CLHLS Data Analysis by R

Simonzhou

2025-02-22

1 CLHLS

R CLHLS

1.1

DVN/WBO7LK_2020

SAS Vintage Car R

2

```
# 1.
raw_data <- read_sav("C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/clhls_2018_15874.sav")

# 2.       proc contents
str(raw_data)

# 3.
selected_data <- raw_data %>%
  mutate(
    #
    SHEALTH = ifelse(!b12 %in% c(NA, 8, 9), b12, NA),

    # ADL
    ADLsum = rowSums(select(., e1:e6), na.rm = TRUE),
    ADL = (ADLsum == 6),
```

```
    # IADL
    IADLsum = rowSums(select(., e7:e14), na.rm = TRUE),
    IADL = (IADLsum == 8),

    #    -
    economic_support = pmap_dbl(
      list(f12a, f12b, f12c),
      function(a, b, c) {
        sum = 0
        for (x in c(a, b, c)) {
          if (!is.na(x)) {
            if (x == 99998) sum = sum + 10000
            else if (!x %in% c(88888, 99999)) sum = sum + x
          }
        }
        return(sum)
      }
    ),

    #
    residence = a51,
    living = a52,
    visit_fren = apply(select(., starts_with("f103") & ends_with("5")), 1,
                       function(x) max(x, na.rm = TRUE) == 1),
    care_support = (residence == 1 | visit_fren),

    #
    emotion_support = apply(select(., starts_with("f103") & ends_with("6")), 1,
                            function(x) max(x, na.rm = TRUE) == 1),

    #
    age = trueage,
    gender = a1,
    education = f1,
    job_type = f2,
    marriage_status = (f41 == 1),
    hukou_type = hukou,
    social_insurance = (nf64a == 0 | f64b == 1 | f64c == 1 | f64i == 1),
    medical_insurance = (f64d == 1 | f64e == 1 | f64g == 1 | f64h == 1),

    #
    chronic_disease = apply(select(., starts_with("g15") & ends_with("1")), 1,
```

```
                            function(x) max(x, na.rm = TRUE) == 1),

    smoking = g151,
    drinking = g161,
    exercise = (d91 == 1 | d92 == 1)
  ) %>%
  select(SHEALTH, ADL, ADLsum, IADL, IADLsum, economic_support, residence, living,
         visit_fren, emotion_support, f10, age, gender, education, job_type,
         marriage_status, hukou_type, social_insurance, medical_insurance,
         chronic_disease, smoking, drinking, exercise, care_support)

# 4.
temp_data <- selected_data %>%
  filter(
    complete.cases(SHEALTH, ADL, ADLsum, IADL, IADLsum, economic_support,
                   residence, living, visit_fren, emotion_support, age, gender,
                   education, job_type, marriage_status, hukou_type,
                   social_insurance, medical_insurance, chronic_disease,
                   smoking, drinking, exercise, care_support),
    f10 > 0 & f10 <= 13
  )

# 5.
final_data <- temp_data %>%
  filter(
    SHEALTH <= 8,
    ADLsum <= 18,
    IADLsum <= 24,
    residence <= 3,
    age >= 60,
    education <= 22,
    smoking <= 2,
    drinking <= 2
  )

# 6.
final_data_grouped <- final_data %>%
  mutate(
    age_group = case_when(
      age < 70 ~ "60-69",
      age < 80 ~ "70-79",
      age < 90 ~ "80-89",
```

```
      TRUE ~ "90+"
    )
  )

# 7.
summary_stats <- final_data_grouped %>%
  summarise(across(
    c(SHEALTH, ADL, ADLsum, IADL, IADLsum, economic_support, care_support,
      emotion_support, age, gender, education, marriage_status, hukou_type,
      social_insurance, medical_insurance, chronic_disease, smoking,
      drinking, exercise),
    list(
      n = ~sum(!is.na(.)),
      mean = ~mean(., na.rm = TRUE),
      std = ~sd(., na.rm = TRUE),
      min = ~min(., na.rm = TRUE),
      median = ~median(., na.rm = TRUE),
      max = ~max(., na.rm = TRUE)
    )
  )) %>%
  pivot_longer(
    everything(),
    names_to = c("var_name", "stat"),
    names_sep = "_",
    values_to = "value"
  ) %>%
  pivot_wider(
    names_from = "stat",
    values_from = "value"
  )

# 8.
label_map <- tibble(
  var_name = c("SHEALTH", "ADL", "ADLsum", "IADL", "IADLsum", "economic_support",
               "care_support", "emotion_support", "age", "gender", "education",
               "marriage_status", "hukou_type", "social_insurance",
               "medical_insurance", "chronic_disease", "smoking", "drinking",
               "exercise"),
  label = c("    ", "    ", "     ", "     ",
            "      ", "   ", "   ", "   ", " ", " ",
            "   ", "   ", "   ", "   ", "   ", "  ",
            " ", " ", "   ")
```

```
)

# 9.
final_summary <- summary_stats %>%
  left_join(label_map, by = "var_name") %>%
  select(label, n, mean, std, min, median, max)

# 10.
print(final_summary)

# 11.
library(knitr)
kable(final_summary,
      digits = 3,
      col.names = c("  ", "  ", "  ", "  ", "  ", "  ", "  "),
      caption = " 3-1     ")
```

2.1

```
library(writexl)
# 12.
write_xlsx(final_summary, "C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/Rsummary0223.xlsx")
# 13.
write_xlsx(final_data, "C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/final_data0223.xlsx")
```

2.2

3

```
library(readxl)
library(dplyr)
library(tidyr)
library(knitr)
library(officer)
library(flextable)
library(car)
```

```r
#
if (!require(officer)) stop("  officer ")
if (!require(flextable)) stop("  flextable ")
if (!require(dplyr)) stop("  dplyr ")

# 1.
final_data <- read_excel("C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/final_data.xlsx",
                         sheet = "final_data")

# 2.
varlist <- c("SHEALTH", "ADLsum", "ADL", "IADLsum", "IADL", "economic_support",
             "residence", "living", "visit_fren", "care_support", "emotion_support",
             "age", "gender", "education", "job_type", "marriage_status",
             "hukou_type", "social_insurance", "medical_insurance", "chronic_disease",
             "smoking", "drinking", "exercise", "f10")

# 3.
label_map <- data.frame(
  variable = varlist,
  label = c("    ", "     ", "    ", "       ",
            "     ", "  ", "  ", "  ", "  ",
            "  ", "  ", " ", " ", "   ", "  ",
            "  ", "  ", "  ", "  ", "  ",
            "  ", "  ", "  ", "  ")
)

# 4.
summary_stats <- final_data[ , varlist]
summary_stats <- summarise(summary_stats, across(
  all_of(varlist),
  list(
    count = ~sum(!is.na(.)),
    mean = ~mean(., na.rm = TRUE),
    sd = ~sd(., na.rm = TRUE),
    min = ~min(., na.rm = TRUE),
    p50 = ~median(., na.rm = TRUE),
    max = ~max(., na.rm = TRUE)
  ),
  .names = "{.col}_{.fn}"
))

#    summarise
```

```r
cat("  summarise   :\n")
print(head(summary_stats))
cat("  summarise    :\n")
print(sapply(summary_stats, class))

#
summary_stats <- pivot_longer(summary_stats,
                              cols = everything(),
                              names_to = c("variable", "stat"),
                              names_pattern = "(.*)_(.*)",  #
                              values_to = "value")

#     pivot_longer
cat("  pivot_longer   :\n")
print(head(summary_stats, 10))
cat("  pivot_longer    :\n")
print(sapply(summary_stats, class))

#
duplicates <- summary_stats %>%
  group_by(variable, stat) %>%
  summarise(n = n()) %>%
  filter(n > 1)
cat("   :\n")
print(duplicates)

#
summary_stats <- pivot_wider(summary_stats,
                             names_from = "stat",
                             values_from = "value",
                             values_fn = list(value = mean))  #

#     pivot_wider
cat("  pivot_wider   :\n")
print(head(summary_stats))
cat("  pivot_wider    :\n")
print(sapply(summary_stats, class))

#
summary_stats <- left_join(summary_stats, label_map, by = "variable")
summary_stats <- select(summary_stats, label, count, mean, sd, min, p50, max)
```

```r
#
cat("    :\n")
print(head(summary_stats))
cat("     :\n")
print(sapply(summary_stats, class))

#
summary_stats <- mutate(summary_stats,
                        count = as.numeric(count),
                        mean = as.numeric(mean),
                        sd = as.numeric(sd),
                        min = as.numeric(min),
                        p50 = as.numeric(p50),
                        max = as.numeric(max))
summary_stats <- mutate(summary_stats,
                        across(c(mean, sd, min, p50, max), ~round(., 2)))

#
cat("   summary_stats  :\n")
print(head(summary_stats))
cat("   summary_stats   :\n")
print(sapply(summary_stats, class))

# 5.   R
cat("   kable  :\n")
print(kable(summary_stats,
            digits = 2,
            col.names = c("  ", "  ", " ", "  ", "  ", "  ", "  "),
            caption = "   "))

# 6.      DOCX
stats_table <- flextable(summary_stats)
stats_table <- set_header_labels(stats_table,
                                 label = "  ",
                                 count = "  ",
                                 mean = " ",
                                 sd = "  ",
                                 min = "  ",
                                 p50 = "  ",
                                 max = "  ")
stats_table <- colformat_double(stats_table, j = 2:7, digits = 2)
stats_table <- set_caption(stats_table, "   ")
```

```r
stats_table <- autofit(stats_table)

doc <- read_docx()
doc <- body_add_flextable(doc, stats_table)
print(doc, target = "C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/   0218.docx")

cat("      C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/   0218.docx\n")

#          ANOVA
# 7.
final_data$age_group <- cut(final_data$age, breaks = c(60, 70, 80, 150),
                            labels = c("60-69", "70-79", "80  "),
                            right = FALSE)
final_data$edu_group <- cut(final_data$education, breaks = c(0, 1, 7, 10, 13, 18, 23),
                            labels = c("  ", " ", " ", " ", " ", " "),
                            right = FALSE)

# 8.
outcomes <- c("SHEALTH", "ADL", "IADL")
controls <- c("age_group", "gender", "edu_group", "marriage_status", "hukou_type",
              "social_insurance", "medical_insurance", "chronic_disease",
              "smoking", "drinking", "exercise")

# 9.
label_map <- rbind(label_map,
                   data.frame(
                     variable = c("age_group", "edu_group"),
                     label = c("  ", "  ")
                   ))

# 10.
freq_tables <- list()
for (ctrl in controls) {
  freq_table <- group_by(final_data, !!sym(ctrl))
  freq_table <- summarise(freq_table,
                          freq = n(),
                          pct = (n() / nrow(final_data)) * 100)
  colnames(freq_table) <- c(label_map$label[label_map$variable == ctrl], " ", "   (%)")
  cat("\n   for", ctrl, ":\n")
  print(freq_table)
  freq_tables[[ctrl]] <- freq_table
}
```

```r
# 11.
anova_results <- list()
for (outcome in outcomes) {
  anova_results[[outcome]] <- data.frame()

  for (ctrl in controls) {
    if (!is.numeric(final_data[[ctrl]])) {
      final_data[[ctrl]] <- as.factor(final_data[[ctrl]])
    }

    formula <- as.formula(paste(outcome, "~", ctrl))
    anova_fit <- tryCatch({
      fit <- aov(formula, data = final_data)
      summary_fit <- summary(fit)[[1]]
      result <- data.frame(
        "  " = label_map$label[label_map$variable == ctrl],
        "  " = summary_fit$"Sum Sq"[1],
        "  " = summary_fit$"Df"[1],
        "F  " = summary_fit$"F value"[1],
        "p  " = summary_fit$"Pr(>F)"[1]
      )
      cat("\nANOVA:", outcome, "vs", ctrl, "\n")
      print(result)
      result
    }, error = function(e) {
      message(paste("ANOVA  :", outcome, "vs", ctrl, " :", e$message))
      return(data.frame(
        "  " = label_map$label[label_map$variable == ctrl],
        "  " = NA,
        "  " = NA,
        "F  " = NA,
        "p  " = NA
      ))
    })

    anova_results[[outcome]] <- rbind(anova_results[[outcome]], anova_fit)
  }
}

# 12.   Word
doc <- read_docx()
```

```r
#
doc <- body_add_par(doc, "      ", style = "heading 1")
for (ctrl in controls) {
  freq_table <- freq_tables[[ctrl]]
  if (nrow(freq_table) > 0) {
    freq_ft <- flextable(freq_table)
    freq_ft <- colformat_double(freq_ft, j = 2, digits = 0)
    freq_ft <- colformat_double(freq_ft, j = 3, digits = 2)
    freq_ft <- autofit(freq_ft)
    doc <- body_add_par(doc, paste(" :", label_map$label[label_map$variable == ctrl]), style
    doc <- body_add_flextable(doc, freq_ft)
  } else {
    cat(" :     -", ctrl, "\n")
  }
}

#   ANOVA
doc <- body_add_par(doc, "     ", style = "heading 1")
for (outcome in outcomes) {
  anova_table <- anova_results[[outcome]]
  if (nrow(anova_table) > 0) {
    anova_ft <- flextable(anova_table)
    anova_ft <- colformat_double(anova_ft, j = 2, digits = 2)
    anova_ft <- colformat_double(anova_ft, j = 3, digits = 0)
    anova_ft <- colformat_double(anova_ft, j = 4, digits = 2)
    anova_ft <- colformat_double(anova_ft, j = 5, digits = 3)
    anova_ft <- autofit(anova_ft)
    doc <- body_add_par(doc, paste("  :", label_map$label[label_map$variable == outcome], "
    doc <- body_add_par(doc, paste("ANOVA    -", label_map$label[label_map$variable == outcom
    doc <- body_add_flextable(doc, anova_ft)
  } else {
    cat(" : ANOVA     -", outcome, "\n")
  }
}

#
print(doc, target = "C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/anova_results0223.docx")

cat("        C:/Users/asus/Desktop/test/CLHLS/Analysis-0214/anova_results0223.docx\n")
```

3.1 logistic