Pre-Analysis Plan for Interest Group Heuristics Survey Experiment

BLINDED

Background

This experiment replicates and extends work by [BLINDED] in 2013:

- That earlier work examined the extent to which interest group endorsements of Members of Congress provide useful information in helping constituents guess their representatives' votes on key issues, finding that they largely do not. In this experiment, we will replicate previous findings using additional interest groups and additional key votes.
- As well, it attempts to identify reasons for the unhelpfulness of interest group cues by testing whether respondents can identify which of two interest groups on opposing sides of an issue is the liberal organization.
- Finally, we address a follow-up question related to approval, and ask whether learning of interest group alignment increases support for one's member of congress.

Experimental Design

We plan to recruit a nationally representative sample to complete our online survey.

This experiment involves the following steps:

1. Identify a respondent's Member of Congress using their 9-digit zip code.
2. Gather respondent's preferences on about 10 policy issues.
3. If respondent is in the control group, show no treatment. If respondent is in the treatment group, show a randomly selected interest group rating for her MoC. These are the real interest group ratings for respondents' real MoCs.
4. Gather respondent's guesses about her MoC's preferences on the same policy issues.
5. If a respondent is in the treatment group, gather respondent's approval of her MoC.
6. Ask all respondents to identify which of two related interest groups supports a given policy.

Outcomes

We collect three outcomes.

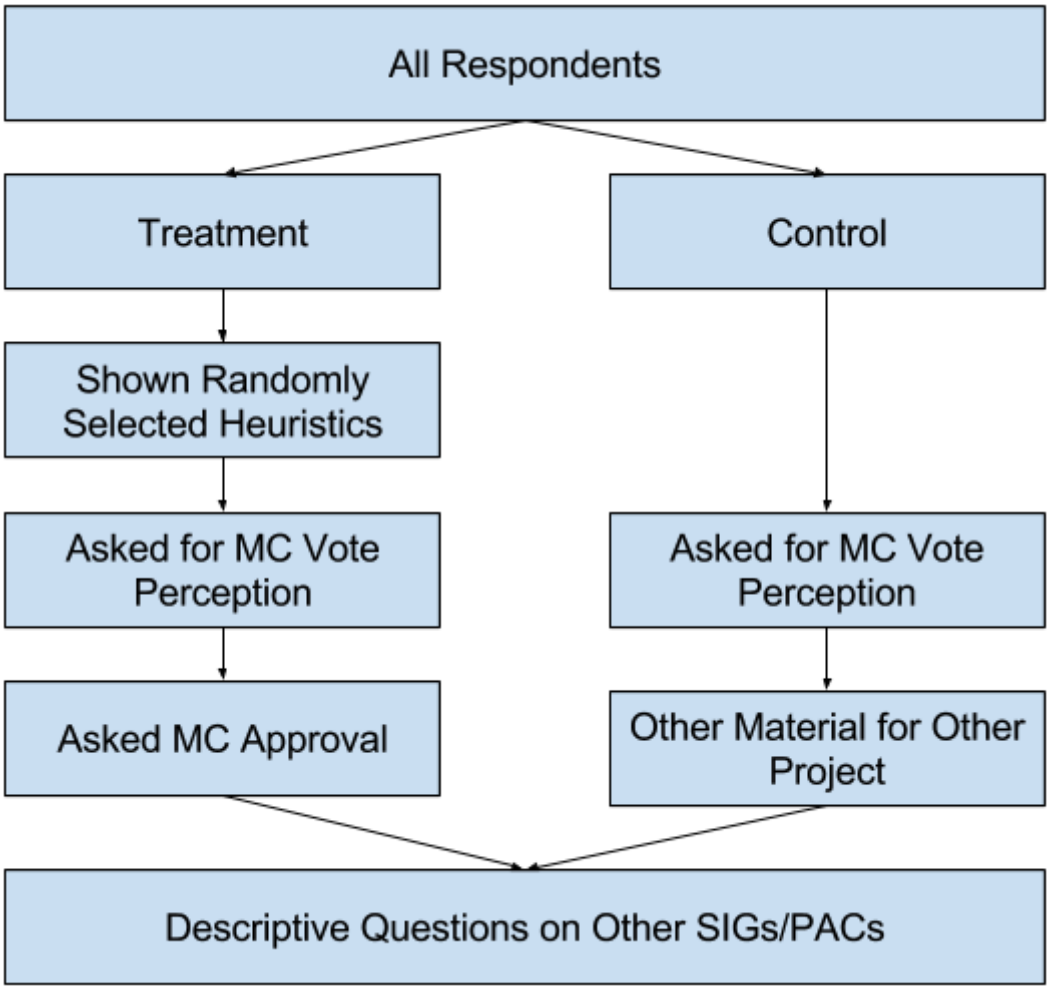
Outcomes in Experiment

- *mcratingscale* measures R's support for her Member of Congress
- *position_correct* indicates whether R correctly guessed her MoC's position on the policy

Descriptive Statistics

- *org_correct* indicates whether R successfully identified which interest groups supports a given policy; each R is shown two

This diagram shows how respondents will flow through the survey and when each outcome is collected.



Data In This PAP

This pre-analysis plan using a limited pilot of this design ran in November 2017. Given the limited sample size, we cannot use it to draw inferences about the results of the experiment to be completed.

```
dat <- read.csv('heuristics_cleaned.csv', stringsAsFactors = FALSE)

# Our control respondents were given a different experiment. The project identifier 'heu
ristics' distinguishes our pure control from our two endorsement-receiving groups.
dat$ratingsshown <- dat$project == 'heuristics'
```

30 respondents completed this pilot pre-survey. To write this PAP, we use this real pre-survey data. We will not use this pre-survey data for the paper.

Experiment: Effect of heuristics on correct perception of MC votes and MC approval

Effect of heuristics on correct perception of MC votes

Does showing the heuristic make people guess the MC's position more accurately on that or other issues?

To answer this question we reshape the dataset to the respondent by issue level. All respondents made guesses about their Member of Congress' position across approximately ten issues each. A randomly selected subset of respondents (the treatment group) were shown heuristics relevant to one randomly selected issue. There are therefore three groups we can compare:

1. How accurate were respondents in the treatment group about the issue on which they were shown a relevant heuristic?
2. How accurate were respondents in the treatment group about the issues on which they were *not* shown relevant heuristics?
3. How accurate were respondents in the control group about all the issues?

Comparing groups 1 and 3 reveal the 'direct effect' of heuristics. Comparing groups 2 and 3 reveal any 'spillover' effect of heuristics. Therefore, we will run a regression with a) an indicator for whether a respondent was shown a heuristic on this issue (i.e., group 1 above), identifying the *direct effect* of a heuristic; and b) an indicator for whether a respondent was shown a heuristic on a different issue (i.e., group 2 above), identifying the *indirect effect* on other issues. The control group will be the base category. We will also use issue fixed effects to increase precision.

```
# Reshape the data frame so the unit of analysis is the respondent * issue area
# To perform this reshape, I use two melt() calls then merge the two.

# A helper function to extract issue area from a variable name
substrRight <- function(x, n){
  substr(x, nchar(x)-n+1, nchar(x))
}

# This first melt call produces a respondent * issue area data frame with the responden
t's guess regarding their Congressmember's vote.
dat_t1 = reshape2::melt(dat[,c(19:35, 47)], id = "workerId")
dat_t1$variable = as.character(dat_t1$variable)
dat_t1$issue = sapply(dat_t1$variable, FUN=function(x) substrRight(x, 5))
dat_t1 = dat_t1[,c(1,3,4)]
colnames(dat_t1) = c("workerId", "guess", "issue")

# This second melt call produces a respondent * issue area data frame with their Congres
smember's actual vote.
dat_t2 = reshape2::melt(dat[,c(67:83, 47)], id = "workerId")
dat_t2$variable = as.character(dat_t2$variable)
dat_t2$issue = sapply(dat_t2$variable, FUN=function(x) substrRight(x, 5))
dat_t2 = dat_t2[,c(1,3,4)]
colnames(dat_t2) = c("workerId", "truth", "issue")

# Then we merge the two data frames together along with an identifier for which issue ar
ea the respondent received a SIG rating for
dat_t = merge(dat_t1, dat_t2)
dat_t = merge(dat_t, dat[,c("randomrating_house_vote_id", "workerId",
                            "project", "num_eligible_ratings", "survey_version")])

# Calculate the dependent variable: whether the respondent guessed the MoC's vote correc
tly on each issue.
dat_t$correct = dat_t$guess == dat_t$truth

dat_t = dat_t[!is.na(dat_t$correct),] # There should be no NAs here in the real data.

# Create variables indicating which treatment group each observation alls in
dat_t$indirect = dat_t$project == "heuristics" & dat_t$issue != as.character(dat_t$rando
mrating_house_vote_id)
dat_t$direct = dat_t$project == "heuristics" & dat_t$issue ==  as.character(dat_t$random
rating_house_vote_id)
dat_t$control =  dat_t$project != "heuristics"

# Regress whether the respondent guessed correctly on whether they were cued, with issue
area fixed effects and respondent-level clustered SEs.
# Respondents in the treatment group are only randomly assigned to one rating among the
 ratings that it makes sense to show them for their MoC (because the MC has both the rat
ing and the corresponding vote). Some MCs are missing different numbers than others, mea
ning that respondent's probability of assignment to each issue depends on how many other
issues there. E.g., if you got issue A and your MC has 3 other issues, your prob of assi
gnment is 1/4, but if your MC has 7 other issues, your prob of assignment is 1/8. So we
 need fixed effects for the number of issues to which you were assigned.
summary(miceadds::lm.cluster(data = dat_t,
```

```
                        formula = correct ~ indirect + direct +
                          factor(issue) + factor(num_eligible_ratings),
                        cluster = 'workerId'))
```

```
## R^2 = 0.18537
##
##                                  Estimate Std. Error    t value
## (Intercept)                    0.61518611 0.14897133  4.1295605
## indirectTRUE                   0.03507197 0.08724910  0.4019752
## directTRUE                     0.02041088 0.13740111  0.1485496
## factor(issue)56612             0.12947236 0.13794377  0.9385880
## factor(issue)58622             0.09022441 0.15765367  0.5722950
## factor(issue)58745             0.09022441 0.16437523  0.5488930
## factor(issue)59180            -0.22351944 0.18001315 -1.2416840
## factor(issue)59189             0.08900265 0.15139017  0.5879025
## factor(issue)59434            -0.07766401 0.17819378 -0.4358402
## factor(issue)59438             0.03567088 0.14332081  0.2488883
## factor(issue)59632            -0.02884819 0.17689180 -0.1630838
## factor(issue)59737            -0.26737983 0.21507530 -1.2431917
## factor(issue)60053             0.26030710 0.16618639  1.5663563
## factor(issue)60153            -0.07766401 0.17118095 -0.4536954
## factor(issue)60587             0.08900265 0.15969201  0.5573394
## factor(issue)60590             0.08642376 0.14190385  0.6090304
## factor(issue)60787             0.22011050 0.15142124  1.4536303
## factor(issue)61746            -0.09766513 0.17944912 -0.5442497
## factor(issue)61788             0.09022441 0.15754915  0.5726747
## factor(num_eligible_ratings)4 -0.18702638 0.05312124 -3.5207460
## factor(num_eligible_ratings)7 -0.02050322 0.08688560 -0.2359795
## factor(num_eligible_ratings)8 -0.01520914 0.09255921 -0.1643180
## factor(num_eligible_ratings)9  0.24246561 0.07358140  3.2952024
##                                   Pr(>|t|)
## (Intercept)                   3.634574e-05
## indirectTRUE                  6.877023e-01
## directTRUE                    8.819091e-01
## factor(issue)56612            3.479423e-01
## factor(issue)58622            5.671221e-01
## factor(issue)58745            5.830789e-01
## factor(issue)59180            2.143532e-01
## factor(issue)59189            5.565978e-01
## factor(issue)59434            6.629527e-01
## factor(issue)59438            8.034472e-01
## factor(issue)59632            8.704524e-01
## factor(issue)59737            2.137972e-01
## factor(issue)60053            1.172652e-01
## factor(issue)60153            6.500481e-01
## factor(issue)60587            5.772955e-01
## factor(issue)60590            5.425043e-01
## factor(issue)60787            1.460488e-01
## factor(issue)61746            5.862697e-01
## factor(issue)61788            5.668650e-01
## factor(num_eligible_ratings)4 4.303347e-04
## factor(num_eligible_ratings)7 8.134486e-01
## factor(num_eligible_ratings)8 8.694808e-01
## factor(num_eligible_ratings)9 9.835082e-04
```

```
# We will test whether there is a positive coefficient on direct, our main effect, or in direct, our spillover effect.
```

We also calculate a variable which indicates whether respondents understand that a MoC receiving a low grade from an interest group means that interest group's policy preferences do not align with the MoC's. This serves as a manipulation check: if we find that respondents do not adequately consider SIG endorsements when guessing their MoC's votes, we can use this mediating variable to rule out the possibility that they do not understand the meaning of an SIG endorsement.

```
# First, we calculate the mediating variable -- whether respondents understand what an interest group endorsement means regarding their Member of Congress.
dat$understandsrating_correct <- ifelse(((dat$randomrating_sig_rating > 50 & dat$understandrating == 1) |(dat$randomrating_sig_rating < 50 & dat$understandrating == 3)), 1, 0)
```

Effect of heuristics on MC approval

Here we address our second question: when people see a heuristic that suggests an MC agrees with them on an issue, do they like the MC more?

```
# Note: for this analysis, can't use respondents in the other project as a control group, since for the other project other information was shown before the MC approval question was asked. So we are just going to use within-subject variation among our treatment group.
dat <- filter(dat, project == 'heuristics')

# Our dependent variable: the respondents' favorability toward their Member of Congress
dat$mcratingscale
```

```
##  [1]  1.19682466 -0.43407833  0.01572741 -0.81655608  2.02910815
##  [6]  0.78068291  0.81434691  0.78068291  0.22123354 -0.81655608
## [11] -0.62531721 -0.81655608  0.01572741  0.39820516 -0.81655608
```

```
# Preliminary step: identify the issues where the liberal position is support and the is
sues where the liberal position is oppose
temp = data.frame(randomrating_sig_name = unique(dat$randomrating_sig_name))
temp$randomrating_sig_libcon = c(-1, -1, 1, 1, -1, 1, 1, -1, 1, -1)
dat = merge(dat, temp)
rm(temp)

# Identify how the respondent voted on the issue relating to the heuristic
# This variable is stored as bill_pref, and is coded as either -1 (liberal) or +1 (conse
rvative)
ownview_heuristic_var = paste0("ownview", dat$randomrating_house_vote_id)
idx = sapply(ownview_heuristic_var, FUN=function(x) which(colnames(dat)==x))
temp0 = c()
for(i in 1:length(idx)){
  temp0[i] = dat[i, idx[i]]
}
temp0[temp0==0] = -1

dat$bill_pref = temp0 * dat$randomrating_sig_libcon
rm(temp0)
rm(idx)
rm(vec)


dat$randomrating_sig_rating[dat$randomrating_sig_rating < 50] = -1
dat$randomrating_sig_rating[dat$randomrating_sig_rating >= 50] = 1
dat$heuristic_candpref = dat$randomrating_sig_rating * dat$randomrating_sig_libcon
dat$heuristic_candpref_binary = dat$heuristic_candpref > 0

# Define heuristicagree as whether the respondent's preference on an issue matches the c
andidate's preferences as indicated by an endorsement
# Note that the endorsement may be misleading! A candidate with a 90% grade from a SIG s
till votes against them 10% of the time.
dat$heuristicagree = dat$bill_pref == dat$heuristic_candpref
by(dat$heuristicagree, dat$mcratingscale, mean)
```

```
## dat$mcratingscale: -0.816556083439987
## [1] 0
## ------------------------------------------------------------
## dat$mcratingscale: -0.625317206839052
## [1] 0
## ------------------------------------------------------------
## dat$mcratingscale: -0.434078330238116
## [1] 0
## ------------------------------------------------------------
## dat$mcratingscale: 0.0157274068804401
## [1] 0.5
## ------------------------------------------------------------
## dat$mcratingscale: 0.2212335403044
## [1] 1
## ------------------------------------------------------------
## dat$mcratingscale: 0.398205160082311
## [1] 0
## ------------------------------------------------------------
## dat$mcratingscale: 0.780682913284182
## [1] 1
## ------------------------------------------------------------
## dat$mcratingscale: 0.814346905242525
## [1] 1
## ------------------------------------------------------------
## dat$mcratingscale: 1.1968246584444
## [1] 0
## ------------------------------------------------------------
## dat$mcratingscale: 2.02910814876482
## [1] 0
```

```
## Calculate the proportion of votes in which the respondent agrees with the member of C
ongress's votes
dat$rep_agree_proportion = sapply(1:nrow(dat), FUN=function(x) mean(dat[x,3:19]==dat[x,2
0:36], na.rm=T))

## Regress respondents' support for their member of Congress on the proportion of votes
 on which they agree, and whether the SIG heuristic suggests policy preference congruenc
e
summary(lm(mcratingscale ~ rep_agree_proportion + heuristicagree, dat))
```

```
##
## Call:
## lm(formula = mcratingscale ~ rep_agree_proportion + heuristicagree,
##     data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73533 -0.30369  0.09695  0.26610  0.58738
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -0.9315     0.1964  -4.742 0.000478 ***
## rep_agree_proportion   2.4655     0.4002   6.161 4.86e-05 ***
## heuristicagreeTRUE    -0.3746     0.2854  -1.312 0.213949
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4355 on 12 degrees of freedom
## Multiple R-squared:  0.7861, Adjusted R-squared:  0.7505
## F-statistic: 22.05 on 2 and 12 DF,  p-value: 9.578e-05
```

```
## If people are learning adequately, there will be a positive coefficient on heuristica
gree independent of allagree

# Secondary hypothesis: recent research (Russell 2017, "US Senators on Twitter: Asymmetr
ic Party Rhetoric in 140 Characters") suggests that Democrats care more about policy ali
gnment than Republicanss
# Test this hypothesis by adding PID to the above regression
by(INDICES = list(dat$heuristicagree, dat$pid), data=dat$mcratingscale, FUN = function(x
) mean(x, na.rm=T))
```

```
## : FALSE
## : 1
## [1] -0.8165561
## ------------------------------------------------------------
## : TRUE
## : 1
## [1] 0.7806829
## ------------------------------------------------------------
## : FALSE
## : 2
## [1] -0.2091755
## ------------------------------------------------------------
## : TRUE
## : 2
## [1] 0.01572741
## ------------------------------------------------------------
## : FALSE
## : 4
## [1] -0.6253172
## ------------------------------------------------------------
## : TRUE
## : 4
## [1] 0.8143469
## ------------------------------------------------------------
## : FALSE
## : 6
## [1] 0.01572741
## ------------------------------------------------------------
## : TRUE
## : 6
## [1] NA
## ------------------------------------------------------------
## : FALSE
## : 7
## [1] 2.029108
## ------------------------------------------------------------
## : TRUE
## : 7
## [1] NA
```

```
summary(lm(mcratingscale ~ rep_agree_proportion + heuristicagree + pid, dat))
```

```
##
## Call:
## lm(formula = mcratingscale ~ rep_agree_proportion + heuristicagree +
##     pid, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54745 -0.10752  0.03072  0.19735  0.34248
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.115139   0.186707  -5.973 0.000333 ***
## rep_agree_proportion  2.024957   0.381840   5.303 0.000726 ***
## heuristicagreeTRUE   -0.006571   0.288451  -0.023 0.982385
## pid                   0.110972   0.058056   1.911 0.092323 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3283 on 8 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.9057, Adjusted R-squared:  0.8703
## F-statistic: 25.61 on 3 and 8 DF,  p-value: 0.0001873
```

Do people correctly perceive PACs and SIGs?

Here we produce a table indicating the interest groups for which respondents most often correctly guess their positions on a related policy. We showed respondents pairs of interest groups and PACs on different sides of a policy issue and asked them to guess which was on which side.

```
dat <- read.csv('heuristics_cleaned.csv', stringsAsFactors = FALSE) # Reread the origina
l dataset since we subsetted it above
dat$orgs <- with(dat, paste0(org1, ' and ', org2))
by(dat$org_correct, dat$orgs, mean)
```

```
## dat$orgs: J Street PAC and Security PAC
## [1] 0.75
## ------------------------------------------------------------
## dat$orgs: Jobs, Opportunities, and Education PAC (Joe PAC) and Prosperity PAC
## [1] 0.5
## ------------------------------------------------------------
## dat$orgs: League of Conservation Voters and the Committee for the Preservation of Cap
italism
## [1] 0.75
## ------------------------------------------------------------
## dat$orgs: NARAL Pro-Choice America and Susan B Anthony List
## [1] 1
## ------------------------------------------------------------
## dat$orgs: National Committee to Preserve Social Security & Medicare PAC and FreedomWo
rks for America
## [1] 0.75
## ------------------------------------------------------------
## dat$orgs: Progressive Change Campaign and the Campaign for Working Families
## [1] 0.75
## ------------------------------------------------------------
## dat$orgs: San Franciscans for Good Government and Conservative Victory Fund
## [1] 0.75
## ------------------------------------------------------------
## dat$orgs: The Brady Campaign to Prevent Gun Violence and The National Rifle Associati
on
## [1] 1
```

Guess Correction

Many of our outcomes are whether R correctly guessed her MoC's policy preferences. This is subject to measurement error, since an uninformed respondent may correctly guess without knowing any real information. Therefore we perform statistical guess correction on our outcomes to estimate the treatment effect; otherwise we will underestimate it. Our guess correction formula is S = R - W / (k-1): corrected score is equal to the number of correct answers minus the number of incorrect answers, divided by the number of choices for each question minus one.