SP Data Cleaning

Ruijian Maggie Lin

2025-02-25

Data Cleaning

Step 1: Setup & Data Import

- Imports longitudinal survey data from 9 rounds

```
round_1_data <- read_dta("Round_1_rename.dta") %>% arrange(spid)
round_2_data <- read_dta("Round_2_rename.dta") %>% arrange(spid)
round_3_data <- read_dta("Round_3_rename.dta") %>% arrange(spid)
round_4_data <- read_dta("Round_4_rename.dta") %>% arrange(spid)
round_5_data <- read_dta("Round_5_rename.dta") %>% arrange(spid)
round_6_data <- read_dta("Round_6_rename.dta") %>% arrange(spid)
round_7_data <- read_dta("Round_7_rename.dta") %>% arrange(spid)
round_8_data <- read_dta("Round_8_rename.dta") %>% arrange(spid)
round_9_data <- read_dta("Round_9_rename.dta") %>% arrange(spid)

#client_data <- read_dta("Finalmodeldataset.dta") %>% arrange(spid)
#client_data <- client_data %>% select(spid, round, finalres, fmhcx, finalres, fparlim, flsn, fnewhx, f
```

Step 2: Data Combination

- Creates standardized processing pipeline for each survey round
- Adds round numbering and ensures consistent column order
- Vertically stacks all rounds into one dataset

```
# Function to read, add round column, and arrange variables
process_round <- function(file, round_number) {
  read_dta(file) %>%
    mutate(round = round_number) %>%  # Add round column
    relocate(spid, round)  # Ensure spid and round are first
}

# Process and combine all rounds into one dataset
combined_data <- bind_rows(
  process_round("Round_1_rename.dta", 1),
  process_round("Round_2_rename.dta", 2),
  process_round("Round_3_rename.dta", 3),
  process_round("Round_4_rename.dta", 4),
```

```r
  process_round("Round_5_rename.dta", 5),
  process_round("Round_6_rename.dta", 6),
  process_round("Round_7_rename.dta", 7),
  process_round("Round_8_rename.dta", 8),
  process_round("Round_9_rename.dta", 9)
) %>%
  arrange(spid)  # Sort the final dataset by spid
```

```
## Warning: `..1$dresid` and `..2$dresid` have conflicting value labels.
## i Labels for these values will be taken from `..1$dresid`.
## x Values: 2, 3, and 4

## Warning: `..1$d2intvrage` and `..2$d2intvrage` have conflicting value labels.
## i Labels for these values will be taken from `..1$d2intvrage`.
## x Values: 1, 2, 3, 4, 5, and 6

## Warning: `..1$health` and `..2$health` have conflicting value labels.
## i Labels for these values will be taken from `..1$health`.
## x Values: 1, 2, 3, 4, and 5

## Warning: `..1$disescn1` and `..2$disescn1` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn1`.
## x Values: 1 and 2

## Warning: `..1$disescn2` and `..2$disescn2` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn2`.
## x Values: 1 and 2

## Warning: `..1$disescn3` and `..2$disescn3` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn3`.
## x Values: 1 and 2

## Warning: `..1$disescn4` and `..2$disescn4` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn4`.
## x Values: 1 and 2

## Warning: `..1$disescn5` and `..2$disescn5` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn5`.
## x Values: 1 and 2

## Warning: `..1$disescn6` and `..2$disescn6` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn6`.
## x Values: 1 and 2

## Warning: `..1$disescn7` and `..2$disescn7` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn7`.
## x Values: 1 and 2

## Warning: `..1$disescn8` and `..2$disescn8` have conflicting value labels.
## i Labels for these values will be taken from `..1$disescn8`.
## x Values: 1 and 2
```

Step 3: Variable Engineering

- Handles missing data using forward-fill for demographic variables
- Creates key variables:
  - **Dependent**: `finalres` (institutionalization status)
  - **Independent**: Mental health score (`fmhcx`), social participation (`fparlim`), health conditions (`fnewhx`), etc.
  - **Covariates**: Demographics (race, age), healthcare access, cognition
- Converts categorical variables to binary/ordinal scales
- Handles special missing value codes (-9, -8, etc.)

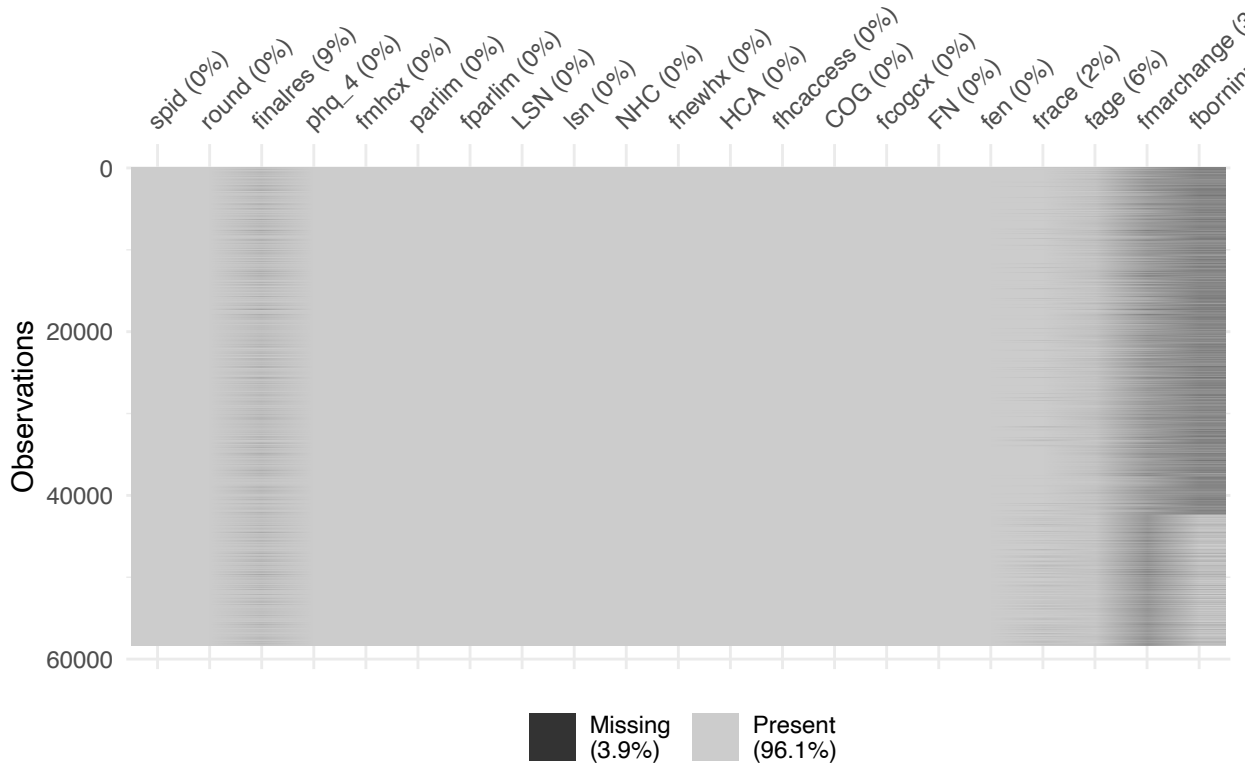
```
# Final Dataset
filtered_data <- combined_data %>%
  # Fill NA for the race variable
  group_by(spid) %>%
  arrange(round) %>%
  fill(dracehisp, .direction = "down") %>%
  fill(borninus, .direction = "down") %>%
  ungroup() %>%
  arrange(spid) %>%

  mutate(
    # Dependent Variable
    ## Transition to skilled care
    finalres = case_when(
      dresid %in% 1:3 ~ 0,    # Living in the community
      dresid %in% 4:5 ~ 1,    # Living in skilled nursing or hospitalized
      dresid %in% 6:8 ~ NA_real_ # Missing data
    ),

    # Mental Health Condition
    depresan1 = ifelse(depresan1 %in% c(-9, -8, -7, -1), NA, depresan1),
    depresan2 = ifelse(depresan2 %in% c(-9, -8, -7, -1), NA, depresan2),
    depresan3 = ifelse(depresan3 %in% c(-9, -8, -7, -1), NA, depresan3),
    depresan4 = ifelse(depresan4 %in% c(-9, -8, -7, -1), NA, depresan4),

    phq_4 = coalesce(depresan1, 0) + coalesce(depresan2, 0)
          + coalesce(depresan3, 0) + coalesce(depresan4, 0),
    fmhcx = ifelse(phq_4 >= 3, 1, 0),

    # Independent Variable
    ## Social participation
    hlkepfvst = ifelse(hlkepfvst %in% c(-9, -8, -7, -1), NA, hlkepfvst),
    trkpfrvis = ifelse(trkpfrvis %in% c(-9, -8, -7, -1), NA, trkpfrvis),
    htkfrrlsr = ifelse(htkfrrlsr %in% c(-9, -8, -7, -1), NA, htkfrrlsr),
    trprrelsr = ifelse(trprrelsr %in% c(-9, -8, -7, -1), NA, trprrelsr),
    hlkpfrclb = ifelse(hlkpfrclb %in% c(-9, -8, -7, -1), NA, hlkpfrclb),
    trprkpfgr = ifelse(trprkpfgr %in% c(-9, -8, -7, -1), NA, trprkpfgr),
    hlkpgoenj = ifelse(hlkpgoenj %in% c(-9, -8, -7, -1), NA, hlkpgoenj),
    trprgoout = ifelse(trprgoout %in% c(-9, -8, -7, -1), NA, trprgoout),
```

```
    helmfvact = ifelse(helmfvact %in% c(-9, -8, -7, -1, 95), NA, helmfvact),
    fvactlimyr = ifelse(fvactlimyr %in% c(-9, -8, -7, -1), NA, fvactlimyr),

    hlkepfvst = ifelse(hlkepfvst == 1, 1, ifelse(hlkepfvst == 2, 0, hlkepfvst)),
    trkpfrvis = ifelse(trkpfrvis == 1, 1, ifelse(trkpfrvis == 2, 0, trkpfrvis)),
    htkfrrlsr = ifelse(htkfrrlsr == 1, 1, ifelse(htkfrrlsr == 2, 0, htkfrrlsr)),
    trprrelsr = ifelse(trprrelsr == 1, 1, ifelse(trprrelsr == 2, 0, trprrelsr)),
    hlkpfrclb = ifelse(hlkpfrclb == 1, 1, ifelse(hlkpfrclb == 2, 0, hlkpfrclb)),
    trprkpfgr = ifelse(trprkpfgr == 1, 1, ifelse(trprkpfgr == 2, 0, trprkpfgr)),
    hlkpgoenj = ifelse(hlkpgoenj == 1, 1, ifelse(hlkpgoenj == 2, 0, hlkpgoenj)),
    trprgoout = ifelse(trprgoout == 1, 1, ifelse(trprgoout == 2, 0, trprgoout)),
    helmfvact = ifelse(helmfvact == 1, 1, ifelse(helmfvact == 2, 0, helmfvact)),
    fvactlimyr = ifelse(fvactlimyr == 1, 1, ifelse(fvactlimyr == 2, 0, fvactlimyr)),

    parlim = coalesce(hlkepfvst, 0) + coalesce(trkpfrvis, 0) +
         coalesce(htkfrrlsr, 0) + coalesce(trprrelsr, 0) +
         coalesce(hlkpfrclb, 0) + coalesce(trprkpfgr, 0) +
         coalesce(hlkpgoenj, 0) + coalesce(trprgoout, 0) +
         coalesce(helmfvact, 0) + coalesce(fvactlimyr, 0),
    fparlim = ifelse(parlim >= 1, 1, 0),

    ## Limited social network
    noonetalk = ifelse(noonetalk %in% c(-9, -8, -7, -1), NA, noonetalk),
    dnumsn = ifelse(dnumsn %in% c(-9, -8, -7, -1), NA, dnumsn),

    LSN = coalesce(noonetalk, 0) + coalesce(dnumsn, 0),
    lsn = ifelse(LSN >= 1, 1, 0),

    ## Development of a new health or condition/ hospitalization
    health = ifelse(health %in% c(-9, -8, -7, -1), NA, health),
    disescn1 = ifelse(disescn1 %in% c(-9, -8, -7, -1), NA, disescn1),
    disescn2 = ifelse(disescn2 %in% c(-9, -8, -7, -1, 7), NA, disescn2),
    disescn3 = ifelse(disescn3 %in% c(-9, -8, -1, 7), NA, disescn3),
    disescn4 = ifelse(disescn4 %in% c(-9, -8, -7, -1, 7), NA, disescn4),
    disescn5 = ifelse(disescn5 %in% c(-9, -8, -7, -1, 7), NA, disescn5),
    disescn6 = ifelse(disescn6 %in% c(-9, -8, -7, -1, 7), NA, disescn6),
    disescn7 = ifelse(disescn7 %in% c(-9, -8, -7, -1, 7), NA, disescn7),
    disescn8 = ifelse(disescn8 %in% c(-9, -8, -7, -1), NA, disescn8),
    disescn9 = ifelse(disescn9 %in% c(-9, -8, -7, -1, 7), NA, disescn9),
    disescn10 = ifelse(disescn10 %in% c(-9, -8, -7, -1), NA, disescn10),
    hosptstay = ifelse(hosptstay %in% c(-9, -8, -7, -1), NA, hosptstay),

    disescn1 = ifelse(disescn1 == 1, 1, ifelse(disescn1 == 2, 0, disescn1)),
    disescn2 = ifelse(disescn2 == 1, 1, ifelse(disescn2 == 2, 0, disescn2)),
    disescn3 = ifelse(disescn3 == 1, 1, ifelse(disescn3 == 2, 0, disescn3)),
    disescn4 = ifelse(disescn4 == 1, 1, ifelse(disescn4 == 2, 0, disescn4)),
    disescn5 = ifelse(disescn5 == 1, 1, ifelse(disescn5 == 2, 0, disescn5)),
    disescn6 = ifelse(disescn6 == 1, 1, ifelse(disescn6 == 2, 0, disescn6)),
    disescn7 = ifelse(disescn7 == 1, 1, ifelse(disescn7 == 2, 0, disescn7)),
    disescn8 = ifelse(disescn8 == 1, 1, ifelse(disescn8 == 2, 0, disescn8)),
    disescn9 = ifelse(disescn9 == 1, 1, ifelse(disescn9 == 2, 0, disescn9)),
    disescn10 = ifelse(disescn10 == 1, 1, ifelse(disescn10 == 2, 0, disescn10)),
    hosptstay = ifelse(hosptstay == 1, 1, ifelse(hosptstay == 2, 0, hosptstay)),
```

```
  NHC = coalesce(health, 0) + coalesce(disescn1, 0) +
    coalesce(disescn2, 0) + coalesce(disescn3, 0) +
    coalesce(disescn4, 0) + coalesce(disescn5, 0) +
    coalesce(disescn6, 0) + coalesce(disescn7, 0) +
    coalesce(disescn8, 0) + coalesce(disescn9, 0) +
    coalesce(disescn10, 0) + coalesce(hosptstay, 0),
  fnewhx = ifelse(NHC >= 1, 1, 0),

  ## Health care access
  havregdoc = ifelse(havregdoc %in% c(-9, -8, -7, -1), NA, havregdoc),
  regdoclyr = ifelse(regdoclyr %in% c(-9, -8, -7, -1), NA, regdoclyr),

  havregdoc = ifelse(havregdoc == 1, 1, ifelse(havregdoc == 2, 0, havregdoc)),
  regdoclyr = ifelse(regdoclyr == 1, 1, ifelse(regdoclyr == 2, 0, regdoclyr)),

  HCA = coalesce(havregdoc, 0) + coalesce(regdoclyr, 0),
  fhcaccess = ifelse(HCA >= 1, 1, 0),

  ## Cognition
  ratememry = ifelse(ratememry %in% c(-9, -8, -7, -1), NA, ratememry),
  ofmemprob = ifelse(ofmemprob %in% c(-9, -8, -7, -1), NA, ofmemprob),
  memcom1yr = ifelse(memcom1yr %in% c(-9, -8, -7, -1), NA, memcom1yr),

  COG = coalesce(ratememry, 0) + coalesce(ofmemprob, 0) + coalesce(memcom1yr, 0),
  fcogcx = ifelse(COG >= 1, 1, 0),

  ## Financial need
  progneed1 = ifelse(progneed1 %in% c(-9, -8, -7, -1), NA, progneed1),
  progneed2 = ifelse(progneed2 %in% c(-9, -8, -7, -1), NA, progneed2),
  progneed3 = ifelse(progneed3 %in% c(-9, -8, -7, -1), NA, progneed3),

  progneed1 = ifelse(progneed1 == 1, 1, ifelse(progneed1 == 2, 0, progneed1)),
  progneed2 = ifelse(progneed2 == 1, 1, ifelse(progneed2 == 2, 0, progneed2)),
  progneed3 = ifelse(progneed3 == 1, 1, ifelse(progneed3 == 2, 0, progneed3)),

  FN = coalesce(progneed1, 0) + coalesce(progneed2, 0) + coalesce(progneed3, 0),
  fen = ifelse(FN >= 1, 1, 0),

  ## Race
  frace = case_when(
    dracehisp == 1 ~ "White, non-Hispanic",
    dracehisp == 2 ~ "Black, non-Hispanic",
    dracehisp == 3 ~ "Hispanic",
    dracehisp == 4 ~ "Other",
    TRUE ~ NA_character_  # Default case for missing values
  ),

  ## Age
  fage = case_when(
    d2intvrage == 1 ~ "65-69",
    d2intvrage == 2 ~ "70-74",
    d2intvrage == 3 ~ "75-79",
    d2intvrage == 4 ~ "80-84",
```

```
      d2intvrage == 5 ~ "85-89",
      d2intvrage == 6 ~ "90+",
      TRUE ~ NA_character_  # Default case for missing values
    ),

    ## Change in marital status
    marchange = ifelse(marchange %in% c(-9, -8, -7, -1), NA, marchange),
    fmarchange = ifelse(marchange == 1, 1, ifelse(marchange == 2, 0, marchange)),

    ## Born in the US
    borninus = ifelse(borninus %in% c(-9, -8, -7, -1), NA, borninus),
    fborninus = ifelse(borninus == 1, 1, ifelse(borninus == 2, 0, borninus)),

  ) %>%
  select(spid, round, finalres, fmhcx, finalres, fparlim, lsn, fnewhx, fhcaccess,
         fcogcx, fen, frace, fage, fmarchange, fborninus, dresid, phq_4,
         depresan1, depresan2, depresan3, depresan4, parlim, hlkepfvst, trkpfrvis,
         htkfrrlsr, trprrelsr, hlkpfrclb, trprkpfgr, hlkpgoenj, trprgoout,
         helmfvact, fvactlimyr, LSN, noonetalk, dnumsn, NHC, health, disescn1,
         disescn2, disescn3, disescn4, disescn5, disescn6, disescn7, disescn8,
         disescn9, disescn10, hosptstay, HCA, havregdoc, regdoclyr, COG,
         rateememry, ofmemprob, memcom1yr, FN, progneed1, progneed2, progneed3,
         dracehisp, d2intvrage, marchange, borninus)

write.csv(filtered_data, "SP_rename.csv", row.names = FALSE)
```

```
final_data <- read.csv("SP_rename.csv")

final_data <- final_data %>%
  select(spid, round, finalres, phq_4, fmhcx, finalres, parlim, fparlim, LSN, lsn, NHC, fnewhx, HCA, fh

# Missingness of raw dataset
vis_miss(final_data, warn_large_data = FALSE)
```




```
number_of_participants <- final_data %>%
  summarize(num_participants = n_distinct(spid))

print(number_of_participants)
```

Number of Participants in Raw Dataset

```
##   num_participants
## 1            12427
```

There are total 12427 participants within the dataset.

Inclusion & Exclusion

```
# Participants with mental health condition
with_mhc_data <- final_data %>%
  group_by(spid) %>%
  filter(any(fmhcx == 1)) %>%
  ungroup()
```

```r
# Count participants with mental health condition
count_mhc_participants <- with_mhc_data %>%
  distinct(spid) %>%
  count()
count_mhc_participants
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1 11551
```

- Focuses on participants with mental health conditions (fmhcx == 1)

There are 11551 participants with in mental health condition.

```r
# Replace NA in finalres with 0
with_mhc_data <- with_mhc_data %>%
  mutate(finalres = ifelse(is.na(finalres), 0, finalres))

# Filter participants to exclude rounds AFTER first institutionalization
# and retain only those with >= 2 valid rounds
with_multiple_rounds <- with_mhc_data %>%
  group_by(spid) %>%
# Identify the first round where institutionalization occurred
  mutate(
    first_inst_round = if_else(
      any(finalres == 1),
      min(round[finalres == 1], na.rm = TRUE),  # First round with finalres = 1
      NA_integer_  # NA if never institutionalized
    )
  ) %>%
  # Keep:
  # - All rounds for never-institutionalized participants (first_inst_round = NA)
  # - Rounds <= first_inst_round for institutionalized participants (includes the first detection)
  filter(is.na(first_inst_round) | round <= first_inst_round) %>%
  # Remove participants with fewer than 2 remaining rounds
  filter(n() >= 2) %>%
  ungroup()
```

```
## Warning: There were 11070 warnings in `mutate()`.
## The first warning was:
## i In argument: `first_inst_round = if_else(...)`.
## i In group 1: `spid = 10000003`.
## Caused by warning in `min()`:
## ! no non-missing arguments to min; returning Inf
## i Run `dplyr::last_dplyr_warnings()` to see the 11069 remaining warnings.
```

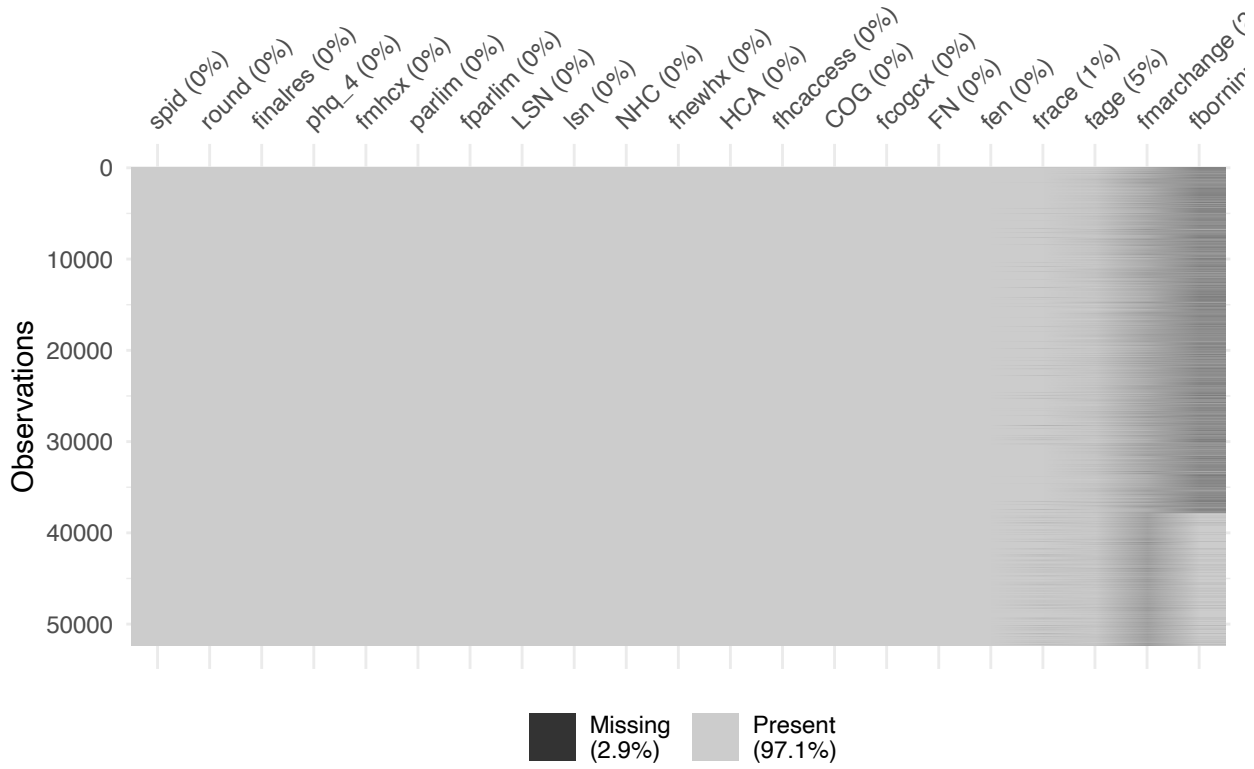
```r
# Count unique participants meeting criteria
count_2_participants <- with_multiple_rounds %>%
  distinct(spid) %>%
  tally(name = "n_participants")

count_2_participants
```

```
## # A tibble: 1 x 1
##   n_participants
##            <int>
## 1           9844
```

- Excludes survey rounds after first institutionalization
- Requires 2 valid observations per participant

There are 9844 participants with mental health condition & Complete at least 2 rounds of Survey before institutionalization.

```
final_data <- with_multiple_rounds%>%
  select(-first_inst_round)

write.csv(final_data, "SP_rename_final.csv", row.names = FALSE)

vis_miss(final_data, warn_large_data = FALSE)
```



- Saves final analysis-ready dataset