

# Social Distance

## Load the data

```
## Load the data
library(readxl)

file_path <- "/Users/daryani/Desktop/data.xlsx"
data <- read_excel(file_path)

Sys.setenv(RGL_USE_NULL = "TRUE")
options(rgl.useNULL = TRUE)

# Packages
library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(car)
```

Loading required package: carData

Attaching package: 'car'

The following object is masked from 'package:dplyr':

recode

```
library(heplots)
```

Loading required package: broom

```
library(MVN)
library(emmeans)
```

Welcome to emmeans.

Caution: You lose important information if you filter this package's results.

See '? untidy'

```
library(effectsize)
```

## Ensure factors are factors

```
data <- data %>%
  mutate(
    group_clean = case_when(
      group %in% c(1, "1", "Prolife", "prolife") ~ "Prolife",
      group %in% c(2, "2", "Prochoice", "prochoice") ~ "Prochoice",
      TRUE ~ NA_character_
    ),
    condition_clean = case_when(
      condition %in% c(1, "1", "Outgroup", "outgroup", "Actual", "Actual Perception") ~ "Outgroup",
      condition %in% c(2, "2", "Metaperception", "metaperception", "Meta") ~ "Metaperception",
      condition %in% c(3, "3", "Ingroup", "ingroup") ~ "Ingroup",
      TRUE ~ NA_character_
    ),
    group = factor(group_clean, levels = c("Prolife", "Prochoice")),
    condition = factor(condition_clean, levels = c("Outgroup", "Metaperception", "Ingroup"))
  ) %>%
```

```
select(-group_clean, -condition_clean)

print(with(data, table(group, condition, useNA = "ifany")))
```

	condition	
group	Outgroup	Metaperception
Prolife	89	85
Prochoice	96	97

## Create Descriptives

```
desc_distance <- data %>%
  group_by(group, condition) %>%
  summarise(
    n = dplyr::n(),
    mean_distance = mean(distance_1, na.rm = TRUE),
    sd_distance = sd(distance_1, na.rm = TRUE),
    .groups = "drop"
  )

print(as.data.frame(desc_distance), row.names = FALSE)
```

group	condition	n	mean_distance	sd_distance
Prolife	Outgroup	89	3.966292	1.256342
Prolife	Metaperception	85	5.341176	1.286865
Prolife	Ingroup	89	2.977528	1.437940
Prochoice	Outgroup	96	4.875000	1.250263
Prochoice	Metaperception	97	5.216495	1.445012
Prochoice	Ingroup	96	3.104167	1.235264

```
# Optional: Wide format to see group x condition more clearly
distance_wide <- tidyr::pivot_wider(desc_distance,
                                   names_from = condition, values_from = mean_distance)
cat("\nSocial Distance (means):\n"); print(as.data.frame(distance_wide), row.names = FALSE)
```

Social Distance (means):

	group	n	sd_distance	Outgroup	Metaperception	Ingroup
	Prolife	89	1.256342	3.966292	NA	NA
	Prolife	85	1.286865	NA	5.341176	NA
	Prolife	89	1.437940	NA	NA	2.977528
	Prochoice	96	1.250263	4.875000	NA	NA
	Prochoice	97	1.445012	NA	5.216495	NA
	Prochoice	96	1.235264	NA	NA	3.104167

## Fit ANCOVA

```
fit_distance <- lm(distance_1 ~ group * condition + ideology, data = data)
Anova(fit_distance, type = 3)
```

Anova Table (Type III tests)

Response: distance\_1

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	448.22	1	256.6841	< 2.2e-16 ***
group	21.15	1	12.1113	0.0005413 ***
condition	243.40	2	69.6950	< 2.2e-16 ***
ideology	2.19	1	1.2519	0.2636887
group:condition	27.09	2	7.7570	0.0004767 ***
Residuals	951.69	545		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
eta_squared(Anova(fit_distance, type = 3), partial = TRUE, ci = 0.95)
```

Type 3 ANOVAs only give sensible and informative results when covariates are mean-centered and factors are coded with orthogonal contrasts (such as those produced by ``contr.sum``, ``contr.poly``, or ``contr.helmert``, but *not* by the default ``contr.treatment``).

# Effect Size for ANOVA (Type III)

Parameter	Eta2 (partial)	95% CI
group	0.02	[0.01, 1.00]

condition		0.20		[0.16, 1.00]
ideology		2.29e-03		[0.00, 1.00]
group:condition		0.03		[0.01, 1.00]

- One-sided CIs: upper bound fixed at [1.00].

## Planned Contrasts – RQ1: Ingroup vs Outgroup

```
emm_dist <- emmeans(fit_distance, ~ group * condition)

labs_d <- with(as.data.frame(emm_dist), paste(group, condition, sep = "."))

make_d <- function(pos_names, neg_names, labels = labs_d) {
  w <- setNames(rep(0, length(labels)), labels)
  w[pos_names] <- w[pos_names] + 1
  w[neg_names] <- w[neg_names] - 1
  w
}

# Ingroup vs Outgroup (RQ1)
h1_distance <- contrast(
  emm_dist,
  method = list(
    "Prolife Ingroup vs Prochoice Outgroup" = make_d("Prolife.Ingroup", "Prochoice.Outgroup"),
    "Prochoice Ingroup vs Prolife Outgroup" = make_d("Prochoice.Ingroup", "Prolife.Outgroup")
  ),
  adjust = "none"
)

summary(h1_distance, infer = c(TRUE, TRUE))
```

contrast	estimate	SE	df	lower.CL	upper.CL
Prolife Ingroup vs Prochoice Outgroup	-1.761	0.230	545	-2.21	-1.310
Prochoice Ingroup vs Prolife Outgroup	-0.986	0.224	545	-1.43	-0.547
t.ratio	p.value				
-7.669	<.0001				
-4.404	<.0001				

Confidence level used: 0.95

```
eff_size(h1_distance, sigma = sigma(fit_distance), edf = df.residual(fit_distance))
```

contrast

Prolife Ingroup vs Prochoice Outgroup - Prochoice Ingroup vs Prolife Outgroup

effect.size	SE	df	lower.CL	upper.CL
-0.586	0.273	545	-1.12	-0.0489

sigma used for effect sizes: 1.321

Confidence level used: 0.95

## Planned Contrasts – RQ2: Outgroup vs Metaperception

```
# Outgroup vs Metaperception (RQ2)
```

```
h2_distance <- contrast(
```

```
  emm_dist,
```

```
  method = list(
```

```
    "Prolife Outgroup vs Prochoice Meta" = make_d("Prolife.Outgroup", "Prochoice.Metapercept.
```

```
    "Prochoice Outgroup vs Prolife Meta" = make_d("Prochoice.Outgroup", "Prolife.Metapercept.
```

```
  ),
```

```
  adjust = "none"
```

```
)
```

```
summary(h2_distance, infer = c(TRUE, TRUE))
```

contrast	estimate	SE	df	lower.CL	upper.CL
Prolife Outgroup vs Prochoice Meta	-1.121	0.226	545	-1.56	-0.677
Prochoice Outgroup vs Prolife Meta	-0.597	0.229	545	-1.05	-0.147
t.ratio	-4.963				
p.value	<.0001				
	-2.608	0.0094			

Confidence level used: 0.95

```
eff_size(h2_distance, sigma = sigma(fit_distance), edf = df.residual(fit_distance))
```

contrast

Prolife Outgroup vs Prochoice Meta - Prochoice Outgroup vs Prolife Meta

effect.size	SE	df	lower.CL	upper.CL
-------------	----	----	----------	----------

-0.396 0.274 545    -0.934    0.141

sigma used for effect sizes: 1.321

Confidence level used: 0.95