Poland Analysis Code

2024-01-21

Intro

These are the preliminary results from our Survey in December 2023. Our first set of analyses, conducted on the trial level, concern the trust index. The index is measured on a scale from 0 to 1 with higher values indicating higher trust.

The trust index is the mean of the following items on a 0 to 1 scale:

Untrustworthy – – – – – – – Trustworthy
Misleading – – – – – – – Truthful
Biased – – – – – – – Objective

Preliminary Summary Stats

Sample demographics and survey responses

```
## Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if
## `.name_repair` is omitted as of tibble 2.0.0.
## i Using compatibility `.name_repair`.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
print(xtable::xtable(summary_dat, caption = "Summary Statistics"), include.rownames = F, caption.placem
```

% latex table generated in R 4.2.2 by xtable 1.8-4 package % Sun Jan 21 21:23:26 2024

Table 1: Summary Statistics

demovars	V1
Female	0.54
Age	43.92
BA Graduate	0.16
Employed Full Time	0.59
Employed Part Time	0.08
Retired	0.17
Voted	0.84
PiS	0.24
Democratic Opposition	0.56
Confederation	0.09
Political Interest	0.68
TVP Watcher	0.31
TVN Watcher	0.53

Blind Condition Outcome Mean by Party and Source

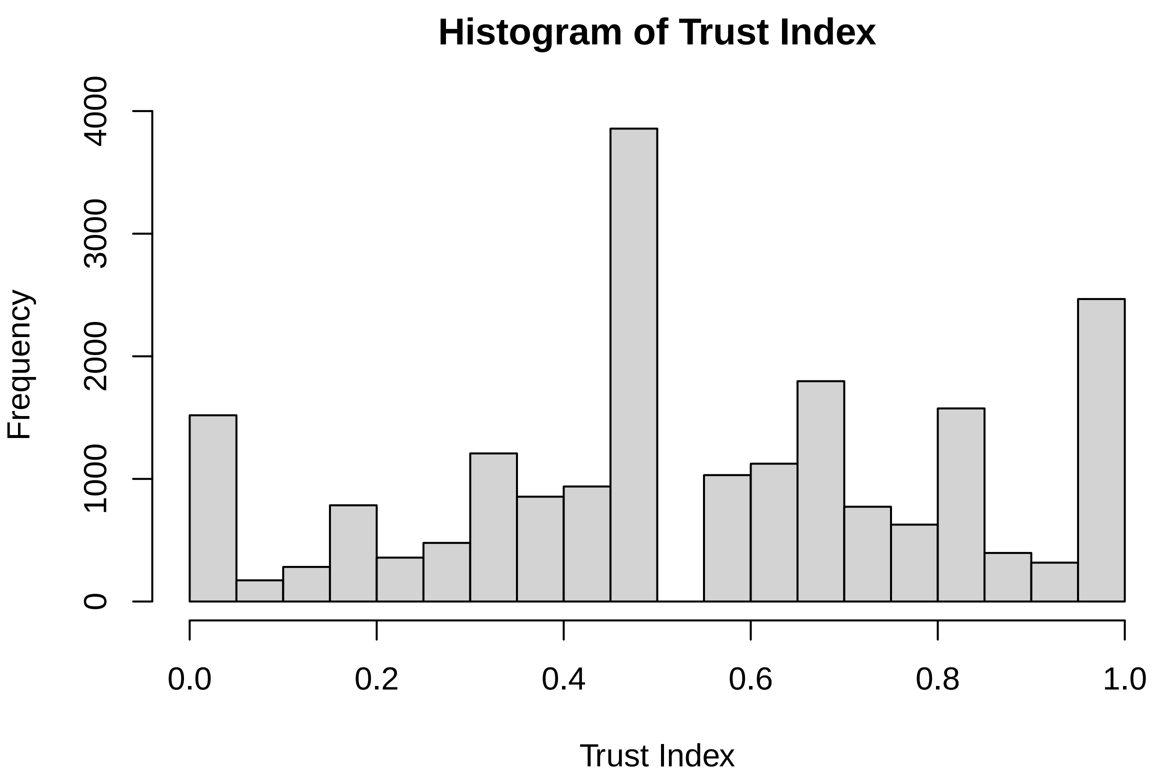
pro_pis	trust_index	trustworthy	truthful	objective
0	0.57	0.57	0.57	0.55
1	0.59	0.60	0.59	0.58

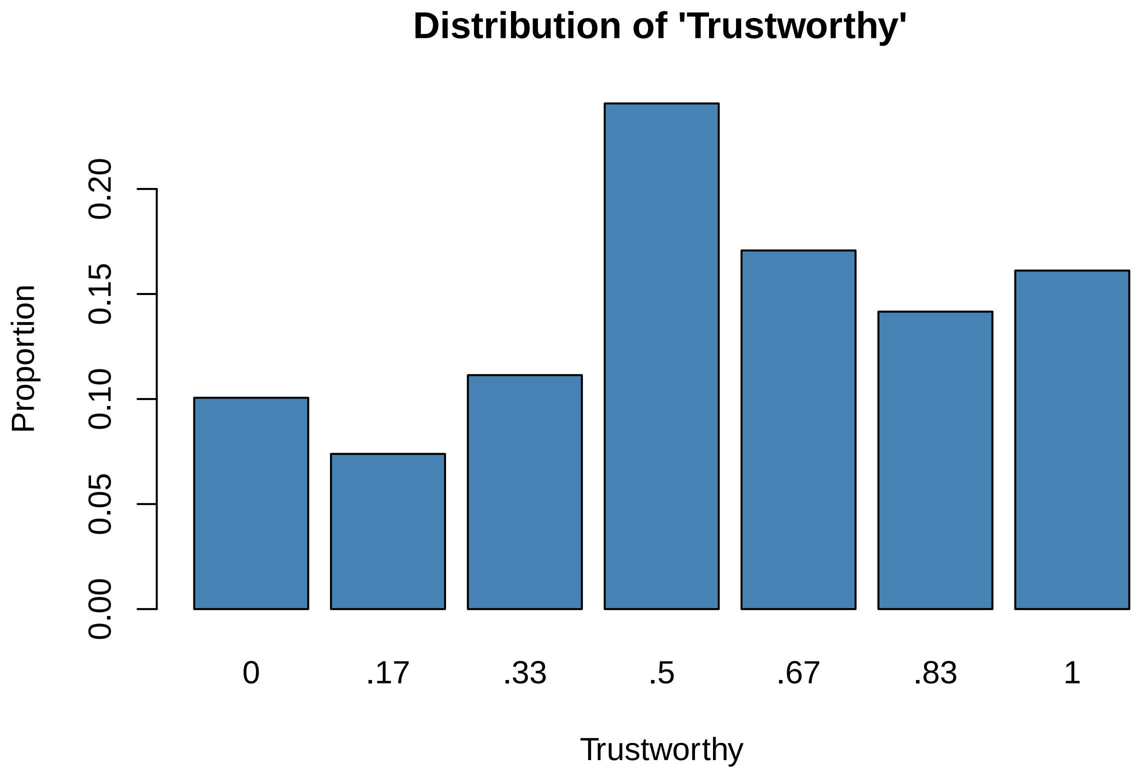
source	trust_index	trustworthy	truthful	objective
tvn	0.59	0.59	0.59	0.57
tvp	0.56	0.57	0.57	0.54

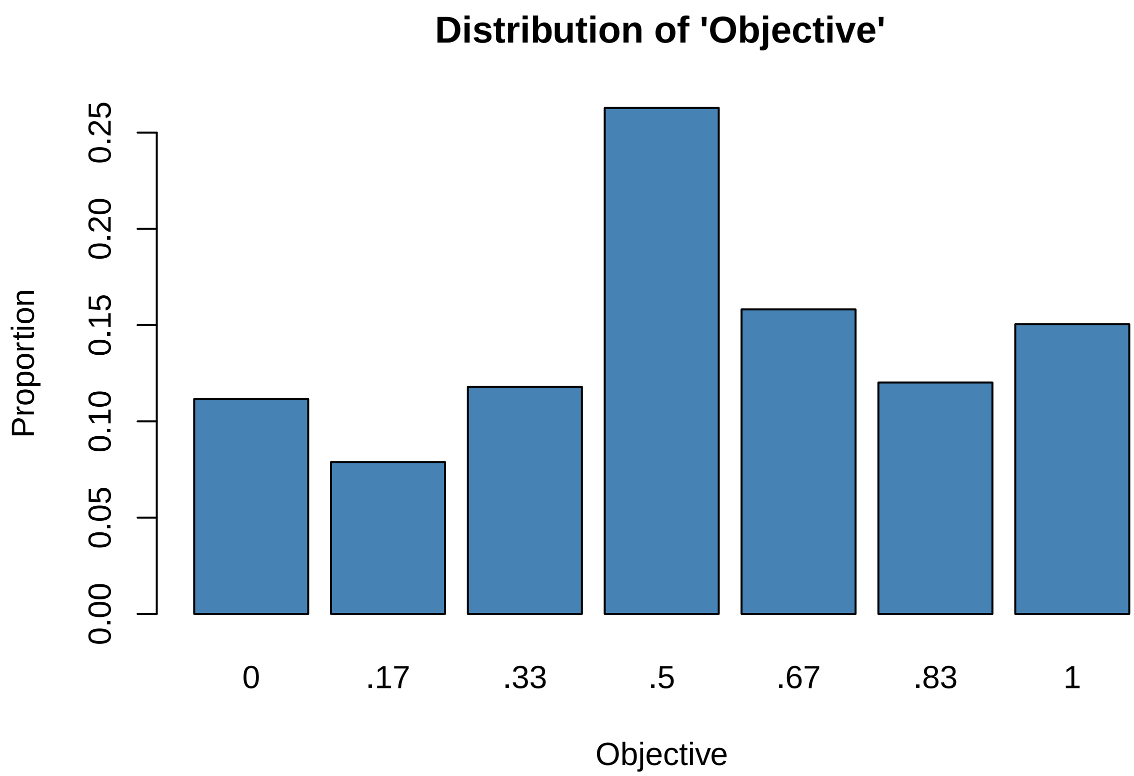
```r
#Std. Dev Trust Index
sd(dat_long$trust_index, na.rm = T)
```

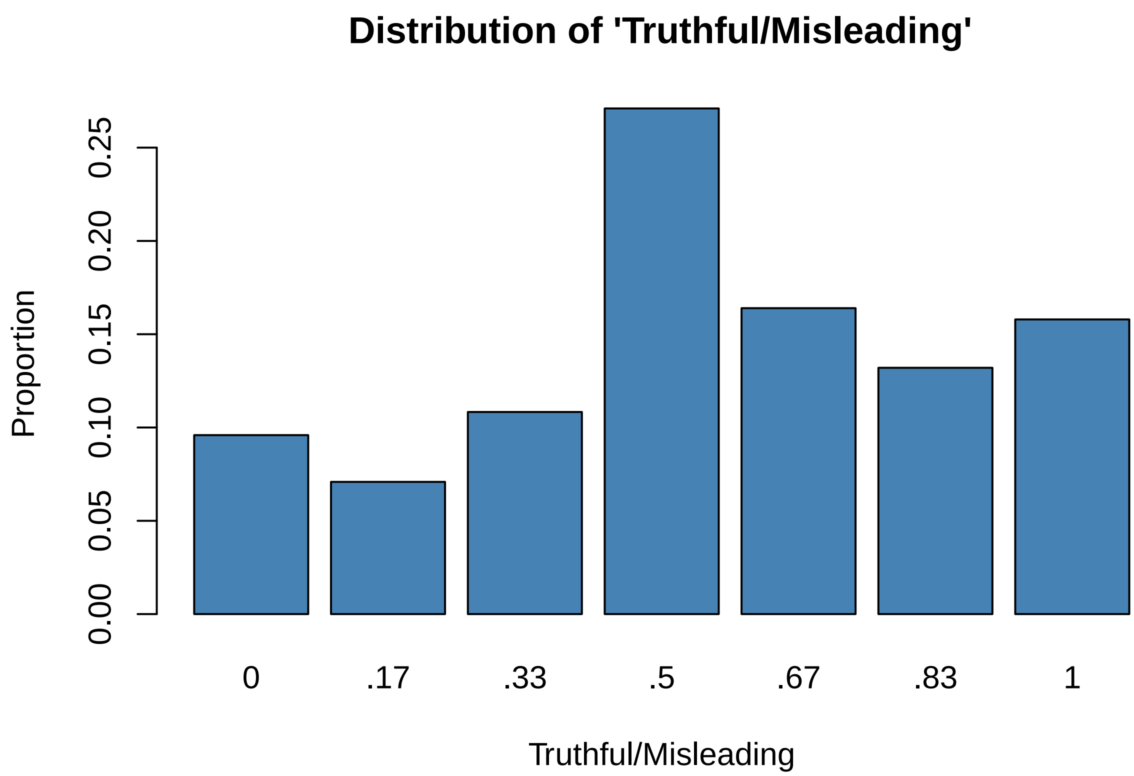
```
## [1] 0.2832011
```

Distribution of the Outcome Variable




Distribution of 'Trustworthy'
Proportion
0.00
0.05
0.10
0.15
0.20
0
.17
.33
.5
.67
.83
1
Trustworthy


Distribution of 'Objective'
Proportion
0.00
0.05
0.10
0.15
0.20
0.25
0
.17
.33
.5
.67
.83
1
Objective


Distribution of 'Truthful/Misleading'
Proportion
0.00
0.05
0.10
0.15
0.20
0.25
0
.17
.33
.5
.67
.83
1
Truthful/Misleading

Main Source Cue Effects

These are the main results from our test of the source cue effect. In all cases, we are comparing responses in the blind condition to responses in the revealed condition, conditional on source. As pre-specified, we fit separate models for respondents that voted/would have voted for PiS and respondents that voted/would have voted for parties in the Democratic Opposition coalition. Note that Poland has a multi-party system and we are not including respondents identifying with a party other than one associated with these largest two coalitions.

Here is the code for fitting the models. Note that for each we fit an OLS model regressing the trust index on treated with excerpt fixed effects. Standard errors are clustered on respondent. Note that in this study, unlike the previous study, respondents were assigned to see either TVP or TVN.

```
dat_long <- dat_long %>%
  mutate(treated = case_when(treatment == "blind" ~ 0,
                             treatment == "revealed" ~ 1))

dat <- dat %>%
  mutate(treated = case_when(treatment == "blind" ~ 0,
                             treatment == "revealed" ~ 1))

#fit main models, OLS

#note I'll be using p for pro-pis and d for Democratic Opposition


#TVP effect on pro-pis respondents
m1_p_tvp <- lm(trust_index ~ treated + excerpt, #models include excerpt FEs
              data = dat_long[dat_long$pro_pis == 1 &  #subsetting to pro-pis
                                dat_long$source == "tvp",]) #only including those in TVP condition
#SEs clustered on respondents (8 trials per respondent)
m1_p_tvp_cl <- coeftest(m1_p_tvp, vcov = vcovCL, cluster = ~ResponseId)



#TVN effect on anti-pis/ Democratic Opposition respondents
m1_d_tvn <- lm(trust_index ~ treated + excerpt,
              data = dat_long[dat_long$anti_pis == 1 &
                                dat_long$source == "tvn",])
m1_d_tvn_cl <- coeftest(m1_d_tvn, vcov = vcovCL, cluster = ~ResponseId)


#TVN cue effect on pro-pis
m1_p_tvn <- lm(trust_index ~ treated + excerpt,
              data = dat_long[dat_long$pro_pis == 1 &
                                dat_long$source == "tvn",])
m1_p_tvn_cl <- coeftest(m1_p_tvn, vcov = vcovCL, cluster = ~ResponseId)



#tvp effect on anti-pis
m1_d_tvp <- lm(trust_index ~ treated + excerpt,
              data = dat_long[dat_long$anti_pis == 1 &
                                dat_long$source == "tvp",])
m1_d_tvp_cl <- coeftest(m1_d_tvp, vcov = vcovCL, cluster = ~ResponseId)
```

The results can be found in Table 2. We see that, consistent with our hypotheses, the TVP brand has a positive effect on trust for those who support the PiS party but it has a negative effect on trust for those who supported the opposition coalition. By contrast, the TVN brand has a negative effective on trust for those who support the PiS party but a positive effect on those who support the opposition. The negative effect of TVP for the opposition supporters is especially strong.

Table 2: Main Source Effect Results

	Dependent variable:			
	Trust Index			
	TVP|PiS	TVN|PiS	TVP|DO	TVN|DO
	(1)	(2)	(3)	(4)
SourceRevealed	0.059^{***}	-0.061^{***}	-0.127^{***}	0.051^{***}
	(0.019)	(0.022)	(0.014)	(0.013)
Excerpt FE	Yes	Yes	Yes	Yes
Observations	2,608	2,400	5,688	5,936
Adjusted R^2	0.067	0.057	0.114	0.065
Note:	*p<0.1; **p<0.05; ***p<0.01			
	Trust index measured on 0 (low trust) to 1 (high trust)			

Source Cue Effect Size

Here we look at the Cohen's D for these source cue effects. Note: for some of these I include both a manual calculation of cohen's d and one calculated by a package - just to check that it's correct.

Counter Source Effect Size

```
#Effect Size, Counter Source
g0_counter <- na.omit(dat_long$trust_index[dat_long$counter_source == 1 &
                                             dat_long$treated == 0])
g1_counter <- na.omit(dat_long$trust_index[dat_long$counter_source == 1 &
                                             dat_long$treated == 1])

mean(g0_counter)
```

```
## [1] 0.5397953
```

```
mean(g1_counter)
```

```
## [1] 0.4354789
```

```
dim_counter <- mean(g1_counter) - mean(g0_counter) #calc dif in means
dim_counter
```

```
## [1] -0.1043165
```

```
#dif in means as percentage of control mean
dim_counter / mean(g0_counter, na.rm = T)
```

```
## [1] -0.1932519
```

```
sd_pooled_counter <- effectsize::sd_pooled(g0_counter, g1_counter)

#cohens'd - counter source
dim_counter / sd_pooled_counter
```

```
## [1] -0.3628144
```

```
##use packages to calculate cohen's d
cohensD_counter <- effsize::cohen.d(g0_counter, g1_counter) #calc cohens d with effsize packages
cohensD_counter
```

```
##
## Cohen's d
##
## d estimate: 0.3628144 (small)
## 95 percent confidence interval:
##     lower     upper
## 0.3187847 0.4068441
```

Aligned source effect size

```
#Effect Size Aligned Source
##control aligned trials - aligned
g0_aligned <- na.omit(dat_long$trust_index[dat_long$aligned_source == 1 &
                                             dat_long$treated == 0])
```

```
## Warning: Unknown or uninitialised column: `aligned_source`.
```

```
##treated aligned trials - aligne
g1_aligned <- na.omit(dat_long$trust_index[dat_long$aligned_source == 1 &
                                             dat_long$treated == 1])
```

```
## Warning: Unknown or uninitialised column: `aligned_source`.
```

```
cohensD_aligned <- effsize::cohen.d(g0_aligned, g1_aligned) #calc cohens d with effsize packages
```

```
## Warning in qt((1 - conf.level)/2, df): NaNs produced
```

```
cohensD_aligned
```

```
##
## Cohen's d
##
## d estimate: NaN (NA)
## 95 percent confidence interval:
## lower upper
##   NaN   NaN
```

PiS supporters - counter source effect sizes

```
#Effect Size, Split By Party

##PiS Supporters - TVN effect

g0_p_tvn <- na.omit(dat_long$trust_index[dat_long$pro_pis == 1 &
                                           dat_long$source == "tvn" &
                                           dat_long$treated == 0])
g1_p_tvn <- na.omit(dat_long$trust_index[dat_long$pro_pis == 1 &
                                           dat_long$source == "tvn" &
                                           dat_long$treated == 1])

mean(g0_p_tvn)
```

```
## [1] 0.5557598
```

```
mean(g1_p_tvn)
```

```
## [1] 0.498518
```

```
dim_rep_tvn <- mean(g1_p_tvn) - mean(g0_p_tvn) #dif in means
dim_rep_tvn
```

```
## [1] -0.05724185
```

```
sd_pooled_rep_tvn <- effectsize::sd_pooled(g0_p_tvn, g1_p_tvn) #pooled sd
dim_rep_tvn / sd_pooled_rep_tvn #cohens d manual
```

```
## [1] -0.1942964
```

```
cohensD_rep_tvn <- effsize::cohen.d(g0_p_tvn, g1_p_tvn) #calc cohens d with effsize packages
cohensD_rep_tvn
```

```
##
## Cohen's d
##
```

```
## d estimate: 0.1942964 (negligible)
## 95 percent confidence interval:
##     lower     upper
## 0.1137019 0.2748908
```

Dem. Oppo. supporters - counter source effect sizes

```r
##Opposition - tvp Effect

g0_d_tvp <- na.omit(dat_long$trust_index[dat_long$anti_pis == 1 &
                                           dat_long$source == "tvp" &
                                           dat_long$treated == 0])
g1_d_tvp <- na.omit(dat_long$trust_index[dat_long$anti_pis == 1 &
                                           dat_long$source == "tvp" &
                                           dat_long$treated == 1])

mean(g0_d_tvp)
```

```
## [1] 0.5333907
```

```r
mean(g1_d_tvp)
```

```
## [1] 0.4076874
```

```r
dim_d_tvp <- mean(g1_d_tvp) - mean(g0_d_tvp) #dif in means
sd_pooled_d_tvp <- effectsize::sd_pooled(g0_d_tvp, g1_d_tvp) #pooled sd
dim_d_tvp / sd_pooled_d_tvp #cohens d manual
```

```
## [1] -0.4456371
```

```r
cohensD_d_tvp <- effsize::cohen.d(g0_d_tvp, g1_d_tvp) #calc cohens d with effsize packages
cohensD_d_tvp
```

```
##
## Cohen's d
##
## d estimate: 0.4456371 (small)
## 95 percent confidence interval:
##     lower     upper
## 0.3929539 0.4983203
```

Blind Taste Test

Here we look at whether people have greater trust in TVP/TVN even in the blind condition. We ask whether pro-PiS respondents prefer TVP to TVN, even without knowing its source, and whether Dem Opposition respondents prefer TVN to TVP even not knowing its source. Respondents were randomly assigned to one of three topic groups: general (all politically relevant news, including articles included in the other topics), globalization and outsiders, and domestic politics. Within a given topic, articles shown to a respondent were randomly selected (although it doesn't appear they were actually shown evenly.)

In these preliminary models, I just fit separate models for the three topic groups, with no adjusting.

```
#Note: I use m_btt to refer to blind taste test models in the code.

#preference for TVP content blind, general topic, pro-pis
m_btt_p_gen <- lm(trust_index ~ I(source == "tvp"), data = dat_long[dat_long$treatment == "blind"
                                                        & dat_long$pro_pis == 1
                                                        & dat_long$topic == "general",])
m_btt_p_gen_cl <- coeftest(m_btt_p_gen, vcov = vcovCL, cluster = ~ResponseId)

#preference for TVP content blind, general topic, anti-pis
m_btt_d_gen <- lm(trust_index ~ I(source == "tvp"), data = dat_long[dat_long$treatment == "blind"
                                                        & dat_long$anti_pis == 1
                                                        & dat_long$topic == "general",])
m_btt_d_gen_cl <- coeftest(m_btt_d_gen, vcov = vcovCL, cluster = ~ResponseId)


#preference for TVP content blind, outsiders topic, pro-pis
m_btt_p_out <- lm(trust_index ~ I(source == "tvp"), data = dat_long[dat_long$treatment == "blind"
                                                        & dat_long$pro_pis == 1
                                                        & dat_long$topic == "outsiders",])
m_btt_p_out_cl <- coeftest(m_btt_p_out, vcov = vcovCL, cluster = ~ResponseId)

#preference for TVP content blind, outsiders topic, anti-pis
m_btt_d_out <- lm(trust_index ~ I(source == "tvp"), data = dat_long[dat_long$treatment == "blind"
                                                        & dat_long$anti_pis == 1
                                                        & dat_long$topic == "outsiders",])
m_btt_d_out_cl <- coeftest(m_btt_d_out, vcov = vcovCL, cluster = ~ResponseId)


#preference for TVP content blind, politics topic, pro-pis
m_btt_p_pol <- lm(trust_index ~ I(source == "tvp"), data = dat_long[dat_long$treatment == "blind"
                                                        & dat_long$pro_pis == 1
                                                        & dat_long$topic == "politics",])
m_btt_p_pol_cl <- coeftest(m_btt_p_pol, vcov = vcovCL, cluster = ~ResponseId)

#preference for TVP content blind, politics topic, anti-pis
m_btt_d_pol <- lm(trust_index ~ I(source == "tvp"), data = dat_long[dat_long$treatment == "blind"
                                                        & dat_long$anti_pis == 1
                                                        & dat_long$topic == "politics",])
m_btt_d_pol_cl <- coeftest(m_btt_d_pol, vcov = vcovCL, cluster = ~ResponseId)
```

Blind taste test - weight adjusted

We can use our full pool of articles as a proxy for the prevalence of a topic during the time period we studied, and then use this to construct weights to down-weight articles that showed up more frequently in our data

Table 3: Blind Source Preferences: Pro-PiS

	Dependent variable:		
	Trust Index		
	General	Globalization	Politics
	(1)	(2)	(3)
TVP	0.038	0.025	0.120^{***}
	(0.030)	(0.034)	(0.038)
Constant	0.565^{***}	0.583^{***}	0.516^{***}
	(0.021)	(0.023)	(0.032)
Observations	840	784	872
Adjusted R^2	0.003	0.001	0.039
Note:			*p<0.1; **p<0.05; ***p<0.01
			Trust index measured on 0 (low trust) to 1 (high trust)

Table 4: Blind Source Preferences: Anti-PiS

	Dependent variable:		
	Trust Index		
	General	Globalization	Politics
	(1)	(2)	(3)
TVP	-0.076^{***}	-0.032	-0.117^{***}
	(0.020)	(0.023)	(0.021)
Constant	0.606^{***}	0.611^{***}	0.615^{***}
	(0.015)	(0.015)	(0.014)
Observations	1,984	1,760	2,016
Adjusted R^2	0.020	0.003	0.044
Note:			*p<0.1; **p<0.05; ***p<0.01
			Trust index measured on 0 (low trust) to 1 (high trust)

than they should have and upweight articles that showed up less frequently than they should have if we were assigning articles completely at random. This allows us to get a more accurate measure of preference for a given source with a representative sample of articles from that source. Here we use data for all the topics together. Trials for excerpts included in the special topics have lower weight, because those articles were over-sampled.

These results suggest that people have a preference for content of an aligned source over a counter source, even in the blind condition. Again, we see evidence of a stronger preference for aligned content from the Democratic Opposition respondents compared to the PiS respondents.

```
#count how frequently a given article showed up in the
weight_dat <- dat_long %>%
  group_by(excerpt) %>%
  summarise(
    article_count = n()
  )

# 124 unique tvn articles, 99 unique tvp articles

tvn_n <- nrow(dat_long[dat_long$source == "tvn",]) / 124

tvp_n <- nrow(dat_long[dat_long$source == "tvp",]) / 99

numerator <- c(rep(tvn_n, 124), rep(tvp_n, 99))

weight_dat <- weight_dat %>%
  mutate(excerpt_prev_weight = numerator / article_count)

dat_long <- left_join(dat_long, weight_dat)
```

```
## Joining with `by = join_by(excerpt)`
```

```
#preference for TVP content blind, weighted by article frequency, pro-pis
m_btt_p_w <- lm(trust_index ~ I(source == "tvp"), weights = excerpt_prev_weight, data = dat_long[dat_lon
                                                                              & dat_long$pro_pis == 1,])
m_btt_p_w_cl <- coeftest(m_btt_p_w, vcov = vcovCL, cluster = ~ResponseId)

#preference for TVP content blind, weighted by article frequency,, anti-pis
m_btt_d_w <- lm(trust_index ~ I(source == "tvp"), weights = excerpt_prev_weight, data = dat_long[dat_lon
                                                                              & dat_long$anti_pis == 1,])
m_btt_d_w_cl <- coeftest(m_btt_d_w, vcov = vcovCL, cluster = ~ResponseId)
```

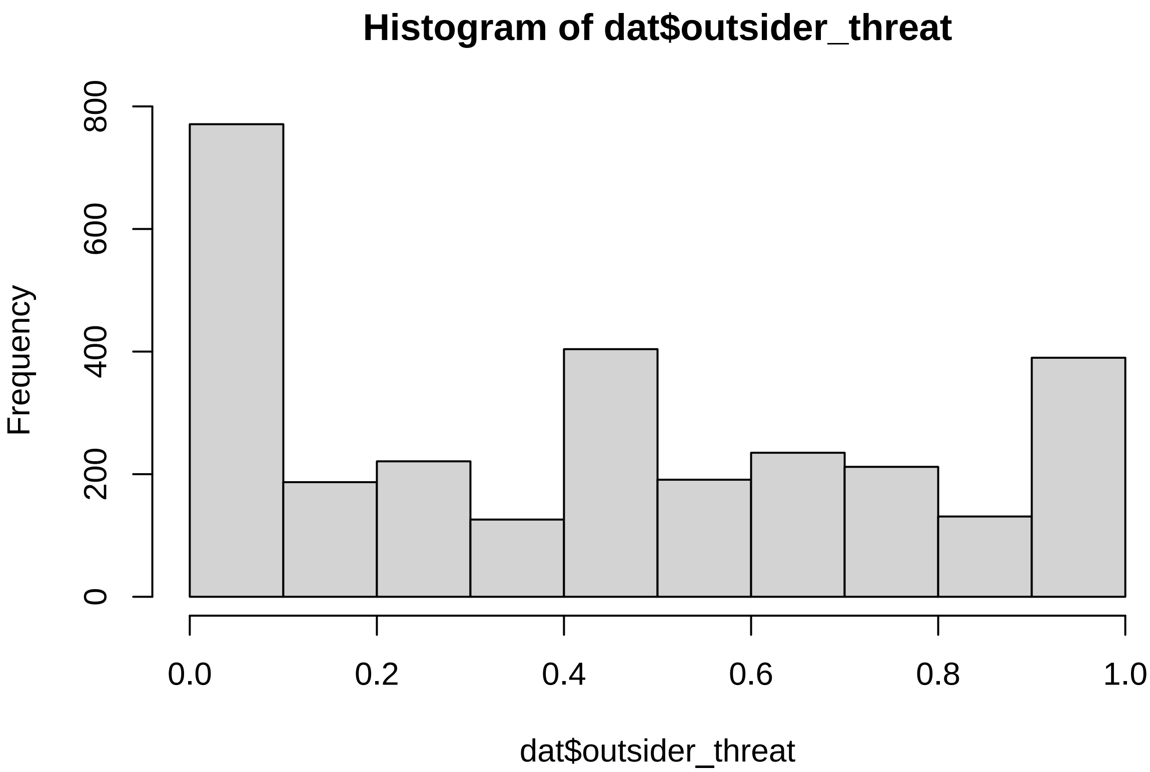
Table 5: Blind Source Preferences: Article Frequency Adjusted

	Dependent variable:	
	Trust Index	
	PiS Respondent	Dem. Oppo. Respondent
	(1)	(2)
TVP	0.053^{**}	-0.071^{***}
	(0.024)	(0.013)
Constant	0.566^{***}	0.613^{***}
	(0.018)	(0.010)
Observations	2,496	5,760
Adjusted R^2	0.008	0.018
Note:		*p<0.1; **p<0.05; ***p<0.01
		Trust index measured on 0 (low trust) to 1 (high trust)

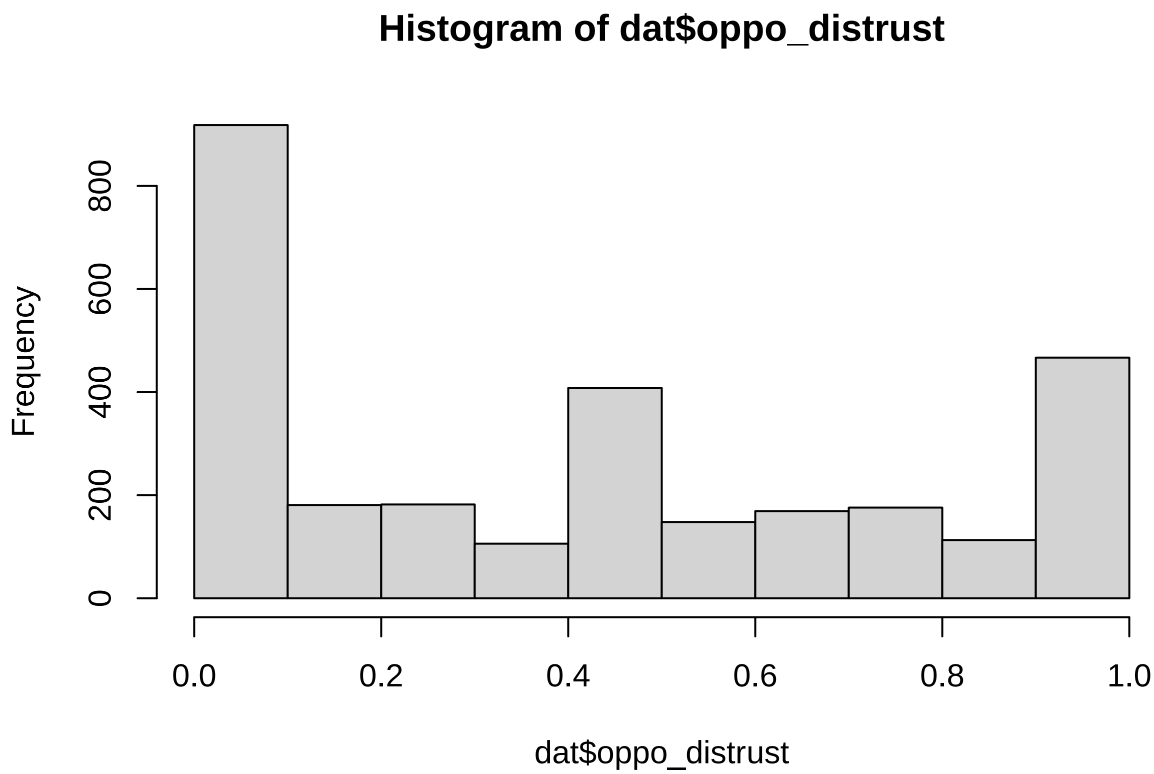
#Exposure effects on attitudes

#distribution of measures

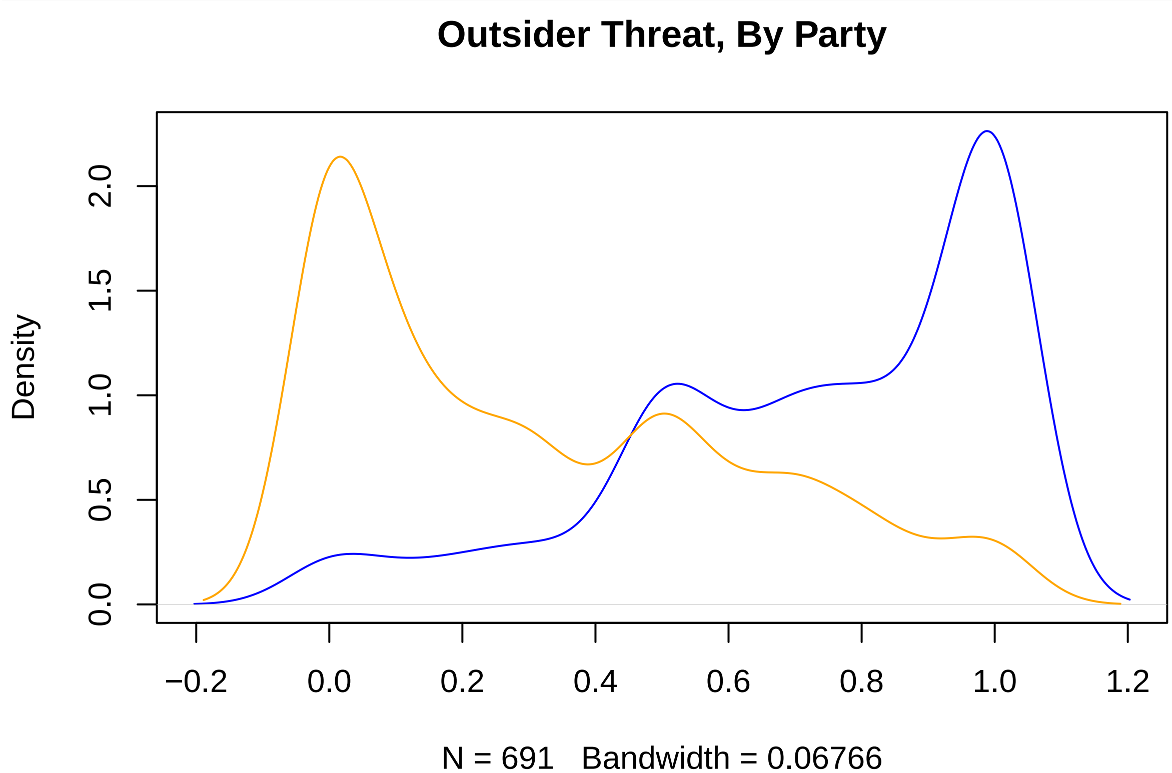
```
hist(dat$outsider_threat)
```



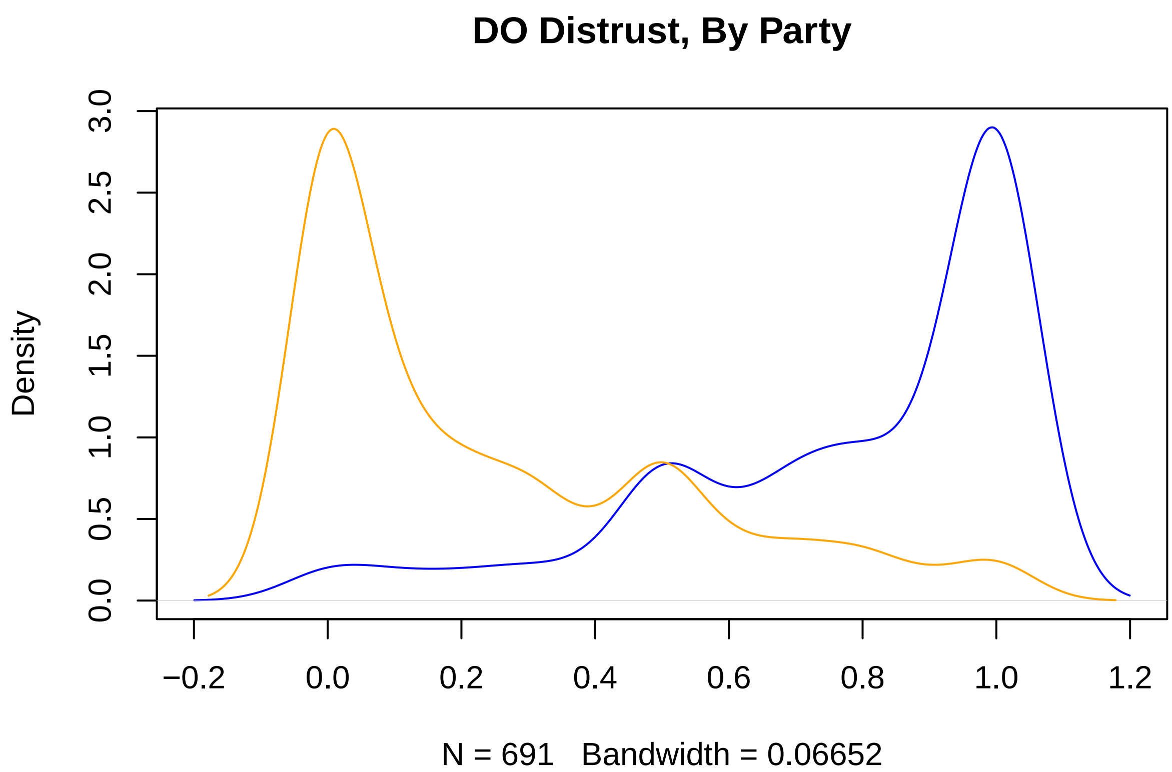

```
hist(dat$oppo_distrust)
```




```r
plot(density((dat$outsider_threat[dat$pro_pis == 1])), col="blue", main = "Outsider Threat, By Party")
lines(density((dat$outsider_threat[dat$anti_pis == 1])), col="orange")
```

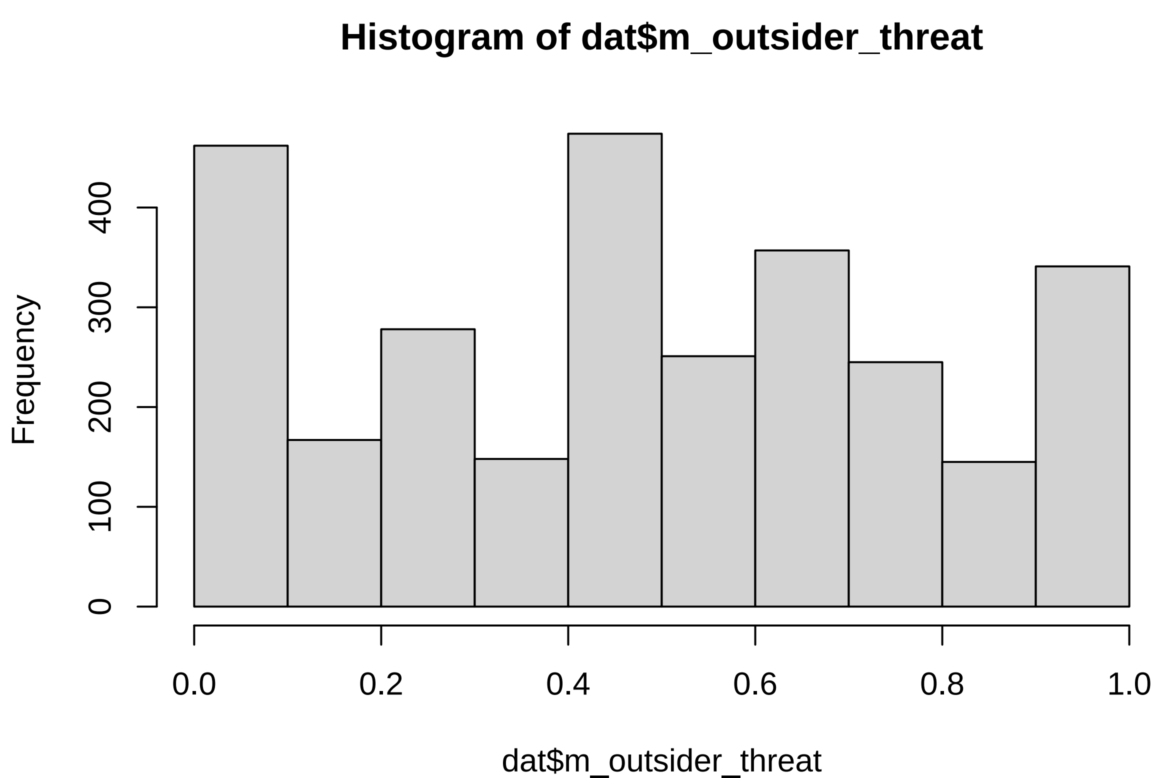



```r
plot(density((dat$oppo_distrust[dat$pro_pis == 1])), col="blue", main = "DO Distrust, By Party")
lines(density((dat$oppo_distrust[dat$anti_pis == 1])), col="orange")
```

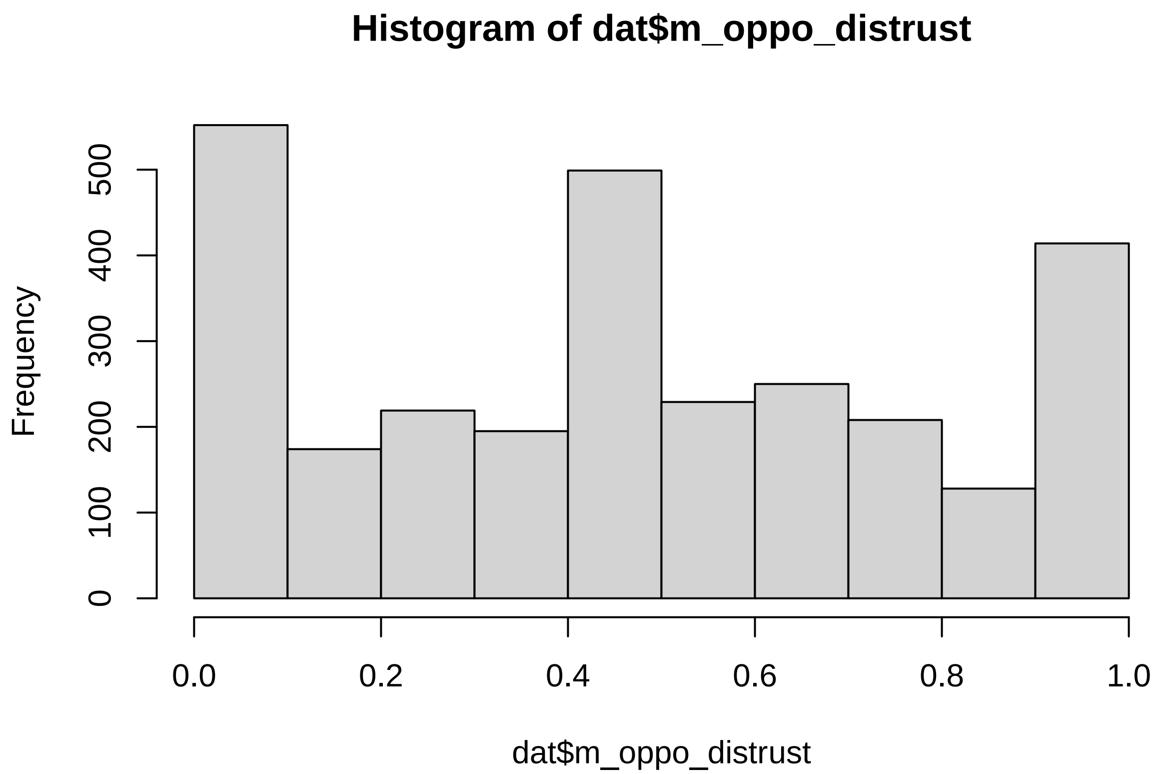


2nd Order Beliefs

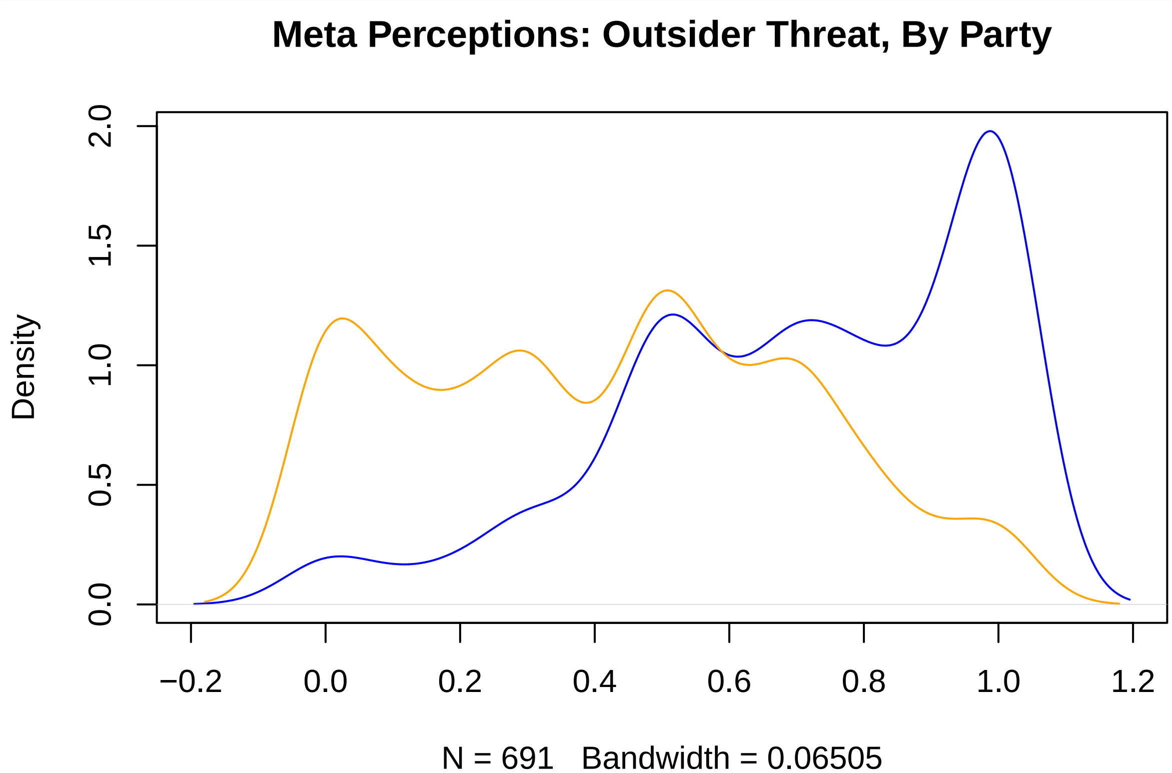
```
hist(dat$m_outsider_threat)
```



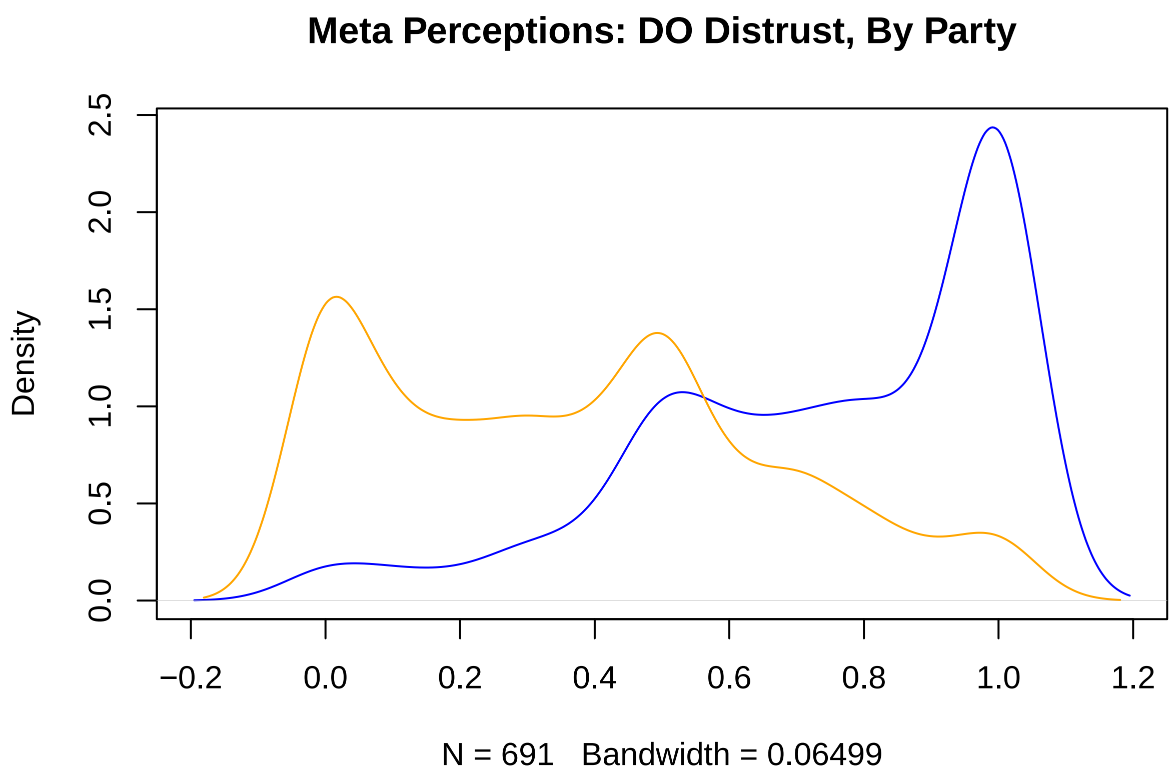

```
hist(dat$m_oppo_distrust)
```




```
plot(density((dat$m_outsider_threat[dat$pro_pis == 1])), col="blue", main = "Meta Perceptions: Outsider
lines(density((dat$m_outsider_threat[dat$anti_pis == 1])), col="orange")
```




```
plot(density((dat$m_oppo_distrust[dat$pro_pis == 1])), col="blue", main = "Meta Perceptions: DO Distrus
lines(density((dat$m_oppo_distrust[dat$anti_pis == 1])), col="orange")
```



ANOVAs for control and treatment conditions

Compares control conditions, i.e., pure control and TVN source with any topic (general, on-topic, and off-topic) both for blind and revealed conditions; and treatment conditions, i.e., TVP source with on-topic topics and both blind and revealed conditions.

```
# create condition var for ANOVA
dat <- dat %>%
  mutate(condition = as.factor(case_when(
    articles == 1 ~ (paste0(source, "_", treatment, "_", topic)),
    articles == 0 ~ "pure_control")),
    pro_pis = as.factor(pro_pis))
```

DV: outsider threat

```
aov_outsider_control <- aov(data = dat %>% filter(condition == "pure_control" | source == "tvn"),
               outsider_threat ~ condition * pro_pis)
summary(aov_outsider_control) # party main effect
```

Control conditions

```
##                      Df Sum Sq Mean Sq F value Pr(>F)
## condition             6   0.59    0.10   1.004  0.421
## pro_pis               1  33.55   33.55 345.376 <2e-16 ***
## condition:pro_pis     6   0.61    0.10   1.054  0.388
## Residuals          1571 152.60    0.10
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
tukey_outsider_control <- TukeyHSD(aov_outsider_control)
tukey_outsider_control_df <- as.data.frame(tukey_outsider_control[[3]]) %>%
  arrange(`p adj`)
#print(tukey_outsider_control_df) # differences only for pro-pis and anti-pis respondents
```

```r
aov_outsider_treatment <- aov(data = dat %>% filter(source == "tvp" & topic == "outsiders"),
                    outsider_threat ~ condition * pro_pis)
summary(aov_outsider_treatment) # party main effect only
```

Treatment conditions

```
##                     Df Sum Sq Mean Sq F value Pr(>F)
## condition            1   0.31   0.306   2.842 0.0926 .
## pro_pis              1  10.45  10.446  97.181 <2e-16 ***
## condition:pro_pis    1   0.00   0.000   0.001 0.9704
## Residuals          412  44.28   0.107
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
tukey_outsider_treatment <- TukeyHSD(aov_outsider_treatment)
tukey_outsider_treatment_df <- as.data.frame(tukey_outsider_treatment[[3]]) %>%
  arrange(`p adj`)
#print(tukey_outsider_treatment_df)
```

DV: democratic opposition distrust

```r
aov_opposition_control <- aov(data = dat %>% filter(condition == "pure_control" | source == "tvn"),
                         oppo_distrust ~ condition * pro_pis)
summary(aov_opposition_control) # party main effect
```

Control conditions

```
##                     Df Sum Sq Mean Sq F value Pr(>F)
## condition            6   0.93    0.15   1.555  0.157
## pro_pis              1  51.58   51.58 519.702 <2e-16 ***
## condition:pro_pis    6   0.43    0.07   0.717  0.636
## Residuals         1571 155.94    0.10
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
tukey_opposition_control <- TukeyHSD(aov_opposition_control)
tukey_opposition_control_df <- as.data.frame(tukey_opposition_control[[3]]) %>%
  arrange(`p adj`)
#print(tukey_opposition_control_df) # differences only for pro-pis and anti-pis respondents
```

```r
aov_opposition_treatment <- aov(data = dat %>% filter(source == "tvp" & topic == "politics"),
                         oppo_distrust ~ condition * pro_pis)
summary(aov_opposition_treatment) # party main effect only
```

Treatment conditions

```
##                     Df Sum Sq Mean Sq F value Pr(>F)
## condition            1   0.04   0.043   0.442  0.506
```

```
## pro_pis             1  16.11  16.107 165.428 <2e-16 ***
## condition:pro_pis   1   0.07   0.073   0.753  0.386
## Residuals         419  40.80   0.097
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_opposition_treatment <- TukeyHSD(aov_opposition_treatment)
tukey_opposition_treatment_df <- as.data.frame(tukey_opposition_treatment[[3]]) %>%
  arrange(`p adj`)
#print(tukey_opposition_treatment_df)
```

DV: meta-perception of outsider threat

```
aov_m_outsider_control <- aov(data = dat %>% filter(condition == "pure_control" | source == "tvn"),
                              m_outsider_threat ~ condition * pro_pis)
summary(aov_m_outsider_control) # party main effect
```

Control conditions

```
##                     Df Sum Sq Mean Sq F value Pr(>F)
## condition            6   0.49   0.081   0.971  0.444
## pro_pis              1  19.21  19.210 229.139 <2e-16 ***
## condition:pro_pis    6   0.68   0.114   1.357  0.229
## Residuals         1571 131.70   0.084
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_m_outsider_control <- TukeyHSD(aov_m_outsider_control)
tukey_m_outsider_control_df <- as.data.frame(tukey_m_outsider_control[[3]]) %>%
  arrange(`p adj`)
#print(tukey_m_outsider_control_df) # differences only for pro-pis and anti-pis respondents
```

```
aov_m_outsider_treatment <- aov(data = dat %>% filter(source == "tvp" & topic == "outsiders"),
                                m_outsider_threat ~ condition * pro_pis)
summary(aov_m_outsider_treatment) # party main effect only
```

Treatment conditions

```
##                    Df Sum Sq Mean Sq F value   Pr(>F)
## condition           1   0.54   0.538   6.136   0.0136 *
## pro_pis             1   5.04   5.042  57.473 2.29e-13 ***
## condition:pro_pis   1   0.01   0.013   0.151   0.6975
## Residuals         412  36.15   0.088
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_m_outsider_treatment <- TukeyHSD(aov_m_outsider_treatment)
tukey_m_outsider_treatment_df <- as.data.frame(tukey_m_outsider_treatment[[3]]) %>%
  arrange(`p adj`)
#print(tukey_m_outsider_treatment_df)
```

DV: meta-perception of democratic opposition distrust

```
aov_m_opposition_control <- aov(data = dat %>% filter(condition == "pure_control" | source == "tvn"),
                                m_oppo_distrust ~ condition * pro_pis)
summary(aov_m_opposition_control) # party main effect
```

Control conditions

```
##                     Df Sum Sq Mean Sq F value Pr(>F)
## condition            6   0.50   0.084   0.966  0.447
## pro_pis              1  28.20  28.204 324.745 <2e-16 ***
## condition:pro_pis    6   0.38   0.063   0.729  0.627
## Residuals         1571 136.44   0.087
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_m_opposition_control <- TukeyHSD(aov_m_opposition_control)
tukey_m_opposition_control_df <- as.data.frame(tukey_m_opposition_control[[3]]) %>%
  arrange(`p adj`)
#print(tukey_m_opposition_control_df) # differences only for pro-pis and anti-pis respondents
```

```
aov_m_opposition_treatment <- aov(data = dat %>% filter(source == "tvp" & topic == "politics"),
                                  m_oppo_distrust ~ condition * pro_pis)
summary(aov_m_opposition_treatment) # party main effect only
```

Treatment conditions

```
##                    Df Sum Sq Mean Sq F value Pr(>F)
## condition           1   0.22   0.221   2.656  0.104
## pro_pis             1   6.81   6.815  81.977 <2e-16 ***
## condition:pro_pis   1   0.07   0.070   0.845  0.358
## Residuals         419  34.83   0.083
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_m_opposition_treatment <- TukeyHSD(aov_m_opposition_treatment)
tukey_m_opposition_treatment_df <- as.data.frame(tukey_m_opposition_treatment[[3]]) %>%
  arrange(`p adj`)
#print(tukey_m_opposition_treatment_df)
```

DV: preference for inter-party cooperation

```
aov_coop_control <- aov(data = dat %>% filter(condition == "pure_control" | source == "tvn"),
                        party_coop ~ condition * pro_pis)
summary(aov_coop_control) # party main effect
```

Control conditions

```
##                     Df Sum Sq Mean Sq F value   Pr(>F)
## condition            6   0.33   0.056   0.573    0.752
## pro_pis              1   5.85   5.847  60.061 1.64e-14 ***
## condition:pro_pis    6   0.79   0.132   1.357    0.228
## Residuals         1571 152.94   0.097
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_coop_control <- TukeyHSD(aov_coop_control)
tukey_coop_control_df <- as.data.frame(tukey_coop_control[[3]]) %>%
  arrange(`p adj`)
#print(tukey_coop_control_df) # only 14 significant comparisons (for other analyzes abour 43)
```

```
aov_coop_treatment <- aov(data = dat %>% filter(source == "tvp" & topic == "politics"),
                          party_coop ~ condition * pro_pis)
summary(aov_coop_treatment) # party main effect only
```

Treatment conditions

```
##                    Df Sum Sq Mean Sq F value   Pr(>F)    
## condition           1   0.14  0.1436   1.490    0.223    
## pro_pis             1   2.00  1.9961  20.707 7.02e-06 ***
## condition:pro_pis   1   0.08  0.0839   0.871    0.351    
## Residuals         419  40.39  0.0964                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_coop_treatment <- TukeyHSD(aov_coop_treatment)
tukey_coop_treatment_df <- as.data.frame(tukey_coop_treatment[[3]]) %>%
  arrange(`p adj`)
#print(tukey_coop_treatment_df) # only 3 significant differences (for other analyzes 4)
```

Treatment effects

Based on the anova results, we are fitting separate models for PiS and Democratic Opposition respondents. We are defining the control condition as all respondents in a) the pure control condition (no articles seen) b) the TVN condition. We are defining the treatment condition as respondents in the TVP on-topic condition. Respondents in the tvp condition who were assigned to a different topic are left out. This is I think what we said we do in the PAP.

```
dat <- dat %>%
  mutate(outsider_treated = case_when(
    articles == 0 ~ 0,
    source == "tvn" ~ 0,
    source == "tvp" & topic == "outsiders" ~ 1
  ))


m_outsider_p <- lm(outsider_threat ~ outsider_treated, data = dat[dat$pro_pis == 1,])
m_outsider_d <- lm(outsider_threat ~ outsider_treated, data = dat[dat$anti_pis == 1,])
m_outsider_int <- lm(outsider_threat ~ outsider_treated * pro_pis, data = dat)


m_outsider_p_2 <- lm(m_outsider_threat ~ outsider_treated, data = dat[dat$pro_pis == 1,])
m_outsider_d_2 <- lm(m_outsider_threat ~ outsider_treated, data = dat[dat$anti_pis == 1,])
m_outsider_int_2 <- lm(m_outsider_threat ~ outsider_treated * pro_pis, data = dat)
```

```
dat <- dat %>%
  mutate(politics_treated = case_when(
    articles == 0 ~ 0,
    source == "tvn" ~ 0,
    source == "tvp" & topic == "politics" ~ 1
```

```
  ))

m_politics_p <- lm(oppo_distrust ~ politics_treated, data = dat[dat$pro_pis == 1,])
m_politics_d <- lm(oppo_distrust ~ politics_treated, data = dat[dat$anti_pis == 1,])
m_politics_int <- lm(oppo_distrust ~ politics_treated * pro_pis, data = dat)

#we then look at 2nd order beliefs
m_politics_p_2 <- lm(m_oppo_distrust ~ politics_treated, data = dat[dat$pro_pis == 1,])
m_politics_d_2 <- lm(m_oppo_distrust ~ politics_treated, data = dat[dat$anti_pis == 1,])
m_politics_int_2 <- lm(m_oppo_distrust ~ politics_treated * pro_pis, data = dat)
```

Table 6: Effect of Messaging on Attitudes: PiS Supporters

	Dependent variable:			
	Outsider Threat	Oppo. Distrust	Outsider Threat Meta	Oppo. Distrust Meta
	(1)	(2)	(3)	(4)
outsider_treated	0.006		0.004	
	(0.031)		(0.029)	
politics_treated		−0.007		−0.035
		(0.029)		(0.028)
Constant	0.734***	0.774***	0.726***	0.752***
	(0.015)	(0.014)	(0.014)	(0.014)
Observations	467	485	467	485
Adjusted R^2	−0.002	−0.002	−0.002	0.001
Note:				*p<0.1; **p<0.05; ***p<0.01
				Trust index measured on 0 (low trust) to 1 (high trust)

Table 7: Effect of Messaging on Attitudes: Dem. Oppo. Supporters

	Dependent variable:			
	Outsider Threat	Oppo. Distrust	Outsider Threat Meta	Oppo. Distrust Meta
	(1)	(2)	(3)	(4)
outsider_treated	−0.034		−0.003	
	(0.023)		(0.022)	
politics_treated		−0.016		−0.010
		(0.022)		(0.022)
Constant	0.324***	0.255***	0.421***	0.375***
	(0.010)	(0.010)	(0.010)	(0.010)
Observations	1,130	1,132	1,130	1,132
Adjusted R^2	0.001	−0.0004	−0.001	−0.001
Note:				*p<0.1; **p<0.05; ***p<0.01 Trust index measured on 0 (low trust) to 1 (high trust)

Table 8: Effect of Messaging on Attitudes: Interaction with Partisanship

	Dependent variable:			
	Outsider Threat	Oppo. Distrust	Outsider Threat Meta	Oppo. Distrust Meta
	(1)	(2)	(3)	(4)
outsider_treated	−0.018		0.007	
	(0.020)		(0.018)	
politics_treated		−0.010		0.001
		(0.020)		(0.019)
pro_pis1	0.346***	0.430***	0.262***	0.319***
	(0.019)	(0.019)	(0.017)	(0.017)
outsider_treated:pro_pis1	0.024		−0.003	
	(0.041)		(0.037)	
politics_treated:pro_pis1		0.003		−0.036
		(0.039)		(0.036)
Constant	0.388***	0.343***	0.464***	0.433***
	(0.009)	(0.009)	(0.008)	(0.008)
Observations	2,001	2,008	2,001	2,008
Adjusted R^2	0.182	0.255	0.125	0.169
Note:				*p<0.1; **p<0.05; ***p<0.01 Trust index measured on 0 (low trust) to 1 (high trust)