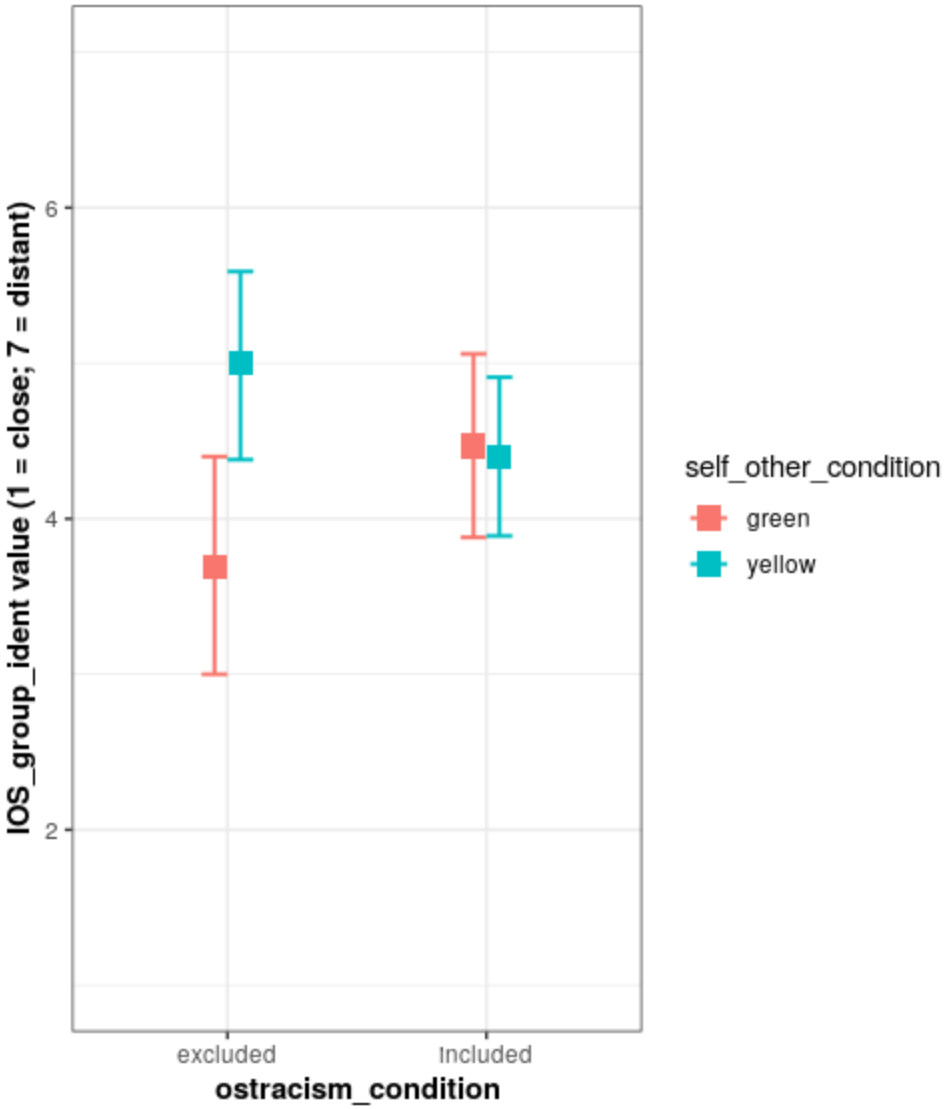
```
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  2.54 0.959
2         2    64  2.09 0.955
>
> # Plot age P3
> Sum_age_green <- rcompanion::groupwiseMean(age_green ~ cyberball,
+                                            data   = master_ost,
+                                            conf   = 0.95,
+                                            digits = 3,
+                                            traditional = FALSE,
+                                            percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_age_green, ggplot2::aes(x = cyberball,
+                                             y = Mean)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 7) +
+   ggplot2::ylab("Age_green value (1 = <18; 7 = >65)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
>
> ## P3 Similarity to others from Group Green
> # Test
> stats::t.test(master_ost$similar_green[master_ost$cyberball == 1],
+               master_ost$similar_green[master_ost$cyberball == 2])

        Welch Two Sample t-test
```

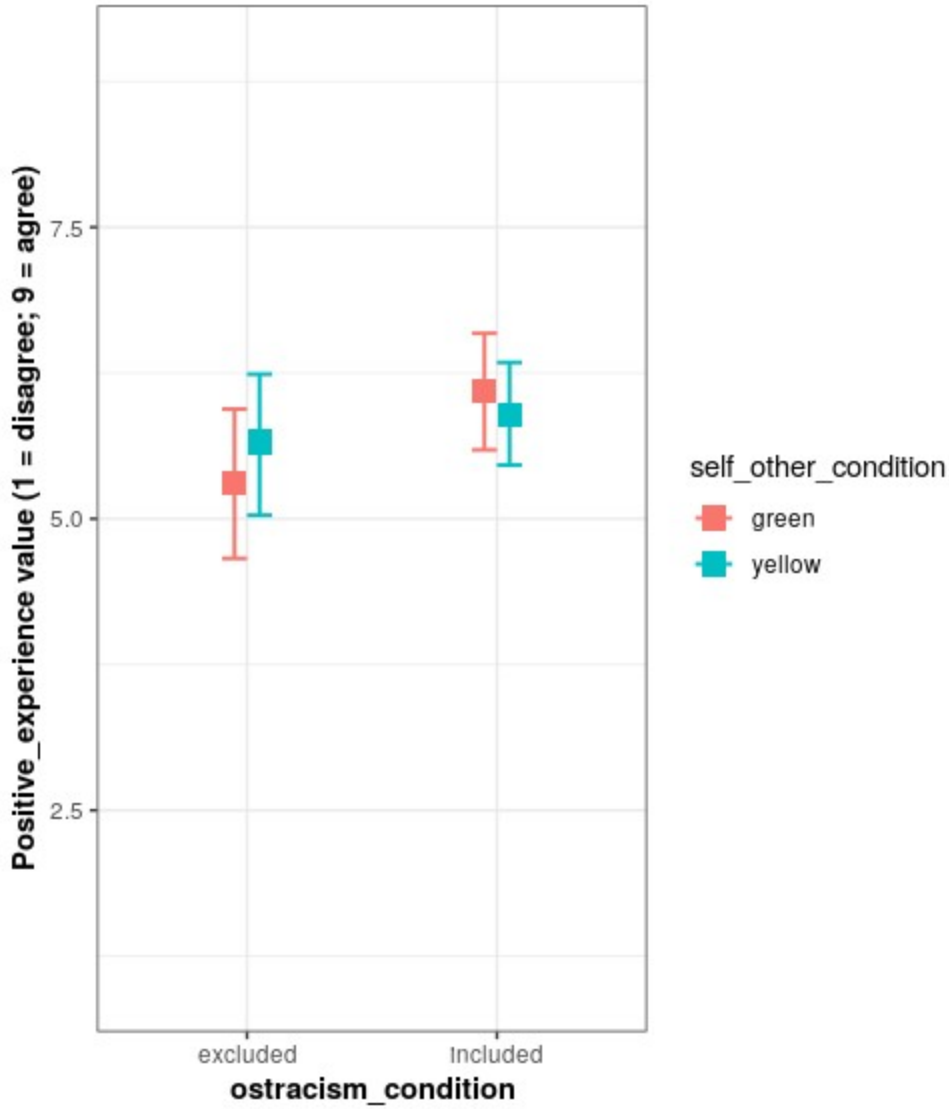
```
data:  master_ost$similar_green[master_ost$cyberball == 1] and master_ost$similar_green[master_ost$cyberball ==
2]
t = -0.040834, df = 128.58, p-value = 0.9675
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.634328  0.608675
sample estimates:
mean of x mean of y
 5.940299  5.953125

>
> # Display mean & sd of groups
> dplyr::group_by(master_ost, cyberball) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(similar_green, na.rm = TRUE),
+                    sd = stats::sd(similar_green, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  cyberball count  mean    sd
      <int> <int> <dbl> <dbl>
1         1    67  5.94  1.89
2         2    64  5.95  1.70
>
> # Plot similarity
> Sum_similar_green ← rcompanion::groupwiseMean(similar_green ~ cyberball,
+                                               data   = master_ost,
+                                               conf   = 0.95,
+                                               digits = 3,
+                                               traditional = FALSE,
+                                               percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_similar_green, ggplot2::aes(x = cyberball,
+                                                 y = Mean)) +
```

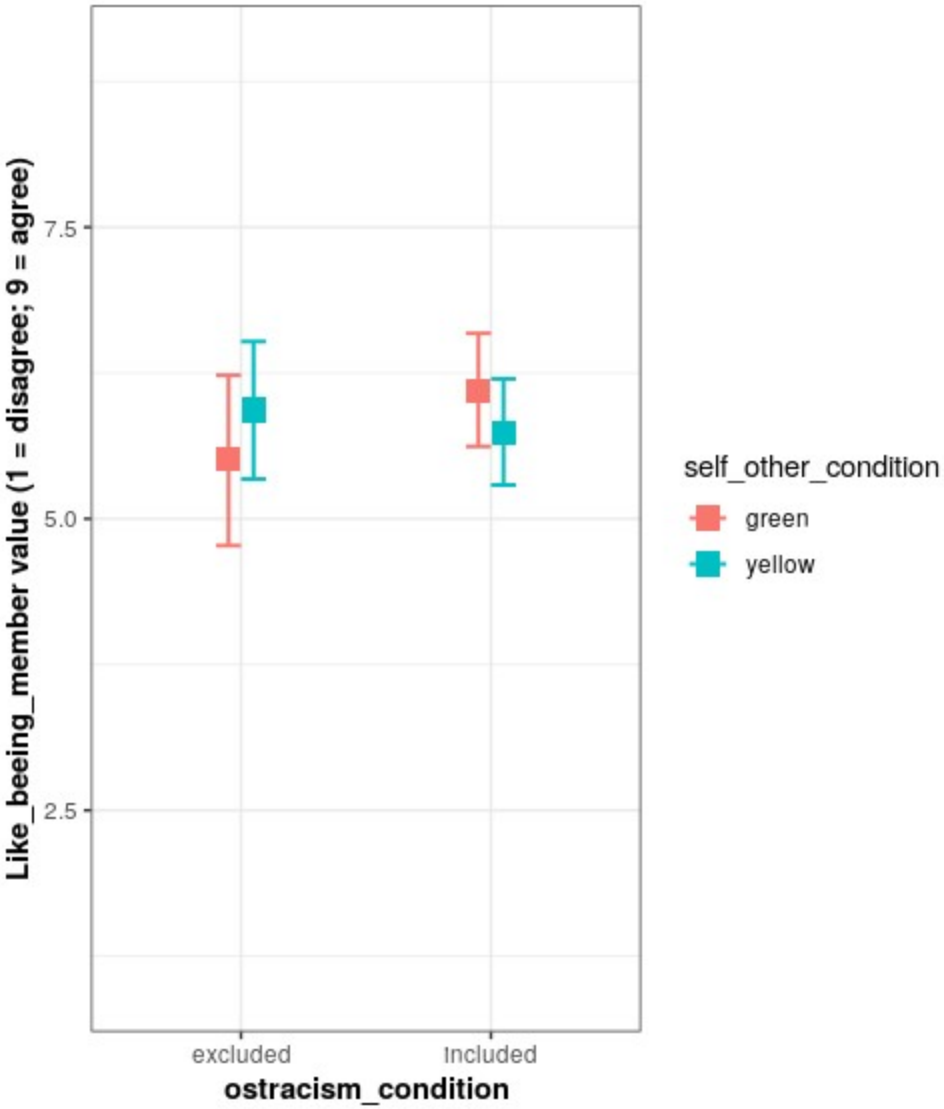
```
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 9) +
+   ggplot2::ylab("Similarity value (1 = low; 9 = high)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
```

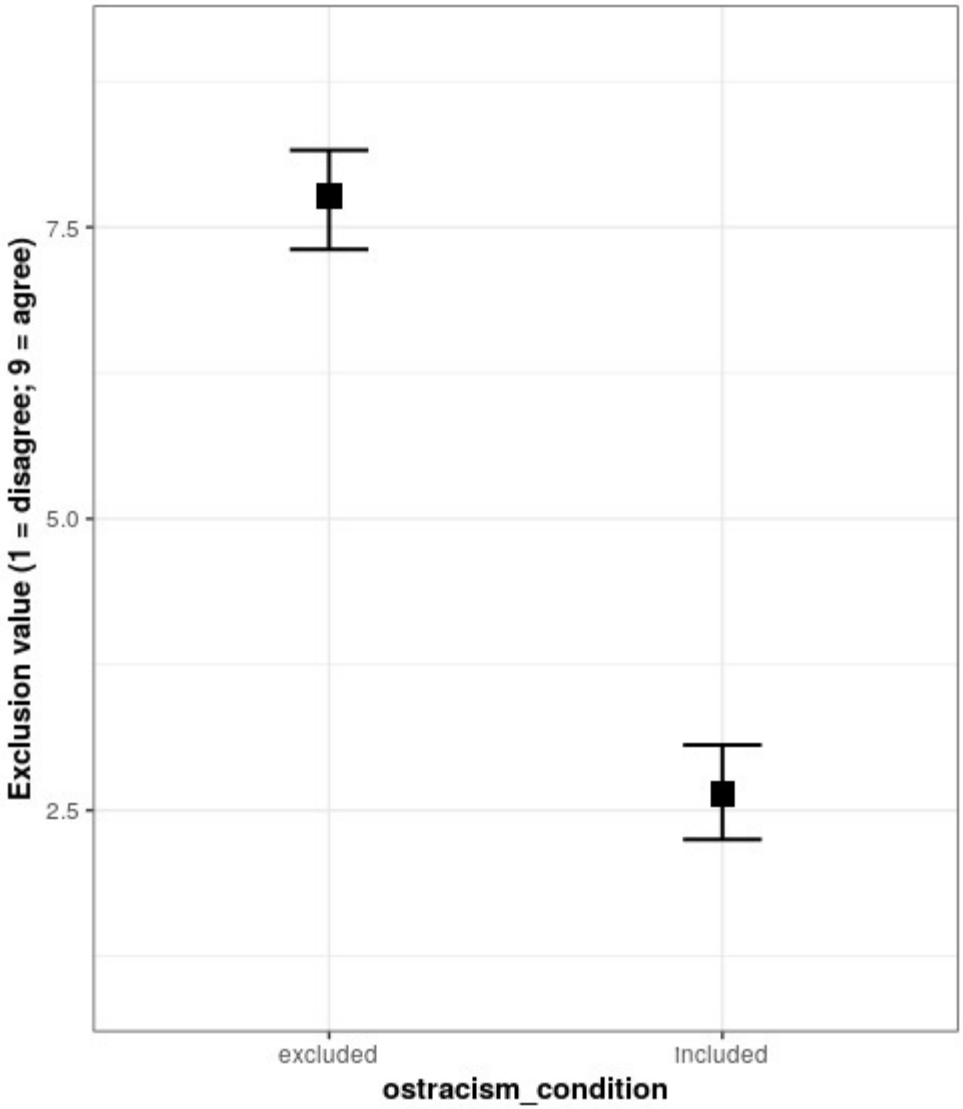

Mood value (1 = bad; 9 = good)
7.5
5.0
2.5
excluded
Included
ostracism_condition
self_other_condition
green
yellow

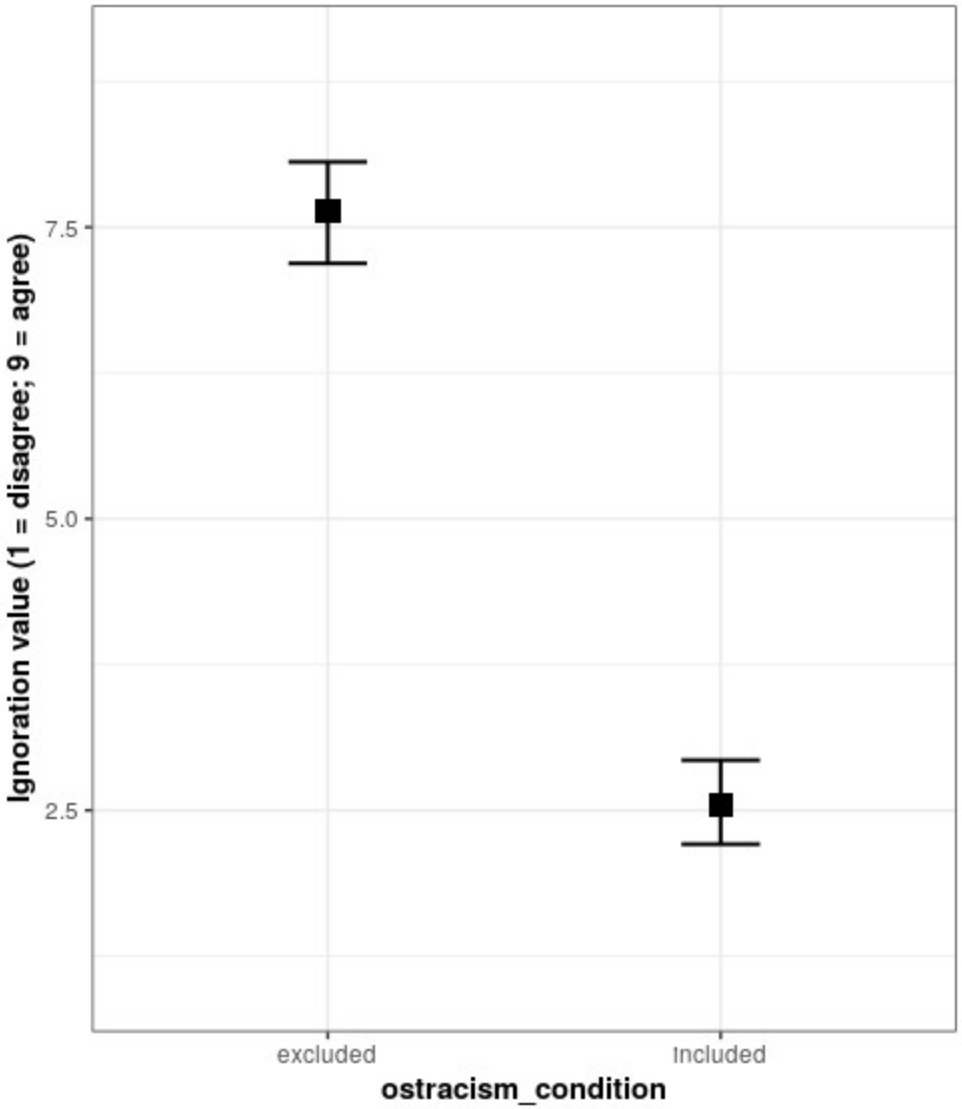

IOS_outcome value (1 = close; 7 = distant)
7.5
5.0
2.5
excluded
Included
ostracism_condition
self_other_condition
green
yellow

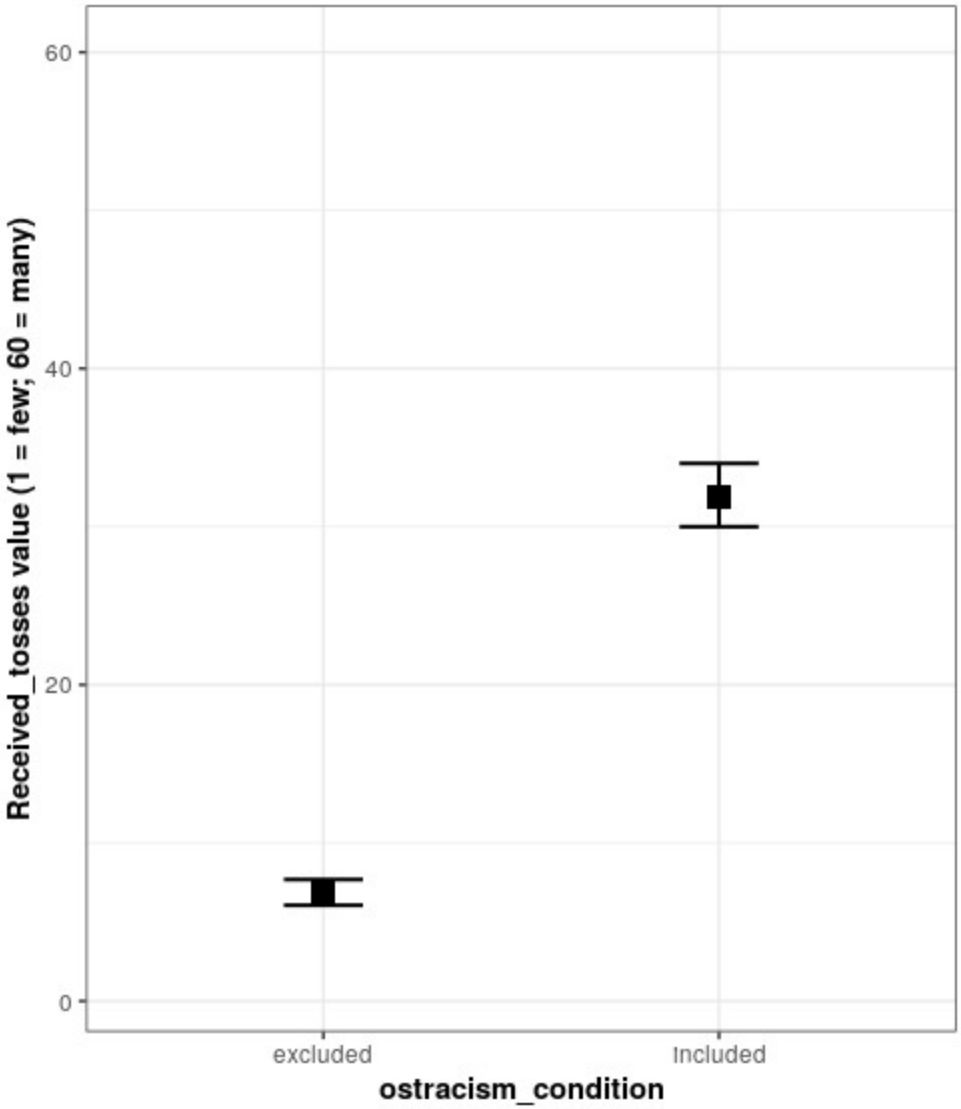

IOS_group_ident value (1 = close; 7 = distant)
6
4
2
excluded
Included
ostracism_condition
self_other_condition
green
yellow

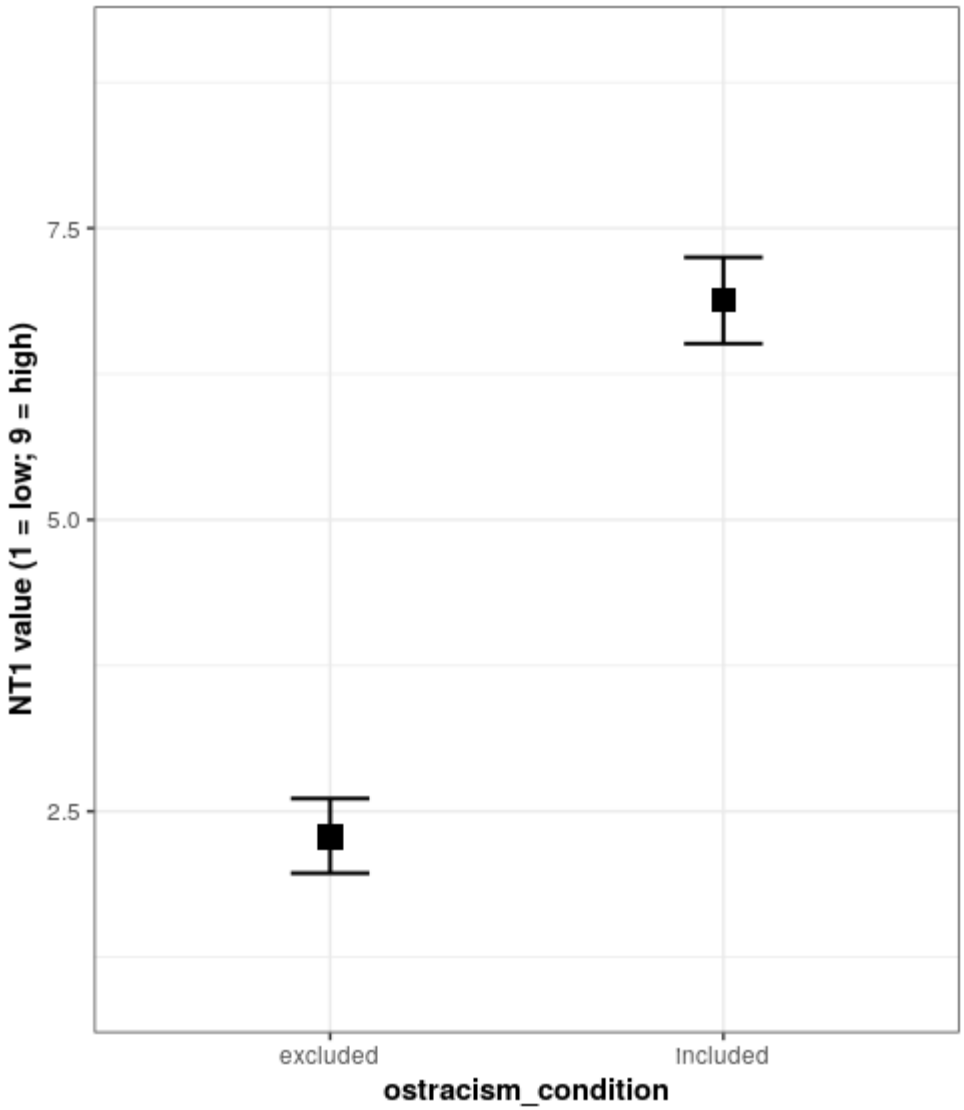

Positive_experience value (1 = disagree; 9 = agree)
7.5
5.0
2.5
excluded
Included
ostracism_condition
self_other_condition
green
yellow

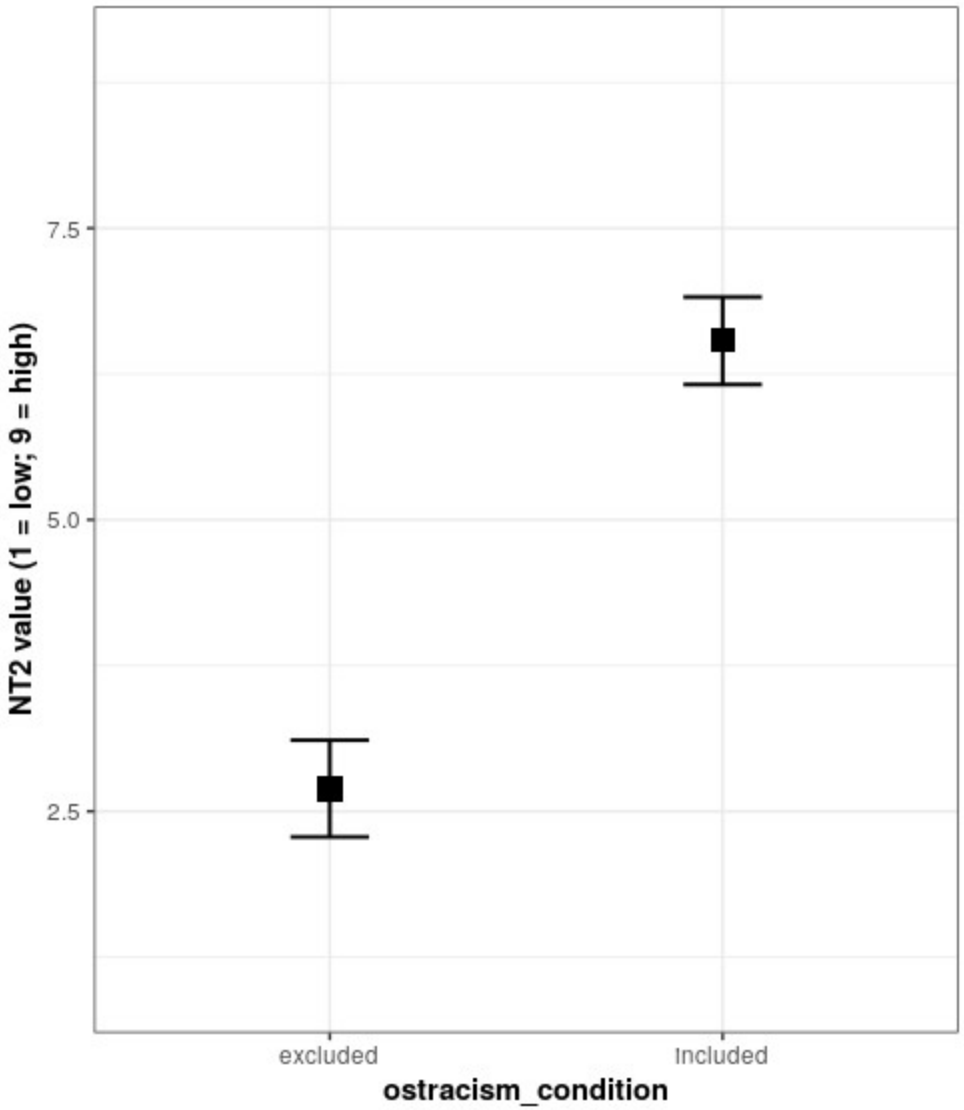

Like_beeing_member value (1 = disagree; 9 = agree)
7.5
5.0
2.5
excluded
Included
ostracism_condition
self_other_condition
green
yellow

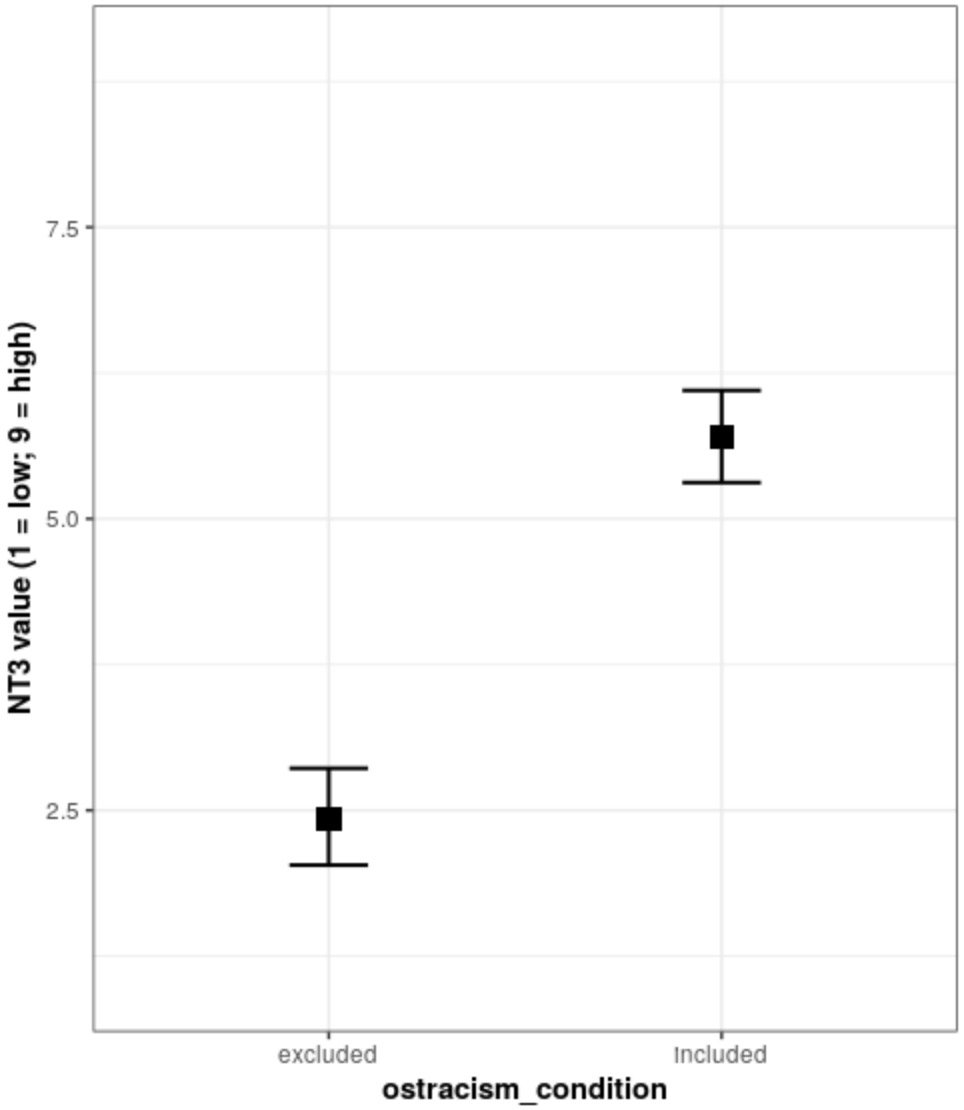

Exclusion value (1 = disagree; 9 = agree)
7.5
5.0
2.5
excluded
Included
ostracism_condition

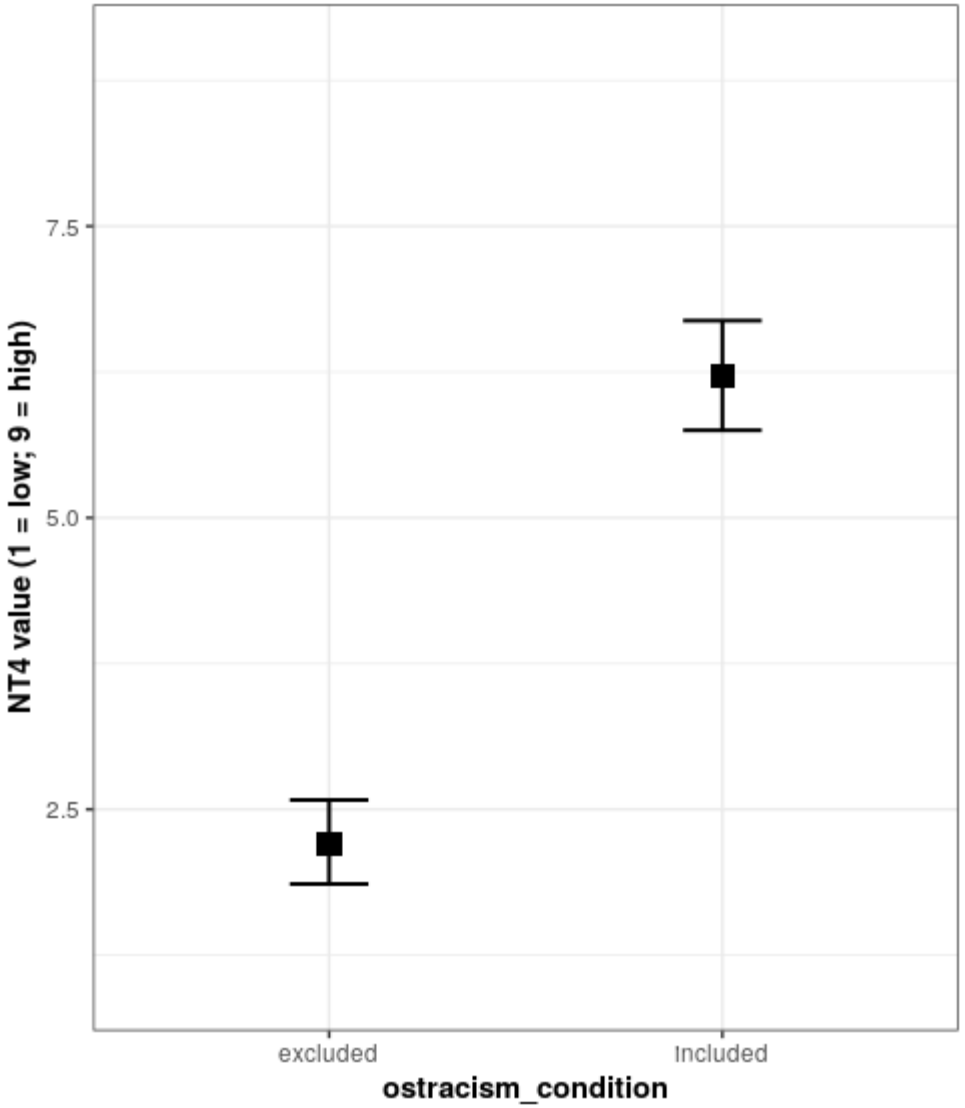

Ignoration value (1 = disagree; 9 = agree)
7.5
5.0
2.5
excluded
Included
ostracism_condition

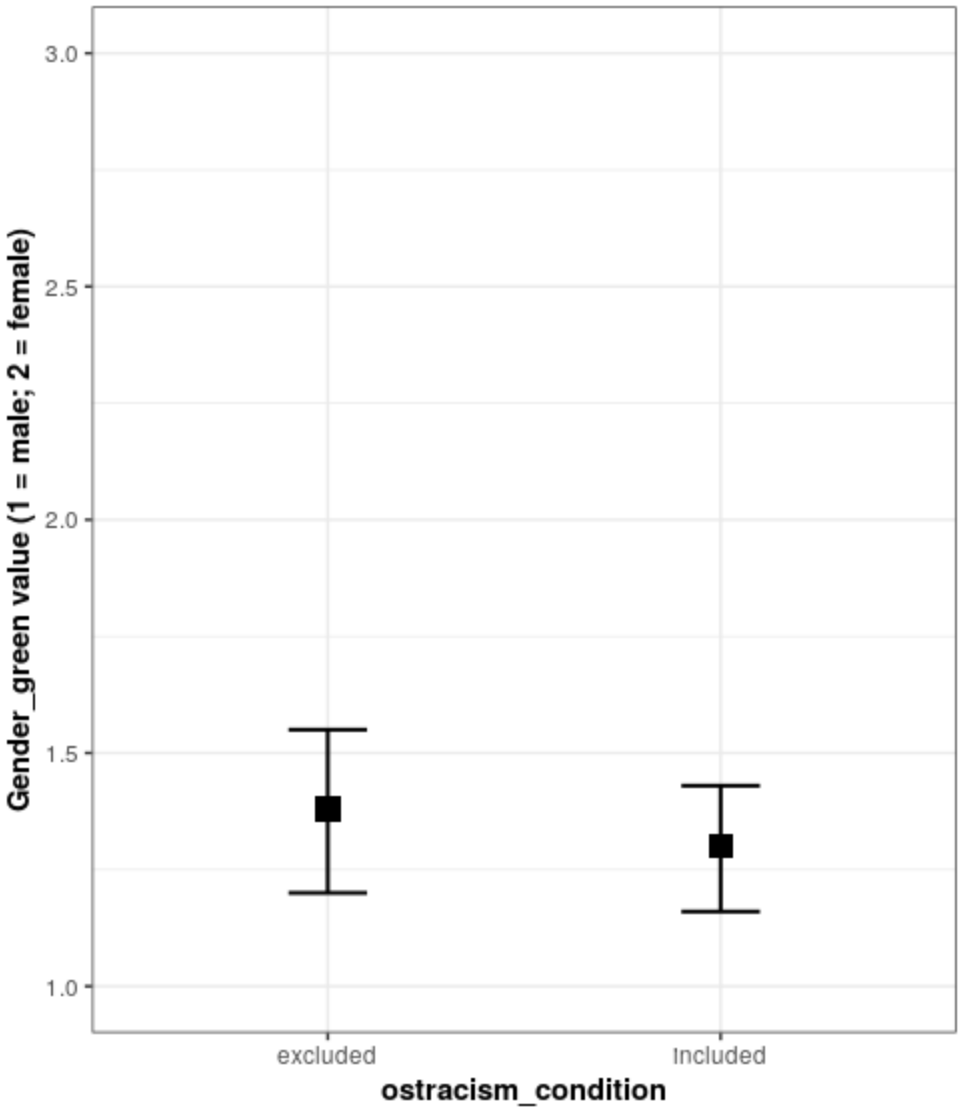

Received_tosses value (1 = few; 60 = many)
60
40
20
0
excluded
Included
ostracism_condition

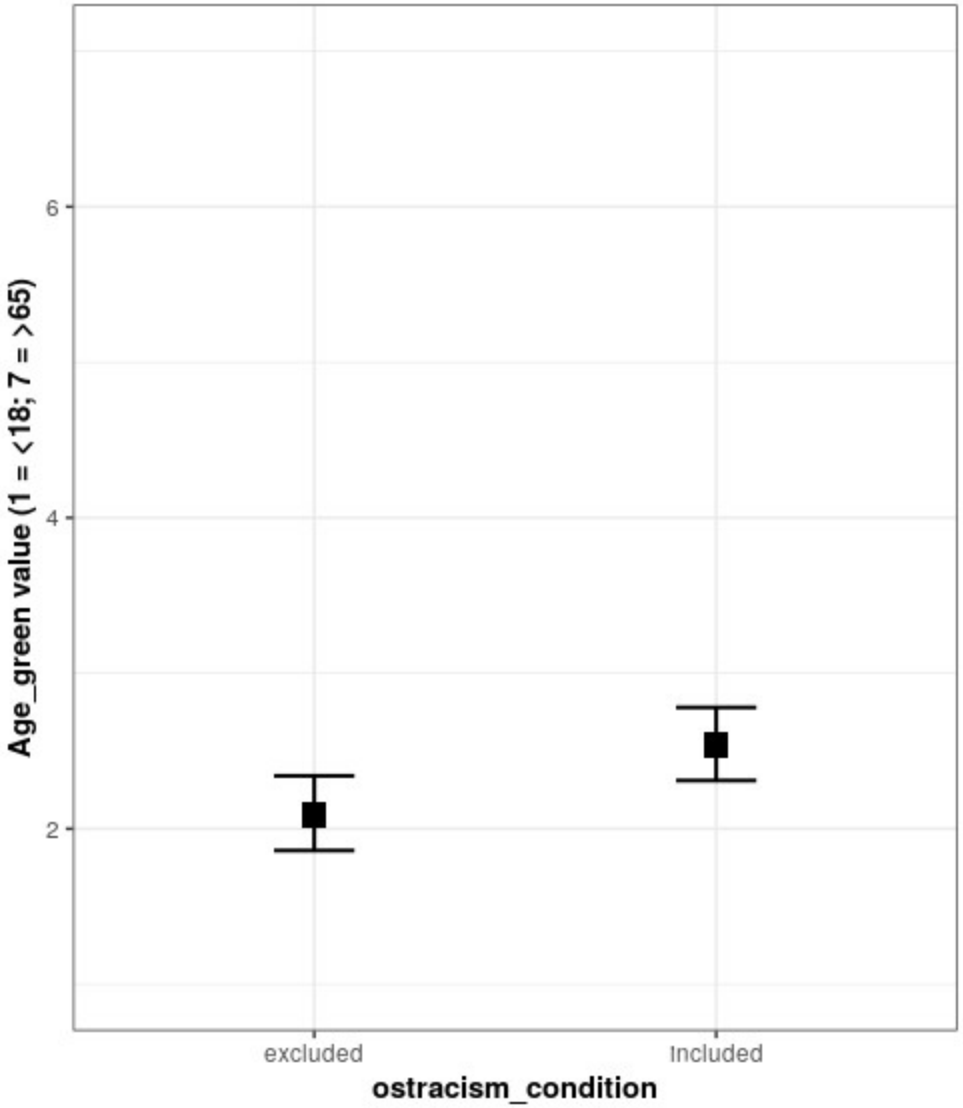

NT1 value (1 = low; 9 = high)
7.5
5.0
2.5
excluded
Included
ostracism_condition

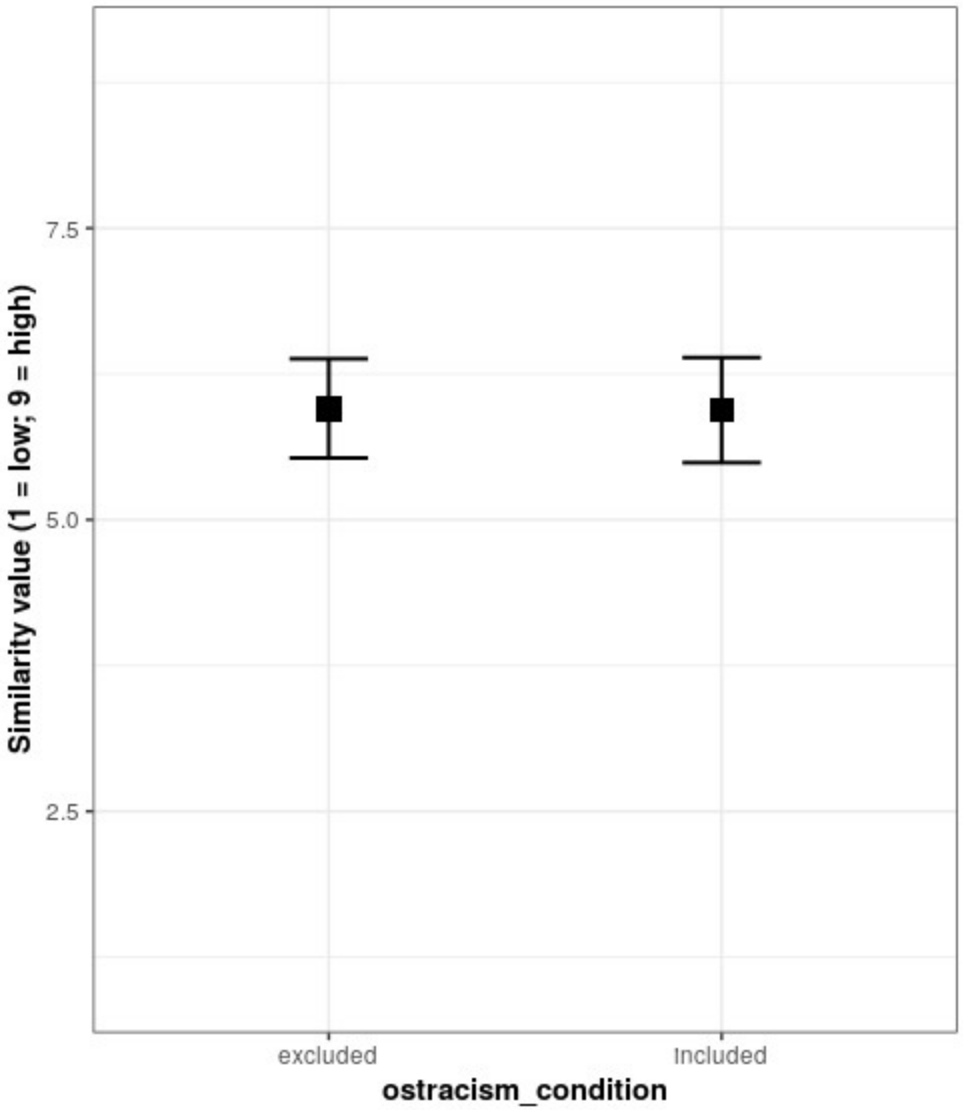

NT2 value (1 = low; 9 = high)
7.5
5.0
2.5
excluded
Included
ostracism_condition


7.5
5.0
2.5
NT3 value (1 = low; 9 = high)
excluded
Included
ostracism_condition


7.5
5.0
2.5
NT4 value (1 = low; 9 = high)
excluded
Included
ostracism_condition


3.0
2.5
2.0
1.5
1.0
Gender_green value (1 = male; 2 = female)
excluded
Included
ostracism_condition


6
4
2
Age_green value (1 = <18; 7 = >65)
excluded
Included
ostracism_condition


Similarity value (1 = low; 9 = high)
7.5
5.0
2.5
excluded
Included
ostracism_condition

```
> ########################################
> ####### OST-CONST-04_Analyses2 ########
> ########################################
>
> # Load packages
> library(ggplot2)
> library(magrittr)
> library("ggpubr")
> library(Matrix)
> library(languageR)
> library(dplyr)
> library(lsr)
> library(tidyr)
> library(reshape)
> library(lme4)
> library(emmeans)
> library(lmerTest)
> library(lsmeans)
> library(rcompanion)
> library(car)
Lade nötiges Paket: carData

Attache Paket: ‘car’

The following object is masked from ‘package:dplyr’:

    recode

> library(rstatix)

Attache Paket: ‘rstatix’

The following object is masked from ‘package:stats’:
```

```
    filter

>
>
> #### Mixed Models Analysis
> # Conditions need to be centered to -0.5 and 0.5 (instead of 1 and 2)
> long_data_clean %<>%
+   dplyr::mutate(ostracism_condition_c = dplyr::if_else(condition = ostracism_condition == "included",
+                                                        true = -0.5,
+                                                        false = 0.5)) %>%
+   dplyr::mutate(self_other_condition_c = dplyr::if_else(condition = self_other_condition == "yellow",
+                                                         true = -0.5,
+                                                         false = 0.5))
>
> # We only look at negative outcomes (as stated in our preregistration)
> long_neg <-
+   long_data_clean %>%
+   dplyr::filter(outcome %in% c(1:5))
>
> # Check if the centering is done correclty
> table(long_neg$ostracism_condition, long_neg$ostracism_condition_c)

           -0.5 0.5
  excluded    0 285
  included  315   0
> table(long_neg$self_other_condition, long_neg$self_other_condition_c)

         -0.5 0.5
  green     0 315
  yellow  285   0
>
> # How many people are in the study?
> length(unique(long_neg$id))
```

```
[1] 130
>
> # Stability analyses
> # First overall overview
> dplyr::group_by(long_neg, ostracism_condition, self_other_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(stability_value, na.rm = TRUE),
+                    sd = stats::sd(stability_value, na.rm = TRUE))
`summarise()` regrouping output by 'ostracism_condition' (override with `.groups` argument)
# A tibble: 4 x 5
# Groups:   ostracism_condition [2]
  ostracism_condition self_other_condition count  mean    sd
  <chr>              <chr>                <int> <dbl> <dbl>
1 excluded           green                  159  3.26 1.12
2 excluded           yellow                 126  3.67 0.986
3 included           green                  156  3.51 1.01
4 included           yellow                 159  3.64 0.895
>
> # Plot stability
> Sum_stability <- rcompanion::groupwiseMean(stability_value ~ ostracism_condition + self_other_condition,
+                                            data   = long_neg,
+                                            conf   = 0.95,
+                                            digits = 3,
+                                            traditional = FALSE,
+                                            percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_stability, ggplot2::aes(x = ostracism_condition,
+                                             y = Mean,
+                                             color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
```

```
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 5) +
+   ggplot2::ylab("Mean stability value (1 = stable; 5 = variable)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> # Is self_other_condition a good predictor for Stability?
> fit_stab_1 <- lme4::lmer (stability_value ~ 1 + (1 | id) + (1 | outcome), data = long_neg)
> fit_stab_2 <- lme4::lmer (stability_value ~ self_other_condition_c + (1 | id) + (1 | outcome), data = long_neg)
> stats::anova(fit_stab_1, fit_stab_2) # Fit2 better
refitting model(s) with ML (instead of REML)
Data: long_neg
Models:
fit_stab_1: stability_value ~ 1 + (1 | id) + (1 | outcome)
fit_stab_2: stability_value ~ self_other_condition_c + (1 | id) + (1 | outcome)
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
fit_stab_1    4 1668.8 1686.4 -830.40   1660.8
fit_stab_2    5 1665.1 1687.1 -827.57   1655.1 5.6612  1    0.01734 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Haupteffekt self_other_condition auf Stability
> summary(fit_stab_2)
Linear mixed model fit by REML ['lmerMod']
Formula: stability_value ~ self_other_condition_c + (1 | id) + (1 | outcome)
   Data: long_neg

REML criterion at convergence: 1660.6

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.8554 -0.5309  0.0228  0.6251  2.5416
```

```
Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.23266  0.4823
 outcome  (Intercept) 0.03262  0.1806
 Residual             0.75573  0.8693
Number of obs: 600, groups:  id, 130; outcome, 5

Fixed effects:
                      Estimate Std. Error t value
(Intercept)            3.51927    0.09803  35.901
self_other_condition_c -0.26597    0.11108  -2.394

Correlation of Fixed Effects:
            (Intr)
slf_thr_cn_ -0.024
>
> dplyr::group_by(long_neg, self_other_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(stability_value, na.rm = TRUE),
+                    sd = stats::sd(stability_value, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  self_other_condition count  mean    sd
  <chr>                <int> <dbl> <dbl>
1 green                  315  3.38 1.07
2 yellow                 285  3.65 0.935
> # Mean difference Group Yellow - Group Green
> # 0.27
>
> # Is Ostracism a good predictor for Stability?
> fit_stab_1 <- lme4::lmer (stability_value ~ 1 + (1 | id) + (1 | outcome), data = long_neg)
> fit_stab_3 <- lme4::lmer (stability_value ~ ostracism_condition_c + (1 | id) + (1 | outcome), data = long_neg)
> stats::anova (fit_stab_1, fit_stab_3)
```

```
refitting model(s) with ML (instead of REML)
Data: long_neg
Models:
fit_stab_1: stability_value ~ 1 + (1 | id) + (1 | outcome)
fit_stab_3: stability_value ~ ostracism_condition_c + (1 | id) + (1 | outcome)
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
fit_stab_1    4 1668.8 1686.4 -830.40   1660.8
fit_stab_3    5 1669.3 1691.3 -829.66   1659.3 1.4737  1     0.2248
>
> # Haupteffekt ostracism_condition auf Stability
> summary(fit_stab_3)
Linear mixed model fit by REML ['lmerMod']
Formula: stability_value ~ ostracism_condition_c + (1 | id) + (1 | outcome)
   Data: long_neg

REML criterion at convergence: 1664.7

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.9434 -0.5385  0.0456  0.6364  2.5341

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.24575  0.4957
 outcome  (Intercept) 0.03255  0.1804
 Residual             0.75578  0.8694
Number of obs: 600, groups:  id, 130; outcome, 5

Fixed effects:
                     Estimate Std. Error t value
(Intercept)           3.51079    0.09848  35.650
ostracism_condition_c -0.13676    0.11292  -1.211
```

```
Correlation of Fixed Effects:
            (Intr)
ostrcsm_cn_ 0.024
>
> dplyr::group_by(long_neg, ostracism_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(stability_value, na.rm = TRUE),
+                    sd = stats::sd(stability_value, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  ostracism_condition count  mean    sd
  <chr>               <int> <dbl> <dbl>
1 excluded              285  3.44 1.08
2 included              315  3.58 0.953
> # Mean difference Group Excluded - Group Included
> # 0.14
>
> # Interaction model
> fit_stability <- lme4::lmer (stability_value ~ self_other_condition_c * ostracism_condition_c + (1 |id) + (1 |
outcome), data = long_neg)
> summary(fit_stability)
Linear mixed model fit by REML ['lmerMod']
Formula: stability_value ~ self_other_condition_c * ostracism_condition_c +      (1 | id) + (1 | outcome)
   Data: long_neg

REML criterion at convergence: 1661.7

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.9129 -0.5332  0.0336  0.6435  2.5935

Random effects:
 Groups   Name        Variance Std.Dev.
```

```
 id       (Intercept) 0.23039  0.4800
 outcome  (Intercept) 0.03267  0.1807
 Residual             0.75587  0.8694
Number of obs: 600, groups:  id, 130; outcome, 5

Fixed effects:
                                            Estimate Std. Error t value
(Intercept)                                  3.52161    0.09811  35.894
self_other_condition_c                      -0.26422    0.11119  -2.376
ostracism_condition_c                       -0.11192    0.11120  -1.006
self_other_condition_c:ostracism_condition_c -0.27175    0.22238  -1.222

Correlation of Fixed Effects:
            (Intr) slf___ ostr__
slf_thr_cn_ -0.027
ostrcsm_cn_  0.028 -0.074
slf_th__:__ -0.042  0.049 -0.048
>
> dplyr::group_by(long_neg, ostracism_condition, self_other_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(stability_value, na.rm = TRUE),
+                    sd = stats::sd(stability_value, na.rm = TRUE))
`summarise()` regrouping output by 'ostracism_condition' (override with `.groups` argument)
# A tibble: 4 x 5
# Groups:   ostracism_condition [2]
  ostracism_condition self_other_condition count  mean    sd
  <chr>               <chr>                <int> <dbl> <dbl>
1 excluded            green                  159  3.26 1.12
2 excluded            yellow                 126  3.67 0.986
3 included            green                  156  3.51 1.01
4 included            yellow                 159  3.64 0.895
>
> # Mean difference Interaction Stability
```

```
> # (Excluded-Yellow - Excluded-Green) - ((Included-Yellow - Included-Green))
> # (3.67 - 3.26) - (3.64 - 3.51) = 0.28
>
> ### Try to also calculate the effect sizes fore mixed models
> ## For effect sizes, use https://jakewestfall.shinyapps.io/crossedpower/
> # Participants within condition
> # Unstandardized
> # 0 for all residuals we did not include in our model
> # Power = X
>
> # effect size self_other_condition: 0.647
> # effect size ostracism_condition: 0.22
> # effect size interaktion: 0.692
>
> # Hypothese 1 verworfen!
> # H1. While in both conditions (exclusion and inclusion) the participants attribute the behavior of an out-
group-member more to stable causes
> # than the behavior of an ingroup member, this difference is greater in the exclusion condition.
>
> ### Locus Analyses
> # First overall overview
> dplyr::group_by(long_neg, ostracism_condition, self_other_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(locus_value, na.rm = TRUE),
+                    sd = stats::sd(locus_value, na.rm = TRUE))
`summarise()` regrouping output by 'ostracism_condition' (override with `.groups` argument)
# A tibble: 4 x 5
# Groups:   ostracism_condition [2]
  ostracism_condition self_other_condition count  mean    sd
  <chr>              <chr>                <int> <dbl> <dbl>
1 excluded           green                  159  2.60  1.25
2 excluded           yellow                 126  2.80  1.23
3 included           green                  156  2.89  1.13
```

```
4 included            yellow                159  3.03  1.22
>
> # Plot locus
> Sum_locus ← rcompanion::groupwiseMean(locus_value ~ ostracism_condition + self_other_condition,
+                                       data   = long_neg,
+                                       conf   = 0.95,
+                                       digits = 3,
+                                       traditional = FALSE,
+                                       percentile  = TRUE)
> pd = ggplot2::position_dodge(.2)
> ggplot2::ggplot(Sum_locus, ggplot2::aes(x = ostracism_condition,
+                                         y = Mean,
+                                         color = self_other_condition)) +
+   ggplot2::geom_errorbar(ggplot2::aes(ymin=Percentile.lower,
+                                       ymax=Percentile.upper),
+                          width=.2, size=0.7, position=pd) +
+   ggplot2::geom_point(shape=15, size=4, position=pd) +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))  +
+   ggplot2::geom_point(shape=15, size=4, position=pd) + ggplot2::ylim(1, 5) +
+   ggplot2::ylab("Mean locus value (1 = stable; 5 = variable)") +
+   ggplot2::theme_bw() +
+   ggplot2::theme(axis.title = ggplot2::element_text(face = "bold"))
>
> # Is self_other_condition a good predictor for Locus?
> fit_locus_1 ← lme4::lmer (locus_value ~ 1 + (1 | id) + (1 | outcome), data = long_neg)
> fit_locus_2 ← lme4::lmer (locus_value ~ self_other_condition_c + (1 | id) + (1 | outcome), data = long_neg)
> stats::anova (fit_locus_1, fit_locus_2)
refitting model(s) with ML (instead of REML)
Data: long_neg
Models:
fit_locus_1: locus_value ~ 1 + (1 | id) + (1 | outcome)
fit_locus_2: locus_value ~ self_other_condition_c + (1 | id) + (1 | outcome)
            npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
```

```
fit_locus_1    4 1903.2 1920.8 -947.61   1895.2
fit_locus_2    5 1903.3 1925.2 -946.63   1893.3 1.9554  1      0.162
>
> # Haupteffekt self_other_condition auf Locus
> summary(fit_locus_2)
Linear mixed model fit by REML ['lmerMod']
Formula: locus_value ~ self_other_condition_c + (1 | id) + (1 | outcome)
   Data: long_neg

REML criterion at convergence: 1898.1

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.39013 -0.67137 -0.06569  0.67495  2.44957

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.2602   0.5101
 outcome  (Intercept) 0.0518   0.2276
 Residual             1.1668   1.0802
Number of obs: 600, groups:  id, 130; outcome, 5

Fixed effects:
                      Estimate Std. Error t value
(Intercept)             2.8423     0.1198  23.725
self_other_condition_c -0.1764     0.1264  -1.396

Correlation of Fixed Effects:
            (Intr)
slf_thr_cn_ -0.023
>
> dplyr::group_by(long_neg, self_other_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
```

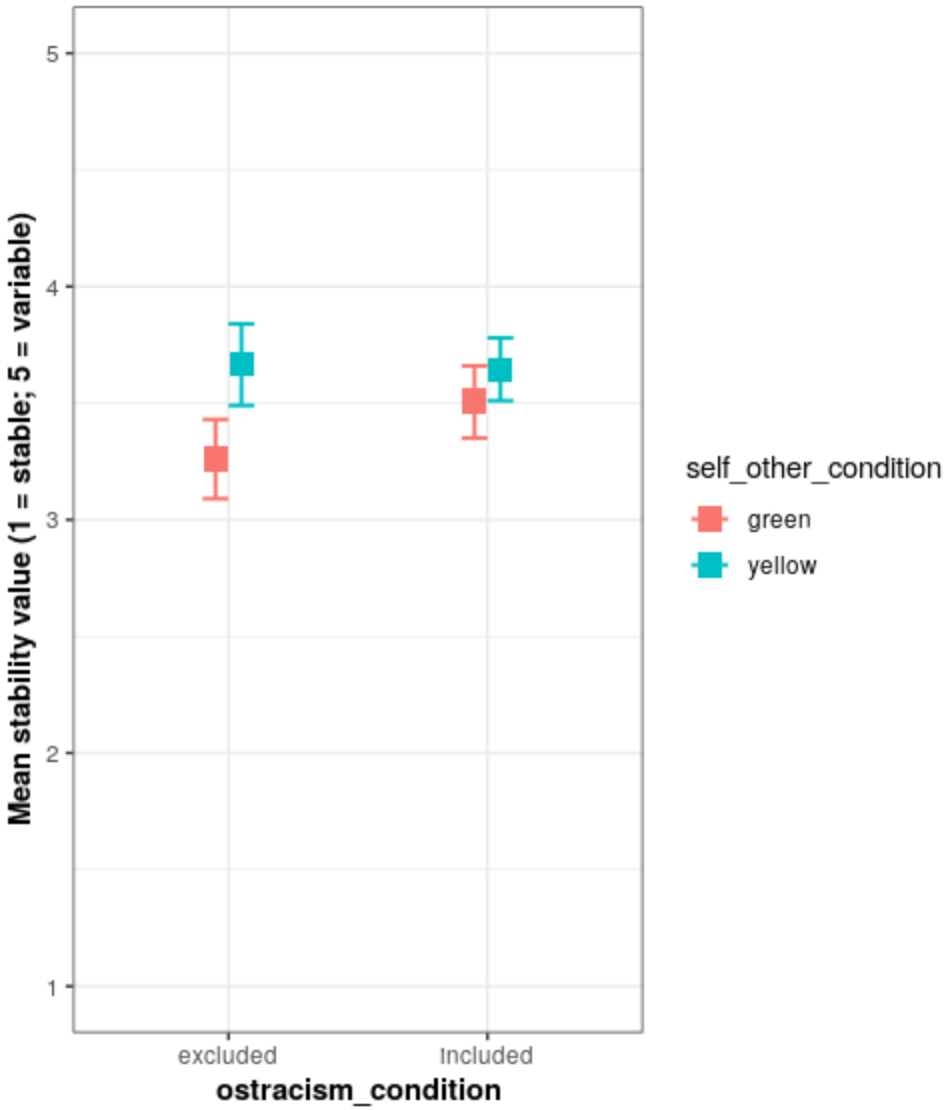
```
+                    mean = mean(locus_value, na.rm = TRUE),
+                    sd = stats::sd(locus_value, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  self_other_condition count  mean    sd
  <chr>                <int> <dbl> <dbl>
1 green                  315  2.74  1.20
2 yellow                 285  2.93  1.23
> # Mean difference Group Yellow - Group Green
> # 0.19
>
> # Is Ostracism a good predictor for Locus?
> fit_locus_1 <- lme4::lmer (locus_value ~ 1 + (1 | id) + (1 | outcome), data = long_neg)
> fit_locus_3 <- lme4::lmer (locus_value ~ ostracism_condition_c + (1 | id) + (1 | outcome), data = long_neg)
> stats::anova(fit_locus_1, fit_locus_3)
refitting model(s) with ML (instead of REML)
Data: long_neg
Models:
fit_locus_1: locus_value ~ 1 + (1 | id) + (1 | outcome)
fit_locus_3: locus_value ~ ostracism_condition_c + (1 | id) + (1 | outcome)
            npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
fit_locus_1    4 1903.2 1920.8 -947.61   1895.2
fit_locus_3    5 1900.5 1922.5 -945.27   1890.5 4.6816  1    0.03049 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Haupteffekt ostracism_condition auf Locus
> summary(fit_locus_3)
Linear mixed model fit by REML ['lmerMod']
Formula: locus_value ~ ostracism_condition_c + (1 | id) + (1 | outcome)
   Data: long_neg

REML criterion at convergence: 1895.4
```

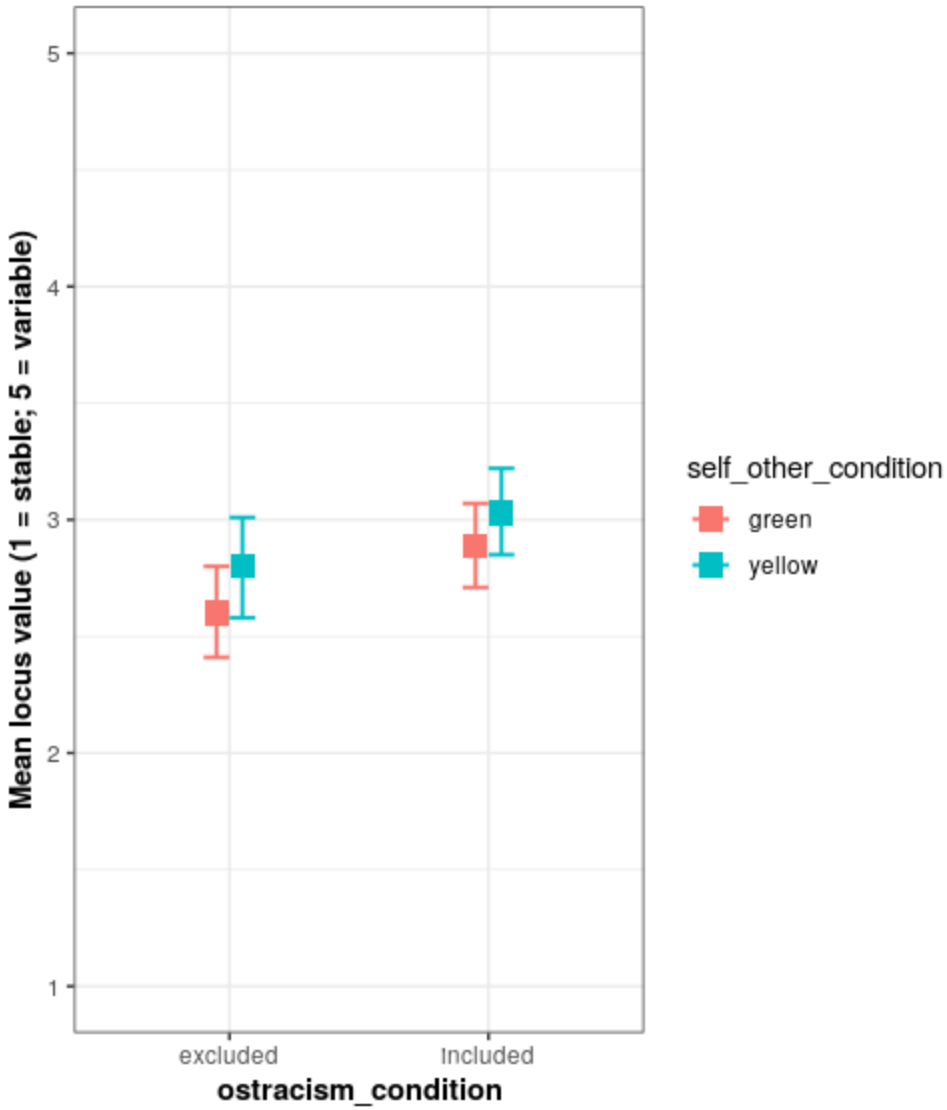
```
Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.39881 -0.70248 -0.06395  0.68038  2.47165

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.25173  0.5017
 outcome  (Intercept) 0.05104  0.2259
 Residual             1.16564  1.0797
Number of obs: 600, groups:  id, 130; outcome, 5

Fixed effects:
                     Estimate Std. Error t value
(Intercept)            2.8325     0.1189  23.825
ostracism_condition_c -0.2716     0.1253  -2.167

Correlation of Fixed Effects:
            (Intr)
ostrcsm_cn_ 0.023
>
> dplyr::group_by(long_neg, ostracism_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(locus_value, na.rm = TRUE),
+                    sd = stats::sd(locus_value, na.rm = TRUE))
`summarise()` ungrouping output (override with `.groups` argument)
# A tibble: 2 x 4
  ostracism_condition count  mean    sd
  <chr>               <int> <dbl> <dbl>
1 excluded              285  2.69  1.24
2 included              315  2.96  1.18
> # Mean difference Group Excluded - Group Included
> # 0.27
```

```
>
> # Interaction model
> fit_locus ← lme4::lmer (locus_value ~ self_other_condition_c * ostracism_condition_c + (1 |id) + (1 |
outcome), data = long_neg)
> summary(fit_locus)
Linear mixed model fit by REML ['lmerMod']
Formula: locus_value ~ self_other_condition_c * ostracism_condition_c +      (1 | id) + (1 | outcome)
   Data: long_neg

REML criterion at convergence: 1897

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.42907 -0.67271 -0.04669  0.65225  2.50532

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.25236  0.5024
 outcome  (Intercept) 0.05101  0.2258
 Residual             1.16619  1.0799
Number of obs: 600, groups:  id, 130; outcome, 5

Fixed effects:
                                            Estimate Std. Error t value
(Intercept)                                   2.8367     0.1190  23.836
self_other_condition_c                       -0.1590     0.1258  -1.263
ostracism_condition_c                        -0.2598     0.1259  -2.064
self_other_condition_c:ostracism_condition_c  -0.0275     0.2517  -0.109

Correlation of Fixed Effects:
            (Intr) slf___ ostr__
slf_thr_cn_ -0.026
ostrcsm_cn_  0.027 -0.072
```

```
slf_th__:__ -0.039  0.051 -0.050
>
> dplyr::group_by(long_neg, ostracism_condition, self_other_condition) %>%
+   dplyr::summarise(count = dplyr::n(),
+                    mean = mean(locus_value, na.rm = TRUE),
+                    sd = stats::sd(locus_value, na.rm = TRUE))
`summarise()` regrouping output by 'ostracism_condition' (override with `.groups` argument)
# A tibble: 4 x 5
# Groups:   ostracism_condition [2]
  ostracism_condition self_other_condition count  mean    sd
  <chr>              <chr>                <int> <dbl> <dbl>
1 excluded           green                  159  2.60  1.25
2 excluded           yellow                 126  2.80  1.23
3 included           green                  156  2.89  1.13
4 included           yellow                 159  3.03  1.22
>
> # Mean difference Interaction Locus
> # (Excluded-Yellow - Excluded-Green) - ((Included-Yellow - Included-Green))
> # (2.80 - 2.6) - (3.03 - 2.89) = 0.06
>
> ### Try to also calculate the effect sizes fore mixed models
> ## For effect sizes, use https://jakewestfall.shinyapps.io/crossedpower/
> # Participants within condition
> # Unstandardized
> # 0 for all residuals we did not include in our model
> # Power = X
>
> # effect size self_other_condition: 0.269
> # effect size ostracism_condition: 0.548
> # effect size interaktion: 0.075
>
> # Hypothese 2 verworfen!
```

> # H2. While in both conditions (exclusion and inclusion) the participants attribute the behavior of an out-group-member more to internal causes
> # than the behavior of an in-group-member, this difference is greater in the exclusion condition.


Mean stability value (1 = stable; 5 = variable)
5
4
3
2
1
excluded
Included
ostracism_condition
self_other_condition
green
yellow


Mean locus value (1 = stable; 5 = variable)
5
4
3
2
1
excluded
Included
ostracism_condition
self_other_condition
green
yellow