Analysis EMG (no outliers)

Antonio Schettino

2020-03-23

Contents

Participants who did not believe the cover story

Eleven participants did not believe the experimental manipulation, and the responses of 3 participants are missing. Here we confirm that the results are qualitatively similar with and without these 14 outliers.

```r
EMG <- read_csv("EMG.csv") %>%
  pivot_longer(
    Reappraisal_Unexpected:Appraisal_Expected,
    names_to = "condition",
    values_to = "amplitude"
  ) %>%
  separate(
    condition,
    c("ER_Strategy", "Expectation"),
    sep = "_",
    remove = TRUE
  ) %>%
  mutate(
    Participant = as_factor(Participant),
    Believed_Story = as_factor(Believed_Story),
    Stimulus = as_factor(Stimulus),
    ER_Strategy = as_factor(ER_Strategy),
    Expectation = as_factor(Expectation)
```

Table 1: Descriptive statistics of EMG amplitude, separately for each condition.

Stimulus	ER_Strategy	Expectation	median	mad
cue	Reappraisal	Unexpected	0.00099	0.0032
cue	Reappraisal	Expected	0.00056	0.0025
cue	Appraisal	Unexpected	-0.00065	0.0043
cue	Appraisal	Expected	0.00112	0.0042
target	Reappraisal	Unexpected	-0.00022	0.0029
target	Reappraisal	Expected	0.00037	0.0024
target	Appraisal	Unexpected	0.00153	0.0051
target	Appraisal	Expected	0.00122	0.0036

```
  ) %>%
  filter(Believed_Story == "yes")

# summary
EMG_summary <-
  EMG %>%
  group_by(Stimulus, ER_Strategy, Expectation) %>%
  summarize(
    median = median(amplitude),
    mad = mad(amplitude)
  )

kable(
  EMG_summary,
  digits = c(0, 0, 0, 5, 5),
  caption = "Descriptive statistics of EMG amplitude, separately for each condition."
) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

Plot

```
# plot
EMG %>%
  ggplot(
    aes(
      x = ER_Strategy,
      y = amplitude,
      color = Expectation,
      fill = Expectation
    )
  ) +
  geom_pirate(
    bars = FALSE,
    cis = TRUE,
    lines = TRUE, lines_params = list(color = "black", alpha = .3),
    points = TRUE, points_params = list(color = "black", shape = 21, size = 5, alpha = .2),
    violins = TRUE, violins_params = list(size = 1),
    show.legend = TRUE
  ) +
  scale_color_viridis_d(option = "cividis") +
  scale_fill_viridis_d(option = "cividis") +
  scale_y_continuous(limits = c(-.05, .05)) +
  ggtitle("EMG") +
  facet_wrap(. ~ Stimulus) +
  theme_custom +
  theme(
    legend.position = c(.55, .8),
    axis.title.x = element_blank()
  )
```

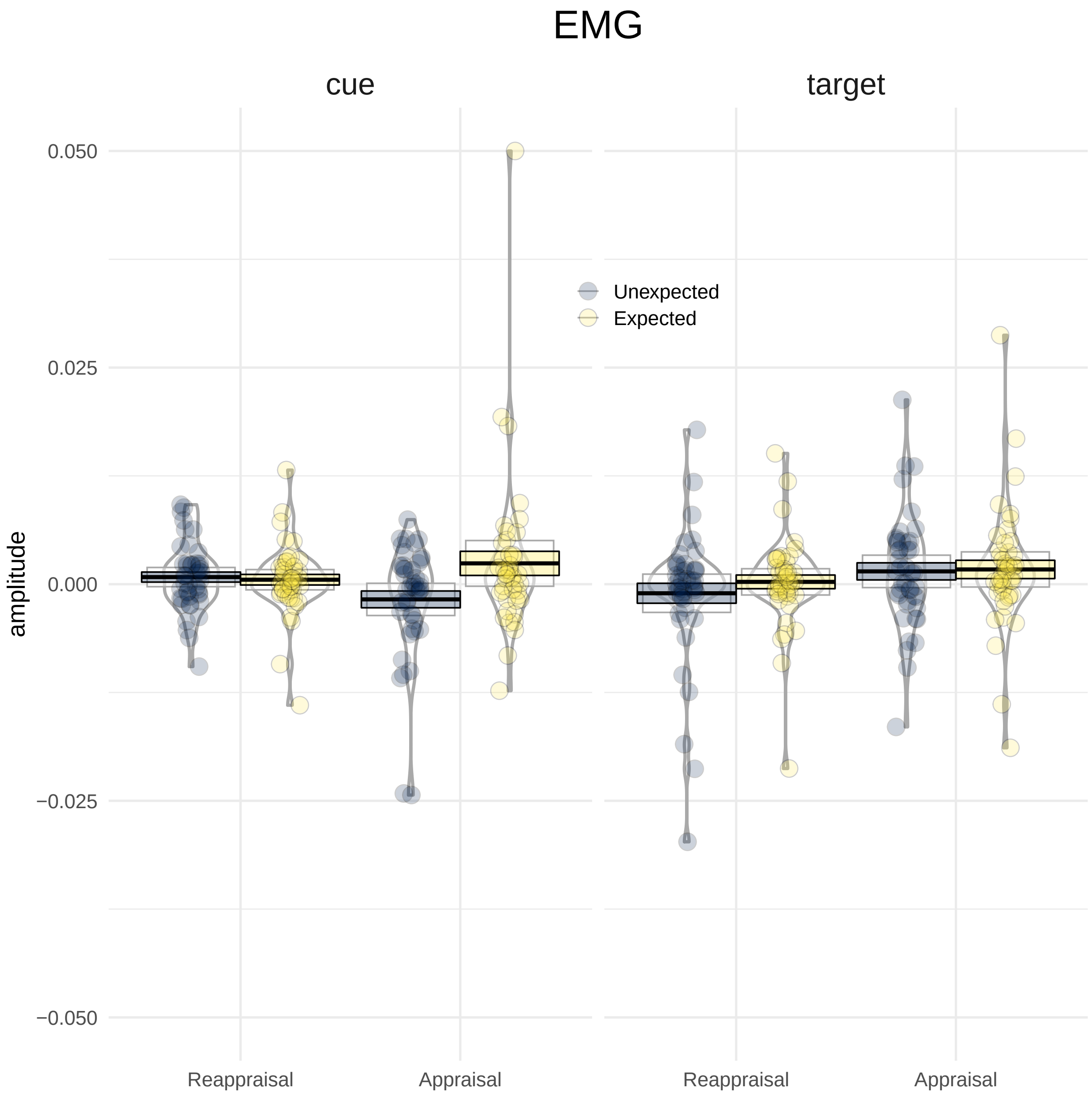

Figure 1: EMG amplitude in response to cue and target. **Note**: 18 values are outside the plot.

Table 2: Repeated measures nonparametric ANOVA (Aligned Rank Transform).

Term	Df	Df.res	F	p.value	eta.sq.part
Stimulus	1	322	2.29	0.131	0.007
ER_Strategy	1	322	1.58	0.210	0.005
Expectation	1	322	3.56	0.060	0.011
Stimulus:ER_Strategy	1	322	3.46	0.064	0.011
Stimulus:Expectation	1	322	0.43	0.513	0.001
ER_Strategy:Expectation	1	322	0.01	0.909	0.000
Stimulus:ER_Strategy:Expectation	1	322	7.72	0.006	0.023

Analysis

```r
# repeated measures nonparametric ANOVA, Aligned Rank Transform
all_ART <- anova(
  art(amplitude ~ Stimulus * ER_Strategy * Expectation + Error(Participant),
    data = EMG
  )
) %>%
  mutate(
    eta.sq.part = `Sum Sq` / (`Sum Sq` + `Sum Sq.res`) # partial eta squared
  )

all_ART %>%
  dplyr::select(`Term`, `Df`, `Df.res`, `F` = `F value`, p.value = `Pr(>F)`, `eta.sq.part`) %>%
  kable(.,
    digits = c(0, 0, 0, 2, 3, 3),
    caption = "Repeated measures nonparametric ANOVA (Aligned Rank Transform)."
  ) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

Cue

Analyze EMG following cue presentation.

```r
# keep only EMG in response to cue
EMG_cue <- filter(EMG, Stimulus == "cue") %>%
  dplyr::select(-Stimulus)
```

Plot

```r
# plot
EMG_cue %>%
  ggplot(
    aes(
      x = ER_Strategy,
      y = amplitude,
      color = Expectation,
      fill = Expectation
    )
  ) +
  geom_pirate(
```

Table 3: Repeated measures nonparametric ANOVA (Aligned Rank Transform).

Term	Df	Df.res	F	p.value	eta.sq.part
ER_Strategy	1	138	0.38	0.539	0.003
Expectation	1	138	4.21	0.042	0.030
ER_Strategy:Expectation	1	138	6.53	0.012	0.045

```
    bars = FALSE,
    cis = TRUE,
    lines = TRUE, lines_params = list(color = "black", alpha = .3),
    points = TRUE, points_params = list(color = "black", shape = 21, size = 5, alpha = .2),
    violins = TRUE, violins_params = list(size = 1),
    show.legend = TRUE
  ) +
  scale_color_viridis_d(option = "cividis") +
  scale_fill_viridis_d(option = "cividis") +
  scale_y_continuous(limits = c(-.05, .05)) +
  ggtitle("Cue EMG") +
  theme_custom +
  theme(
    legend.position = c(.5, .15),
    axis.title.x = element_blank()
  )
```

Analysis

Repeated measures nonparametric ANOVA (Aligned Rank Transform), post-hoc Paired Samples Wilcoxon Tests.

```
# repeated measures nonparametric ANOVA, Aligned Rank Transform
cue_ART <- anova(
  art(amplitude ~ ER_Strategy * Expectation + Error(Participant),
    data = EMG_cue
  )
) %>%
  mutate(
    eta.sq.part = `Sum Sq` / (`Sum Sq` + `Sum Sq.res`) # partial eta squared
  )

cue_ART %>%
  dplyr::select(`Term`, `Df`, `Df.res`, `F` = `F value`, p.value = `Pr(>F)`, `eta.sq.part`) %>%
  kable(.,
    digits = c(0, 0, 0, 2, 3, 3),
    caption = "Repeated measures nonparametric ANOVA (Aligned Rank Transform)."
  ) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

```
## post-hoc analyses
# Appraisal, Expected vs. Unexpected
# Paired Samples Wilcoxon Test
EMG_Appraisal_ExpectedUnexpected_Wilcoxon <-
  wilcox.test(filter(EMG_cue, ER_Strategy == "Appraisal" & Expectation == "Expected")$amplitude,
    filter(EMG_cue, ER_Strategy == "Appraisal" & Expectation == "Unexpected")$amplitude,
```

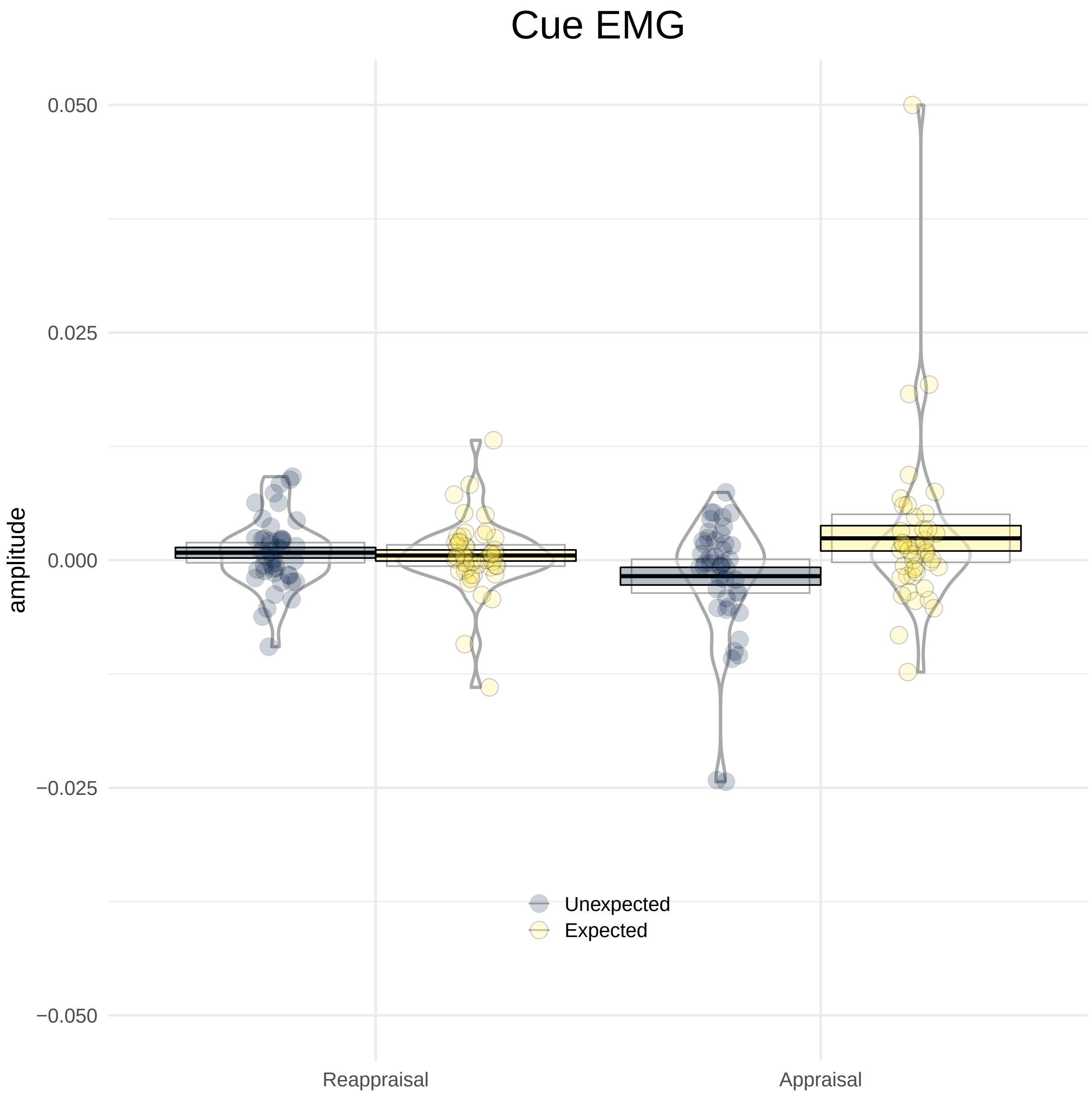

Figure 2: EMG amplitude in response to cue. **Note**: 9 values are outside the plot.

```
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Appraisal_ExpectedUnexpected_r <-
  wilcoxonPairedR(filter(EMG_cue, ER_Strategy == "Appraisal" & Expectation %in% c("Expected", "Unexpecte
    filter(EMG_cue, ER_Strategy == "Appraisal" & Expectation %in% c("Expected", "Unexpected"))$Expectati
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# Reappraisal, Expected vs. Unexpected
# Paired Samples Wilcoxon Test
EMG_Reappraisal_ExpectedUnexpected_Wilcoxon <-
  wilcox.test(filter(EMG_cue, ER_Strategy == "Reappraisal" & Expectation == "Expected")$amplitude,
    filter(EMG_cue, ER_Strategy == "Reappraisal" & Expectation == "Unexpected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Reappraisal_ExpectedUnexpected_r <-
  wilcoxonPairedR(filter(EMG_cue, ER_Strategy == "Reappraisal" & Expectation %in% c("Expected", "Unexpe
    filter(EMG_cue, ER_Strategy == "Reappraisal" & Expectation %in% c("Expected", "Unexpected"))$Expecta
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# Expected, Appraisal vs. Reappraisal
# Paired Samples Wilcoxon Test
EMG_Expected_AppraisalReappraisal_Wilcoxon <-
  wilcox.test(filter(EMG_cue, ER_Strategy == "Appraisal" & Expectation == "Expected")$amplitude,
    filter(EMG_cue, ER_Strategy == "Reappraisal" & Expectation == "Expected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
```

```r
EMG_Expected_AppraisalReappraisal_r <-
  wilcoxonPairedR(filter(EMG_cue, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Expe
    filter(EMG_cue, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Expected")$ER_Strat
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# Unexpected, Appraisal vs. Reappraisal
# Paired Samples Wilcoxon Test
EMG_Unexpected_AppraisalReappraisal_Wilcoxon <-
  wilcox.test(filter(EMG_cue, ER_Strategy == "Appraisal" & Expectation == "Unexpected")$amplitude,
    filter(EMG_cue, ER_Strategy == "Reappraisal" & Expectation == "Unexpected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Unexpected_AppraisalReappraisal_r <-
  wilcoxonPairedR(filter(EMG_cue, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Unexp
    filter(EMG_cue, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Unexpected")$ER_Str
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# summary
cue_ART_posthoc <- tibble(
  "comparison" = c(
    "Appraisal, Expected vs. Unexpected", "Reappraisal, Expected vs. Unexpected",
    "Expected, Appraisal vs. Reappraisal", "Unexpected, Appraisal vs. Reappraisal"
  ),
  "V" = c(
    EMG_Appraisal_ExpectedUnexpected_Wilcoxon$statistic, EMG_Reappraisal_ExpectedUnexpected_Wilcoxon$sta
    EMG_Expected_AppraisalReappraisal_Wilcoxon$statistic, EMG_Unexpected_AppraisalReappraisal_Wilcoxon$
  ),
  "p" = c(
    EMG_Appraisal_ExpectedUnexpected_Wilcoxon$p.value, EMG_Reappraisal_ExpectedUnexpected_Wilcoxon$p.val
    EMG_Expected_AppraisalReappraisal_Wilcoxon$p.value, EMG_Unexpected_AppraisalReappraisal_Wilcoxon$p.
  ),
  "r" = c(
    EMG_Appraisal_ExpectedUnexpected_r$r, EMG_Reappraisal_ExpectedUnexpected_r$r,
    EMG_Expected_AppraisalReappraisal_r$r, EMG_Unexpected_AppraisalReappraisal_r$r
  ),
  "CI95_lower" = c(
    EMG_Appraisal_ExpectedUnexpected_r$lower.ci, EMG_Reappraisal_ExpectedUnexpected_r$lower.ci,
    EMG_Expected_AppraisalReappraisal_r$lower.ci, EMG_Unexpected_AppraisalReappraisal_r$lower.ci
  ),
  "CI95_upper" = c(
```

Table 4: Post-hoc Repeated Samples Wilcoxon tests (Bonferroni-Holm p-value correction), bootstrapped effect size.

comparison	V	p	r	CI95_lower	CI95_upper
Appraisal, Expected vs. Unexpected	802	0.044	-0.37	-0.59	-0.08
Reappraisal, Expected vs. Unexpected	529	0.738	0.05	-0.24	0.34
Expected, Appraisal vs. Reappraisal	650	0.738	-0.13	-0.41	0.15
Unexpected, Appraisal vs. Reappraisal	370	0.119	0.30	0.02	0.55

```
    EMG_Appraisal_ExpectedUnexpected_r$upper.ci, EMG_Reappraisal_ExpectedUnexpected_r$upper.ci,
    EMG_Expected_AppraisalReappraisal_r$upper.ci, EMG_Unexpected_AppraisalReappraisal_r$upper.ci
  )
) %>%
  mutate("p" = p.adjust(p, method = "holm")) # Bonferroni-Holm correction on post-hoc tests

kable(cue_ART_posthoc,
  digits = c(0, 0, 3, 2, 2, 2),
  caption = "Post-hoc Repeated Samples Wilcoxon tests (Bonferroni-Holm p-value correction), bootstrappe
) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

Target

Analyze EMG following target presentation.

```
# keep only EMG in response to target
EMG_target <- filter(EMG, Stimulus == "target") %>%
  dplyr::select(-Stimulus)
```

Plot

```
# plot
EMG_target %>%
  ggplot(
    aes(
      x = ER_Strategy,
      y = amplitude,
      color = Expectation,
      fill = Expectation
    )
  ) +
  geom_pirate(
    bars = FALSE,
    cis = TRUE,
    lines = TRUE, lines_params = list(color = "black", alpha = .3),
    points = TRUE, points_params = list(color = "black", shape = 21, size = 5, alpha = .2),
    violins = TRUE, violins_params = list(size = 1),
    show.legend = TRUE
  ) +
  scale_color_viridis_d(option = "cividis") +
  scale_fill_viridis_d(option = "cividis") +
```

```r
scale_y_continuous(limits = c(-.05, .05)) +
ggtitle("Cue EMG") +
theme_custom +
theme(
  legend.position = c(.5, .15),
  axis.title.x = element_blank()
)
```

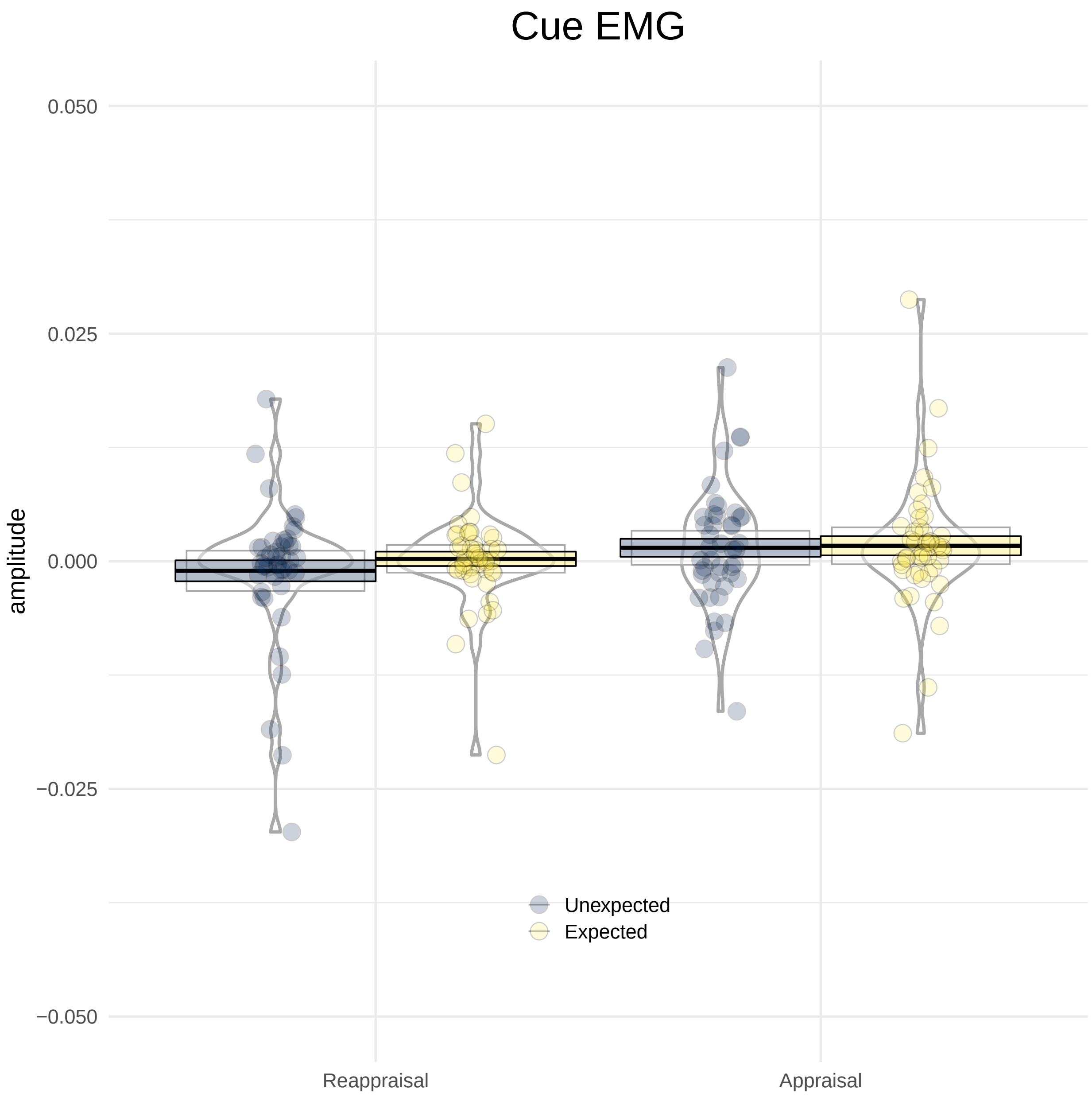

Figure 3: EMG amplitude in response to target. **Note**: 9 values are outside the plot.

Table 5: Repeated measures nonparametric ANOVA (Aligned Rank Transform).

Term	Df	Df.res	F	p.value	eta.sq.part
ER_Strategy	1	138	5.58	0.020	0.039
Expectation	1	138	0.35	0.553	0.003
ER_Strategy:Expectation	1	138	4.36	0.039	0.031

Analysis

Repeated measures nonparametric ANOVA (Aligned Rank Transform), post-hoc Paired Samples Wilcoxon Tests.

```
# repeated measures nonparametric ANOVA, Aligned Rank Transform
target_ART <- anova(
  art(amplitude ~ ER_Strategy * Expectation + Error(Participant),
    data = EMG_target
  )
) %>%
  mutate(
    eta.sq.part = `Sum Sq` / (`Sum Sq` + `Sum Sq.res`) # partial eta squared
  )

target_ART %>%
  dplyr::select(`Term`, `Df`, `Df.res`, `F` = `F value`, p.value = `Pr(>F)`, `eta.sq.part`) %>%
  kable(.,
    digits = c(0, 0, 0, 2, 3, 3),
    caption = "Repeated measures nonparametric ANOVA (Aligned Rank Transform)."
  ) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

```
## post-hoc analyses
# Appraisal, Expected vs. Unexpected
# Paired Samples Wilcoxon Test
EMG_Appraisal_ExpectedUnexpected_Wilcoxon <-
  wilcox.test(filter(EMG_target, ER_Strategy == "Appraisal" & Expectation == "Expected")$amplitude,
    filter(EMG_target, ER_Strategy == "Appraisal" & Expectation == "Unexpected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Appraisal_ExpectedUnexpected_r <-
  wilcoxonPairedR(filter(EMG_target, ER_Strategy == "Appraisal" & Expectation %in% c("Expected", "Unexpe
    filter(EMG_target, ER_Strategy == "Appraisal" & Expectation %in% c("Expected", "Unexpected"))$Expect
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# Reappraisal, Expected vs. Unexpected
```

```r
# Paired Samples Wilcoxon Test
EMG_Reappraisal_ExpectedUnexpected_Wilcoxon <-
  wilcox.test(filter(EMG_target, ER_Strategy == "Reappraisal" & Expectation == "Expected")$amplitude,
    filter(EMG_target, ER_Strategy == "Reappraisal" & Expectation == "Unexpected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Reappraisal_ExpectedUnexpected_r <-
  wilcoxonPairedR(filter(EMG_target, ER_Strategy == "Reappraisal" & Expectation %in% c("Expected", "Une
    filter(EMG_target, ER_Strategy == "Reappraisal" & Expectation %in% c("Expected", "Unexpected"))$Expe
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# Expected, Appraisal vs. Reappraisal
# Paired Samples Wilcoxon Test
EMG_Expected_AppraisalReappraisal_Wilcoxon <-
  wilcox.test(filter(EMG_target, ER_Strategy == "Appraisal" & Expectation == "Expected")$amplitude,
    filter(EMG_target, ER_Strategy == "Reappraisal" & Expectation == "Expected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Expected_AppraisalReappraisal_r <-
  wilcoxonPairedR(filter(EMG_target, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "E
    filter(EMG_target, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Expected")$ER_St
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# Unexpected, Appraisal vs. Reappraisal
# Paired Samples Wilcoxon Test
EMG_Unexpected_AppraisalReappraisal_Wilcoxon <-
  wilcox.test(filter(EMG_target, ER_Strategy == "Appraisal" & Expectation == "Unexpected")$amplitude,
    filter(EMG_target, ER_Strategy == "Reappraisal" & Expectation == "Unexpected")$amplitude,
    alternative = "two.sided",
    mu = 0,
    paired = TRUE,
    exact = TRUE,
    correct = TRUE,
    conf.int = TRUE, conf.level = 0.95
  )

# effect size r
EMG_Unexpected_AppraisalReappraisal_r <-
  wilcoxonPairedR(filter(EMG_target, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Un
    filter(EMG_target, ER_Strategy %in% c("Appraisal", "Reappraisal") & Expectation == "Unexpected")$ER_
    ci = TRUE, conf = 0.95,
    type = "bca", R = 5000,
    cases = TRUE
  )

# summary
target_ART_posthoc <- tibble(
  "comparison" = c(
    "Appraisal, Expected vs. Unexpected", "Reappraisal, Expected vs. Unexpected",
    "Expected, Appraisal vs. Reappraisal", "Unexpected, Appraisal vs. Reappraisal"
  ),
  "V" = c(
    EMG_Appraisal_ExpectedUnexpected_Wilcoxon$statistic, EMG_Reappraisal_ExpectedUnexpected_Wilcoxon$sta
    EMG_Expected_AppraisalReappraisal_Wilcoxon$statistic, EMG_Unexpected_AppraisalReappraisal_Wilcoxon$s
  ),
  "p" = c(
    EMG_Appraisal_ExpectedUnexpected_Wilcoxon$p.value, EMG_Reappraisal_ExpectedUnexpected_Wilcoxon$p.val
    EMG_Expected_AppraisalReappraisal_Wilcoxon$p.value, EMG_Unexpected_AppraisalReappraisal_Wilcoxon$p.v
  ),
  "r" = c(
    EMG_Appraisal_ExpectedUnexpected_r$r, EMG_Reappraisal_ExpectedUnexpected_r$r,
    EMG_Expected_AppraisalReappraisal_r$r, EMG_Unexpected_AppraisalReappraisal_r$r
  ),
  "CI95_lower" = c(
    EMG_Appraisal_ExpectedUnexpected_r$lower.ci, EMG_Reappraisal_ExpectedUnexpected_r$lower.ci,
    EMG_Expected_AppraisalReappraisal_r$lower.ci, EMG_Unexpected_AppraisalReappraisal_r$lower.ci
  ),
  "CI95_upper" = c(
    EMG_Appraisal_ExpectedUnexpected_r$upper.ci, EMG_Reappraisal_ExpectedUnexpected_r$upper.ci,
    EMG_Expected_AppraisalReappraisal_r$upper.ci, EMG_Unexpected_AppraisalReappraisal_r$upper.ci
  )
) %>%
  mutate("p" = p.adjust(p, method = "holm")) # Bonferroni-Holm correction on post-hoc tests

kable(target_ART_posthoc,
  digits = c(0, 0, 3, 2, 2, 2),
  caption = "Post-hoc Repeated Samples Wilcoxon tests (Bonferroni-Holm p-value correction), bootstrappe
) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

Table 6: Post-hoc Repeated Samples Wilcoxon tests (Bonferroni-Holm p-value correction), bootstrapped effect size.

comparison	V	p	r	CI95_lower	CI95_upper
Appraisal, Expected vs. Unexpected	514	0.604	0.08	-0.22	0.34
Reappraisal, Expected vs. Unexpected	667	0.562	-0.16	-0.43	0.13
Expected, Appraisal vs. Reappraisal	692	0.537	-0.20	-0.46	0.09
Unexpected, Appraisal vs. Reappraisal	836	0.014	-0.42	-0.62	-0.13

Session Info

```r
sessionInfo()
```

```
## R version 3.6.3 (2020-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.4 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=nl_NL.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=nl_NL.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=nl_NL.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=nl_NL.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] cowplot_1.0.0     ggpirate_0.1.1    viridis_0.5.1     viridisLite_0.3.0
##  [5] ez_4.4-0          ARTool_0.10.7     rcompanion_2.3.25 forcats_0.5.0
##  [9] stringr_1.4.0     dplyr_0.8.5       purrr_0.3.3       readr_1.3.1
## [13] tidyr_1.0.2       tibble_2.1.3      ggplot2_3.3.0     tidyverse_1.3.0
## [17] kableExtra_1.1.0  knitr_1.28
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-144       matrixStats_0.56.0 fs_1.3.2           lubridate_1.7.4
##  [5] webshot_0.5.2      httr_1.4.1         tools_3.6.3        backports_1.1.5
##  [9] R6_2.4.1           mgcv_1.8-31        nortest_1.0-4      DBI_1.1.0
## [13] colorspace_1.4-1   withr_2.1.2        gridExtra_2.3      tidyselect_1.0.0
## [17] curl_4.3           compiler_3.6.3     cli_2.0.2          rvest_0.3.5
## [21] expm_0.999-4       xml2_1.2.5         sandwich_2.5-1     labeling_0.3
## [25] scales_1.1.0       lmtest_0.9-37      mvtnorm_1.1-0      multcompView_0.1-8
## [29] digest_0.6.25      minqa_1.2.4        foreign_0.8-75     rmarkdown_2.1
## [33] rio_0.5.16         pkgconfig_2.0.3    htmltools_0.4.0    lme4_1.1-21
## [37] highr_0.8          dbplyr_1.4.2       rlang_0.4.5        readxl_1.3.1
## [41] rstudioapi_0.11    farver_2.0.3       generics_0.0.2     zoo_1.8-7
## [45] jsonlite_1.6.1     zip_2.0.4          car_3.0-7          magrittr_1.5
## [49] modeltools_0.2-23  Matrix_1.2-18      Rcpp_1.0.4         DescTools_0.99.34
## [53] munsell_0.5.0      fansi_0.4.1        abind_1.4-5        lifecycle_0.2.0   
## [57] stringi_1.4.6      multcomp_1.4-12    yaml_2.2.1         carData_3.0-3     
## [61] MASS_7.3-51.5      plyr_1.8.6         grid_3.6.3         parallel_3.6.3    
## [65] crayon_1.3.4       lattice_0.20-40    haven_2.2.0        splines_3.6.3     
## [69] hms_0.5.3          pillar_1.4.3       EMT_1.1            boot_1.3-24       
## [73] reshape2_1.4.3     codetools_0.2-16   stats4_3.6.3       reprex_0.3.0      
## [77] glue_1.3.2         evaluate_0.14      data.table_1.12.8  modelr_0.1.6      
## [81] nloptr_1.2.2.1     vctrs_0.2.4        cellranger_1.1.0   gtable_0.3.0      
## [85] assertthat_0.2.1   xfun_0.12          openxlsx_4.1.4     coin_1.3-1        
## [89] libcoin_1.0-5      broom_0.5.5        survival_3.1-11    ellipsis_0.3.0    
## [93] TH.data_1.0-10
```