Generate clean US data

ddd

11/7/2020

- go back and delete dashes
- Mutual friends Tom edit
- decide what to do with unclassifiable race posts

NF

```
to.char.list <- c("participant_id", "post_tenure", "nf_group..y.n.", "event..y.n.")
```

```
participant.log.cdr1 <- read.csv("Data/RA Protocol/CDR/Wave 1 Participant Log.csv")

participant.log.cdr1$source <- "CDR 1"
participant.log.cdr1$participant_id <- as.character(participant.log.cdr1$participant_id)

newsfeed.df.cdr1 <- read.csv("Data/RA Protocol/CDR/Wave 1 NF.csv")
newsfeed.df.cdr1$source <- "CDR 1"

for (var in to.char.list) {
  newsfeed.df.cdr1[[var]] <- as.character(newsfeed.df.cdr1[[var]])
}

nrow(participant.log.cdr1) - length(unique(participant.log.cdr1$participant_id))
```

```
## [1] 3
```

```
participant.log.hdsl <- read.csv("Data/RA Protocol/HDSL/Facebook Study Master Sheet - HDSL - Participan

participant.log.hdsl$source <- "HDSL"
participant.log.hdsl$participant_id <- as.character(participant.log.hdsl$participant_id)

newsfeed.df.hdsl <- read.csv("Data/RA Protocol/HDSL/Facebook Study Master Sheet - HDSL - NF.csv")
newsfeed.df.hdsl$source <- "HDSL"

for (var in to.char.list) {
  newsfeed.df.hdsl[[var]] <- as.character(newsfeed.df.hdsl[[var]])
}

nrow(participant.log.hdsl) - length(unique(participant.log.hdsl$participant_id))
```

```
## [1] 2
```

```
participant.log.cdr2 <- read.csv("Data/RA Protocol/CDR/Wave 2 Participant Log.csv")
```

```
participant.log.cdr2$source <- "CDR 2"
participant.log.cdr2$participant_id <- as.character(participant.log.cdr2$participant_id)

newsfeed.df.cdr2 <- read.csv("Data/RA Protocol/CDR/Wave 2 NF.csv")
newsfeed.df.cdr2$source <- "CDR 2"

for (var in to.char.list) {
  newsfeed.df.cdr2[[var]] <- as.character(newsfeed.df.cdr2[[var]])
}

nrow(participant.log.cdr2) - length(unique(participant.log.cdr2$participant_id))
```

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

```
participant.log.cdr3 <- read.csv("Data/RA Protocol/CDR/Wave 3 Participant Log.csv")

participant.log.cdr3$source <- "CDR 3"
participant.log.cdr3$participant_id <- as.character(participant.log.cdr3$participant_id)

newsfeed.df.cdr3 <- read.csv("Data/RA Protocol/CDR/Wave 3 NF.csv")
newsfeed.df.cdr3$source <- "CDR 3"

for (var in to.char.list) {
  if(var %in% colnames(newsfeed.df.cdr3)) {
    newsfeed.df.cdr3[[var]] <- as.character(newsfeed.df.cdr3[[var]])
  }
}

nrow(participant.log.cdr3) - length(unique(participant.log.cdr3$participant_id))
```

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

```
participant.log.cdr4 <- read.csv("Data/RA Protocol/CDR/Wave 4 Participant Log.csv")

participant.log.cdr4$source <- "CDR 4"
participant.log.cdr4$participant_id <- as.character(participant.log.cdr4$participant_id)

newsfeed.df.cdr4 <- read.csv("Data/RA Protocol/CDR/Wave 4 NF.csv")
newsfeed.df.cdr4$source <- "CDR 4"

for (var in to.char.list) {
  if(var %in% colnames(newsfeed.df.cdr3)) {
  newsfeed.df.cdr4[[var]] <- as.character(newsfeed.df.cdr4[[var]])
  }
}

nrow(participant.log.cdr4) - length(unique(participant.log.cdr4$participant_id))
```

```
## [1] 0
```

```
newsfeed.combined.raw <- bind_rows(newsfeed.df.cdr1,
                                   newsfeed.df.hdsl,
                                   newsfeed.df.cdr2,
                                   newsfeed.df.cdr3,
                                   newsfeed.df.cdr4)
```

```
p.log.combined.raw <- bind_rows(participant.log.cdr1,
                                participant.log.hdsl,
                                participant.log.cdr2,
                                participant.log.cdr3,
                                participant.log.cdr4)

newsfeed.combined.raw <- newsfeed.combined.raw %>% filter(participant_id != "")
p.log.combined.raw <- p.log.combined.raw %>% filter(participant_id != "")

merged.combined.raw <- merge(p.log.combined.raw, newsfeed.combined.raw, by = c("participant_id", "source

(length(unique(newsfeed.combined.raw$participant_id)) - length(unique(merged.combined.raw$participant_i
```

```
## [1] 0.03896104
```

```
nrow(p.log.combined.raw)
```

```
## [1] 693
```

```
(nrow(newsfeed.combined.raw) - nrow(merged.combined.raw))/nrow(newsfeed.combined.raw)
```

```
## [1] 0.01845009
```

```
table(p.log.combined.raw$source)
```

```
##
## CDR 1 CDR 2 CDR 3 CDR 4  HDSL
##   250    64   125    52   202
```

```
# delete useless columns
merged.combined.raw$X <- NULL
merged.combined.raw$X.1 <- NULL
merged.combined.raw$X.2 <- NULL
merged.combined.raw$Mutual.Friends..Tom.Edit. <- NULL

# ignore the phantom row if there's no RA_id
#merged.combined.raw <- merged.combined.raw %>% filter(RA_id != "")

# remove commas and safely convert factor to numeric
  merged.combined.raw$total_friends <- as.numeric(gsub(",", "", merged.combined.raw$total_friends))
  #as.numeric(levels(participant.log$total_friends))[participant.log$total_friends] .... if it happens

# deal with gender variations
  merged.combined.raw$subject_gender_RA <- tolower(merged.combined.raw$subject_gender_RA)

# deal with race variations
    merged.combined.raw$subject_race_RA <- tolower(merged.combined.raw$subject_race_RA)

# ignore phantom rows
merged.combined.raw <- merged.combined.raw[!is.na(merged.combined.raw$preference..1.7.) & !is.na(merged

# clean up names
  colnames(merged.combined.raw) <- c("participant.id",
                                     "source",
                                     "RA_id",
                                     "date",
```

```
                            "start_time",
                            "end_time",
                            "subject_race_RA",
                            "subject_gender_RA",
                            "total_friends",
                            "p_log_notes",
                            "study_about",

                    "nf.order",
                    "preference",
                    "post.tenure",
                    "tenure.units",
                    "human",
                    "nf.group",
                    "unrest",
                    "event",
                    "poster.nf.race.1.ra",
                    "poster.nf.race.2.ra",
                    "poster.nf.gender",
                    "nf.ra.notes",
                    "familiarity",
                    "relationship",
                    "common.friends"
                    )

  merged.combined.raw$poster.nf.race.1.ra <- tolower(merged.combined.raw$poster.nf.race.1.ra)
  merged.combined.raw$poster.nf.race.2.ra <- tolower(merged.combined.raw$poster.nf.race.2.ra)
  merged.combined.raw$poster.nf.gender <- tolower(merged.combined.raw$poster.nf.gender)

# drop more phantom rows

  merged.combined.raw <- merged.combined.raw %>% filter(poster.nf.race.1.ra != 'wjoye')

  merged.combined.raw$poster.nf.race.1.ra <- ifelse(merged.combined.raw$poster.nf.race.1.ra %in% c("bla

merged.combined.raw$poster.nf.race.1.ra <- ifelse(merged.combined.raw$poster.nf.race.1.ra %in% c("asian

merged.combined.raw$race.in.group <- ((merged.combined.raw$poster.nf.race.1.ra == merged.combined.raw$su
merged.combined.raw$gender.in.group <- merged.combined.raw$poster.nf.gender == merged.combined.raw$subj

merged.combined.raw <- merged.combined.raw %>% filter(subject_race_RA != "")
merged.combined.raw <- merged.combined.raw %>% filter(preference <= 7)

merged.combined.raw <- merged.combined.raw %>% filter(nf.order <= 60)

merged.combined.raw$date <- paste0(merged.combined.raw$date, "/2020")
merged.combined.raw$date <- mdy(merged.combined.raw$date)

merged.combined.raw$primary.id <- paste0(merged.combined.raw$date, merged.combined.raw$participant.id)

length(unique(merged.combined.raw$primary.id))
```

```
## [1] 662
```

```
to.missing <- c("", "`", "rs", "Years", ",")
to.mins <- c("m", "maleinutes", "min", "mins", "minutes", "Minutes", "minw")
to.hours <- c("hours", "Hours")
to.days <- c("days", "Days", "day", "fays")

merged.combined.raw$tenure.units <- as.character(merged.combined.raw$tenure.units)

merged.combined.raw$tenure.units <- ifelse(merged.combined.raw$tenure.units %in% to.missing,
                                           NA,
                                           merged.combined.raw$tenure.units)

merged.combined.raw$tenure.units <- ifelse(merged.combined.raw$tenure.units %in% to.mins,
                                           "mins",
                                           merged.combined.raw$tenure.units)

merged.combined.raw$tenure.units <- ifelse(merged.combined.raw$tenure.units %in% to.hours,
                                           "hours",
                                           merged.combined.raw$tenure.units)

merged.combined.raw$tenure.units <- ifelse(merged.combined.raw$tenure.units %in% to.days,
                                           "days",
                                           merged.combined.raw$tenure.units)

merged.combined.raw$post.tenure <- as.numeric(as.character(merged.combined.raw$post.tenure))
```

```
## Warning: NAs introduced by coercion
```

```
# clean up
merged.combined.raw$nf.group <- as.numeric(as.character(merged.combined.raw$nf.group))
```

```
## Warning: NAs introduced by coercion
```

```
merged.combined.raw$nf.group <- with(merged.combined.raw, ifelse(nf.group %in% 0:1,
                                                                 nf.group,
                                                                 NA))

write.csv(merged.combined.raw, file = "Temp/Clean US Data NF.csv")
```

PYMK

```
pymk.df <- read.csv("Data/RA Protocol/CDR/PYMK.csv")
pymk.df$source <- "CDR 1"
```

```
pymk.df.hdsl <- read.csv("Data/RA Protocol/HDSL/Facebook Study Master Sheet - HDSL - PYMK.csv")
pymk.df.hdsl$source <- "HDSL"
```

```
pymk.combined.raw <- bind_rows(pymk.df.hdsl,
                               pymk.df)

pymk.combined.raw <- pymk.combined.raw[!is.na(pymk.combined.raw$familiarity..1.7.) & !is.na(pymk.combine

# clean up names
  colnames(pymk.combined.raw) <- c("participant_id",
```

```
                    "pymk.order",
                    "familiarity",
                    "mutual.friends",
                    "poster.pymk.race.1.ra",
                    "poster.pymk.race.2.ra",
                    "poster.pymk.gender",
                    "pymk.ra.notes",
                    "source")

  pymk.combined.raw$poster.pymk.race.1.ra <- tolower(pymk.combined.raw$poster.pymk.race.1.ra)
  pymk.combined.raw$poster.pymk.race.2.ra <- tolower(pymk.combined.raw$poster.pymk.race.2.ra)
  pymk.combined.raw$poster.pymk.gender <- tolower(pymk.combined.raw$poster.pymk.gender)

# drop more phantom rows

  pymk.combined.raw <- pymk.combined.raw %>% filter(participant_id != "")

  merged.combined.raw <- merge(p.log.combined.raw, pymk.combined.raw, by = c("participant_id", "source"

     merged.combined.raw$poster.pymk.race.1.ra <- ifelse(merged.combined.raw$poster.pymk.race.1.ra %in%

     merged.combined.raw$poster.pymk.race.2.ra <- ifelse(merged.combined.raw$poster.pymk.race.2.ra %in%

   merged.combined.raw$subject_race_RA <- tolower(merged.combined.raw$subject_race_RA)
   merged.combined.raw$subject_gender_RA <- tolower(merged.combined.raw$subject_gender_RA)

  merged.combined.raw$race.in.group <- ((merged.combined.raw$poster.pymk.race.1.ra == merged.combined.ra
merged.combined.raw$gender.in.group <- merged.combined.raw$poster.pymk.gender == merged.combined.raw$sub

merged.combined.raw$date <- paste0(merged.combined.raw$date, "/2020")
merged.combined.raw$date <- mdy(merged.combined.raw$date)

merged.combined.raw$primary.id <- paste0(merged.combined.raw$date, merged.combined.raw$participant_id)

merged.combined.raw <- merged.combined.raw %>% filter(familiarity <= 7)
merged.combined.raw <- merged.combined.raw %>% filter(!is.na(pymk.order))

  write.csv(merged.combined.raw, file = "Temp/Clean US Data PYMK.csv")

  dim(merged.combined.raw)
```

```
## [1] 25593    21
```

```
  length(unique(merged.combined.raw$participant_id))
```

```
## [1] 413
```

Recent

```
recent.cdr.1 <- read.csv("Data/RA Protocol/CDR/Recent 1.csv")
recent.cdr.1$source <- "CDR 2"
```

```
recent.cdr.2 <- read.csv("Data/RA Protocol/CDR/Recent 2.csv")
recent.cdr.2$source <- "CDR 4"

recent.combined.raw <- bind_rows(recent.cdr.1, recent.cdr.2)

recent.combined.raw <- recent.combined.raw[!is.na(recent.combined.raw$react) & !is.na(recent.combined.ra

# clean up names
  colnames(recent.combined.raw) <- c("participant_id",
                    "recent.order",
                    "react",
                    "comment",
                    "post_tenure",
                    "tenure_units",
                    "human",
                    "poster.recent.race",
                    "poster.recent.gender",
                    "notes",
                    "source")

  recent.combined.raw$poster.recent.race <- tolower(recent.combined.raw$poster.recent.race)
  recent.combined.raw$poster.recent.gender <- tolower(recent.combined.raw$poster.recent.gender)

# drop more phantom rows
  recent.combined.raw <- recent.combined.raw %>% filter(participant_id != "")
  recent.combined.raw$participant_id <- as.character(recent.combined.raw$participant_id)
   recent.combined.raw <- merge(p.log.combined.raw, recent.combined.raw, by = c("participant_id", "sour

   recent.combined.raw$poster.recent.race <- ifelse(recent.combined.raw$poster.recent.race %in% c("asia

   recent.combined.raw$subject_race_RA <- tolower(recent.combined.raw$subject_race_RA)
   recent.combined.raw$subject_gender_RA <- tolower(recent.combined.raw$subject_gender_RA)

    recent.combined.raw$race.in.group <- ((recent.combined.raw$poster.recent.race == recent.combined.ra
    recent.combined.raw$gender.in.group <- recent.combined.raw$poster.recent.gender == recent.combined.

  write.csv(recent.combined.raw, file = "Temp/Clean US Data Recent.csv")

  length(unique(recent.combined.raw$participant_id))
```

```
## [1] 102
```

Qualtrics data

```
nf.data.raw <- read.csv("Temp/Clean US Data NF.csv")
pymk.data.raw <- read.csv("Temp/Clean US Data PYMK.csv")

nf.data <- nf.data.raw %>% group_by(primary.id,
                                    subject_gender_RA,
                                    subject_race_RA,
                                    source,
                                    total_friends) %>% dplyr::summarise(recorded.posts = max(nf.order, na.rm =
```

```
                                                                    mean.rating = mean(preference, na.rm =
                                                                    sd.rating = sd(preference, na.rm = T)
                                                                    median.rating = median(preference, na
```

```
## `summarise()` regrouping output by 'primary.id', 'subject_gender_RA', 'subject_race_RA', 'source' (ov
```

```
pymk.data <- pymk.data.raw %>% group_by(primary.id) %>% dplyr::summarise(pymk.recorded.posts = max(pymk
                                                                    pymk.mean.rating = mean(familiarity, r
                                                                    pymk.sd.rating = sd(familiarity, na.rm
                                                                    pymk.median.rating = median(familiarit
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
nf.data <- nf.data %>% dplyr::rename(uni.id = primary.id)

qualtrics1 <- read.csv("Data/Self-Assessments/FB Standard.csv")

qualtrics2 <- read.csv("Data/Self-Assessments/FB Network Structure.csv")

combined.qualtrics <- bind_rows(qualtrics1, qualtrics2)

combined.qualtrics$date <- date(combined.qualtrics$StartDate)
```

```
## Warning: tz(): Don't know how to compute timezone for object of class factor;
## returning "UTC". This warning will become an error in the next major version of
## lubridate.
```

```
combined.qualtrics <- combined.qualtrics %>% filter(Status == "IP Address" &
                                                    Finished == "True" &
                                                    Q2 != "")

combined.qualtrics$primary.id <- paste0(combined.qualtrics$date, combined.qualtrics$subject_id)

merged <- merge(combined.qualtrics, nf.data, by.x = "primary.id", by.y = "uni.id", all.x = F, all.y = T

merged <- merge(merged, pymk.data, by = "primary.id", all.x = T, all.y = F)

merged <- merged %>% dplyr::rename(self.race = Q3,
                                   self.race.free = Q4,
                                   gender = Q5,
                                   age = Q6,
                                   education = Q7,
                                   covid.usage = Q9,
                                   last.login = Q10,
                                   pre.covid.usage = Q11)

merged$self.race <- as.character(merged$self.race)
merged$self.race <- ifelse(grepl(",",merged$self.race),"Two or more",merged$self.race)

dim(merged)
```

```
## [1] 666  42
```

```
write.csv(merged, file = "Temp/Clean US qualtrics data.csv")
```