scicur-study1

# Libraries

library(rio)  
library(here)  
library(tidyverse)  
library(janitor)  
library(ggplot2)

# Import Joined Data (df1)

Loading a single dataframe that has already been prepped in a different document. It includes both the baseline and study 1 data, matched by MTurk ID.

It is already **excluding**: \* Duplicates \* Unfinished responses \* Responses where Ss reported cheating \* Responses where Ss reported bad response \* Unapproved responses \* Responses where Ss did not consent \* Responses that were not in both dfs (b and s1)

df1 <- import(here("prepped-data", "joined-data.csv"))

# Cleaning (df2)

df2 <- df1 %>% # organize df cols  
 select(id, batch, contains("duration\_in\_seconds"), contains("2\_1\_sc"),   
 contains("news1"), contains("act"), contains("books"), contains("convo"),  
 contains("share\_freq"), contains("read\_article"), contains("topic\_interest"),   
 contains("1\_1\_sc"), contains("time\_page\_submit\_sc"),   
 contains("rav"), contains("cc\_risk"), contains("ccr"),  
 contains("dem"), contains("start\_date"), contains("end\_date"),   
 contains("finished"), contains("consent"), contains("open"), contains("fl\_")  
 )

# make numeric variables numeric  
df2[3:170] <- lapply(df2[3:170], as.numeric)

## Warning in lapply(df2[3:170], as.numeric): NAs introduced by coercion  
  
## Warning in lapply(df2[3:170], as.numeric): NAs introduced by coercion  
  
## Warning in lapply(df2[3:170], as.numeric): NAs introduced by coercion

df2[172:177] <- lapply(df2[172:177], as.numeric)

## Duration

df2 <- df2 %>%   
 mutate(duration\_z = scale(duration\_in\_seconds\_sc),  
 duration\_ex = case\_when(  
 duration\_z >= 3 ~ "1", # z score duration times to exclude outliers  
 duration\_z <= -3 ~ "1",  
 TRUE ~ "0"  
 ))

## Political Ideology

df2 <- df2 %>%   
 mutate(conservatism = dem\_annes\_id\_b, # 1 (ext L), 7 (ext C), 4 = mod  
 ideology\_lean = dem\_annes\_id2\_b, # 1 = R lean, 2 = neither, 4 = D lean  
 ideology\_lean2 = case\_when(  
 dem\_annes\_id2\_b == 1 ~ 4.5,  
 dem\_annes\_id2\_b == 2 ~ 4,  
 dem\_annes\_id2\_b == 4 ~ 3.5,  
 TRUE ~ 999  
 ),  
 ideology\_lean2 = ifelse(ideology\_lean2 == 999, NA, ideology\_lean2),  
 conservatism\_f = ifelse(conservatism == 4, ideology\_lean2, conservatism),  
 con = case\_when(  
 conservatism\_f > 4 ~ 1,  
 conservatism\_f < 4 ~ -1,  
 conservatism\_f == 4 ~ 0,  
 TRUE ~ 999  
 ),  
 con = ifelse(con == 999, NA, con),  
 party = dem\_annes\_party\_b, # 1 = D, 2 = R, 3 = Indep, 4 = other  
 party\_str = dem\_annes\_party\_str\_b, # 1 = strong, 2 = not very strong  
 party\_lean = dem\_annes\_party\_lean\_b)

# checking if batch and ideology match -- it doesn't for all??  
df2 <- df2 %>%   
 mutate(batchcon = case\_when(  
 batch == "con1" ~ 1,  
 batch == "con2" ~ 1,  
 batch == "lib1" ~ -1,  
 batch == "lib2" ~ -1,  
 TRUE ~ 999  
 ),  
 batchcon = ifelse(batchcon == 999, NA, batchcon)  
 )  
  
df2\_picheck <- df2 %>%   
 select(id, con, batchcon) %>%   
 mutate(batchmatch = ifelse(con == batchcon, 1, 0))  
  
df2\_picheck %>%   
 filter(batchmatch == 0) %>%   
 nrow() # all of these ppl got sorted into the wrong HIT,

## [1] 92

# not too much of an issue though  
  
df2\_picheck %>%   
 filter(batchmatch == 0) %>%   
 head(10)

## id con batchcon batchmatch  
## 1 A2615YW1YERQBO 1 -1 0  
## 2 A1CRE2Z623JZ6V 0 1 0  
## 3 A2EH7IQCDH2Q5D 0 1 0  
## 4 A1A46CKE3F5JAJ 0 1 0  
## 5 A2FRKLXGF4118Q 0 1 0  
## 6 A1HPECU3OD1G9D 0 1 0  
## 7 AQKC1VIZAZ6VI 0 1 0  
## 8 A1ZEOAAKKV1NLP 0 1 0  
## 9 A19OCD2Q0K2U6M 0 1 0  
## 10 A2D98V3ZYWDMJO 0 1 0

# Updated N

Ncon = 227, Nmod = 52, Nlib = 266. Ntot = 545

df2 %>%   
 group\_by(con) %>%   
 summarise(n = n())

## # A tibble: 3 × 2  
## con n  
## <dbl> <int>  
## 1 -1 266  
## 2 0 52  
## 3 1 227

## Demographics

Haven’t touched these yet.

# gender: dem\_gender (\_1 = man, \_2 = nb, \_5 = woman, \_3 = other,   
# \_4 = prefer not to answer)  
  
# race: dem\_race (\_1 = white/caucasian, \_2 = black/african american,  
# \_3 = hispanic, \_4 = asian, \_5 = native hawaiian/pacific islander,  
# \_6 = american indian, \_7 = eskimo aleut, \_8 = other, \_9 = prefer not to answer)  
  
# native english speaker: dem\_lang (1 = yes, 2 = no)  
  
# edu: dem\_ed (\_1 = <hs, \_2 = some hs, \_3 = hs/ged, \_4 = 2yr college,  
# \_5 = 4yr college, \_6 = masters, \_7 = doctoral)  
  
# income: dem\_inc (\_1 = <10k, \_2 = 10-20k, \_3 = 20-30k, (...), \_11 = 100-150k,  
# \_12 = 150-200k, \_13 = 200-250k, \_14 > 250k)

# Scoring/Recoding

## Score Science Curiosity Scale (df3)

df3 <- df2 %>%  
 mutate(sc\_likert = news1\_2\_sc + news1\_6\_sc + convo\_2\_sc + convo\_10\_sc +  
 topic\_interest\_3\_sc + topic\_interest\_5\_sc + topic\_interest\_8\_sc,  
 sc\_books = as.numeric(case\_when(  
 books\_3\_sc > 0 ~ "1",  
 is.na(books\_3\_sc) == T ~ "0")),  
 sc\_lecture = as.numeric(case\_when(  
 act10c1\_2\_sc > 0 ~ "1",  
 is.na(act10c1\_2\_sc) == T ~ "0")),  
 sc\_scimu = if\_else(act\_scib\_sc > 0, 1, 0),  
 sc\_scimu\_c19 = if\_else(act\_scic\_sc > 0, 1, 0),  
 sc\_read = if\_else(read\_article\_sc == 2, 1, 0),  
 sc\_share\_yn = if\_else(share\_freq1\_sc == 1, 0, 1),  
 sc\_share\_tech\_na = recode(share\_freq2\_6\_sc, `1` = 1,  
 `2` = 2,  
 `3` = 2,  
 `4` = 2,  
 `5` = 3,  
 `6` = 3,  
 `7` = 4,  
 `8` = 4,  
 `9` = 5,  
 `10` = 5),  
 sc\_share\_tech = ifelse(is.na(sc\_share\_tech\_na) == T, 0, sc\_share\_tech\_na),  
 sc\_share\_sci\_na = recode(share\_freq2\_11\_sc, `1` = 1,  
 `2` = 2,  
 `3` = 2,  
 `4` = 2,  
 `5` = 3,  
 `6` = 3,  
 `7` = 4,  
 `8` = 4,  
 `9` = 5,  
 `10` = 5),  
 sc\_share\_sci = ifelse(is.na(sc\_share\_sci\_na) == T, 0, sc\_share\_sci\_na)  
 )

df3 %>%   
 select(contains("sc\_")) %>%   
 head(10)

## sc\_likert sc\_books sc\_lecture sc\_scimu sc\_scimu\_c19 sc\_read sc\_share\_yn  
## 1 24 0 0 0 0 0 1  
## 2 19 1 0 0 0 1 1  
## 3 14 0 0 1 0 0 0  
## 4 17 0 0 0 0 0 1  
## 5 19 1 0 0 0 0 0  
## 6 18 0 0 0 0 1 1  
## 7 23 0 0 0 1 0 1  
## 8 21 0 0 0 0 1 1  
## 9 22 0 0 1 0 0 0  
## 10 17 0 0 0 0 0 1  
## sc\_share\_tech\_na sc\_share\_tech sc\_share\_sci\_na sc\_share\_sci  
## 1 5 5 2 2  
## 2 NA 0 4 4  
## 3 NA 0 NA 0  
## 4 2 2 NA 0  
## 5 2 2 5 5  
## 6 1 1 5 5  
## 7 5 5 5 5  
## 8 2 2 1 1  
## 9 NA 0 2 2  
## 10 4 4 3 3

df3 <- df3 %>% # score sci cur  
 mutate(sc\_score1 = sc\_likert + sc\_books + sc\_lecture + sc\_scimu +   
 sc\_scimu\_c19 + sc\_read,  
 sc\_score2a = sc\_share\_tech + sc\_share\_sci,  
 sc\_score2b = if\_else(share\_freq1\_sc == 0, 0, sc\_score2a),  
 sc\_scored = sc\_score1 + sc\_score2b,  
 sc\_scoredz = scale(sc\_scored)) # z score

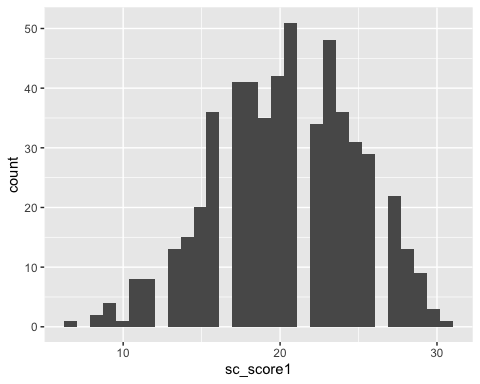
df3$sc\_scored %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 544 24.69 5.96 25 24.75 5.93 7 44 37 -0.06 0.04 0.26

df3 %>%   
 ggplot(aes(x = sc\_score1)) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1 rows containing non-finite values (stat\_bin).



## Score RPM (df4)

df4 <- df3 %>%   
 mutate(rav\_scored = ravens1\_b + ravens2\_b + ravens3\_b + ravens4\_b + ravens5\_b +  
 ravens6\_b + ravens7\_b + ravens8\_b + ravens9\_b + ravens10\_b,  
 rav\_scoredz = scale(rav\_scored))  
  
# df4 %>%   
# select(contains("rav"), -contains("click"), -contains("submit"))  
  
df4 %>%   
 filter(is.na(rav\_scored) == T) %>%   
 nrow() # gotta check this out

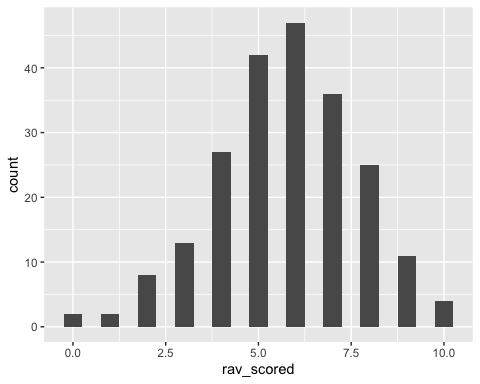
## [1] 328

df4$rav\_scored %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 217 5.75 1.93 6 5.81 1.48 0 10 10 -0.26 0.02 0.13

df4 %>%  
 ggplot(aes(x = rav\_scored)) +  
 geom\_histogram(binwidth = .5)

## Warning: Removed 328 rows containing non-finite values (stat\_bin).



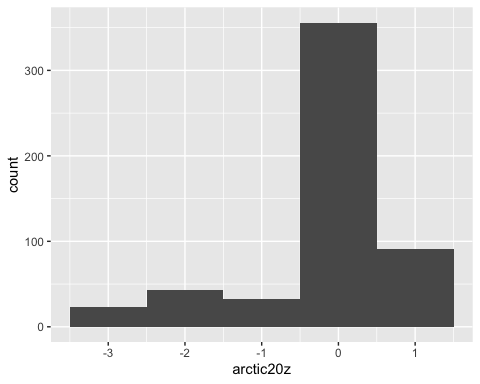
## MR Task (df5)

df5 <- df4 %>%   
 mutate(arctic20e = dplyr::recode(arctic2\_1\_sc, `1` = 3,  
 `2` = 2,  
 `3` = 1,  
 `4` = 0,  
 `5` = -1,  
 `6` = -2,  
 `7` = -3),  
 arctic20z = scale(arctic20e),  
 temp20e = dplyr::recode(temp2\_1\_sc,`1` = -3,  
 `2` = -2,  
 `3` = -1,  
 `4` = 0,  
 `5` = 1,  
 `6` = 2,  
 `7` = 3),  
 temp20z = scale(temp20e),  
 ozone20e = dplyr::recode(ozone2\_1\_sc, `1` = 3,  
 `2` = 2,  
 `3` = 1,  
 `4` = 0,   
 `5` = -1,  
 `6` = -2,  
 `7` = -3),  
 ozone20z = scale(ozone20e),  
 airqual20e = dplyr::recode(air\_qual2\_1\_sc, `1` = 3,  
 `2` = 2,  
 `3` = 1,  
 `4` = 0,  
 `5` = -1,  
 `6` = -2,  
 `7` = -3),  
 airqual20z = scale(airqual20e),  
 diatho20e = dplyr::recode(dia\_tho2\_1\_sc, `1` = 3,  
 `2` = 2,  
 `3` = 1,  
 `4` = 0,  
 `5` = -1,  
 `6` = -2,  
 `7` = -3),  
 diatho20z = scale(diatho20e),  
 co220e = dplyr::recode(co22\_1\_sc, `1` = -3,  
 `2` = -2,  
 `3` = -1,  
 `4` = 0,  
 `5` = 1,  
 `6` = 2,  
 `7` = 3),  
 co220z = scale(co220e),  
 icesheets20e = dplyr::recode(ice\_sheets2\_1\_sc, `1` = 3,  
 `2` = 2,  
 `3` = 1,  
 `4` = 0,  
 `5` = -1,  
 `6` = -2,  
 `7` = -3),  
 icesheets20z = scale(icesheets20e),  
 quake20e = if\_else(quake2\_1\_sc == 0, 0, 1)  
 )

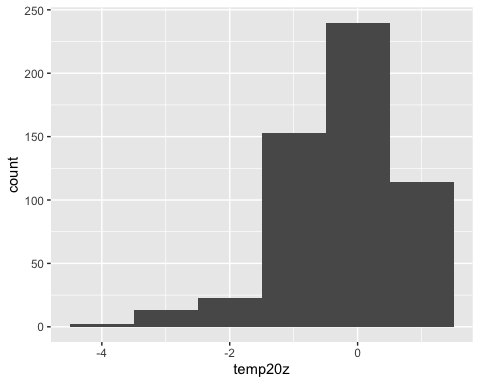
df5 %>%   
 select(contains("20e")) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew  
## arctic20e 1 545 1.41 1.32 2 1.57 1.48 -3 3 6 -1.10  
## temp20e 2 545 1.65 1.16 2 1.81 1.48 -3 3 6 -1.30  
## ozone20e 3 545 0.28 1.41 1 0.35 1.48 -3 3 6 -0.42  
## airqual20e 4 545 0.86 1.57 1 1.00 1.48 -3 3 6 -0.80  
## diatho20e 5 545 0.29 1.24 0 0.32 1.48 -3 3 6 -0.23  
## co220e 6 545 2.08 1.03 2 2.24 1.48 -3 3 6 -1.65  
## icesheets20e 7 545 2.07 1.21 2 2.30 1.48 -3 3 6 -1.86  
## quake20e 8 545 1.00 0.00 1 1.00 0.00 1 1 0 NaN  
## kurtosis se  
## arctic20e 0.81 0.06  
## temp20e 2.02 0.05  
## ozone20e -0.55 0.06  
## airqual20e -0.45 0.07  
## diatho20e -0.42 0.05  
## co220e 3.75 0.04  
## icesheets20e 3.86 0.05  
## quake20e NaN 0.00

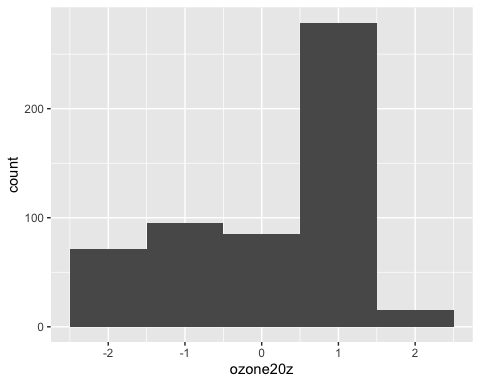
df5 %>%  
 ggplot(aes(x = arctic20z)) +  
 geom\_histogram(binwidth = 1)



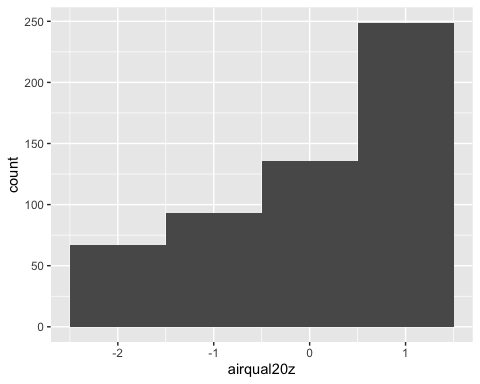
df5 %>%  
 ggplot(aes(x = temp20z)) +  
 geom\_histogram(binwidth = 1)



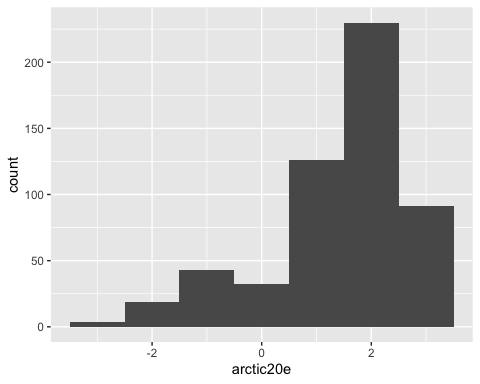
df5 %>%  
 ggplot(aes(x = ozone20z)) +  
 geom\_histogram(binwidth = 1)



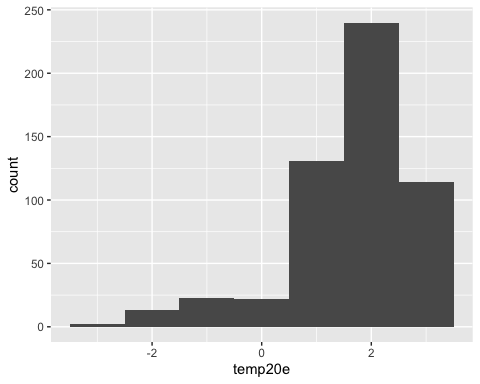
df5 %>%  
 ggplot(aes(x = airqual20z)) +  
 geom\_histogram(binwidth = 1)



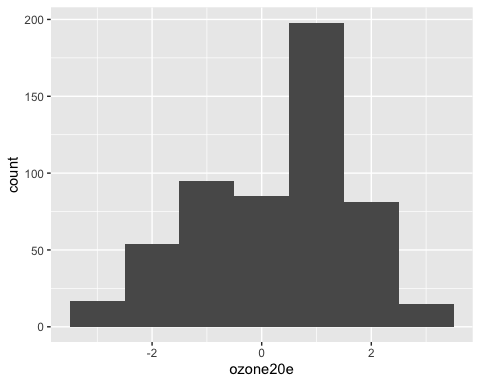
df5 %>%  
 ggplot(aes(x = arctic20e)) +  
 geom\_histogram(binwidth = 1)



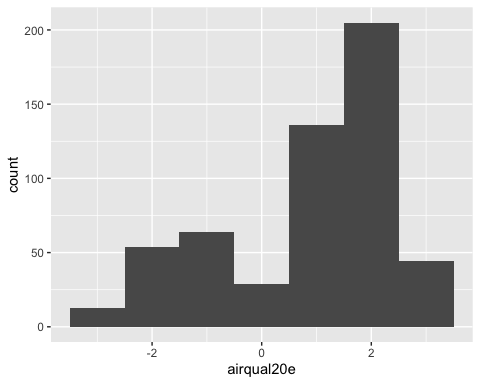
df5 %>%  
 ggplot(aes(x = temp20e)) +  
 geom\_histogram(binwidth = 1)



df5 %>%  
 ggplot(aes(x = ozone20e)) +  
 geom\_histogram(binwidth = 1)



df5 %>%  
 ggplot(aes(x = airqual20e)) +  
 geom\_histogram(binwidth = 1)



All of the MR items are very negatively skewed; most people exhibited endpoint bias. We preregistered coding the IMRI so that it was based on z > 1.5, but that isn’t helpful here with this distribution.

df5 <- df5 %>%   
 mutate(n\_plot\_score = diatho20e + co220e + icesheets20e,  
 n\_plot\_score\_z = scale(n\_plot\_score),  
 nq\_plot\_score = n\_plot\_score + quake20e,   
 nq\_plot\_score\_z = scale(nq\_plot\_score),  
 con\_plot = arctic20e + temp20e,  
 con\_plot\_z = scale(con\_plot),  
 lib\_plot = ozone20e + airqual20e,  
 lib\_plot\_z = scale(lib\_plot))  
  
df5 <- df5 %>%   
 mutate(bad\_plot\_score = ifelse(n\_plot\_score\_z <= -3, 1, 0),  
 bad\_plot\_score\_q = ifelse(nq\_plot\_score\_z <= -3, 1, 0)  
 ) # turns out no diff btwn no q and \_q so can just use no q

### Poor Plot Performance

Only 6 people (~1%) performed exceptionally poorly on the filler plot items.

### Attention Check (Practice Item)

A ton of people failed the attention check, which leads me to believe it was probably not the best attention check. I’ll see if this correlates with anything else important.

df5 <- df5 %>%   
 mutate(attn\_check = if\_else(prac2\_1\_sc == 6, 1, 0),  
 attn\_check\_12 = if\_else(prac1\_1\_sc == prac2\_1\_sc, 1, 0))  
  
df5 %>%   
 select(contains("attn\_check")) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew kurtosis  
## attn\_check 1 545 0.30 0.46 0 0.24 0 0 1 1 0.89 -1.20  
## attn\_check\_12 2 545 0.34 0.47 0 0.30 0 0 1 1 0.69 -1.52  
## se  
## attn\_check 0.02  
## attn\_check\_12 0.02

## Score IMRI

So this gets tricky because of the negative skew of the responses for these items. No one actually has z > 1.5 on any of them, so that scoring method isn’t helpful.

df5 <- df5 %>%  
 mutate(ebiasiz15\_arctic = ifelse(arctic20z >= 1.5, 1, 0),  
 ebiasiz15\_temp = ifelse(temp20z >= 1.5, 1, 0),  
 ebiasiz15\_ozone = ifelse(ozone20z >= 1.5, 1, 0),  
 ebiasiz15\_airqual = ifelse(airqual20z >= 1.5, 1, 0))  
  
df5 %>%   
 select(contains("ebiasiz15")) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew  
## ebiasiz15\_arctic 1 545 0.00 0.00 0 0 0 0 0 0 NaN  
## ebiasiz15\_temp 2 545 0.00 0.00 0 0 0 0 0 0 NaN  
## ebiasiz15\_ozone 3 545 0.03 0.16 0 0 0 0 1 1 5.76  
## ebiasiz15\_airqual 4 545 0.00 0.00 0 0 0 0 0 0 NaN  
## kurtosis se  
## ebiasiz15\_arctic NaN 0.00  
## ebiasiz15\_temp NaN 0.00  
## ebiasiz15\_ozone 31.24 0.01  
## ebiasiz15\_airqual NaN 0.00

### Coding for Above-Average

Here’s what happens if we code it based on whether or not they exhibit above-average endpoint bias (e.g., z > 0).

df5 <- df5 %>%   
 mutate(ebiasiz0\_arctic = ifelse(arctic20z >= 0, 1, 0),  
 ebiasiz0\_temp = ifelse(temp20z >= 0, 1, 0),  
 ebiasiz0\_ozone = ifelse(ozone20z >= 0, 1, 0),  
 ebiasiz0\_airqual = ifelse(airqual20z >= 0, 1, 0))  
  
df5 %>%   
 select(contains("ebiasiz0")) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew  
## ebiasiz0\_arctic 1 545 0.59 0.49 1 0.61 0 0 1 1 -0.36  
## ebiasiz0\_temp 2 545 0.65 0.48 1 0.69 0 0 1 1 -0.63  
## ebiasiz0\_ozone 3 545 0.54 0.50 1 0.55 0 0 1 1 -0.16  
## ebiasiz0\_airqual 4 545 0.71 0.46 1 0.76 0 0 1 1 -0.90  
## kurtosis se  
## ebiasiz0\_arctic -1.87 0.02  
## ebiasiz0\_temp -1.61 0.02  
## ebiasiz0\_ozone -1.98 0.02  
## ebiasiz0\_airqual -1.18 0.02

df5 <- df5 %>%   
 mutate(  
 ebiasiz0\_con = ebiasiz0\_arctic + ebiasiz0\_temp,  
 ebiasiz0\_lib = ebiasiz0\_ozone + ebiasiz0\_airqual,  
 ebiasiz0\_con\_yn = ifelse(ebiasiz0\_con == 2, 1, 0),  
 ebiasiz0\_lib\_yn = ifelse(ebiasiz0\_lib == 2, 1, 0),  
 )  
  
df5 %>%   
 select(ebiasiz0\_con, ebiasiz0\_lib, ebiasiz0\_con\_yn, ebiasiz0\_lib\_yn) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew  
## ebiasiz0\_con 1 545 1.24 0.80 1 1.30 1.48 0 2 2 -0.46  
## ebiasiz0\_lib 2 545 1.25 0.79 1 1.31 1.48 0 2 2 -0.46  
## ebiasiz0\_con\_yn 3 545 0.47 0.50 0 0.46 0.00 0 1 1 0.12  
## ebiasiz0\_lib\_yn 4 545 0.46 0.50 0 0.46 0.00 0 1 1 0.14  
## kurtosis se  
## ebiasiz0\_con -1.31 0.03  
## ebiasiz0\_lib -1.26 0.03  
## ebiasiz0\_con\_yn -1.99 0.02  
## ebiasiz0\_lib\_yn -1.98 0.02

df5 <- df5 %>%   
 mutate(imri\_iz0 = case\_when(  
 con == 1 ~ ifelse(ebiasiz0\_con\_yn == 1 & ebiasiz0\_lib\_yn == 0, 1, 0),  
 con == 0 ~ ifelse(ebiasiz0\_con\_yn == 0 & ebiasiz0\_lib\_yn == 1, 1, 0),  
 TRUE ~ 0  
 ))  
  
df5 %>%   
 filter(imri\_iz0 == 1) %>%   
 nrow()

## [1] 57

### Coding for Endpoint Bias

Here’s what happens if we code it solely based on whether or not they exhibit endpoint bias at all (e.g., score > 0; the higher the number, the more extreme their bias)

df5 <- df5 %>%   
 mutate(ebiasie\_arctic = ifelse(arctic20e > 0, 1, 0),  
 ebiasie\_temp = ifelse(temp20e > 0, 1, 0),  
 ebiasie\_ozone = ifelse(ozone20e > 0, 1, 0),  
 ebiasie\_airqual = ifelse(airqual20e > 0, 1, 0)  
 )  
  
df5 %>%   
 select(contains("ebiasie")) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew  
## ebiasie\_arctic 1 545 0.82 0.38 1 0.90 0 0 1 1 -1.66  
## ebiasie\_temp 2 545 0.89 0.31 1 0.99 0 0 1 1 -2.48  
## ebiasie\_ozone 3 545 0.54 0.50 1 0.55 0 0 1 1 -0.16  
## ebiasie\_airqual 4 545 0.71 0.46 1 0.76 0 0 1 1 -0.90  
## kurtosis se  
## ebiasie\_arctic 0.77 0.02  
## ebiasie\_temp 4.18 0.01  
## ebiasie\_ozone -1.98 0.02  
## ebiasie\_airqual -1.18 0.02

df5 <- df5 %>%   
 mutate(  
 ebiasie\_con = ebiasie\_arctic + ebiasie\_temp,  
 ebiasie\_lib = ebiasie\_ozone + ebiasie\_airqual,  
 ebiasie\_con\_yn = ifelse(ebiasie\_con == 2, 1, 0),  
 ebiasie\_lib\_yn = ifelse(ebiasie\_lib == 2, 1, 0),  
 )  
  
df5 %>%   
 select(ebiasie\_con, ebiasie\_lib, ebiasie\_con\_yn, ebiasie\_lib\_yn) %>%   
 psych::describe()

## vars n mean sd median trimmed mad min max range skew  
## ebiasie\_con 1 545 1.71 0.58 2 1.84 0.00 0 2 2 -1.87  
## ebiasie\_lib 2 545 1.25 0.79 1 1.31 1.48 0 2 2 -0.46  
## ebiasie\_con\_yn 3 545 0.78 0.42 1 0.84 0.00 0 1 1 -1.32  
## ebiasie\_lib\_yn 4 545 0.46 0.50 0 0.46 0.00 0 1 1 0.14  
## kurtosis se  
## ebiasie\_con 2.33 0.02  
## ebiasie\_lib -1.26 0.03  
## ebiasie\_con\_yn -0.25 0.02  
## ebiasie\_lib\_yn -1.98 0.02

df5 <- df5 %>%   
 mutate(imri\_ie = case\_when(  
 con == 1 ~ ifelse(ebiasie\_con\_yn == 1 & ebiasie\_lib\_yn == 0, 1, 0),  
 con == 0 ~ ifelse(ebiasie\_con\_yn == 0 & ebiasie\_lib\_yn == 1, 1, 0),  
 TRUE ~ 0  
 ))  
  
df5 %>%   
 filter(imri\_ie == 1) %>%   
 nrow()

## [1] 87

# Exclude (df6)

**Now excluding:** \* Duplicates \* Unfinished \* Reported cheating \* Reported bad response \* Unapproved \* Did not consent \* Was not in both dfs (b and s1) \* **Outlier duration** \* **Poor performance on plots**

df6 <- df5 %>%   
 filter(duration\_ex != 1 & bad\_plot\_score != 1)  
  
df6 %>%   
 nrow()

## [1] 538

df6 %>%   
 group\_by(con) %>%   
 summarise(n = n())

## # A tibble: 3 × 2  
## con n  
## <dbl> <int>  
## 1 -1 263  
## 2 0 51  
## 3 1 224

**Now excluding:** \* Duplicates \* Unfinished \* Reported cheating \* Reported bad response \* Unapproved \* Did not consent \* Was not in both dfs (b and s1) \* **Outlier duration** \* **Poor performance on plots** \* **Failed attention check**

df6attn <- df5 %>%   
 filter(attn\_check != 1) # basically not feasible bc it's so many ppl  
  
df6attn %>%   
 nrow()

## [1] 384

df6attn %>%   
 group\_by(con) %>%   
 summarise(n = n())

## # A tibble: 3 × 2  
## con n  
## <dbl> <int>  
## 1 -1 181  
## 2 0 38  
## 3 1 165

**Final df excluding:** \* Duplicates \* Unfinished \* Reported cheating \* Reported bad response \* Unapproved \* Did not consent \* Was not in both dfs (b and s1) \* Outlier duration \* Performed poorly on filler plot items

# IMRI Log Model

## Coding for Above Average

### w/o contol

imriiz0\_model1 = glm(data = df6, formula = as.factor(imri\_iz0) ~ sc\_scored, family = "binomial")  
summary(imriiz0\_model1)

##   
## Call:  
## glm(formula = as.factor(imri\_iz0) ~ sc\_scored, family = "binomial",   
## data = df6)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.6205 -0.4888 -0.4535 -0.4126 2.3823   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.1933 0.5775 -2.066 0.0388 \*  
## sc\_scored -0.0396 0.0237 -1.671 0.0948 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 359.14 on 536 degrees of freedom  
## Residual deviance: 356.34 on 535 degrees of freedom  
## (1 observation deleted due to missingness)  
## AIC: 360.34  
##   
## Number of Fisher Scoring iterations: 5

### w/ control

imriiz0\_model2 = glm(data = df6, formula = as.factor(imri\_iz0) ~ sc\_scored + conservatism\_f, family = "binomial")  
summary(imriiz0\_model2)

##   
## Call:  
## glm(formula = as.factor(imri\_iz0) ~ sc\_scored + conservatism\_f,   
## family = "binomial", data = df6)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.9814 -0.5370 -0.3312 -0.2487 2.3576   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.56978 0.76829 -4.646 3.38e-06 \*\*\*  
## sc\_scored -0.02656 0.02457 -1.081 0.28   
## conservatism\_f 0.47930 0.09138 5.245 1.56e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 359.14 on 536 degrees of freedom  
## Residual deviance: 324.18 on 534 degrees of freedom  
## (1 observation deleted due to missingness)  
## AIC: 330.18  
##   
## Number of Fisher Scoring iterations: 5

## Coding for Endpoint Bias

### w/o control

imriie\_model1 = glm(data = df6, formula = as.factor(imri\_ie) ~ sc\_scored, family = "binomial")  
summary(imriie\_model1)

##   
## Call:  
## glm(formula = as.factor(imri\_ie) ~ sc\_scored, family = "binomial",   
## data = df6)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.7074 -0.6080 -0.5798 -0.5395 2.0442   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.02462 0.49038 -2.089 0.0367 \*  
## sc\_scored -0.02591 0.01975 -1.312 0.1896   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 472.47 on 536 degrees of freedom  
## Residual deviance: 470.75 on 535 degrees of freedom  
## (1 observation deleted due to missingness)  
## AIC: 474.75  
##   
## Number of Fisher Scoring iterations: 4

### w/ control

imriie\_model2 = glm(data = df6, formula = as.factor(imri\_ie) ~ sc\_scored + conservatism\_f, family = "binomial")  
summary(imriie\_model2)

##   
## Call:  
## glm(formula = as.factor(imri\_ie) ~ sc\_scored + conservatism\_f,   
## family = "binomial", data = df6)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.3315 -0.5931 -0.2570 -0.1740 2.0775   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.75982 0.73226 -6.500 8.02e-11 \*\*\*  
## sc\_scored -0.00784 0.02181 -0.359 0.719   
## conservatism\_f 0.74191 0.09161 8.099 5.55e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 472.47 on 536 degrees of freedom  
## Residual deviance: 378.03 on 534 degrees of freedom  
## (1 observation deleted due to missingness)  
## AIC: 384.03  
##   
## Number of Fisher Scoring iterations: 6