GMS6025C: Final Project 1 Part 1 Preprocessing Data

Emely Gazarov, Ryo Iwata, Maria Ramirez

05 May, 2024

About the data

Name: Addiction Patient Assessment Measures

Description: Data are gathered from a residential treatment facility for substance use disorders. Data include item-level responses for all questionnaires listed below. Data are available at initiation of treatment, after one month of treatment, and at treatment discharge.

Inputting the data

```
# Reading in the data

# Predictor
## Basic demographics (age, gender, education)
## Duration of sobriety prior to treatment
raw_demographics <- read_delim("./data/demo.damon.csv",
                               delim = ",",
                               progress = FALSE,
                               show_col_types = FALSE)

## Specific SUDs (e.g., alcohol use disorder)
raw_SUD <- read_delim("./data/SUDdiagnosis.damon.csv",
                               delim = ",",
                               progress = FALSE,
                               show_col_types = FALSE)

## Social Support (MSPSS)
raw_social <- read_delim("./data/mspss.damon.csv",
                               delim = ",",
                               progress = FALSE,
                               show_col_types = FALSE)

## Substance Use History
raw_sub_history <- read_delim("./data/subuse.damon.csv",
                               delim = ",",
                               progress = FALSE,
                               show_col_types = FALSE)

# problems(raw_sub_history)
```

```r
## AA/NA Affiliation
raw_aana_affiliation <- read_delim("./data/aana.damon.csv",
                                   delim = ",",
                                   progress = FALSE,
                                   show_col_types = FALSE)

# problems(raw_aana_affiliation)
```

```r
## Stressful Life Experiences (LEC-5) lec.damon

# *The structure of this dataframe is really bonkers.
#You probably want to use the variable "toyou_total"
# which is a sum of event types that the patients endorsed as having happened to them.
# We sometimes also use "toyou_wit_total", which is a similar sum score,
# but includes events that have happened to the participant
#AND events that the participant has witnessed.

raw_stress  <- read_delim("./data/lec.damon.csv",
                                   delim = ",",
                                   progress = FALSE,
                                   show_col_types = FALSE)

## Spiritual Experiences (Brief R-COPE)
raw_spiritual <- read_delim("./data/r_cope.damon.csv",
                                   delim = ",",
                                   progress = FALSE,
                                   show_col_types = FALSE)

## Childhood Experiences (ACE)
raw_childhood <- read_delim("./data/aces.damon.csv",
                                   delim = ",",
                                   progress = FALSE,
                                   show_col_types = FALSE)
```

```r
# Things to predict

## Quality of Life (WHOQOL-BREF):
# Evaluates general, physical, psych health, social relationships, and environment.

raw_life_quality <- read_delim("./data/QOL.damon.csv",
                                   delim = ",",
                                   progress = FALSE,
                                   show_col_types = FALSE)

raw_commitment <- read_delim("./data/change.damon.csv",
                                  delim = ",",
                                  progress = FALSE,
                                  show_col_types = FALSE)
## Alcohol/Drug Craving (PACS): Measures the frequency and intensity of cravings.
raw_craving <- read_delim("./data/craving.damon.csv",
                             delim = ",",
                             progress = FALSE,
                             show_col_types = FALSE)
```

```
## Treatment dropout
## demo.damon (dropout_yn)
## Length of stay in treatment
## demo.damon (days_in_tx_clean)

# Other
raw_data_dictionary <- read_delim("./data/Data Dictionary.csv",
                                  delim = ",",
                                  progress = FALSE,
                                  show_col_types = FALSE)
```

Getting baseline, followup, discharge for all datasets

```
# Filtering for subjects that are in each time

discharged_demo <- raw_demographics |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_demo <- raw_demographics |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_demo <- raw_demographics |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

# Joining each filtered timepoint within a dataset based on subject

joined_demographics <- baseline_demo |>
  inner_join(discharged_demo, by="record_id") |>
  inner_join(followup_demo, by="record_id")
```

```
# Filtering for subjects that are in each time

discharged_SUD <- raw_SUD |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_SUD <- raw_SUD |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_SUD <- raw_SUD |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_SUD <- baseline_SUD |>
  inner_join(discharged_SUD, by="record_id") |>
  inner_join(followup_SUD, by="record_id")
```

```
# Filtering for subjects that are in each time
```

```r
discharged_social <- raw_social |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_social <- raw_social |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_social <- raw_social |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_social <- baseline_social |>
  inner_join(discharged_social, by="record_id") |>
  inner_join(followup_social, by="record_id")
```

```r
# Filtering for subjects that are in each time

discharged_sub_history <- raw_sub_history |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_sub_history <- raw_sub_history |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_sub_history <- raw_sub_history |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_sub_history <- baseline_sub_history |>
  inner_join(discharged_sub_history, by="record_id") |>
  inner_join(followup_sub_history, by="record_id")
```

```r
# Filtering for subjects that are in each time

discharged_aana_affiliation <- raw_aana_affiliation |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_aana_affiliation <- raw_aana_affiliation |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_aana_affiliation <- raw_aana_affiliation |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_aana_affiliation <- baseline_aana_affiliation |>
  inner_join(discharged_aana_affiliation, by="record_id") |>
  inner_join(followup_aana_affiliation, by="record_id")
```

```r
# Filtering for subjects that are in each time
```

```r
discharged_stress <- raw_stress |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_stress <- raw_stress |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_stress <- raw_stress |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_stress <- baseline_stress |>
  inner_join(discharged_stress, by="record_id") |>
  inner_join(followup_stress, by="record_id")
```

```r
# Filtering for subjects that are in each time

discharged_spiritual <- raw_spiritual |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_spiritual <- raw_spiritual |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_spiritual <- raw_spiritual |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_spiritual <- baseline_spiritual |>
  inner_join(discharged_spiritual, by="record_id") |>
  inner_join(followup_spiritual, by="record_id")
```

```r
# Filtering for subjects that are in each time

discharged_childhood <- raw_childhood |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_childhood <- raw_childhood |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_childhood <- raw_childhood |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_childhood <- baseline_childhood |>
  inner_join(discharged_childhood, by="record_id") |>
  inner_join(followup_childhood, by="record_id")
```

```r
# Filtering for subjects that are in each time
```

```r
discharged_life_quality <- raw_life_quality |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_life_quality <- raw_life_quality |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_life_quality <- raw_life_quality |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_life_quality <- baseline_life_quality |>
  inner_join(discharged_life_quality, by="record_id") |>
  inner_join(followup_life_quality, by="record_id")
```

```r
# Filtering for subjects that are in each time

discharged_craving <- raw_craving |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_craving <- raw_craving |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_craving <- raw_craving |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_craving <- baseline_craving |>
  inner_join(discharged_craving, by="record_id") |>
  inner_join(followup_craving, by="record_id")
```

```r
# Filtering for subjects that are in each time

discharged_commitment <- raw_commitment |>
  filter(str_detect(redcap_event_name,'discharge')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".discharge")))

baseline_commitment <- raw_commitment |>
  filter(str_detect(redcap_event_name,'baseline')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".baseline")))

followup_commitment <- raw_commitment |>
  filter(str_detect(redcap_event_name,'followup')) |>
  rename_with(~ ifelse(.x == "record_id", .x, paste0(.x, ".followup")))

joined_commitment <- baseline_commitment |>
  inner_join(discharged_commitment, by="record_id") |>
  inner_join(followup_commitment, by="record_id")
```

Joining all

```
# Combining all the datasets by subject ID

all_addiction_data <- inner_join(joined_demographics,
                                 joined_aana_affiliation,
                                 by="record_id") |>
  inner_join(joined_commitment, by="record_id") |>
  inner_join(joined_childhood, by="record_id") |>
  inner_join(joined_craving, by="record_id") |>
  inner_join(joined_life_quality, by="record_id") |>
  inner_join(joined_social, by="record_id") |>
  inner_join(joined_spiritual, by="record_id") |>
  inner_join(joined_stress, by="record_id") |>
  inner_join(joined_sub_history, by="record_id") |>
  inner_join(joined_SUD, by="record_id")
```

Creating/renaming columns to remove redundancy and factoring as needed

```
factored_addiction_data <- all_addiction_data |>

  # Demographics
  mutate(age = age_today.baseline) |>
  mutate(gender = as_factor(gender.baseline)) |>
  mutate(education = factor(ed_summary.baseline, levels =
                              c("High School/GED or less",
                                "Associate's/Some College",
                                "Bachelor's",
                                "Master's",
                                "Doctoral"))) |>

  # 17a. # of sobriety days (calculated)
  mutate(sober_days = sobriety_calc.baseline) |>
  # B4. Number of days since baseline.
  mutate(treatment_days = days_in_tx_clean.baseline) |>
  mutate(dropout_yn = as_factor(dropout_yn.baseline)) |>

  # Making factors from the substance of concern
  mutate(SUD_is_Alchohol = as_factor(sa_dx___0.baseline)) |>
  mutate(SUD_is_Opioid = as_factor(sa_dx___1.baseline)) |>
  mutate(SUD_is_Cannabis = as_factor(sa_dx___2.baseline)) |>
  mutate(SUD_is_depressants_anxiolytic = as_factor(sa_dx___3.baseline)) |>
  mutate(SUD_is_Cocaine = as_factor(sa_dx___4.baseline)) |>
  mutate(SUD_is_Other_stimulant = as_factor(sa_dx___5.baseline)) |>
  mutate(SUD_is_Hallucinogen = as_factor(sa_dx___6.baseline)) |>
  mutate(SUD_is_Nicotine = as_factor(sa_dx___7.baseline)) |>
  mutate(SUD_is_Inhalant = as_factor(sa_dx___8.baseline)) |>
  mutate(SUD_is_psychoactive = as_factor(sa_dx___9.baseline)) |>
  mutate(SUD.sum = sa_dx_sum.baseline) |>

  # Feature engineering
```

```
## Substances

### Summing up the number of legal substances
mutate(SUD.sum_legal = as.numeric(as.character(SUD_is_Alchohol)) +
         as.numeric(as.character(SUD_is_Cannabis)) +
         as.numeric(as.character(SUD_is_Nicotine)) +
         as.numeric(as.character(SUD_is_Inhalant))) |>

### Boolean of if legal substance is used or not
mutate(SUD.uses_legal = as_factor(if_else(SUD.sum_legal >= 1, TRUE, FALSE))) |>

### Summing up the number of illegal substances
mutate(SUD.sum_illegal = as.numeric(as.character(SUD_is_Opioid)) +
         as.numeric(as.character(SUD_is_depressants_anxiolytic)) +
         as.numeric(as.character(SUD_is_Cocaine)) +
         as.numeric(as.character(SUD_is_Other_stimulant)) +
         as.numeric(as.character(SUD_is_psychoactive)) +
         as.numeric(as.character(SUD_is_Hallucinogen))
       ) |>

### Boolean of if illegal substance is used or not
mutate(SUD.uses_illegal = as_factor(if_else(SUD.sum_illegal >= 1, TRUE, FALSE))) |>


## AA/NA affiliation and regard

# 1. People at AA/NA could give me a lot of support
# 2. Going to AA/NA meetings can help me use some of my free time.
# 3. Going to AA/NA meetings would help me remember why I want to stay sober.
# 4. I could learn a lot by working on the Twelve Steps of AA or NA.
# 5. Being part of AA/NA would make me feel more hopeful.
# 6. Many people have encouraged me to go to AA or NA.
# 7. I would get bored easily at AA/NA meetings.
# 8. I would feel embarrassed going to an AA/NA meeting.
# 9. Going to AA or NA would depress me.
# 10. I would feel very nervous going to an AA/NA meeting.
# 11. I would not want to speak in front of a group at an AA/NA meeting.
# 12. I do not think I would like the people I meet at AA/NA.
# 13. I don't want people at AA or NA telling me how I should lead my life.
# 14. I don't want to hear other people talk about their problems at AA/NA meetings.
# 15. I feel very uncomfortable with the religious (or spiritual) aspects of AA/NA.
# 16. I don't have enough time to attend AA/NA meetings.
# TSPE Positive Score sum([tspe_1]+[tspe_2]+[tspe_3]+[tspe_4]+[tspe_5]+[tspe_6])
# TSPE Negative Score
# sum([tspe_7]+[tspe_8]+[tspe_9]+[tspe_10]+[tspe_11]+
# [tspe_12]+[tspe_13]+[tspe_14]+[tspe_15]+[tpse_16])
mutate(aana_positive = tspe_positive.baseline) |>
mutate(aana_negative = tspe_negative.baseline) |>
mutate(aana_life = as.factor(aaas_calc_lifetime.baseline)) |>
mutate(aana_past_year = as.factor(aaas_calc_past_year.baseline))  |>


## Childhood
```

```
#   "1. Did a parent or other adult in the household often or very often...
#
# Swear at you, insult you, put you down, or humiliate you?
#
#           OR
#
# Act in a way that made you afraid that you might be physically hurt?"
mutate(childhood.verbal_abuse = as_factor(ace1.baseline)) |>

# "2. Did a parent or other adult in the household often or very often...
#
# Push, grab, slap, or throw something at you?
#
#            OR
#
# Ever hit you so hard that you had marks or were injured?"
mutate(childhood.physical_abuse = as_factor(ace2.baseline)) |>

# "3. Did an adult or person at least 5 years older than you ever...
#
# Touch or fondle you or have you touch their body in a sexual way?
#
#            OR
#
# Attempt or actually have oral, anal, or vaginal intercourse with you?"
mutate(childhood.sexual_abuse = as_factor(ace3.baseline)) |>

# "4. Did you often or very often feel that ...
#
# No one in your family loved you or thought you were important or special?
#
#            OR
#
# Your family didn't look out for each other,
# feel close to each other, or support each other?"
mutate(childhood.alone = as_factor(ace4.baseline)) |>

# "5. Did you often or very often feel that ...
#
# You didn't have enough to eat, had to wear dirty clothes,
# and had no one to protect you?
#
#            OR
#
# Your parents were too drunk or high to take care of you or
# take you to the doctor if you needed it?"
mutate(childhood.neglected = as_factor(ace5.baseline)) |>

# 6. Were your parents ever separated or divorced?
mutate(childhood.divorced = as_factor(ace6.baseline)) |>

# "7. Was your mother or stepmother (or father/stepfather):
#
```

```
# Often or very often pushed, grabbed, slapped,
# or had something thrown at her/him?
#
#            OR
#
# Sometimes, often, or very often kicked, bitten, hit with a fist,
# or hit with something hard?
#
#            OR
#
# Ever repeatedly hit at least a few minutes
# or threatened with a gun or knife?"
mutate(childhood.parent_was_abused = as_factor(ace7.baseline)) |>

# 7a. For Q7, indicate which parent was violated
# 8. Did you live with anyone who was a problem drinker
# or alcoholic or who used street drugs?
mutate(childhood.other_was_addicted = as_factor(ace8.baseline)) |>

# 9. Was a household member depressed or mentally ill,
# or did a household member attempt suicide?
mutate(childhood.other_was_stressed = as_factor(ace9.baseline)) |>

# 10. Did a household member go to prison?
mutate(childhood.other_was_prisoned = as_factor(ace10.baseline)) |>

# Total childhood
mutate(childhood_sum = total_ace_score.baseline) |>


## Commitment

# 1. Staying sober is the most important thing in my life.
# 2. I am totally committed to staying off alcohol/drugs.
# 3. I will do whatever it takes to recover from my addiction.
# 4. I never want to return to alcohol/drug use again.
# 5. I have had enough alcohol and drugs.
# Commitment to Change Total
# sum([change_1], [change_2], [change_3],[change_4], [change_5])
mutate(commit.BL = commit_to_change.total.baseline) |>
mutate(commit.DC = commit_to_change.total.discharge) |>
mutate(commit.FU = commit_to_change.total.followup) |>


## Confidence in abstaining

# Please read each statement
# and indicate how confident you are RIGHT NOW
# that you would not choose to use a drug or drugs
# if they were readily available to you TODAY.
# 1, Not at all confident | 2, Not Very confident | 3, Moderately confident |
# 4, Very confident | 5, Extremely confident
```

```
# 1. When I am in agony because of stopping or withdrawing from drug use.
# 2. When I have a headache.
# 3. When I am feeling depressed.
# 4. When I am on vacation and want to relax.
# 5. When I am concerned about someone.
# 6. When I am worried.
# 7. When I have the urge to use drugs to see what happens.
# 8. When I am being offered drugs in a social situation.
# 9. When I dream about using drugs.
# 10. When I want to test my will power over using drugs.
# 11. When I am feeling a physical need or craving for drugs.
# 12. When I am physically tired.
# 13. When I am experiencing some physical pain or injury.
# 14. When I feel like blowing up because of frustration.
# 15. When I see others using drugs at a bar or a party.
# 16. When I sense everything is going wrong for me.
# 17. When people I used to use drugs with encourage me to use drugs.
# 18. When I am feeling angry inside.
# 19. When I experience an urge or
# impulse to use drugs that catches me unprepared.
# 20. When I am excited or celebrating with others.
# DASE Neg Affect Subscale
# sum([dase3],[dase6],[dase14],[dase16],[dase18])
# DASE Social/Positive Subscale
# sum([dase4],[dase8],[dase15],[dase17],[dase20])
# DASE Physical Subscale  sum([dase2],[dase5],[dase9],[dase12],[dase13])
# DASE Cravings and Urges Subscale
# sum([dase1],[dase7],[dase10],[dase11],[dase19])
mutate(abstain_neg.BL = daase_low_neg_mean.baseline) |>
mutate(abstain_pos.BL = daase_low_pos_mean.baseline) |>
mutate(abstain_phy.BL = daase_low_phy_mean.baseline) |>
mutate(abstain_crv.BL = daase_low_crv_mean.baseline) |>
mutate(abstain_total.BL = daase_low_tot_mean.baseline) |>
mutate(abstain_neg.FU = daase_low_neg_mean.followup) |>
mutate(abstain_pos.FU = daase_low_pos_mean.followup) |>
mutate(abstain_phy.FU = daase_low_phy_mean.followup) |>
mutate(abstain_crv.FU = daase_low_crv_mean.followup) |>
mutate(abstain_total.FU = daase_low_tot_mean.followup) |>
mutate(abstain_neg.DC = daase_low_neg_mean.discharge) |>
mutate(abstain_pos.DC = daase_low_pos_mean.discharge) |>
mutate(abstain_phy.DC = daase_low_phy_mean.discharge) |>
mutate(abstain_crv.DC = daase_low_crv_mean.discharge) |>
mutate(abstain_total.DC = daase_low_tot_mean.discharge) |>

# 1. During the past week
# HOW OFTEN have you thought about drinking or using drugs,
# or about how good drinking/using would make you feel?
# 2. At its most severe point, HOW STRONG was your craving during the past week?
# 3. During the past week HOW MUCH TIME have you spent thinking about drinking
# or using drugs, or about how good drinking/using would make you feel?
# 4. During the past week HOW DIFFICULT WOULD IT HAVE BEEN TO RESIST
# taking a drink or using drugs if you had known that alcohol
#or drugs were in your current place of residence?
```

```
# 5. Keeping in mind your responses to the previous questions,
# please rate your overall AVERAGE ALCOHOL AND DRUG CRAVING for the past week.
# PACS Total Score
mutate(craving.BL = pacs_mean.baseline) |>
mutate(craving.FU = pacs_mean.followup) |>
mutate(craving.DC = pacs_mean.discharge) |>


## Life Quality

# 1. How would you rate your quality of life?
# 2. How satisfied are you with your health?
# 3. To what extent do you feel that physical pain
# prevents you from doing what you need to do?
# 4. How much do you need any medical treatment to function in your daily life?
# 5. How much do you enjoy life?
# 6. To what extent do you feel your life to be meaningful?
# 7. How well are you able to concentrate?
# 8. How safe do you feel in your daily life?
# 9. How healthy is your physical environment?
# 10. Do you have enough energy for everyday life?
# 11. Are you able to accept your bodily appearance?
# 12. Have you enough money to meet your needs?
# 13. How available to you is the information you need in your day-to-day life?
# 14. To what extent do you have the opportunity for leisure activities?
# 15. How well are you able to get around?
# 16. How satisfied are you with your sleep?
# 17. How satisfied are you
# with your ability to perform your daily living activities?
# 18. How satisfied are you with your capacity for work?
# 19. How satisfied are you with yourself?
# 20. How satisfied are you with your personal relationships?
# 21. How satisfied are you with your sex life?
# 22. How satisfied are you with the support you get from your friends?
# 23. How satisfied are you with the conditions of your living place?
# 24. How satisfied are you with your access to health services?
# 25. How satisfied are you with your mode of transportation?
# 26. How often do you have negative feelings,
# such as blue mood, despair, anxiety, or depression?

# General Health  sum([who_1],[who_2])
# psych Health    sum([who_5],[who_6],[who_7],[who_11],[who_19],[who_26])
# Physical Health
# sum([who_3],[who_4],[who_10],[who_15],[who_16],[who_17],[who_18])
# Social Relationships    sum([who_20],[who_21],[who_22])
# Environment
# sum([who_8],[who_9],[who_12],[who_13],[who_14],[who_23],[who_24],[who_25])
mutate(qol_general_health.BL =  who_qol_gh_total.baseline) |>
mutate(qol_psych.BL =  who_psy_total.baseline) |>
mutate(qol_physical.BL =  who_ph_total.baseline) |>
mutate(qol_social.BL =  who_soc_rel_total.baseline) |>
mutate(qol_env.BL =  who_env_total.baseline) |>
mutate(qol.BL =  qol_mean.baseline) |>
```

```
mutate(qol_general_health.FU =  who_qol_gh_total.followup) |>
mutate(qol_psych.FU =  who_psy_total.followup) |>
mutate(qol_physical.FU =  who_ph_total.followup) |>
mutate(qol_social.FU =  who_soc_rel_total.followup) |>
mutate(qol_env.FU =  who_env_total.followup) |>
mutate(qol.FU =  qol_mean.followup) |>
mutate(qol_general_health.DC =  who_qol_gh_total.discharge) |>
mutate(qol_psych.DC =  who_psy_total.discharge) |>
mutate(qol_physical.DC =  who_ph_total.discharge) |>
mutate(qol_social.DC =  who_soc_rel_total.discharge) |>
mutate(qol_env.DC =  who_env_total.discharge) |>
mutate(qol.DC =  qol_mean.discharge) |>


## Social support

# 1. There is a special person who is around when I am in need.
# 2. There is a special person with whom I can share joys and sorrows.
# 3. My family really tries to help me.
# 4. I get the emotional help & support I need from my family.
# 5. I have a special person who is a real source of comfort to me.
# 6. My friends really try to help me.
# 7. I can count on my friends when things go wrong.
# 8. I can talk about my problems with my family.
# 9. I have friends with whom I can share my joys and sorrows.
# 10. There is a special person in my life who cares about my feelings.
# 11. My family is willing to help me make decisions.
# 12. I can talk about my problems with my friends.
# MSPSS Family Subscale   sum([mspss3],[mspss4],[mspss8],[mspss11])/4
# MSPSS Friends Subscale  sum([mspss6],[mspss7],[mspss9],[mspss12])/4
# MSPSS Significant Other Subscale
# sum([mspss1], [mspss2], [mspss5], [mspss10])/4
# MSPSS Total round(sum( [mspss1],[mspss2],[mspss3],[mspss4],[mspss5],
# [mspss6],[mspss7],[mspss8],[mspss9],[mspss10],[mspss11],[mspss12] )/12,2)
mutate(social.family = mspss_fam_sub_total.baseline) |>
mutate(social.friends = mspss_friends_sub_total.baseline) |>
mutate(social.family_and_friends = mspss_fam_friends_tot.baseline) |>
mutate(social.sig_other = mspss_sig_other_sub_total.baseline) |>
mutate(social = mspss_total.baseline) |>


## Spirituality and religion

# 0, Atheist | 1, Agnostic | 2, Protestant | 3, Catholic | 4, Muslim |
# 5, Jewish | 6, Hindu | 7, Buddhist | 8, Baptist |
# 9, No religious affiliation | 10, Non-denominational Christian | 15, Other
mutate(rel.is_Atheist = as_factor(brc_rel___0.baseline)) |>
mutate(rel.is_Agnostic = as_factor(brc_rel___1.baseline)) |>
mutate(rel.is_Protestant = as_factor(brc_rel___2.baseline)) |>
mutate(rel.is_Catholic = as_factor(brc_rel___3.baseline)) |>
mutate(rel.is_Muslim = as_factor(brc_rel___4.baseline)) |>
mutate(rel.is_Jewish = as_factor(brc_rel___5.baseline)) |>
mutate(rel.is_Hindu = as_factor(brc_rel___6.baseline)) |>
```

```
mutate(rel.is_Buddhist = as_factor(brc_rel___7.baseline)) |>
mutate(rel.is_Baptist = as_factor(brc_rel___8.baseline)) |>
mutate(rel.is_No_affiliation = as_factor(brc_rel___9.baseline)) |>
mutate(rel.is_Non_denominational_Christian =
         as_factor(brc_rel___10.baseline)) |>
mutate(rel.is_Other = as_factor(brc_rel___15.baseline)) |>
mutate(religion_pos = pos_cope.baseline) |>
mutate(religion_neg = neg_cope.baseline) |>


## Religion

mutate(rel.is_religious = as.numeric(as.character(rel.is_Other)) +
         as.numeric(as.character(rel.is_Non_denominational_Christian)) +
         as.numeric(as.character(rel.is_Baptist)) +
         as.numeric(as.character(rel.is_Buddhist)) +
         as.numeric(as.character(rel.is_Hindu)) +
         as.numeric(as.character(rel.is_Jewish)) +
         as.numeric(as.character(rel.is_Muslim)) +
                   as.numeric(as.character(rel.is_Catholic)) +
                   as.numeric(as.character(rel.is_Protestant))) |>

mutate(rel.is_religious =
         as_factor(if_else(rel.is_religious >= 1, TRUE, FALSE))) |>

mutate(rel.is_not_religious =
         as.numeric(as.character(rel.is_No_affiliation)) +
       as.numeric(as.character(rel.is_Agnostic)) +
       as.numeric(as.character(rel.is_Atheist))) |>

mutate(rel.is_not_religious =
         as_factor(if_else(rel.is_not_religious >= 1, TRUE, FALSE))) |>


## Stressful life events

# 1, Happened to you | 2, Witnessed it happen |
# 3, Learned about it happening | 4, Exposed as part of job | 5, Not sure |
# 6, Doesn't apply

# 1. Natural disaster
mutate(stress.natural_disaster.to_subj = as_factor(lec_2_1___1.baseline)) |>
mutate(stress.natural_disaster.witnessed = as_factor(lec_2_1___2.baseline)) |>
mutate(stress.natural_disaster.learned = as_factor(lec_2_1___3.baseline)) |>
mutate(stress.natural_disaster.exposed = as_factor(lec_2_1___4.baseline)) |>
# 2. Fire or explosion
mutate(stress.fire.to_subj = as_factor(lec_2_2___1.baseline)) |>
mutate(stress.fire.witnessed = as_factor(lec_2_2___2.baseline)) |>
mutate(stress.fire.learned = as_factor(lec_2_2___3.baseline)) |>
mutate(stress.fire.exposed = as_factor(lec_2_2___4.baseline)) |>
# 3. Transportation accident
mutate(stress.transportation_accident.to_subj = as_factor(lec_2_3___1.baseline)) |>
mutate(stress.transportation_accident.witnessed = as_factor(lec_2_3___2.baseline)) |>
```

```
mutate(stress.transportation_accident.learned = as_factor(lec_2_3___3.baseline)) |>
mutate(stress.transportation_accident.exposed = as_factor(lec_2_3___4.baseline)) |>
# 4. Serious accident
mutate(stress.serious_accident.to_subj = as_factor(lec_2_4___1.baseline)) |>
mutate(stress.serious_accident.witnessed = as_factor(lec_2_4___2.baseline)) |>
mutate(stress.serious_accident.learned = as_factor(lec_2_4___3.baseline)) |>
mutate(stress.serious_accident.exposed = as_factor(lec_2_4___4.baseline)) |>
# 5. Toxic substance
mutate(stress.toxic.to_subj = as_factor(lec_2_5___1.baseline)) |>
mutate(stress.toxic.witnessed = as_factor(lec_2_5___2.baseline)) |>
mutate(stress.toxic.learned = as_factor(lec_2_5___3.baseline)) |>
mutate(stress.toxic.exposed = as_factor(lec_2_5___4.baseline)) |>
# 6. Physical assault
mutate(stress.physical_assault.to_subj = as_factor(lec_2_6___1.baseline)) |>
mutate(stress.physical_assault.witnessed = as_factor(lec_2_6___2.baseline)) |>
mutate(stress.physical_assault.learned = as_factor(lec_2_6___3.baseline)) |>
mutate(stress.physical_assault.exposed = as_factor(lec_2_6___4.baseline)) |>
# 7. Assault with weapon
mutate(stress.weapon_assault.to_subj = as_factor(lec_2_7___1.baseline)) |>
mutate(stress.weapon_assault.witnessed = as_factor(lec_2_7___2.baseline)) |>
mutate(stress.weapon_assault.learned = as_factor(lec_2_7___3.baseline)) |>
mutate(stress.weapon_assault.exposed = as_factor(lec_2_7___4.baseline)) |>
# 8. Sexual assault
mutate(stress.sexual_assault.to_subj = as_factor(lec_2_8___1.baseline)) |>
mutate(stress.sexual_assault.witnessed = as_factor(lec_2_8___2.baseline)) |>
mutate(stress.sexual_assault.learned = as_factor(lec_2_8___3.baseline)) |>
mutate(stress.sexual_assault.exposed = as_factor(lec_2_8___4.baseline)) |>
# 9. Unwanted sexual experience
mutate(stress.unwanted_sexual.to_subj = as_factor(lec_2_9___1.baseline)) |>
mutate(stress.unwanted_sexual.witnessed = as_factor(lec_2_9___2.baseline)) |>
mutate(stress.unwanted_sexual.learned = as_factor(lec_2_9___3.baseline)) |>
mutate(stress.unwanted_sexual.exposed = as_factor(lec_2_9___4.baseline)) |>
# 10. Combat or war exposure
mutate(stress.combat.to_subj = as_factor(lec_2_10___1.baseline)) |>
mutate(stress.combat.witnessed = as_factor(lec_2_10___2.baseline)) |>
mutate(stress.combat.learned = as_factor(lec_2_10___3.baseline)) |>
mutate(stress.combat.exposed = as_factor(lec_2_10___4.baseline)) |>
# 11. Captivity
mutate(stress.captivity.to_subj = as_factor(lec_2_11___1.baseline)) |>
mutate(stress.captivity.witnessed = as_factor(lec_2_11___2.baseline)) |>
mutate(stress.captivity.learned = as_factor(lec_2_11___3.baseline)) |>
mutate(stress.captivity.exposed = as_factor(lec_2_11___4.baseline)) |>
# 12. Life-threatening illness
mutate(stress.illness.to_subj = as_factor(lec_2_12___1.baseline)) |>
mutate(stress.illness.witnessed = as_factor(lec_2_12___2.baseline)) |>
mutate(stress.illness.learned = as_factor(lec_2_12___3.baseline)) |>
mutate(stress.illness.exposed = as_factor(lec_2_12___4.baseline)) |>
# 13. Severe human suffering
mutate(stress.severe_suffering.to_subj = as_factor(lec_2_13___1.baseline)) |>
mutate(stress.severe_suffering.witnessed = as_factor(lec_2_13___2.baseline)) |>
mutate(stress.severe_suffering.learned = as_factor(lec_2_13___3.baseline)) |>
mutate(stress.severe_suffering.exposed = as_factor(lec_2_13___4.baseline)) |>
# 14. Sudden violent death
```

```
  mutate(stress.sudden_violent_death.to_subj = as_factor(lec_2_14___1.baseline)) |>
  mutate(stress.sudden_violent_death.witnessed = as_factor(lec_2_14___2.baseline)) |>
  mutate(stress.sudden_violent_death.learned = as_factor(lec_2_14___3.baseline)) |>
  mutate(stress.sudden_violent_death.exposed = as_factor(lec_2_14___4.baseline)) |>
  # 15. Sudden accidental death
  mutate(stress.sudden_accidental_death.to_subj = as_factor(lec_2_15___1.baseline)) |>
  mutate(stress.sudden_accidental_death.witnessed = as_factor(lec_2_15___2.baseline)) |>
  mutate(stress.sudden_accidental_death.learned = as_factor(lec_2_15___3.baseline)) |>
  mutate(stress.sudden_accidental_death.exposed = as_factor(lec_2_15___4.baseline)) |>
  # 16. Serious injury or harm caused to someone else
  mutate(stress.harm_to_others.to_subj = as_factor(lec_2_16___1.baseline)) |>
  mutate(stress.harm_to_others.witnessed = as_factor(lec_2_16___2.baseline)) |>
  mutate(stress.harm_to_others.learned = as_factor(lec_2_16___3.baseline)) |>
  mutate(stress.harm_to_others.exposed = as_factor(lec_2_16___4.baseline)) |>
  # 17. Any other very stressful experience
  mutate(stress.other.to_subj = as_factor(lec_2_17___1.baseline)) |>
  mutate(stress.other.witnessed = as_factor(lec_2_17___2.baseline)) |>
  mutate(stress.other.learned = as_factor(lec_2_17___3.baseline)) |>
  mutate(stress.other.exposed = as_factor(lec_2_17___4.baseline)) |>
  # Total stress
  mutate(stress_to_subj = toyou_total.baseline) |>
  mutate(stress_to_subj_and_wit = toyou_wit_total.baseline) |>
  mutate(stress_wit = witnessed_total.baseline) |>


  ## Substance use history

  # 18. Please indicate which of these substances you have EVER TRIED:
  # 1, Tobacco (including e-cigarettes or vaping) | 2, Alcohol | 10, Other drug(s)
  # Tobacco (including e-cigarettes or vaping):
  # Age you first tried any tobacco/nicotine product.
  # Alcohol: Age you first tried any type of alcohol.
  # Other drug(s): Age you first tried any type of mood altering substance.
  mutate(history.tobacco = as_factor(newace18a___1.baseline)) |>
  mutate(history.alcohol = as_factor(newace18a___2.baseline)) |>
  mutate(history.other = as_factor(newace18a___10.baseline)) |>

  ## Other feature engineering
  ### Calculating the difference of commitment to sobriety score between discharge and baseline
  mutate(commit_change = (commit.DC - commit.BL)) |>
  ### Recalculating quality of life scores without social component
  mutate(no_social_qol.BL = (qol_general_health.BL
                             + qol_psych.BL
                             + qol_physical.BL
                             + qol_env.BL)/23)
```

Filtering redundant columns and removing rows with NANs

```
filtered_addiction_data <- factored_addiction_data |>
  # Removing redundant columns and columns that won't be used
  select(-contains(".baseline")) |>
  select(-contains(".discharge")) |>
```

```
  select(-contains(".followup")) |>
  select(-contains(".x.x")) |>
  select(-contains(".y.y")) |>

  # Removing weird outlier of negative days
  filter(treatment_days >= 0) |>
  # Removing all rows with NANs in any column
  drop_na()
```

Saving the Tibble to file

```
saveRDS(filtered_addiction_data, "addiction.rds")
```