Original Replication File

# Overview

This document provides the code necessary to replicate the results of “Getting the Message? Choice, Self-Selection, and the Efficacy of Social Movement Arguments” It consist of the following sections

* **Setup** Sets up R environment:
  + Sets working directory and knitr options for display
  + Loads libraries (tidyverse packages, car, Hmisc, kableExtra)
  + Loads data (df\_mtg.rda, df\_qg.rda, power\_simulations.rda)
* **Functions** Defines a set of custom functions to:
  + Calculate treatment effects (diff\_fn(), acte\_fn(), cacte\_fn())
  + Display treatment effects (balance\_fn(), plot\_balance\_fn(), effects\_fn(), plot\_effects\_fn(), format\_ci\_fn(), table\_fn(),table\_app\_fn())
  + Conduct power simulations (data\_fn(), power\_fn(), sim\_power\_fn(), display\_power\_fn())
* **Main Figures** Produces Figures 1-6 as seen in text using functions defined above
* **Main Table** Produces Tables 1-2 as seen in text using functions defined above
* **Online Appendix** Produces tables and figures from Online Appendices C-F using functions defined above

Note: Each power simulations displayed in Figure 2 and Appendix C takes approximately 30-40 minutes to complete. The replication file loads the cached results of a round of power simulations. To conduct simulations, uncomment code.

# Setup

# Set working directory  
wd <- "C:/Users/hanna/Documents/GitHub/ps231b\_reproduction\_group4/original reproduction package"  
setwd(wd)  
  
# Load libraries  
  
# Uncomment to install packages  
# if(!require('tidyverse')){install.packages('tidyverse')}  
# if(!require('car')){install.packages('car')}  
# if(!require('Hmisc')){install.packages('Hmisc')}  
# if(!require('kableExtra')){install.packages('kableExtra')}  
# if(!require('sessioninfo')){install.packages('sessioninfo')}  
  
library(tidyverse)  
library(car)  
library(Hmisc)  
library(kableExtra)  
library(sessioninfo)  
library(formatR)  
  
# Set knitr output options  
knitr::opts\_chunk$set(message = F, warning = F, fig.height = 6, cache = T)  
options(knitr.table.format = "latex")  
  
# Load data  
load("df\_mtg.rda")  
load("df\_qg.rda")  
  
# Load results of power simulations  
load("power\_simulations.rda")  
  
  
# Display session info  
sessioninfo::session\_info()

## - Session info ---------------------------------------------------------------  
## setting value  
## version R version 4.1.3 (2022-03-10)  
## os Windows 10 x64 (build 19042)  
## system x86\_64, mingw32  
## ui RTerm  
## language (EN)  
## collate English\_United States.1252  
## ctype English\_United States.1252  
## tz America/Los\_Angeles  
## date 2022-04-09  
## pandoc 2.11.4 @ C:/Program Files/RStudio/bin/pandoc/ (via rmarkdown)  
##   
## - Packages -------------------------------------------------------------------  
## package \* version date (UTC) lib source  
## abind 1.4-5 2016-07-21 [1] CRAN (R 4.1.1)  
## assertthat 0.2.1 2019-03-21 [1] CRAN (R 4.1.1)  
## backports 1.4.1 2021-12-13 [1] CRAN (R 4.1.2)  
## base64enc 0.1-3 2015-07-28 [1] CRAN (R 4.1.0)  
## broom 0.7.12 2022-01-28 [1] CRAN (R 4.1.3)  
## car \* 3.0-12 2021-11-06 [1] CRAN (R 4.1.2)  
## carData \* 3.0-5 2022-01-06 [1] CRAN (R 4.1.2)  
## cellranger 1.1.0 2016-07-27 [1] CRAN (R 4.1.1)  
## checkmate 2.0.0 2020-02-06 [1] CRAN (R 4.1.2)  
## cli 3.1.0 2021-10-27 [1] CRAN (R 4.1.2)  
## cluster 2.1.3 2022-03-28 [1] CRAN (R 4.1.3)  
## colorspace 2.0-3 2022-02-21 [1] CRAN (R 4.1.3)  
## crayon 1.5.1 2022-03-26 [1] CRAN (R 4.1.3)  
## data.table 1.14.2 2021-09-27 [1] CRAN (R 4.1.2)  
## DBI 1.1.2 2021-12-20 [1] CRAN (R 4.1.3)  
## dbplyr 2.1.1 2021-04-06 [1] CRAN (R 4.1.1)  
## digest 0.6.29 2021-12-01 [1] CRAN (R 4.1.3)  
## dplyr \* 1.0.8 2022-02-08 [1] CRAN (R 4.1.3)  
## ellipsis 0.3.2 2021-04-29 [1] CRAN (R 4.1.1)  
## evaluate 0.15 2022-02-18 [1] CRAN (R 4.1.3)  
## fansi 1.0.3 2022-03-24 [1] CRAN (R 4.1.3)  
## fastmap 1.1.0 2021-01-25 [1] CRAN (R 4.1.1)  
## forcats \* 0.5.1 2021-01-27 [1] CRAN (R 4.1.1)  
## foreign 0.8-82 2022-01-13 [1] CRAN (R 4.1.2)  
## formatR \* 1.12 2022-03-31 [1] CRAN (R 4.1.3)  
## Formula \* 1.2-4 2020-10-16 [1] CRAN (R 4.1.1)  
## fs 1.5.2 2021-12-08 [1] CRAN (R 4.1.3)  
## generics 0.1.2 2022-01-31 [1] CRAN (R 4.1.3)  
## ggplot2 \* 3.3.5 2021-06-25 [1] CRAN (R 4.1.1)  
## glue 1.6.2 2022-02-24 [1] CRAN (R 4.1.3)  
## gridExtra 2.3 2017-09-09 [1] CRAN (R 4.1.2)  
## gtable 0.3.0 2019-03-25 [1] CRAN (R 4.1.1)  
## haven 2.4.3 2021-08-04 [1] CRAN (R 4.1.1)  
## Hmisc \* 4.6-0 2021-10-07 [1] CRAN (R 4.1.2)  
## hms 1.1.1 2021-09-26 [1] CRAN (R 4.1.2)  
## htmlTable 2.4.0 2022-01-04 [1] CRAN (R 4.1.2)  
## htmltools 0.5.2 2021-08-25 [1] CRAN (R 4.1.1)  
## htmlwidgets 1.5.4 2021-09-08 [1] CRAN (R 4.1.1)  
## httr 1.4.2 2020-07-20 [1] CRAN (R 4.1.1)  
## jpeg 0.1-9 2021-07-24 [1] CRAN (R 4.1.1)  
## jsonlite 1.8.0 2022-02-22 [1] CRAN (R 4.1.3)  
## kableExtra \* 1.3.4 2021-02-20 [1] CRAN (R 4.1.1)  
## knitr 1.38 2022-03-25 [1] CRAN (R 4.1.3)  
## lattice \* 0.20-45 2021-09-22 [1] CRAN (R 4.1.2)  
## latticeExtra 0.6-29 2019-12-19 [1] CRAN (R 4.1.2)  
## lifecycle 1.0.1 2021-09-24 [1] CRAN (R 4.1.2)  
## lubridate 1.8.0 2021-10-07 [1] CRAN (R 4.1.2)  
## magrittr 2.0.3 2022-03-30 [1] CRAN (R 4.1.3)  
## Matrix 1.4-1 2022-03-23 [1] CRAN (R 4.1.3)  
## modelr 0.1.8 2020-05-19 [1] CRAN (R 4.1.1)  
## munsell 0.5.0 2018-06-12 [1] CRAN (R 4.1.1)  
## nnet 7.3-17 2022-01-13 [1] CRAN (R 4.1.3)  
## pillar 1.7.0 2022-02-01 [1] CRAN (R 4.1.3)  
## pkgconfig 2.0.3 2019-09-22 [1] CRAN (R 4.1.1)  
## png 0.1-7 2013-12-03 [1] CRAN (R 4.1.1)  
## purrr \* 0.3.4 2020-04-17 [1] CRAN (R 4.1.1)  
## R6 2.5.1 2021-08-19 [1] CRAN (R 4.1.1)  
## RColorBrewer 1.1-3 2022-04-03 [1] CRAN (R 4.1.3)  
## readr \* 2.1.2 2022-01-30 [1] CRAN (R 4.1.3)  
## readxl 1.4.0 2022-03-28 [1] CRAN (R 4.1.3)  
## reprex 2.0.1 2021-08-05 [1] CRAN (R 4.1.1)  
## rlang 1.0.2 2022-03-04 [1] CRAN (R 4.1.3)  
## rmarkdown 2.13 2022-03-10 [1] CRAN (R 4.1.3)  
## rpart 4.1.16 2022-01-24 [1] CRAN (R 4.1.3)  
## rstudioapi 0.13 2020-11-12 [1] CRAN (R 4.1.1)  
## rvest 1.0.2 2021-10-16 [1] CRAN (R 4.1.2)  
## scales 1.1.1 2020-05-11 [1] CRAN (R 4.1.1)  
## sessioninfo \* 1.2.2 2021-12-06 [1] CRAN (R 4.1.2)  
## stringi 1.7.6 2021-11-29 [1] CRAN (R 4.1.2)  
## stringr \* 1.4.0 2019-02-10 [1] CRAN (R 4.1.1)  
## survival \* 3.3-1 2022-03-03 [1] CRAN (R 4.1.3)  
## svglite 2.1.0 2022-02-03 [1] CRAN (R 4.1.3)  
## systemfonts 1.0.4 2022-02-11 [1] CRAN (R 4.1.3)  
## tibble \* 3.1.6 2021-11-07 [1] CRAN (R 4.1.3)  
## tidyr \* 1.2.0 2022-02-01 [1] CRAN (R 4.1.3)  
## tidyselect 1.1.2 2022-02-21 [1] CRAN (R 4.1.3)  
## tidyverse \* 1.3.1 2021-04-15 [1] CRAN (R 4.1.1)  
## tzdb 0.3.0 2022-03-28 [1] CRAN (R 4.1.3)  
## utf8 1.2.2 2021-07-24 [1] CRAN (R 4.1.1)  
## vctrs 0.4.0 2022-03-30 [1] CRAN (R 4.1.3)  
## viridisLite 0.4.0 2021-04-13 [1] CRAN (R 4.1.1)  
## webshot 0.5.2 2019-11-22 [1] CRAN (R 4.1.1)  
## withr 2.5.0 2022-03-03 [1] CRAN (R 4.1.3)  
## xfun 0.30 2022-03-02 [1] CRAN (R 4.1.3)  
## xml2 1.3.3 2021-11-30 [1] CRAN (R 4.1.2)  
## yaml 2.3.5 2022-02-21 [1] CRAN (R 4.1.2)  
##   
## [1] C:/Users/hanna/Documents/R/win-library/4.1  
## [2] C:/Program Files/R/R-4.1.3/library  
##   
## ------------------------------------------------------------------------------

# Functions

## Functions to Calculate Treatment effects

* diff\_fn(): Estimate differences in means

# Difference in Means Function  
diff\_fn <- function(the\_data, dv1="Y",c,weights=F,...){  
 # REQUIRES  
 require(Hmisc)  
 # INPUTS:  
 # the\_data: data frame  
 # dv1: outcome  
 # c: object containing names of treatment conditions  
 # weights: boolean indicating whether to calculated weighted ATE  
 # OUTPUTS:  
 # result: vector containing Difference in Means, SE, 95% ci, 90%ci, and p-value  
  
 tmp <- as.data.frame(the\_data[the\_data$treatment%in%c, ])  
   
 if(weights==F){  
 mu1 <- with(tmp, mean(tmp[treatment==c[1], dv1],na.rm=T))  
 mu2 <- with(tmp, mean(tmp[treatment==c[2], dv1],na.rm=T))  
 sd1 <- with(tmp, sd(tmp[treatment==c[1], dv1],na.rm=T))  
 sd2 <- with(tmp, sd(tmp[treatment==c[2], dv1],na.rm=T))  
 }  
 if(weights==T){  
 mu1 <- with(tmp, Hmisc::wtd.mean(  
 tmp[treatment==c[1], dv1],na.rm=T,weights=tmp[treatment==c[1],"weights"])  
 )  
 mu2 <- with(tmp, Hmisc::wtd.mean(  
 tmp[treatment==c[2], dv1],na.rm=T,weights=tmp[treatment==c[2],"weights"])  
 )  
 sd1 <- sqrt(with(tmp, Hmisc::wtd.var(  
 tmp[treatment==c[1], dv1],na.rm=T,weights = tmp[treatment==c[1],"weights"])  
 ))  
 sd2 <- sqrt(with(tmp, Hmisc::wtd.var(  
 tmp[treatment==c[2], dv1],na.rm=T,weights = tmp[treatment==c[2],"weights"])  
 ))  
 }  
 # Calculate Difference  
 diff <- mu2-mu1  
   
 # Calculate N  
 n1 <- with(tmp, sum(!is.na(tmp[treatment==c[1], dv1])\*tmp[treatment==c[1],"weights"]))  
 n2 <- with(tmp, sum(!is.na(tmp[treatment==c[2], dv1])\*tmp[treatment==c[2],"weights"]))  
   
 # SE of Difference  
 se <- sqrt( sd1^2/n1 + sd2^2/n2)  
   
 # Degrees of Freedom  
 the\_df <- (sd1^2/n1+sd2^2/n2)^2/((sd1^4)/(n1^2\*(n1-1))+ (sd2^4)/(n2^2\*(n2-1)))  
   
 # 95% CI  
 ll <- diff - qt(.975,the\_df)\*se  
 ul <- diff + qt(.975,the\_df)\*se  
 # 90% CI  
 ll90 <- diff - qt(.95,the\_df)\*se  
 ul90 <- diff + qt(.95,the\_df)\*se  
 # t-stat  
 stat <- diff/se  
 # p-value  
 pval = 2 \* pt(-abs(stat),the\_df)  
 # Combine results  
 results <- c(Difference = diff, SE = se, ll = ll, ul = ul,  
 ll90=ll90,ul90=ul90, pval = pval)  
 return(results)  
}

* acte\_fn(): Estimate ACTEs using delta method

# ACTE function   
acte\_fn <- function(dat,dv2="Y",z,w,...){  
 # REQUIRES  
 require(car)  
 require(Hmisc)  
   
 # INPUTS:  
 # dat: data frame  
 # dv2: outcome  
 # z: object containing treatment groups to calculate ACTE-Select or ACTE-Avoid  
 # w: weight argument passed to diff\_fn  
 # OUTPUTS:  
 # result: vector containing ATE, SE, 95% ci, 90%ci, and p-value  
   
 df <- dat  
 N <- dim(df)[1]  
 # N assigned to random assignment  
 n\_exp <- sum(df$C=="Experiment")  
 # N assigned to choice  
 n\_choice <- sum(df$C == "Choice")  
 # N avoiding treatment  
 n\_ch\_a <- sum(df$C=="Choice" & df$avoid01==1)  
 # N selecting treatment  
 n\_select <- sum(df$C=="Choice" & df$avoid01==0, na.rm=T)  
 # N avoiding treatment  
 n\_avoid <- sum(df$C=="Choice" & df$avoid01==1,na.rm = T)  
 # N avoiding treatment who receive no treatment   
 n\_control <- sum(df$D\_ch == "Control",na.rm = T)  
   
 # Weights to reflect fact   
 df$weights <- rep(1,N)  
 df$weights[df$C=="Choice" & df$avoid01==0 & df$D\_ch == "Control"] <-1/(n\_select/n\_choice)  
 df$weights[df$C=="Choice" & df$avoid01==1 & df$D\_ch == "Control"] <-1/(n\_control/n\_avoid)   
   
   
 # Calculate ACTE using Delta Method  
 #ACTE-Select: c = c\_acte\_s = c("Control","Selection")  
 #ACTE-Avoid: c = c\_acte\_a = c("Selection","Treatment")  
   
 c\_acte\_s <- c("Control","Selection")  
 c\_acte\_a <- c("Selection","Treatment")  
   
 tmp <- diff\_fn(the\_data=df, dv1=dv2,c=z, weights=w)  
 x <- as.numeric(tmp["Difference"])  
 se\_x <- as.numeric(tmp["SE"])  
 # ACTE-Select  
 if(z[1]=="Control"){  
 y <- summary(lm(select01~1,df[df$C=="Choice",]))$coef[1,1]  
 se\_y <- summary(lm(select01~1,df[df$C=="Choice",]))$coef[1,2]}  
 # ACTE-Avoid  
 if(z[2]=="Treatment"){  
 y <- summary(lm(avoid01~1,df[df$C=="Choice",]))$coef[1,1]  
 se\_y <- summary(lm(avoid01~1,df[df$C=="Choice",]))$coef[1,2]}  
 mvec <- c(x=x, y= y)  
 V <- diag(c(se\_x,se\_y)^2)  
 est <- car::deltaMethod(mvec,"x/y",V,level=.95)  
 est90 <- car::deltaMethod(mvec,"x/y",V,level=.90)  
 stat <- as.numeric(est[1])/as.numeric(est[2])  
   
 # Return results  
 results <- c(Difference = as.numeric(est[1]),   
 SE = as.numeric(est[2]),   
 ll = as.numeric(est[3]), ul = as.numeric(est[4]),   
 ll90 = as.numeric(est90[3]), ul90 = as.numeric(est90[4]),  
 pval = 2 \* pnorm(-abs(stat)))  
 return(results)  
}

* cacte\_fn(): Estimate CACTEs

# CACTE function   
cacte\_fn <- function(d, dv3="Y", z, w2=F){  
 # INPUTS:  
 # d: data frame  
 # dv3: outcome  
 # z: object containing treatment groups to calculate ACTE-Select or ACTE-Avoid  
 # w2: weight argument passed to diff\_fn  
 # OUTPUTS:  
 # result: Matrix containing Female and Male CACTE, SE, 95% ci, 90%ci, and p-value  
   
 tmp <- as.data.frame(d[d$avoid01 == 1, ])  
 tmp$treatment <- tmp$D\_ch  
  
 # CACTE- Female  
 cacte\_female <- diff\_fn(tmp, dv1=dv3,c=c("Control","Female"),weights=w2)  
   
 # CACTE Male  
 cacte\_male <- diff\_fn(tmp, dv1=dv3, c=c("Control","Male"),weights=w2)  
 results <- rbind(cacte\_female,cacte\_male)  
 return(results)  
   
}

## Functions to display results

* balance\_function(): Function to calculate covariate differences in respondents selecting and avoiding treatment

balance\_fn <- function(the\_data, dv1="Y",c,weights=F,...){  
 # INPUTS  
 # the\_data: data frame  
 # dv1: Variable to calculate difference in means  
 # c: which group to compare  
 # weights: Calcultate weighted differences (T/F)  
 # OUTPUTS  
 # result: summary stats of difference in means  
  
 tmp <- as.data.frame(the\_data[the\_data$balance%in%c, ])  
 if(weights==F){  
 mu1 <- with(tmp, mean(tmp[balance==c[1], dv1],na.rm=T))  
 mu2 <- with(tmp, mean(tmp[balance==c[2], dv1],na.rm=T))  
 }  
 if(weights==T){  
 mu1 <- with(tmp, Hmisc::wtd.mean(  
 tmp[balance==c[1], dv1],na.rm=T,weights=tmp[balance==c[1],"weights"])  
 )  
 mu2 <- with(tmp, Hmisc::wtd.mean(  
 tmp[balance==c[2], dv1],na.rm=T,weights=tmp[balance==c[2],"weights"])  
 )  
 }  
 diff <- mu1-mu2  
 if(weights==F){  
 sd1 <- with(tmp, sd(tmp[balance==c[1], dv1],na.rm=T))  
 sd2 <- with(tmp, sd(tmp[balance==c[2], dv1],na.rm=T))  
 }  
 if(weights==T){  
 sd1 <- sqrt(with(tmp, Hmisc::wtd.var(  
 tmp[balance==c[1], dv1],na.rm=T,weights = tmp[balance==c[1],"weights"])  
 ))  
 sd2 <- sqrt(with(tmp, Hmisc::wtd.var(  
 tmp[balance==c[2], dv1],na.rm=T,weights = tmp[balance==c[2],"weights"])  
 ))  
 }  
 n1 <- with(tmp, sum(!is.na(tmp[balance==c[1], dv1])\*tmp[balance==c[1],"weights"]))  
 n2 <- with(tmp, sum(!is.na(tmp[balance==c[2], dv1])\*tmp[balance==c[2],"weights"]))  
 se <- sqrt( sd1^2/n1 + sd2^2/n2)  
 the\_df <- (sd1^2/n1+sd2^2/n2)^2/((sd1^4)/(n1^2\*(n1-1))+ (sd2^4)/(n2^2\*(n2-1)))  
 ll <- diff - qt(.975,the\_df)\*se  
 ul <- diff + qt(.975,the\_df)\*se  
 ll90 <- diff - qt(.95,the\_df)\*se  
 ul90 <- diff + qt(.95,the\_df)\*se  
 stat <- diff/se  
 pval = 2 \* pt(-abs(stat),the\_df)  
 result <- c(Mu1= mu1, Mu2 = mu2, Difference = diff, SE = se, ll = ll, ul = ul,  
 ll90 = ll90, ul90 = ul90, pval = pval, N1=n1,N2=n2)  
 return(result)  
}

* plot\_balance\_function(): Wrapper function to display results of balance\_function():

plot\_balance\_fn <- function(d,  
 bal\_labs = c("Female", "Non-white", "Education",   
 "Income", "PID", "Ideology",   
 "MeToo Familiarity",   
 "Specific Support", "General Support"),  
 comparison = c("Select Treatment", "Avoid Treatment")  
 ){  
 # INPUTS  
 # d: data frame  
 # bal\_labs: Covariate labels  
 # comparison: which group to compare  
 # OUTPUTS  
 # fig: ggplot of comparisons  
   
   
 # Descriptives Differences in Selecting Treatment - Overall  
 bal\_gen <- data.frame(  
 rbind(  
 balance\_fn(the\_data = d, "gender", comparison),  
 balance\_fn(the\_data = d, "non\_white", comparison),  
 balance\_fn(the\_data = d, "education", comparison),  
 balance\_fn(the\_data = d, "income", comparison),  
 balance\_fn(the\_data = d, "pid", comparison),  
 balance\_fn(the\_data = d, "ideo", comparison),  
 balance\_fn(the\_data = d, "fam\_movement", comparison),  
 balance\_fn(the\_data = d, "dv\_pca\_metoo", comparison),  
 balance\_fn(the\_data = d, "dv\_pca\_general", comparison)  
 ))  
 bal\_gen$Covariate <- factor(bal\_labs, levels=rev(bal\_labs))  
 rownames(bal\_gen) <- bal\_gen$Covariate  
   
 # Descriptives Differences in Selecting Treatment - Men  
 bal\_gen\_male <- data.frame(  
 rbind(  
 balance\_fn(the\_data = d[d$gender==0,],   
 "gender", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "non\_white", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "education", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "income", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "pid", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "ideo", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "fam\_movement", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "dv\_pca\_metoo", comparison),  
 balance\_fn(the\_data = d[d$gender==0,],   
 "dv\_pca\_general", comparison)  
 ))  
 bal\_gen\_male$Covariate <- factor(bal\_labs, levels=rev(bal\_labs))  
 rownames(bal\_gen\_male) <- bal\_gen\_male$Covariate  
 bal\_gen\_male[1,3] <-NA  
   
   
 # Descriptives Differences in Selecting Treatment - Women  
 bal\_gen\_female <- data.frame(  
 rbind(  
 balance\_fn(the\_data = d[d$gender==1,], "gender", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "non\_white", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "education", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "income", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "pid", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "ideo", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "fam\_movement", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "dv\_pca\_metoo", comparison),  
 balance\_fn(the\_data = d[d$gender==1,], "dv\_pca\_general", comparison)  
 ))  
 bal\_gen\_female$Covariate <- factor(bal\_labs, levels=rev(bal\_labs))  
 rownames(bal\_gen\_female) <- bal\_gen\_female$Covariate  
 bal\_gen\_female[1,3] <-NA  
   
 # Descriptives Differences in Selecting Treatment - Democrats  
 bal\_gen\_dem <- data.frame(  
 rbind(  
 balance\_fn(the\_data = d[d$pid<4,], "gender", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "non\_white", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "education", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "income", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "pid", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "ideo", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "fam\_movement", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "dv\_pca\_metoo", comparison),  
 balance\_fn(the\_data = d[d$pid<4,], "dv\_pca\_general", comparison)  
 ))  
 bal\_gen\_dem$Covariate <- factor(bal\_labs, levels=rev(bal\_labs))  
 rownames(bal\_gen\_dem) <- bal\_gen\_dem$Covariate  
   
 # Descriptives Differences in Selecting Treatment - Republicans  
 bal\_gen\_rep <- data.frame(  
 rbind(  
 balance\_fn(the\_data = d[d$pid>4,], "gender", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "non\_white", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "education", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "income", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "pid", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "ideo", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "fam\_movement", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "dv\_pca\_metoo", comparison),  
 balance\_fn(the\_data = d[d$pid>4,], "dv\_pca\_general", comparison)  
 ))  
 bal\_gen\_rep$Covariate <- factor(bal\_labs, levels=rev(bal\_labs))  
 rownames(bal\_gen\_rep) <- bal\_gen\_rep$Covariate  
   
 # Create data frame for plotting  
 fig\_df <- rbind(  
 data.frame(  
 bal\_gen,  
 Group = "Overall",  
 Type = "Overall"  
 ),  
 data.frame(  
 bal\_gen\_male,  
 Group = "Men",  
 Type = "By Gender"  
 ),  
 data.frame(  
 bal\_gen\_female,  
 Group = "Women",  
 Type = "By Gender"  
 ),  
 data.frame(  
 bal\_gen\_rep,  
 Group = "Republicans",  
 Type = "By Partisanship"  
 ),  
 data.frame(  
 bal\_gen\_female,  
 Group = "Democrats",  
 Type = "By Partisanship"  
 )  
 )  
   
 # Set labels as factors for plotting  
 fig\_df$Type <- factor(fig\_df$Type, levels = unique(fig\_df$Type))  
 fig\_df$Group <- factor(fig\_df$Group , levels = unique(fig\_df$Group ))  
  
 # Create Figure   
 fig <- fig\_df %>%   
 ggplot(  
 aes(Covariate, Difference, col = Group, shape = Group)  
 )+  
 geom\_hline(  
 yintercept = 0, linetype = "dashed",alpha = .5  
 )+  
 facet\_wrap(  
 ~Type, ncol = 3, drop = T)+  
 geom\_point(  
 aes(shape=Group),  
 position = position\_dodge(width = .5), size=2  
 )+  
 geom\_linerange(  
 aes(ymin=ll,ymax=ul),  
 size=.3,  
 position = position\_dodge(width = .5))+  
 geom\_linerange(  
 aes(ymin=ll90,ymax=ul90),  
 size=.6,  
 position = position\_dodge(width = .5))+  
 theme(  
 axis.text.x = element\_text(angle = 0, hjust = 1)  
 )+  
 ylab("Difference\n(Treatment Selectors - Treatment Avoiders)")+  
 coord\_flip()+  
 theme\_bw()+  
 scale\_color\_grey(start = 0, end = .75)+  
 scale\_shape\_manual(values = c(16, 17, 15, 23, 4))+  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom"  
 )  
   
 # Display Figure   
 fig  
   
}

* effects\_fn(): Wrapper function to calculated ATE, ACTEs, CACTEs, and CATEs

effects\_fn <- function(the\_dat,  
 the\_dv,  
 the\_lab = c("ATE",  
 "ACTE:","Select Treatment","Avoid Treatment",  
 "CACTE:","Female Treatment","Male Treatment",  
 "CATE:",  
 "White","Non-White",  
 "Male","Female",  
 "Democrat","Independent","Republican",  
 "Liberal","Moderate","Conservative",  
 "College Degree ","No Degree",  
 "Familiar","Unfamiliar"  
 ),...){  
 # INPUTS:  
 # the\_dat: data frame  
 # dv1: outcome  
 # the\_lab: object containing row names  
 # OUTPUTS:  
 # result: dataframe containing ATE, ACTES, CACTEs, CATE  
   
   
 # Calculate ATE, ACTE, CACTEs  
 tmp <- rbind(  
 diff\_fn(the\_dat, dv1=the\_dv,c("Control","Treatment")),  
 rep(NA,7),  
 acte\_fn(the\_dat, dv2=the\_dv,c("Control","Selection"),w = T),  
 acte\_fn(the\_dat, dv2=the\_dv,c("Selection","Treatment"),w = T),  
 rep(NA,7),  
 cacte\_fn(the\_dat, dv3=the\_dv),  
 rep(NA,7),  
 diff\_fn(the\_data = the\_dat[the\_dat$non\_white==0,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$non\_white==1,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$gender==0,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$gender==1,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$pid<4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$pid==4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$pid>4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$ideo<4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$ideo==4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$ideo>4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$education>4,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$education<5,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$fam\_movement>2,],the\_dv,c("Control","Treatment")),  
 diff\_fn(the\_data = the\_dat[the\_dat$fam\_movement<3,],the\_dv,c("Control","Treatment"))  
 )  
  
 results <- data.frame(tmp)  
  
 # Format labels  
 results$Estimate <- as.character(the\_lab)  
 results$Estimate <- factor(results$Estimate,levels=rev(the\_lab))   
 results$Estimand <- c("ATE",  
 rep("ACTE",3),  
 rep("CACTE",3),  
 rep("CATE",15)  
   
 )  
 results$Estimand <- factor(results$Estimand, levels =c("ATE","ACTE","CACTE","CATE"))  
  
 return(results)  
}

* plot\_effects\_fn(): Wrapper to plot results of effects\_fn()

plot\_effects\_fn <- function(d, dv){  
 # INPUTS:  
 # d: data frame  
 # dv: outcome  
 # OUTPUTS:  
 # fig: ggplot of results  
   
 # Create dataframe of effects for plotting  
 fig\_df <- rbind(  
 data.frame(  
 effects\_fn(d, dv),  
 Group = "Overall",  
 Type = "Overall"  
 ),  
 data.frame(  
 effects\_fn(d[d$gender==0,], dv),  
 Group = "Men",  
 Type = "By Gender"),  
 data.frame(  
 effects\_fn(d[d$gender==1,], dv),  
 Group = "Women",  
 Type = "By Gender"),  
 data.frame(  
 effects\_fn(d[d$pid > 4,], dv),  
 Group = "Republicans",  
 Type = "By Partisanship"),  
 data.frame(  
 effects\_fn(d[d$pid < 4,], dv),  
 Group = "Democrats",  
 Type = "By Partisanship")  
 )  
   
 # Set labels as factors for plotting  
 fig\_df$Type <- factor(fig\_df$Type, levels = unique(fig\_df$Type))  
 fig\_df$Group <- factor(fig\_df$Group , levels = unique(fig\_df$Group ))  
   
   
 # Creat Figure   
 fig <- fig\_df %>%  
 filter(Estimand!="CATE") %>%   
 ggplot(  
 aes(Estimate, Difference,col=Group,shape=Group)  
 )+  
 geom\_hline(yintercept = 0,linetype="dashed",alpha=.5)+  
 facet\_grid(~Type)+  
 geom\_point(  
 aes(shape=Group),  
 position = position\_dodge(width = .5),size=2  
 )+  
 geom\_linerange(  
 aes(ymin=ll,ymax=ul),  
 size=.3,  
 position = position\_dodge(width = .5)  
 )+  
 geom\_linerange(  
 aes(ymin=ll90,ymax=ul90),  
 size=.6,  
 position = position\_dodge(width = .5)  
 )+  
 ylab("Difference\n(Treatment Selectors - Treatment Avoiders)")+  
 coord\_flip()+  
 theme\_bw()+  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom"  
 )  
 # Format axis labels  
 fig <- fig +  
 theme(  
 axis.text.y = element\_text(  
 face = ifelse(rev(fig$data$Estimate) %in% c("ATE", "ACTE:", "CACTE:", "CATE:"),   
 "bold", "italic")  
 )  
 ) +  
 scale\_shape\_manual(values = c(16, 17,15,23,4))  
   
 return(fig)  
   
   
}

* format\_ci\_fn(): Helper function to format CIs

format\_ci\_fn <- function(est){  
 paste("[",sprintf("%.2f",est[,"ll"]),", ", sprintf("%.2f",est[,"ul"]),"]",sep="")  
}

* table\_fn(): Wrapper to format results of plot\_effects\_fn() as LaTeX table

table\_fn <- function(d, the\_cap=NULL){  
 # INPUTS:  
 # d: a dataframe of effects  
 # OUTUTS  
 # A LaTeX table  
 tab\_df <- d %>%filter(Estimand!="CATE") %>% na.omit()  
  
   
 tab <- data.frame(matrix(NA,nrow=10,ncol=6))  
 est\_seq <- seq(1,dim(tab)[1], by=2)  
 ci\_seq <- seq(2,dim(tab)[1], by=2)  
 tab[est\_seq,1] <- as.character(tab\_df[tab\_df$Group == "Overall", "Estimate"])  
 tab[ci\_seq,1] <- ""  
   
 tab[est\_seq,2] <- sprintf("%.2f",tab\_df[tab\_df$Group == "Overall", "Difference"])  
 tab[ci\_seq,2] <- format\_ci\_fn(tab\_df[tab\_df$Group == "Overall",])  
   
 tab[est\_seq,3] <- sprintf("%.2f",tab\_df[tab\_df$Group == "Men", "Difference"])  
 tab[ci\_seq, 3] <- format\_ci\_fn(tab\_df[tab\_df$Group == "Men",])  
   
 tab[est\_seq,4] <- sprintf("%.2f",tab\_df[tab\_df$Group == "Women", "Difference"])  
 tab[ci\_seq, 4] <- format\_ci\_fn(tab\_df[tab\_df$Group == "Women",])  
   
 tab[est\_seq,5] <- sprintf("%.2f",tab\_df[tab\_df$Group == "Republicans", "Difference"])  
 tab[ci\_seq, 5] <- format\_ci\_fn(tab\_df[tab\_df$Group == "Republicans",])  
   
 tab[est\_seq,6] <- sprintf("%.2f",tab\_df[tab\_df$Group == "Democrats", "Difference"])  
 tab[ci\_seq, 6] <- format\_ci\_fn(tab\_df[tab\_df$Group == "Democrats",])  
   
 colnames(tab) <- c("","Overall","Men","Women","Republicans","Democrats")  
   
 kable(  
 tab,   
 booktabs = TRUE,   
 caption = the\_cap,   
 digits=2,  
 align = c("l",rep("c",3))  
 ) %>%   
 kable\_styling(latex\_options = c("HOLD\_position",font\_size=10)) %>%  
 kableExtra::group\_rows("ATE",1,2) %>%  
 kableExtra::group\_rows("ACTE",3,6) %>%  
 kableExtra::group\_rows("CACTE",7,10)%>%  
 footnote(general = "The table provides point estimates and 95% confidence intervals for treatment effect estimated from the full sample and separately by gender and partisanship",  
 threeparttable = T,  
 fixed\_small\_size = T)  
}

* table\_app\_fn(): Wrapper to format results of effects\_fn() as LaTeX tables for appendix

table\_app\_fn <- function(d, dv, g,eg,the\_cap="Treatment Effects"){  
 # INPUTS:  
 # d: data frame  
 # dv: outcome  
 # g: variable to group by (Quotes)  
 # eg: variable to group by (No quotes)  
 # the\_cap: caption for table  
 # OUTPUTS:  
 # tab: table of results  
 tmp <- effects\_fn(d,dv)%>%filter(Estimand!="CATE")  
 tmp <- tmp[-c(2,5,8),-c(5:7,9)]  
  
   
   
   
 the\_vals <- na.omit(unlist(unique(d[,g])))  
  
 enquo\_g <- enquo(eg)  
 quo\_g <- quo\_name(enquo(eg))  
   
 tmp1 <- effects\_fn(d[d[,quo\_g]==the\_vals[1],], dv)%>%filter(Estimand!="CATE")  
 tmp1[,g] <- the\_vals[1]  
 tmp1 <- tmp1[-c(2,5,8),-c(5:7,9)]  
   
 tmp2 <- effects\_fn(d[d[,quo\_g]==the\_vals[2],], dv)%>%filter(Estimand!="CATE")  
 tmp2[,g] <- the\_vals[2]  
 tmp2 <- tmp2[-c(2,5,8),-c(5:7,9)]  
  
  
   
 d\_m <- d %>%filter(gender==0)  
 tmp1\_m <- effects\_fn(d\_m[d\_m[,quo\_g]==the\_vals[1],], dv)%>%filter(Estimand!="CATE")  
 tmp1\_m[,g] <- the\_vals[1]  
 tmp1\_m <- tmp1\_m[-c(2,5,8),-c(5:7,9)]  
  
 tmp2\_m <- effects\_fn(d\_m[d\_m[,quo\_g]==the\_vals[2],], dv)%>%filter(Estimand!="CATE")  
 tmp2\_m[,g] <- the\_vals[2]  
 tmp2\_m <- tmp2\_m[-c(2,5,8),-c(5:7,9)]  
   
 d\_f <- d %>%filter(gender==1)  
 tmp1\_f <- effects\_fn(d\_f[d\_f[,quo\_g]==the\_vals[1],], dv)%>%filter(Estimand!="CATE")  
 tmp1\_f[,g] <- the\_vals[1]  
 tmp1\_f <- tmp1\_f[-c(2,5,8),-c(5:7,9)]  
  
 tmp2\_f <- effects\_fn(d\_f[d\_f[,quo\_g]==the\_vals[2],], dv)%>%filter(Estimand!="CATE")  
 tmp2\_f[,g] <- the\_vals[2]  
 tmp2\_f <- tmp2\_f[-c(2,5,8),-c(5:7,9)]  
  
   
 tab <- data.frame(matrix(NA,nrow=dim(tmp)[1]\*2,ncol=7))  
 est\_seq <- seq(1,dim(tab)[1], by=2)  
 ci\_seq <- seq(2,dim(tab)[1], by=2)  
   
 tab[est\_seq,1] <- as.character(tmp[,c("Estimate")])  
 tab[ci\_seq,1] <- ""  
   
 tab[est\_seq,2] <- sprintf("%.2f",tmp1[,c("Difference")])  
 tab[ci\_seq, 2] <- format\_ci\_fn(tmp1)  
 tab[est\_seq, 3] <- sprintf("%.2f",tmp2[,c("Difference")])  
 tab[ci\_seq, 3] <- format\_ci\_fn(tmp2)  
   
 tab[est\_seq,4] <- sprintf("%.2f",tmp1\_m[,c("Difference")])  
 tab[ci\_seq, 4] <- format\_ci\_fn(tmp1\_m)  
 tab[est\_seq, 5] <- sprintf("%.2f",tmp2\_m[,c("Difference")])  
 tab[ci\_seq, 5] <- format\_ci\_fn(tmp2\_m)  
   
   
 tab[est\_seq,6] <- sprintf("%.2f",tmp1\_f[,c("Difference")])  
 tab[ci\_seq, 6] <- format\_ci\_fn(tmp1\_f)  
 tab[est\_seq, 7] <- sprintf("%.2f",tmp2\_f[,c("Difference")])  
 tab[ci\_seq, 7] <- format\_ci\_fn(tmp2\_f)  
 colnames(tab) <- c(" ",the\_vals[1],the\_vals[2],  
 the\_vals[1],the\_vals[2],  
 the\_vals[1],the\_vals[2])  
   
 tab <- kable(tab,  
 booktabs = TRUE,   
 caption = the\_cap,   
 digits=2,  
 align = c("l",rep("c",6))) %>%   
 kable\_styling(latex\_options = c("HOLD\_position",font\_size=10)) %>%  
 kableExtra::group\_rows("ATE",1,2) %>%  
 kableExtra::group\_rows("ACTE",3,6) %>%  
 kableExtra::group\_rows("CACTE",7,10)%>%  
 add\_header\_above(c(" " = 1, "Full Sample" = 2, "Men" = 2, "Women" = 2))  
 return(tab)  
   
}

## Functions to conduct power simulations

* data\_fn(): Simulate data for power simulations

# Generate simulated data  
data\_fn <- function(N=1000,   
 sigma = 2,  
 p\_treat=.5,   
 prop\_select=.5,   
 p\_treat\_select = c(.25,.5,.25),   
 tau\_st = .5,   
 tau\_af = -.5,  
 tau\_am = .5,  
 select\_effect = 0,  
 ...  
){  
 # INPUTS:  
 # N: sample size  
 # sigma: SE of outcome  
 # p\_treat: Prob of treatment assignment in experiment condition  
 # prop\_select: Proportion selecting treatment in choice conditoin  
 # p\_treat\_select: Prob of treatment assignment for CACTE  
 # tau\_st: effect among those selecting treatment  
 # tau\_af: effect among those avoiding treatment assigned to receive female treatment  
 # tau\_am: effect among those avoiding treatment assigned to receive mael treatment  
 # select\_effect: creates correlation between outcome and selecting treatment  
 # OUTPUTS:  
 # df: data frame of simualted responses  
   
 select <- rbinom(n=N, size=1,prob = prop\_select)  
 avoid <- as.numeric(select!=1)  
  
 # Baseline  
 Y0 <- rnorm(n=N, mean = 0, sd = sigma) + select\_effect \* select  
   
 # Condition  
 C <- sample(c("Choice","Experiment"),size=N,  
 prob = c(prop\_select, 1- prop\_select),replace = T)  
 n\_exp <- sum(C=="Experiment")  
 n\_ch\_a <- sum(C=="Choice" & avoid==1)  
   
 # Treatment status in experimental arm  
 D\_exp <- rep("Selection",N)  
 D\_exp[C=="Experiment"] <- sample(c("Treatment","Control"),size=n\_exp,  
 prob = c(p\_treat,1-p\_treat), replace = T)  
   
 D\_ch <- rep("Experiment",N)  
 D\_ch[C=="Choice" & avoid==1] <- sample(c("Treatment","Control", "Alternative"),  
 size=n\_ch\_a,  
 prob = p\_treat\_select,replace = T)  
 n\_choice <- sum(C == "Choice")  
 n\_select <- sum(C=="Choice" & avoid==0)  
 n\_avoid <- sum(C=="Choice" & avoid==1)  
 n\_control <- sum(D\_ch == "Control")  
 weights <- rep(1,N)  
 weights[C=="Choice" & avoid==0 & D\_ch == "Control"] <-1/(n\_select/n\_choice)  
 weights[C=="Choice" & avoid==1 & D\_ch == "Control"] <-1/(n\_control/n\_avoid)  
   
 # Potential outcome is conditional on preferences  
 Y1 <- Y0 + tau\_st\*(select == 1 & D\_exp == "Treatment") + tau\_af\*(avoid == 1 & D\_exp == "Treatment" ) +  
 tau\_st\*(select == 1 & C == "Choice")+  
 tau\_af\*(avoid == 1 & D\_ch == "Treatment" ) + tau\_am\*(avoid == 1 & D\_ch == "Alternative" )  
   
   
   
 # Observed Outcome  
   
 Y <- rep(NA,N)  
 Y[C=="Experiment"] <- Y0[C=="Experiment"]\*(D\_exp[C=="Experiment"]=="Control") +   
 Y1[C=="Experiment"]\*(D\_exp[C=="Experiment"]=="Treatment")  
 Y[C=="Choice" & select == 1] <- Y1[C=="Choice" & select == 1]  
 Y[C=="Choice" & select == 0 & D\_ch == "Control"] <- Y0[C=="Choice" & select == 0 & D\_ch == "Control"]  
 Y[C=="Choice" & select == 0 & D\_ch == "Treatment"] <- Y1[C=="Choice" & select == 0 & D\_ch == "Treatment"]  
 Y[C=="Choice" & select == 0 & D\_ch == "Alternative"] <- Y1[C=="Choice" & select == 0 & D\_ch == "Alternative"]  
   
   
 treatment = rep(NA, N)  
 treatment[C=="Experiment" & D\_exp == "Treatment"] <- "Treatment"  
 treatment[C=="Experiment" & D\_exp == "Control"] <- "Control"  
 treatment[C=="Choice"] <- "Selection"  
 treatment[C=="Choice" & D\_ch == "Treatment" ] <- NA  
 treatment[C=="Choice" & D\_ch == "Alternative" ] <- NA  
   
 avoid01 <- NA  
 avoid01[C == "Choice" & avoid == 1] <- 1  
 avoid01[C == "Choice" & avoid == 0] <- 0  
   
 select01 <- NA  
 select01[C == "Choice" & avoid == 0] <- 1  
 select01[C == "Choice" & avoid == 1] <- 0  
   
 df <- data.frame(Y0,Y1,Y,true\_diff = Y1-Y0,  
 C, treatment, select, avoid,  
 select01,avoid01,   
 D\_exp, D\_ch, weights)  
 return(df)  
}

* power\_fn(): Calculate power for given treatment effects

# Calculate power for given parameters  
power\_fn <- function(sims,  
 p\_N=1000,   
 p\_sigma = 1,  
 p\_p\_treat=.5,   
 p\_prop\_select=.5,   
 p\_p\_treat\_select = c(.25,.5,.25),   
 p\_tau\_st = .5,   
 p\_tau\_af = -.5,  
 p\_tau\_am = .5,  
 p\_select\_effect = 0,  
   
 ...){  
 # INPUTS:  
 # sims: number of simulations  
 # p\_\*: arguments passed to data\_fn()  
 # OUTPUTS:  
 # results: results of power simulation  
   
 ate <- rep(NA,sims)  
 acte\_s <- rep(NA,sims)  
 acte\_a <- rep(NA,sims)  
 cacte\_female <- rep(NA,sims)  
 cacte\_male <- rep(NA,sims)  
   
 sig\_ate <- rep(NA,sims)  
 sig\_acte\_s <- rep(NA,sims)  
 sig\_acte\_a <- rep(NA,sims)  
 sig\_cacte\_female <- rep(NA,sims)  
 sig\_cacte\_male <- rep(NA,sims)  
 cor\_select <- rep(NA,sims)  
   
   
 for(i in 1:sims){  
 df <- data\_fn(N=p\_N,sigma= p\_sigma ,  
 p\_treat=p\_p\_treat,  
 prop\_select = p\_prop\_select,  
 p\_treat\_select = p\_p\_treat\_select ,  
 tau\_st= p\_tau\_st ,  
 tau\_af= p\_tau\_af ,  
 tau\_am= p\_tau\_am ,  
 select\_effect= p\_select\_effect )  
 sig\_ate[i] <- diff\_fn(df,dv1="Y",c=c("Control","Treatment"),weights = T)["pval"]  
 sig\_acte\_s[i] <- acte\_fn(df,z=c("Control","Selection"), w=T)["pval"]  
 sig\_acte\_a[i] <- acte\_fn(df,z=c("Selection","Treatment"), w=T)["pval"]  
 sig\_cacte\_female[i] <- cacte\_fn(df)[1,"pval"]  
 sig\_cacte\_male[i] <- cacte\_fn(df)[2,"pval"]  
 cor\_select[i] <- cor(df$Y,df$select)  
   
 ate[i] <- diff\_fn(df,dv1="Y",c=c("Control","Treatment"))["Difference"]  
 acte\_s[i] <- acte\_fn(df,z=c("Control","Selection"), w=T)["Difference"]  
 acte\_a[i] <- acte\_fn(df,z=c("Selection","Treatment"),w = T)["Difference"]  
 cacte\_female[i] <- cacte\_fn(df)[1,"Difference"]  
 cacte\_male[i] <- cacte\_fn(df)[2,"Difference"]  
 }  
   
 pow\_ate <- mean(sig\_ate<.05)  
 pow\_acte\_s <- mean(sig\_acte\_s<.05)  
 pow\_acte\_a <- mean(sig\_acte\_a<.05)  
 pow\_cacte\_female <- mean(sig\_cacte\_female<.05)  
 pow\_cacte\_male <- mean(sig\_cacte\_male<.05)  
 mn\_cor\_select <- mean(cor\_select)  
   
 mn\_ate <- mean(ate)  
 mn\_acte\_s <- mean(acte\_s)  
 mn\_acte\_a <- mean(acte\_a)  
 mn\_cacte\_female <- mean(cacte\_female)  
 mn\_cacte\_male <- mean(cacte\_male)  
   
   
 results <- rbind(  
 c(mn\_ate,  
 mn\_acte\_s,  
 mn\_acte\_a,  
 mn\_cacte\_female,  
 mn\_cacte\_male,mn\_cor\_select),  
   
 c(pow\_ate,  
 pow\_acte\_s,  
 pow\_acte\_a,  
 pow\_cacte\_female,  
 pow\_cacte\_male,  
 NA  
 ))  
   
 return(results)  
   
}

* sim\_power\_fn(): Conduct power simulations over a range of treatment effects

# Simulate power for range of treatment effects  
sim\_power\_fn <- function(  
 s\_sims = 500,   
 s\_N=1000,   
 s\_sigma = 1,   
 s\_p\_treat=.5, #  
 s\_prop\_select=.5,   
 s\_p\_treat\_select = c(.25,.5,.25),   
 s\_tau\_st = .5,   
 s\_tau\_af = -.5,   
 s\_tau\_am = .5,   
 s\_select\_effect = 0  
   
   
   
) {  
 # INPUTS:  
 # s\_\*: arguments passed to power\_fn()  
 # OUTPUTS:  
 # results: list containing   
 # - data frame of power simulations  
 # - summary statitistics  
 # - range of correlation between selecting treatment and outcome  
   
 # Create matrix to store values  
 power\_mat <- matrix(NA,nrow=5, ncol = length(s\_tau\_st),  
 dimnames = list(c("ATE","ACTE-Select","ACTE-Avoid",  
 "CACTE-Female","CACTE-Male"),  
 s\_tau\_st  
 )  
 )  
 bias\_mat <- matrix(NA,nrow=5, ncol = length(s\_tau\_st),  
 dimnames = list(c("ATE","ACTE-Select","ACTE-Avoid",  
 "CACTE-Female","CACTE-Male"),  
 s\_tau\_st  
 )  
 )  
 tmp <- c()  
 ave\_cor <- c()  
 tmp\_df <- data.frame(Estimate=NULL,Tau=NULL)  
 df <- data.frame(Estimate= NULL,  
 Tau\_Select = NULL,   
 Tau\_Avoid = NULL ,  
 Tau\_Alt = NULL ,  
 Power = NULL )  
 # Loop over possible values  
 for(i in 1:length(s\_tau\_st)){  
 tmp <- power\_fn(sims = s\_sims,   
 p\_N = s\_N,  
 p\_sigma = s\_sigma,  
 p\_p\_treat = s\_p\_treat,  
 p\_prop\_select = s\_prop\_select,  
 p\_p\_treat\_select = s\_p\_treat\_select,  
 p\_tau\_st = s\_tau\_st[i],  
 p\_tau\_af = s\_tau\_af[i],  
 p\_tau\_am = s\_tau\_am[i],  
 p\_select\_effect = s\_select\_effect  
 )  
 ave\_cor[i] <- tmp[1,6]  
 power\_mat[,i] <- tmp[2,1:5]   
 tmp2 <- data.frame(Estimate= c("ATE","ACTE-Select","ACTE-Avoid",  
 "CACTE-Female","CACTE-Male"),  
 Type = c("ATE","ACTE","ACTE","CACTE","CACTE"),  
 Tau\_Select = rep(s\_tau\_st[i],5),  
 Tau\_Avoid = rep(s\_tau\_af[i],5),  
 Tau\_Alt = rep(s\_tau\_am[i],5),  
 Power = tmp[2,1:5]   
 )  
 df <- rbind(tmp2,df)  
 }  
 return(list(df,power\_mat,ave\_cor))  
   
}

* display\_power\_sim\_fn(): Wrapper to display results of power simulations as figure and table

# Display results of power simulations  
display\_power\_sim\_fn <- function(  
 p\_s\_sims = 1000,  
 p\_s\_N=1000,   
 p\_s\_sigma = 1,  
 p\_s\_p\_treat=.4,   
 p\_s\_prop\_select=.5,   
 p\_s\_p\_treat\_select = c(.25,.5,.25),   
 p\_s\_tau\_st = .5,   
 p\_s\_tau\_af = -.5,  
 p\_s\_tau\_am = .5,  
 p\_s\_select\_effect = 0,  
 lab\_effects = "Effects = Equal & offsetting"  
   
){  
 # INPUTS:  
 # p\_s\_\*: arguments passed to sim\_power\_fn()  
 # OUTPUTS:  
 # results: list containing   
 # - figure displaying results  
 # - table formatted for LaTeX  
  
 pow <- sim\_power\_fn(  
 s\_sims = p\_s\_sims,  
 s\_N= p\_s\_N,  
 s\_sigma = p\_s\_sigma,  
 s\_p\_treat= p\_s\_p\_treat,  
 s\_prop\_select= p\_s\_prop\_select,  
 s\_p\_treat\_select = p\_s\_p\_treat\_select,  
 s\_tau\_st = p\_s\_tau\_st,  
 s\_tau\_af = p\_s\_tau\_af,  
 s\_tau\_am = p\_s\_tau\_am,  
 s\_select\_effect = p\_s\_select\_effect  
   
   
 )  
 rhos.min <- round(range(pow[[3]]),2) [1]  
 rhos.max <- round(range(pow[[3]]),2) [2]  
   
 p <- pow[[1]] %>%  
 mutate(  
 Estimate=factor(Estimate,  
 levels = c("ATE","ACTE-Select","ACTE-Avoid",  
 "CACTE-Female","CACTE-Male")  
 ),  
 Type = factor(Type,   
 levels = c("ATE","ACTE","CACTE")  
 )  
 )%>%  
 ggplot(aes(Tau\_Select,Power, col=Estimate,linetype=Type))+  
 geom\_line()+  
 ylim(0,1.05) +  
 xlim(0,.7) +  
 geom\_hline(yintercept = .8,linetype = "dashed",col="grey")+  
 xlab(expression(tau[Select]))+  
 ylab("Power (Probability of Statistical Significance)")+  
 annotate(geom = "text",  
 hjust = 0,  
 y = 1.05,  
 x = 0,  
 label = paste("Simulations =",scales::comma(p\_s\_sims))  
 )+  
 annotate(geom = "text",  
 hjust = 0,  
 y = 1,  
 x = 0,  
 label = lab\_effects  
 )+  
 annotate(geom = "text",  
 hjust = 0,  
 y = .95,  
 x = 0,  
 label =paste("Prop Select =", round(p\_s\_prop\_select,2))  
 ) +  
 annotate(geom = "text",  
 hjust = 0,  
 y = .90,  
 x = 0,  
 label = paste("Cor = [",rhos.min,", ",rhos.max,"]",sep="")  
 )  
 tab <- kable(pow[[2]],  
 caption = "Power Analysis",  
 format = "latex",  
 booktabs=T,   
 linesep = "",  
 digits=2) %>%  
 add\_header\_above(c("","Hypothesized Effect Among Selectors"=dim(pow[[2]])[2])) %>%  
 kable\_styling(latex\_options = c("hold\_position",font\_size=10))  
   
 return(list(p,tab))  
   
   
}

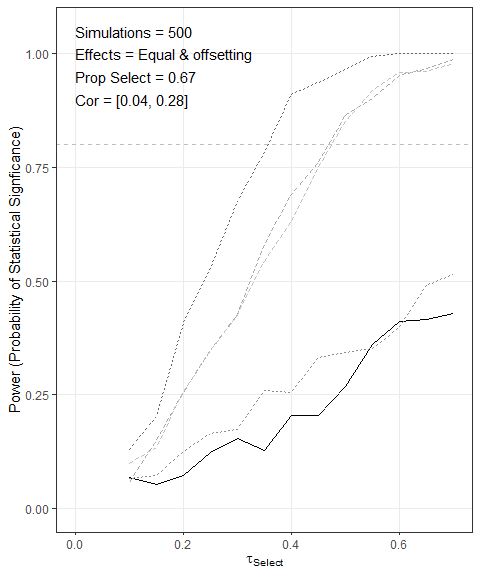
# Main Figures

## Figure 1: Triply Randomized Parallel Design

Note: Figure created using Adobe Illustrator

## Figure 2: Statistical Power with More Selectors than Avoiders, Equal and Offsetting Effects

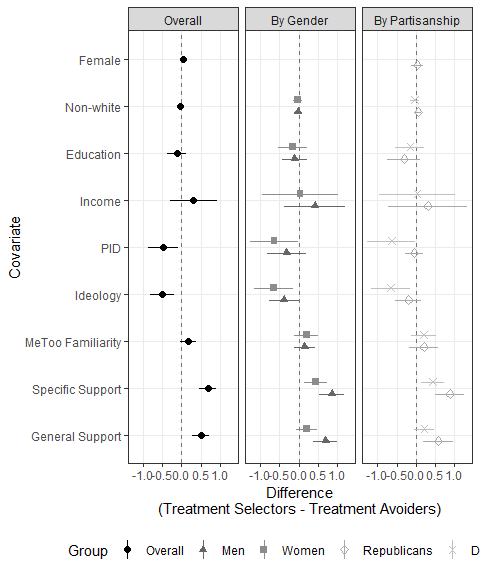
#Uncomment to run. 500 Simulation takes ~ 30-40 minutes  
# Set random seed  
#set.seed(123)  
  
#fig2\_power\_sim <- display\_power\_sim\_fn(  
# p\_s\_sims = 500,  
# p\_s\_prop\_select = 2/3,  
# p\_s\_tau\_st = seq(.1,.7,by=.05),  
# p\_s\_tau\_af = seq(-.1,-.7,by=-.05),  
# p\_s\_tau\_am = seq(.1,.7,by=.05)  
# )  
#print(fig2\_power\_sim)  
  
# Format Figure 2  
fig2 <- fig2\_power\_sim [[1]]+  
 theme\_bw()+  
 theme(  
 panel.grid.minor = element\_blank(),  
 )+  
 scale\_color\_grey(start = 0, end = .75)  
  
# Display Figure 2  
fig2



Statistical Power with More Selectors than Avoiders, Equal and Offsetting Effects

## Figure 3: Who is Likely to Seek Out or Avoid the Message of the #MeToo Movement?

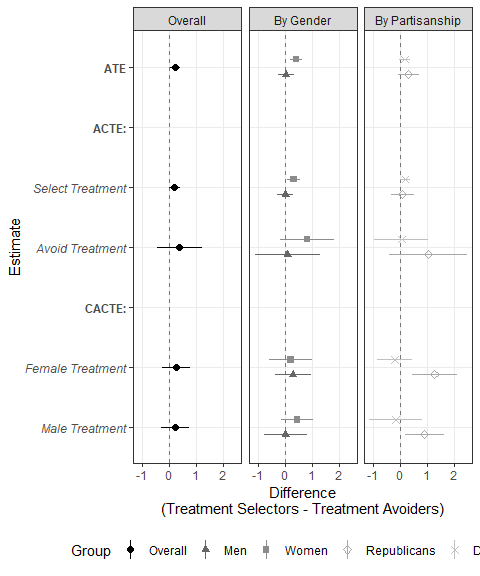
# Create Figure 3  
fig3 <- plot\_balance\_fn(df\_mtg)  
  
# Display Figure 3  
fig3



Who is Likely to Seek Out or Avoid the Message of the #MeToo Movement?

## Figure 4: Heterogeneous Effects in the #MeToo MTurk Study

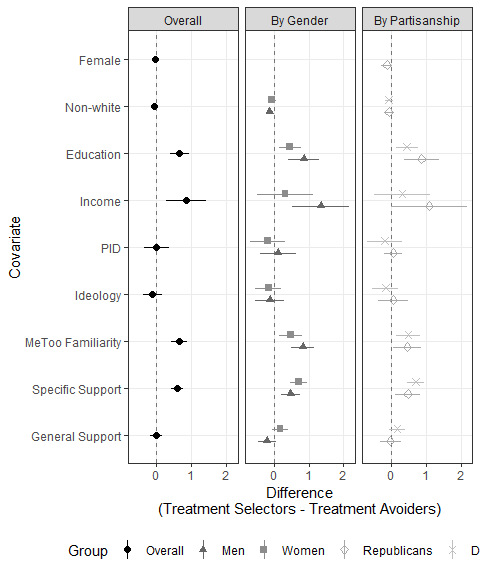
# Create Figure 4  
fig4 <- plot\_effects\_fn(df\_mtg,"dv\_pca\_metoo") + scale\_color\_grey(start = 0, end = .75)   
  
# Display Figure 4  
fig4



Heterogeneous Effects in the #MeToo MTurk Study

## Figure 5: Who is Likely to Seek Out or Avoid the Message of the #MeToo Movement in a More Nationally Representative Sample?

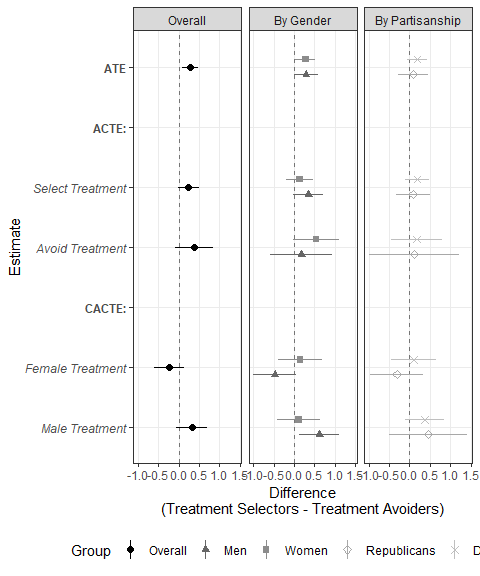
# Create Figure 3  
fig5 <- plot\_balance\_fn(df\_qg)  
  
# Display Figure 3  
fig5



Who Seeks Out or Avoids the Message of the #MeToo Movement in a More Nationally Representative Sample?

## Figure 6: Heterogeneous Effects in the #MeToo Qualtrics Study

# Create Figure 6  
fig6 <- plot\_effects\_fn(df\_qg,"dv\_pca\_metoo") + scale\_color\_grey(start = 0, end = .75)   
  
# Display Figure 6  
fig6



Heterogeneous Effects in the #MeToo Qualtrics Study

# Main Tables

## Table 1

table\_fn(fig4$data,  
 "Table 1: Treatment Effect Estimates on Specific Support for \\#MeToo (MTurk Sample)")

## Table 2

table\_fn(fig6$data,  
 "Table 2: Treatment Effect Estimates on Specific Support for \\#MeToo (Qualtrics Sample)")

# Online Appendix

## Appendix C Power Simulations

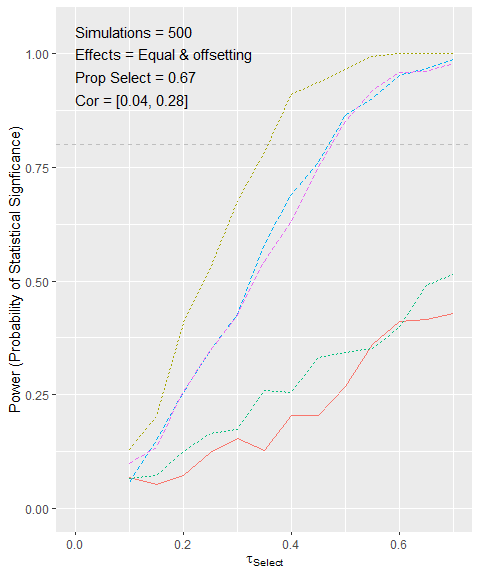
### Figure and Table C.1: Statistical Power with Equal Number of Selectors than Avoiders, Equal and Offsetting Effects

# Uncomment to run. 500 Simulation takes ~ 30-40 minutes  
# Set random seed  
#set.seed(123)  
  
#figC1\_power\_sim <- display\_power\_sim\_fn(  
# p\_s\_sims = 500,  
# p\_s\_prop\_select = .5,  
# p\_s\_tau\_st = seq(.1,.7,by=.05),  
# p\_s\_tau\_af = seq(-.1,-.7,by=-.05),  
# p\_s\_tau\_am = seq(.1,.7,by=.05)  
# )  
#  
#figC1\_power\_sim[[1]]

figC1\_power\_sim[[2]]

### Figure and Table C.2: Statistical Power with More Selectors than Avoiders, Equal and Offsetting Effects

# Same as Figure 2  
fig2\_power\_sim[[1]]



Statistical Power with More Selectors than Avoiders, Equal and Offsetting Effects

fig2\_power\_sim[[2]]

### Figure and Table C.3: Statistical Power with More Selectors than Avoiders, Equal and Offsetting Effects, and Selection Correlated with Outcome

# Uncomment to run. 500 Simulation takes ~ 30-40 minutes  
  
# Set random seed  
#set.seed(123)  
  
#figC3\_power\_sim <- display\_power\_sim\_fn(  
# p\_s\_sims = 500,  
# p\_s\_prop\_select = 2/3,  
# p\_s\_select\_effect = 0.5,  
# p\_s\_tau\_st = seq(.1,.7,by=.05),  
# p\_s\_tau\_af = seq(-.1,-.7,by=-.05),  
# p\_s\_tau\_am = seq(.1,.7,by=.05)  
# )  
  
#figC3\_power\_sim[[1]]

figC3\_power\_sim[[2]]

# Uncomment to save results of power simulations  
#save(figC1\_power\_sim, fig2\_power\_sim,figC3\_power\_sim,file = "power\_simulations.rda")

## Appendix D Descriptive Statistics

### Table D.1: Descriptive Statistics for MTurk Sample

the\_covariates <- c("female01", "age", "income", "education", "pid", "ideo",   
 "black", "latino", "asian", "fam\_movement", "avoid01")  
desc\_tab\_mt<- c()  
for(i in 1:length(the\_covariates)){  
 desc\_tab\_mt <- cbind(desc\_tab\_mt,  
 summary(df\_mtg[,the\_covariates[i]])[1:6]  
 )  
}  
desc\_tab\_mt <- t(round(desc\_tab\_mt,2))  
rownames(desc\_tab\_mt) <- c("Prop. Female", "Age", "Income", "Education",  
 "Party ID", "Ideology",  
 "Prop. Black",  
 "Prop. Latinx",  
 "Prop. Asian",  
 "Familiarity with MeToo",  
 "Prop Avoiding Treatment")  
  
  
desc\_tab\_mt\_tex <- kable(desc\_tab\_mt,  
 booktabs = TRUE,   
 caption = "Descriptive Statistics for MTurk Sample",   
 digits=2,  
 align = "l") %>%   
 kable\_styling(latex\_options = c("hold\_position",font\_size=10))  
desc\_tab\_mt\_tex

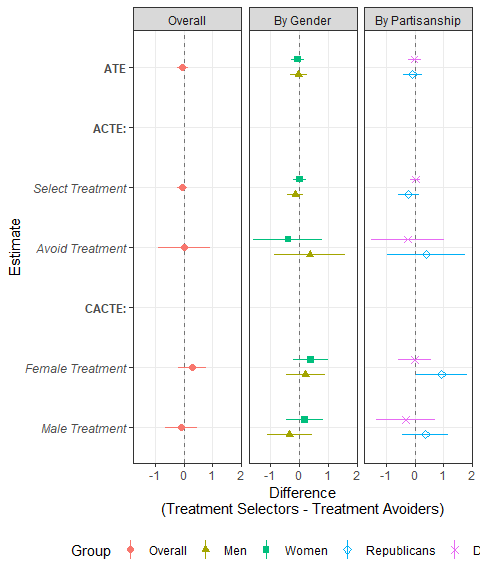
### Table D.2: Descriptive Statistics for Qualtrics Sample

desc\_tab\_q <- c()  
df\_qg <- data.frame(df\_qg)  
for(i in 1:length(the\_covariates)){  
 desc\_tab\_q <- cbind(desc\_tab\_q,  
 summary(na.omit(df\_qg[,the\_covariates[i]])))  
   
}  
desc\_tab\_q <- t(round(desc\_tab\_q,2))  
rownames(desc\_tab\_q) <- c("Prop. Female", "Age", "Income", "Education",  
 "Party ID", "Ideology",  
 "Prop. Black",  
 "Prop. Latinx",  
 "Prop. Asian",  
 "Familiarity with MeToo",  
 "Prop Avoiding Treatment")  
  
  
  
desc\_tab\_q\_tex <- kable(desc\_tab\_q,  
 booktabs = TRUE,   
 caption = "Descriptive Statistics for Qualtrics Sample",   
 digits=2,  
 align = "l") %>%   
 kable\_styling(latex\_options = c("hold\_position",font\_size=10))  
desc\_tab\_q\_tex

## Appendix E Effects on General Support for Gender Equality

### Figure and Table E.1: Effects on General Support for Gender Equality (MTurk Study)

# Create Figure E1  
figE1 <- plot\_effects\_fn(df\_mtg, "dv\_pca\_general")  
  
# Display Figure 4  
figE1

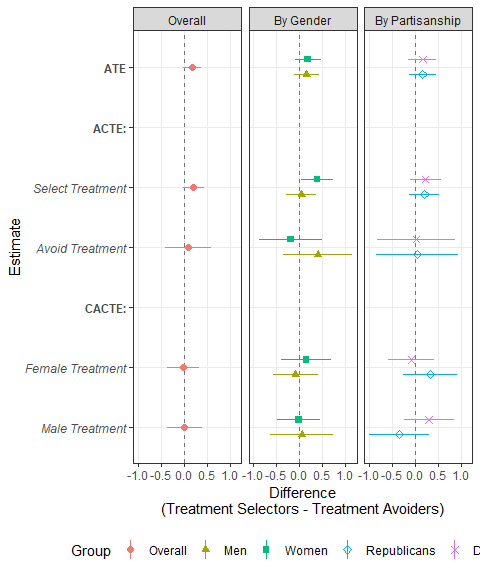


Effects on General Support for Gender Equality (MTurk Study)

table\_fn(figE1$data,  
 "Treatment Effect Estimates on General Support for Gender Equality (MTurk Sample)"  
 )

### Figure and Table E.2: Effects on General Support for Gender Equality (Qualtrics Study)

# Create Figure E2  
figE2 <- plot\_effects\_fn(df\_qg, "dv\_pca\_general")  
  
# Display Figure E2  
figE2



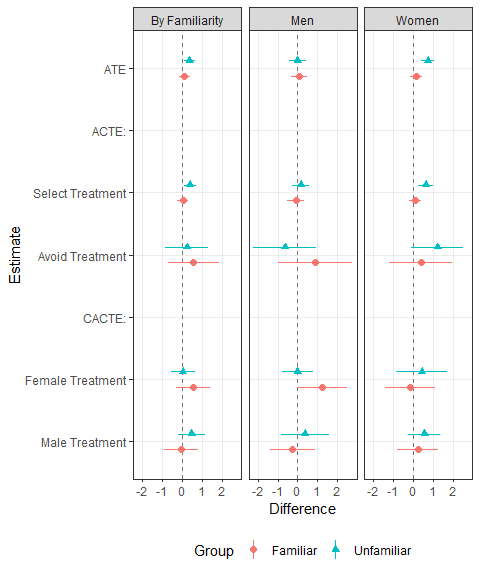
Effects on General Support for Gender Equality (Qualtrics Study)

table\_fn(figE2$data,  
 "Treatment Effect Estimates on General Support for Gender Equality (MTurk Sample)"  
 )

## F Additional Analyses

### Figure and Table F.1: Treatment Effect Estimates on Specific Support for #MeToo Conditional on Familiarity and Gender (MTurk Sample)

figF1\_df <- rbind(  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$fam\_movement>2,], "dv\_pca\_metoo"),  
 Group = "Familiar",  
 Type = "By Familiarity"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$fam\_movement<3,], "dv\_pca\_metoo"),  
 Group = "Unfamiliar",  
 Type = "By Familiarity"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$fam\_movement>2 & df\_mtg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Familiar",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$fam\_movement<3& df\_mtg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Unfamiliar",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$fam\_movement>2 & df\_mtg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Familiar",  
 Type = "Women"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$fam\_movement<3& df\_mtg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Unfamiliar",  
 Type = "Women")  
 )  
  
figF1 <- figF1\_df %>%  
 filter(Estimand != "CATE")%>%  
 ggplot(aes(Estimate, Difference,col=Group,shape=Group))+  
 geom\_hline(yintercept = 0,linetype="dashed",alpha=.5)+  
 facet\_grid(~Type)+  
 geom\_point(aes(shape=Group),  
 position = position\_dodge(width = .5),size=2  
 )+  
 geom\_linerange(aes(ymin=ll,ymax=ul),size=.3,  
 position = position\_dodge(width = .5))+  
 geom\_linerange(aes(ymin=ll90,ymax=ul90),size=.6,  
 position = position\_dodge(width = .5))+  
 coord\_flip()+  
 theme\_bw()+  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom"  
 )  
  
figF1

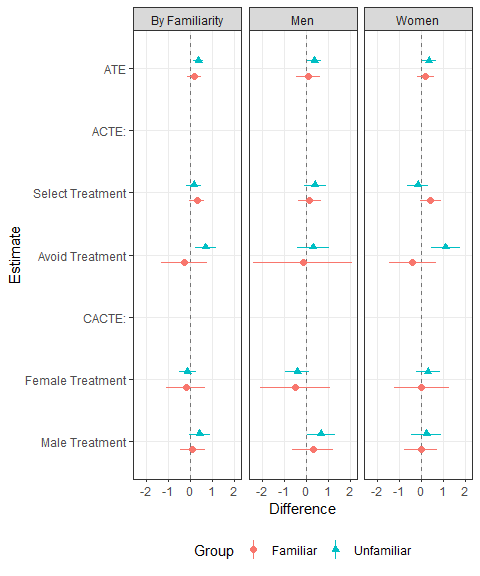


Treatment Effect Estimates on Specific Support for #MeToo Conditional on Familiarity and Gender

# Create Grouping Label   
#df\_mtg$Familiarity <- ifelse(df\_mtg$fam\_movement>2, "Familiar", "Unfamiliar")  
  
table\_app\_fn(df\_mtg, "dv\_pca\_metoo",  
 "Familiarity",  
 Familiarity,  
 the\_cap ="Treatment Effect Estimates on Specific Support for \\#MeToo Conditional On Familiarity and Gender (MTurk Sample)" )%>%  
 footnote(general = "The table provides point estimates and 95% confidence intervals for treatment effect estimated by level of pre-test familiarity with the movement overall and by gender",  
 threeparttable = T,  
 fixed\_small\_size = T)

### Figure and Table F.2: Treatment Effect Estimates on Specific Support for #MeToo Conditional on Familiarity and Gender (Qualtrics Study)

figF2\_df <- rbind(  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$fam\_movement>2,], "dv\_pca\_metoo"),  
 Group = "Familiar",  
 Type = "By Familiarity"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$fam\_movement<3,], "dv\_pca\_metoo"),  
 Group = "Unfamiliar",  
 Type = "By Familiarity"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$fam\_movement>2 & df\_qg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Familiar",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$fam\_movement<3& df\_qg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Unfamiliar",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$fam\_movement>2 & df\_qg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Familiar",  
 Type = "Women"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$fam\_movement<3& df\_qg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Unfamiliar",  
 Type = "Women")  
 )  
  
figF2 <- figF2\_df %>%  
 filter(Estimand != "CATE")%>%  
 ggplot(aes(Estimate, Difference,col=Group,shape=Group))+  
 geom\_hline(yintercept = 0,linetype="dashed",alpha=.5)+  
 facet\_grid(~Type)+  
 geom\_point(aes(shape=Group),  
 position = position\_dodge(width = .5),size=2  
 )+  
 geom\_linerange(aes(ymin=ll,ymax=ul),size=.3,  
 position = position\_dodge(width = .5))+  
 geom\_linerange(aes(ymin=ll90,ymax=ul90),size=.6,  
 position = position\_dodge(width = .5))+  
 coord\_flip()+  
 theme\_bw()+  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom"  
 )  
  
figF2

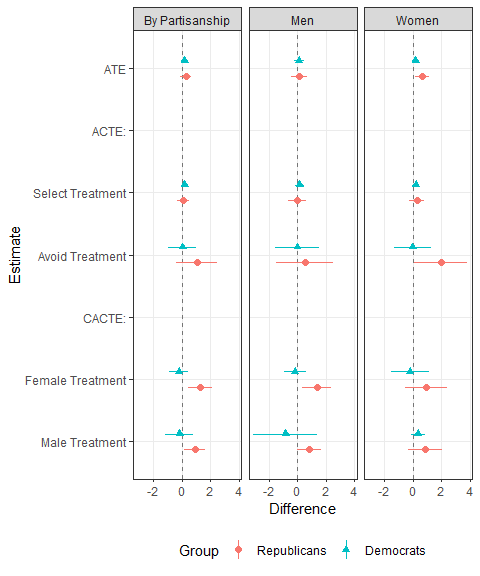


Treatment Effect Estimates on Specific Support for #MeToo Conditional on Familiarity and Gender (Qualtrics Study)

table\_app\_fn(df\_qg, "dv\_pca\_metoo",  
 "Familiarity",  
 Familiarity,  
 the\_cap ="Treatment Effect Estimates on Specific Support for \\#MeToo Conditional On Familiarity and Gender (Qualtrics Sample)" )%>%  
 footnote(general = "The table provides point estimates and 95% confidence intervals for treatment effect estimated by level of pre-test familiarity with the movement overall and by gender",  
 threeparttable = T,  
 fixed\_small\_size = T)

### Figure and Table F.3: Treatment Effect Estimates on Specific Support for #MeToo Conditional On Partisanship and Gender (MTurk Sample)

figF3\_df <- rbind(  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$pid>4,], "dv\_pca\_metoo"),  
 Group = "Republicans",  
 Type = "By Partisanship"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$pid<4,], "dv\_pca\_metoo"),  
 Group = "Democrats",  
 Type = "By Partisanship"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$pid>4 & df\_mtg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Republicans",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$pid<4& df\_mtg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Democrats",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$pid>4 & df\_mtg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Republicans",  
 Type = "Women"),  
 data.frame(  
 effects\_fn(df\_mtg[df\_mtg$pid<4& df\_mtg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Democrats",  
 Type = "Women")  
 )  
  
figF3\_df$Group <- factor(figF3\_df$Group,  
 levels = unique(figF3\_df$Group) )  
  
figF3 <- figF3\_df %>%  
 filter(Estimand != "CATE")%>%  
 ggplot(aes(Estimate, Difference,col=Group,shape=Group))+  
 geom\_hline(yintercept = 0,linetype="dashed",alpha=.5)+  
 facet\_grid(~Type)+  
 geom\_point(aes(shape=Group),  
 position = position\_dodge(width = .5),size=2  
 )+  
 geom\_linerange(aes(ymin=ll,ymax=ul),size=.3,  
 position = position\_dodge(width = .5))+  
 geom\_linerange(aes(ymin=ll90,ymax=ul90),size=.6,  
 position = position\_dodge(width = .5))+  
 coord\_flip()+  
 theme\_bw()+  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom"  
 )  
  
figF3

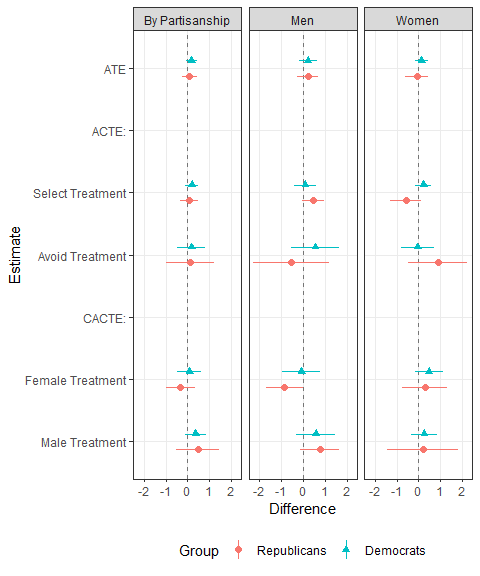


Treatment Effect Estimates on Specific Support for #MeToo Conditional On Partisanship and Gender (MTurk Sample)

# Create Grouping Label   
  
table\_app\_fn(df\_mtg, "dv\_pca\_metoo",  
 "Partisanship",  
 Partisanship,  
 the\_cap ="Treatment Effect Estimates on Specific Support for \\#MeToo Conditional On Partisanship and Gender (MTurk Sample)" )%>%  
 footnote(general = "The table provides point estimates and 95% confidence intervals for treatment effect estimated by partisanship overall and partisanship by gender",  
 threeparttable = T,  
 fixed\_small\_size = T)

### Figure and Table F.4: Treatment Effect Estimates on Specific Support for #MeToo Conditional On Partisanship and Gender (Qualtrics Sample)

figF4\_df <- rbind(  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$pid>4,], "dv\_pca\_metoo"),  
 Group = "Republicans",  
 Type = "By Partisanship"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$pid<4,], "dv\_pca\_metoo"),  
 Group = "Democrats",  
 Type = "By Partisanship"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$pid>4 & df\_qg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Republicans",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$pid<4& df\_qg$gender==0,], "dv\_pca\_metoo"),  
 Group = "Democrats",  
 Type = "Men"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$pid>4 & df\_qg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Republicans",  
 Type = "Women"),  
 data.frame(  
 effects\_fn(df\_qg[df\_qg$pid<4& df\_qg$gender==1,], "dv\_pca\_metoo"),  
 Group = "Democrats",  
 Type = "Women")  
 )  
  
figF4\_df$Group <- factor(figF4\_df$Group,  
 levels = unique(figF4\_df$Group) )  
  
figF4 <- figF4\_df %>%  
 filter(Estimand != "CATE")%>%  
 ggplot(aes(Estimate, Difference,col=Group,shape=Group))+  
 geom\_hline(yintercept = 0,linetype="dashed",alpha=.5)+  
 facet\_grid(~Type)+  
 geom\_point(aes(shape=Group),  
 position = position\_dodge(width = .5),size=2  
 )+  
 geom\_linerange(aes(ymin=ll,ymax=ul),size=.3,  
 position = position\_dodge(width = .5))+  
 geom\_linerange(aes(ymin=ll90,ymax=ul90),size=.6,  
 position = position\_dodge(width = .5))+  
 coord\_flip()+  
 theme\_bw()+  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom"  
 )  
  
figF4

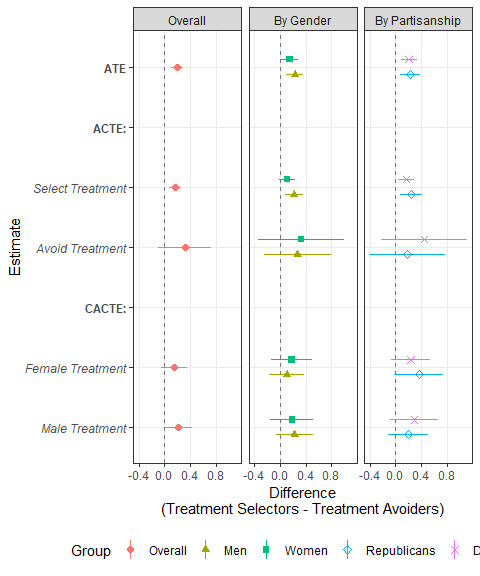


Treatment Effect Estimates on Specific Support for #MeToo Conditional On Partisanship and Gender (Qualtrics Sample)

# Create Grouping Label   
table\_app\_fn(df\_qg, "dv\_pca\_metoo",  
 "Partisanship",  
 Partisanship,  
 the\_cap ="Treatment Effect Estimates on Specific Support for \\#MeToo Conditional On Partisanship and Gender (Qulatrics Sample)" )%>%  
 footnote(general = "The table provides point estimates and 95% confidence intervals for treatment effect estimated by partisanship overall and partisanship by gender",  
 threeparttable = T,  
 fixed\_small\_size = T)

### Figure and Table F.5: Treatment Effect Estimates on Knowledge of Sexual Assault Statistics (MTurk Sample)

# Create Figure 5  
figF5 <- plot\_effects\_fn(df\_mtg,"dv\_fact01")   
  
# Display Figure 5  
figF5

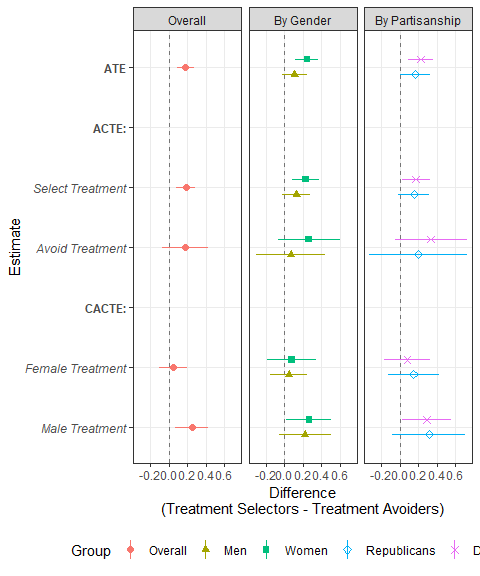


Treatment Effect Estimates on Knowledge of Sexual Assault Statistics (MTurk Sample)

table\_fn(figF5$data,  
 "Treatment Effect Estimates on Knowledge of Sexual Assault Statistics (MTurk Sample)")

### Figure and Table F.6: Treatment Effect Estimates on Knowledge of Sexual Assault Statistics (Qualtrics Sample)

# Create Figure 6  
figF6 <- plot\_effects\_fn(df\_qg,"dv\_fact01")   
  
# Display Figure 6  
figF6

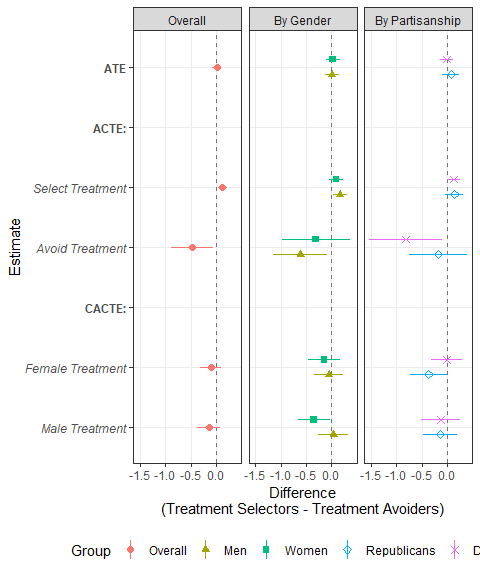


Treatment Effect Estimates on Knowledge of Sexual Assault Statistics (Qualtrics Sample)

table\_fn(figF6$data,  
 "Treatment Effect Estimates on Knowledge of Sexual Assault Statistics (Qualtrics Sample)")

### Figure and Table F.7: Treatment Effect Estimates on Providing Written Responses about #MeToo (MTurk Sample)

# Create Figure 6  
figF8 <- plot\_effects\_fn(df\_mtg,"dv\_response01")   
  
# Display Figure 6  
figF8

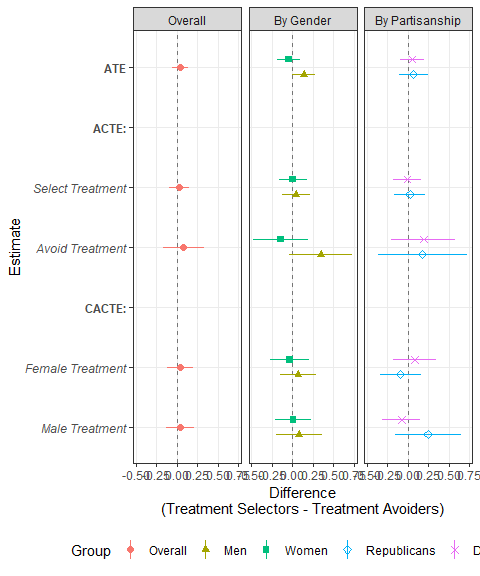


Treatment Effect Estimates on Providing Written Responses about #MeToo (MTurk Sample)

table\_fn(figF8$data,  
 "Treatment Effect Estimates on Providing Written Responses about \\#MeToo (MTurk Sample)")

### Figure and Table F.8: Treatment Effect Estimates on Providing Written Responses about #MeToo (Qualtrics Sample)

# Create Figure 6  
figF8 <- plot\_effects\_fn(df\_qg,"dv\_response01")   
  
# Display Figure 6  
figF8



Treatment Effect Estimates on Providing Written Responses about #MeToo (Qualtrics Sample)

table\_fn(figF8$data,  
 "Treatment Effect Estimates on Providing Written Responses about \\#MeToo (Qualtrics Sample)")