Week 5-6 Data Preprocessing HW

公衛四 b11801033 張藝馨

Github : https://github.com/tt921/Week-5-6-Data-Preprocessing-HW

1. Introduction & Setup

・Briefly describe the purpose of this analysis (NHANES data, 2021–2023).

在此份報告中，利用 NHANES data, 2021–2023。

NHANES 2021–2023 是一項全美代表性調查，結合了問卷訪談與身體檢查，收集美國人口的健康狀況、營養攝取、疾病盛行率、生理與生化測量資料。這些資料常用於公共衛生研究，例如慢性病（如糖尿病、肥胖、高血壓）之流行趨勢分析與營養政策制定。

基於此資料集進行分析並回答下列問題。

Q1. Among adults aged ≥20 years in the 2021–2023 NHANES, observe the association between BMI and mean systolic blood pressure (SBP) and does the association vary between sex ?

Q2. Among all the subjects in 2021-2023 NHANES dataset, observe the distribution of BMI in different races and education levels

I. What is the distribution of educational attainment (EDU) and ethnicity (Race) in your data? (Please calculate the number and proportion of each EDU and Race, and output the table)
(a) For Education levels, please refer to the variable “dmdeduc2”
(b) For Race categories, please refer to the variable “ridreth3”
II. Please use boxplots to visualize the BMI distribution in different races and education levels (2 outputs: BMI as X variable and filled by education and vice versa)

III. Please state your brief conclusion about the plots (Do not need the statistical testsyou’re your inference)

Q2. Among all the subjects in 2021-2023 NHANES dataset, BPX is the data including three times of examination of blood pressure (SBP & DBP). The values were recorded in different columns (bpxosy1-3; bpxodi1-3) (Reminder: please use the “cleaned” BP data).

I. Currently the dataset is stored in a wide format, meaning that each measurement is placed in a separate column. Please reshape the dataset into a long format, so that each row represents a single measurement, and include the following variables:

(a) seqn: Participant ID

(b) measure (new defined): Measurement type (SBP or DBP)

(c) trial (new defined): Trial number (1, 2, or 3)

(d) value (from each BP value): The recorded blood pressure value

II. After reshaping the dataset, create a boxplot to compare the distribution of SBP and DBP across the three trials and facet by the measurement type.

III. Now, suppose we are only interested in the two trials that show the largest difference for each subject. Please complete the tasks aboved.

IV. Please infer whether these blood pressure values were measured at long intervals or on the same day to avoid errors.

以上問題，Q1 將於#2. Week 5 Components (BMI & SBP Cleaning)#中回答，Q2 與 Q3 則在#3. Week 6 Components (EDU, Race, and BP Trials)#與#4. Homework Extensions (if applicable)#回答。

· Load all required packages and datasets.

```
# ================= Class Lab: BMI Cleaning & Visualization ===========
======
# 1) Packages and folders ---------------------------------------------
------
pkgs <- c("tidyverse","haven","janitor","stringr","scales","skimr","nan
iar") # tidyverse: metapackage (including dplyr, tidyr, ggplot2), haven:
 read SAS/XPT files
to_install <- setdiff(pkgs, rownames(installed.packages()))
# if (length(to_install)) install.packages(to_install)
invisible(lapply(pkgs, library, character.only = TRUE))

## ── Attaching core tidyverse packages ──────────────────────── tidyve
rse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.1     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.2     ✓ tibble    3.3.0
## ✓ lubridate 1.9.4     ✓ tidyr     1.3.1
## ✓ purrr     1.1.0
## ── Conflicts ────────────────────────────────────────── tidyverse_co
nflicts() ──
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to for
ce all conflicts to become errors
```

```
##
## Attaching package: 'janitor'
##
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
##
##
##
## Attaching package: 'scales'
##
##
## The following object is masked from 'package:purrr':
##
##     discard
##
##
## The following object is masked from 'package:readr':
##
##     col_factor
##
##
##
## Attaching package: 'naniar'
##
##
## The following object is masked from 'package:skimr':
##
##     n_complete
```

```
dir.create("outputs", showWarnings = FALSE) # where plots will be saved
data_dir <- "data_raw"                      # folder containing .XPT fi
les

getwd()  # check working directory
```

```
## [1] "D:/下載"
```

```
setwd("D:/下載")
```

2. Week 5 Components (BMI & SBP Cleaning)

- Data loading, handling missing values.

```
# 2) Load raw data ---------------------------------------------------
------
demo <- read_xpt(file.path(data_dir,"DEMO_L.XPT")) %>% clean_names()  #
 %>% is one of the most important operators in the tidyverse, it pronou
```

```r
nce as“and then”
bpx  <- read_xpt(file.path(data_dir,"BPXO_L.XPT")) %>% clean_names()  #
 clean_names() from janitor package: make column names consistent (lowe
rcase, no spaces or special characters)
bmx  <- read_xpt(file.path(data_dir,"BMX_L.XPT"))  %>% clean_names()

# quick overviews (on-screen)
skimr::skim(demo); skimr::skim(bpx); skimr::skim(bmx)
```

Data summary

Name	demo
Number of rows	11933
Number of columns	27
_____	
Column type frequency:	
numeric	27
_____	
Group variables	None

Variable type: numeric

Data summary

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
seqn	0	1.00	13644.00	3444.90	13078.00	13361.00	13644.00	13927.00	14310.00	▇▇▇▇▇
sddsrvyr	0	1.00	12.00	0.00	12.00	12.00	12.00	12.00	12.0	▁▁▇▁▁
ridstatr	0	1.00	1.74	0.44	1.00	1.00	2.00	2.00	2.0	▃▁▁▁▇
riagendr	0	1.00	1.53	0.50	1.00	1.00	2.00	2.00	2.0	▇▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
ridageyr	0	1.00	38.32	25.60	0.00	13.00	37.00	62.00	80.0	▇▅▅▅▅▅▅▅▅▅
ridagemn	11556	0.03	11.63	6.81	0.00	6.00	11.00	17.00	24.0	▇▇▇▇▇▇▇▅▅▅
ridreth1	0	1.00	3.10	1.08	1.00	3.00	3.00	4.00	5.0	▅▅▇▅▅▅▅▅▅▅
ridreth3	0	1.00	3.32	1.52	1.00	3.00	3.00	4.00	7.0	▅▅▇▅▅▅▅▅▅▅
ridexmon	3073	0.74	1.52	0.50	1.00	1.00	2.00	2.00	2.0	▇▅▅▅▅▅▅▅▅▅
ridexagm	9146	0.23	121.91	67.16	0.00	66.00	122.00	179.50	239.0	▅▅▅▅▅▅▅▅▅▅
dmqmiliz	3632	0.70	1.92	0.28	1.00	2.00	2.00	2.00	7.0	▅▅▅▅▅▅▅▅▅▅

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
dmdborn4	19	1.00	1.16	0.36	1.00	1.00	1.00	1.00	2.0	
dmdyrusr	10058	0.16	7.33	15.83	1.00	3.00	6.00	6.00	99.0	
dmdeduc2	4139	0.65	3.80	1.15	1.00	3.00	4.00	5.00	9.0	
dmdmartz	4141	0.65	1.78	3.10	1.00	1.00	1.00	2.00	99.0	
ridexprg	10430	0.13	2.24	0.49	1.00	2.00	2.00	3.00	3.0	
dmdhhsiz	0	1.00	3.24	1.70	1.00	2.00	3.00	4.00	7.0	
dmdhrgnd	7818	0.34	1.56	0.50	1.00	1.00	2.00	2.00	2.0	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
dmdhragz	7809	0.35	2.54	0.64	1.00	2.00	2.00	3.00	4.0	▁▇▁▇▁
dmdhredz	8187	0.31	2.17	0.66	1.00	2.00	2.00	3.00	3.0	▃▁▇▁▅
dmdhrmaz	7913	0.34	1.38	0.68	1.00	1.00	1.00	2.00	3.0	▇▁▃▁▁
dmdhsedz	9806	0.18	2.28	0.69	1.00	2.00	2.00	3.00	3.0	▃▁▇▁▇
wtint2yr	0	1.00	27404.14	19449.16	4584.46	14331.75	21670.19	33831.33	170968.3	▇▃▁▁▁
wtmec2yr	0	1.00	27404.14	27962.96	0.00	0.00	21717.85	38341.15	227108.3	▇▃▁▁▁
sdmvstra	0	1.00	179.92	4.31	173.00	176.00	180.00	184.00	187.0	▇▇▇▇▇
sdmvpsu	0	1.00	1.49	0.50	1.00	1.00	1.00	2.00	2.0	▇

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
indfmpir	2041	0.83	2.71	1.67	0.00	1.18	2.50	4.50	5.0	▇▅▅▃▂▇

Name	bpx
Number of rows	7801
Number of columns	12
_____	
Column type frequency:	
character	1
numeric	11
_____	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
bpaoarm	0	1	0	1	147	3	0

Variable type: numeric

Data summary

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
seqn	0	1.00	136349.49	34491.49	130378	133335	136382	139325	142310	▇▇▇▇▇
bpaocsz	190	0.98	3.52	0.67	2	3	4	4	5	▂

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bpxosy1	284	0.96	119.29	18.56	61	106	117	130	232	▇▅▇▃▁
bpxodi1	284	0.96	72.75	11.90	33	64	72	80	142	▁▇▆▂▁
bpxosy2	296	0.96	119.08	18.57	59	106	116	129	233	▁▇▅▂▁
bpxodi2	296	0.96	72.09	11.85	32	64	71	79	139	▁▇▆▂▁
bpxosy3	321	0.96	118.92	18.50	50	106	116	129	232	▁▇▆▂▁
bpxodi3	321	0.96	71.81	11.77	24	64	71	79	136	▁▇▇▂▁
bpxopls1	284	0.96	72.34	12.72	35	63	71	80	158	▃▇▃

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bpxopls2	296	0.96	73.09	12.78	32	64	72	81	141	▁▁▁▇▇▁▁
bpxopls3	321	0.96	73.69	12.89	31	65	73	82	154	▁▁▇▅▁▁

Name	bmx
Number of rows	8860
Number of columns	22
_____	
Column type frequency:	
numeric	22
_____	
Group variables	None

Variable type: numeric

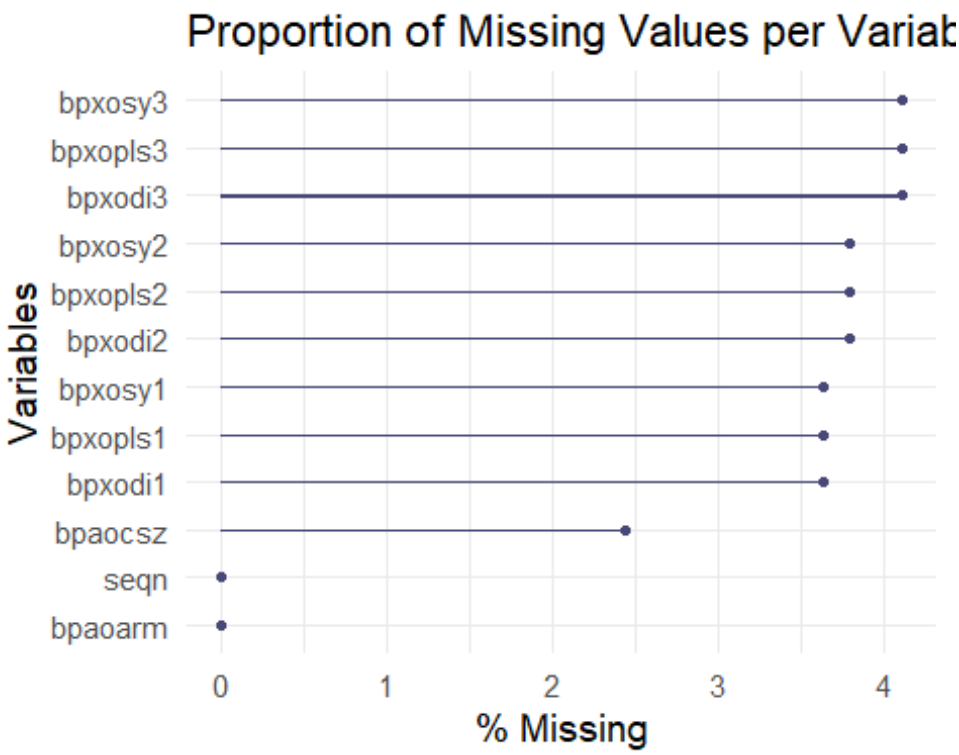
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
seqn	0	1.00	13645.83	3453.78	13078.0	13319.75	13677.5	13936.2	14210.0	▇▇▇▇▇
bmdstats	0	1.00	1.13	0.50	1.0	1.00	1.0	1.0	4.0	▇▁▁▁▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bmxwt	106	0.99	70.55	30.39	2.7	54.20	71.7	89.1	248.2	▅▇▅▂▂
bmiwt	8515	0.04	2.88	0.62	1.0	3.00	3.0	3.0	4.0	▂▂▂▇▂
bmxrecum	8406	0.05	84.33	14.06	48.5	73.48	84.7	96.1	118.8	▅▇▇▇▅
bmirecum	8842	0.00	1.00	0.00	1.0	1.00	1.0	1.0	1.0	▂▂▇▂▂
bmxhead	8790	0.01	41.93	2.80	34.4	40.20	42.4	44.0	46.5	▂▅▇▇▇
bmihead	8860	0.00	NaN	NA	NA	NA	NA	NA	NA	
bmxht	361	0.96	159.66	19.86	79.1	154.40	163.6	172.1	200.7	▂▂▅▇▅
bmiht	8726	0.02	2.31	0.95	1.0	1.00	3.0	3.0	3.0	▅▂▂▂▇
bmxbmi	389	0.96	27.25	8.1	11.1	21.60	26.4	31.7	74.8	▇

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
				4						
bmdbmic	6368	0.28	2.56	0.88	1.0	2.00	2.0	3.0	4.0	
bmxleg	1525	0.83	38.13	3.86	24.9	35.50	38.1	40.8	51.6	
bmileg	8464	0.04	1.00	0.00	1.0	1.00	1.0	1.0	1.0	
bmxarml	292	0.97	35.11	6.18	10.0	33.60	36.5	39.0	49.2	
bmiarml	8660	0.02	1.00	0.00	1.0	1.00	1.0	1.0	1.0	
bmxarmc	298	0.97	30.56	7.37	12.0	26.40	31.2	35.4	63.3	
bmiarmc	8655	0.02	1.00	0.00	1.0	1.00	1.0	1.0	1.0	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bmxwaist	670	0.92	92.12	22.05	39.8	77.50	92.7	107.0	187.0	▁▅▇▇▁▁
bmiwaist	8513	0.04	1.00	0.00	1.0	1.00	1.0	1.0	1.0	▁▁▇▁▁
bmxhip	2084	0.76	106.26	14.66	69.9	96.40	103.7	113.5	187.1	▁▅▇▅▁▁
bmihip	8499	0.04	1.00	0.00	1.0	1.00	1.0	1.0	1.0	▁▁▇▁▁

```r
gg_miss_var(bpx, show_pct = TRUE) +
  theme_minimal(base_size = 14) +
  labs(title = "Proportion of Missing Values per Variable")   # (not saved)
```



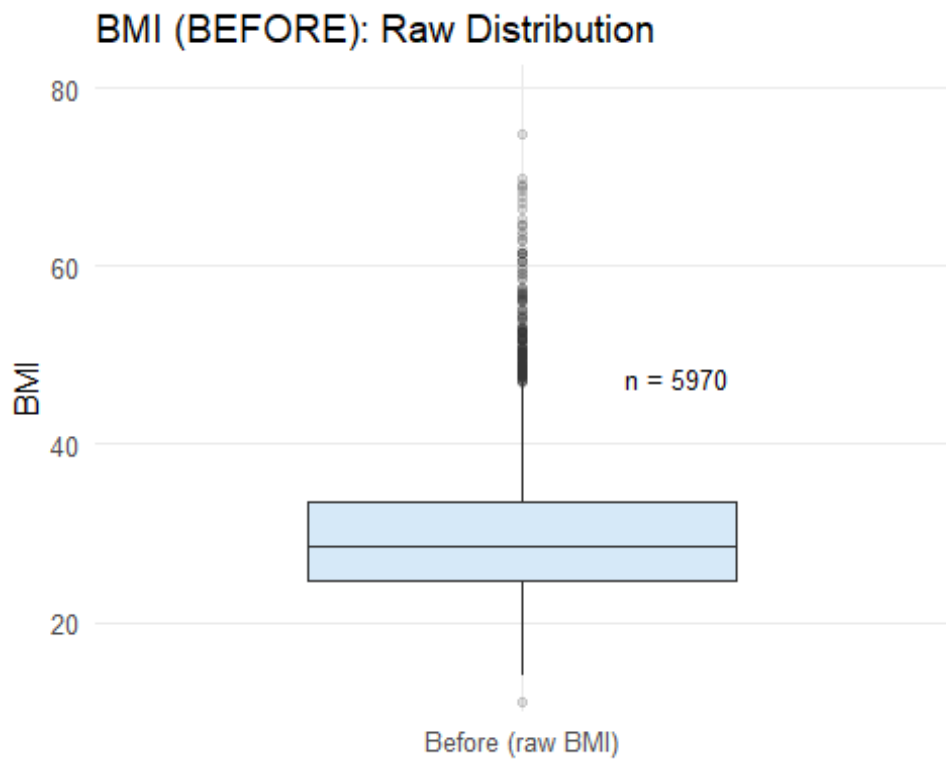

```
# 3) Detect Systolic blood pressure/Diastolic blood pressure reading co
lumns ----------------------------------------------------
#    Support both naming patterns (bpxosy1 or bpxsy1); the 'o' is optio
nal.
sbp_cols <- names(bpx)[stringr::str_detect(names(bpx), "^bpxo?sy[1-3]$
")]  # names() returns the column names of a data frame.(character vect
or)
dbp_cols <- names(bpx)[stringr::str_detect(names(bpx), "^bpxo?di[1-3]$
")]  # str_detect(x, pattern) returns TRUE or FALSE for each element of
 x, depending on whether it matches the regex pattern.
# This code finds the column names in the dataset bpx that correspond t
o the 3 repeated measurements of systolic (sy) or diastolic (di) blood
pressure.


# 4) Build BEFORE (raw) variables and dataset -------------------------
-----------------
#    bmi_raw = original BMI from BMX.
bmi_raw <- bmx %>%
  transmute(seqn, bmi_raw = bmxbmi)  # transmute() keeps only the varia
bles you create, unlike mutate() which keeps all existing variables.

# SBP/DBP
sbpdbp_raw <- bpx %>%
  transmute(seqn,
            sbp_raw = rowMeans(select(., all_of(sbp_cols)), na.rm = TRU
```

```
E),
            dbp_raw = rowMeans(select(., all_of(dbp_cols)), na.rm = TRU
E))

table(demo$riagendr) # $ means "grab" the column from the data frame

##
##    1    2
## 5575 6358

demo <- demo %>%
  mutate(riagendr = as.numeric(riagendr)) %>%       # convert to numeri
c (some values are character)
  filter(is.na(riagendr) | riagendr %in% c(1, 2))   # drop rows with ri
agendr==3 (keep NA and 1/2)

demo_sex <- demo %>%
  transmute(seqn, age = ridageyr,
            sex = factor(riagendr, levels=c(1,2), labels=c("Male","Fema
le")))

dat_raw <- demo_sex %>%
  left_join(bmi_raw, by="seqn") %>%  # join demo (left) with bmi_raw (r
ight) by seqn
  filter(age >= 20) %>%
  mutate(
    # normalize NaN from rowMeans when all readings missing
    bmi_raw = ifelse(is.nan(bmi_raw), NA_real_, bmi_raw) # normalize Na
N to NA
  )

dat_raw <- dat_raw %>%
  left_join(sbpdbp_raw, by = "seqn")  # join demo (left) with bmi_raw
(right) by seqn
```

- Boxplots(Before), Outlier cleaning (BMI, SBP), Boxplots(After).

```
# 5) Draw BEFORE plots ---------------------------------------------
---------------------
# ---- BMI boxplot (BEFORE) ----
bmi_before_df <- dat_raw %>% transmute(stage = "Before (raw BMI)", valu
e = bmi_raw)
x <- bmi_before_df$value
qs <- quantile(x, c(.25,.75), na.rm = TRUE)  # na.rm=TRUE to ignore mis
sing values
iqr <- qs[2]-qs[1]
upper_whisker <- min(max(x, na.rm = TRUE), qs[2] + 1.5*iqr)  # upper wh
isker position, Q3 + 1.5×IQR, capped by max value.
bmi_before_label_y <- upper_whisker + 0.05*iqr
bmi_before_N <- sum(!is.na(x))   # count of non-missing values, !is.na()
```

```
 means "not NA"

p_bmi_before <- ggplot(bmi_before_df, aes(stage, value, fill = stage))
+
  geom_boxplot(width = 0.6, outlier.alpha = 0.15, fatten = 1.2) +
  geom_text(data = tibble(stage="Before (raw BMI)", y=bmi_before_label_
y, N=bmi_before_N),
            aes(stage, y, label=paste0("n = ", N)), hjust = -1, size =
3.5, inherit.aes = FALSE) +
            #size:字型大小
  scale_fill_manual(values = c("Before (raw BMI)" = "#D6E9F8")) +
  labs(title = "BMI (BEFORE): Raw Distribution", x = NULL, y = "BMI") +
  scale_y_continuous(expand = expansion(mult = c(0.02, 0.12))) +
  theme_minimal(base_size = 12) + theme(legend.position = "none", panel.
grid.minor = element_blank())
ggsave("outputs/q1_box_bmi_before.png", p_bmi_before, bg = "white")

## Saving 5 x 4 in image

## Warning: Removed 1839 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).

p_bmi_before

## Warning: Removed 1839 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```

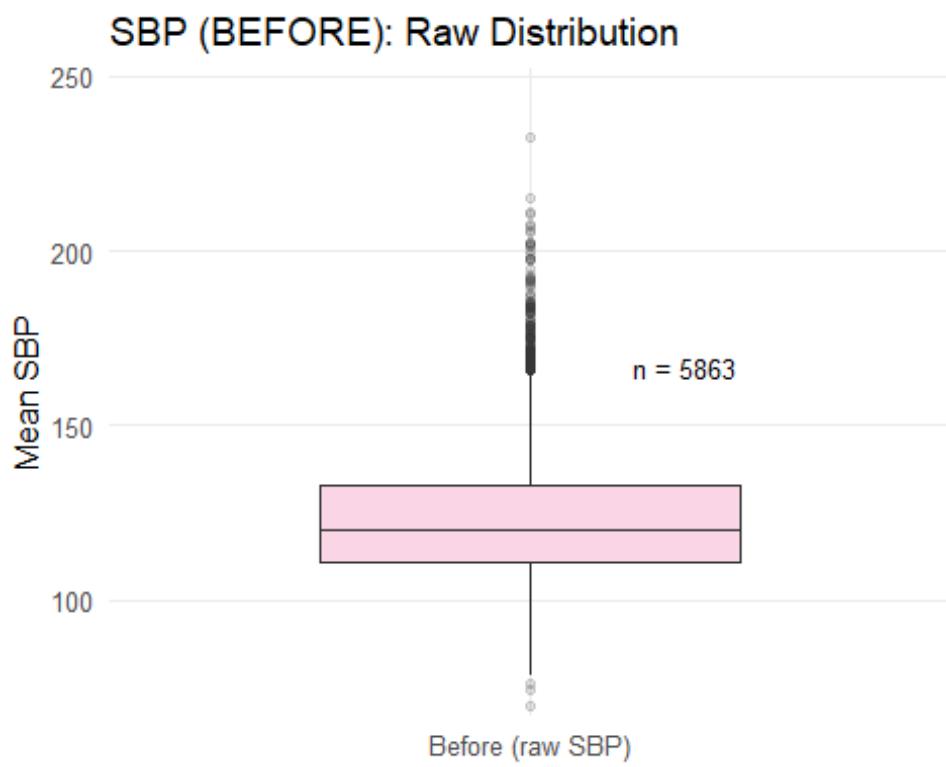



```
# 5b) Draw BEFORE SBP boxplot ----------------------------------------
-----------------
sbp_before_df <- dat_raw %>% transmute(stage = "Before (raw SBP)", valu
e = sbp_raw)
x <- sbp_before_df$value
qs <- quantile(x, c(.25, .75), na.rm = TRUE)
iqr <- qs[2] - qs[1]
upper_whisker <- min(max(x, na.rm = TRUE), qs[2] + 1.5 * iqr)
sbp_before_label_y <- upper_whisker + 0.05 * iqr
sbp_before_N <- sum(!is.na(x))

p_sbp_before <- ggplot(sbp_before_df, aes(stage, value, fill = stage))
+
  geom_boxplot(width = 0.6, outlier.alpha = 0.15, fatten = 1.2) +
  geom_text(data = tibble(stage = "Before (raw SBP)", y = sbp_before_la
bel_y, N = sbp_before_N),
            aes(stage, y, label = paste0("n = ", N)), hjust = -1, size
= 3.5, inherit.aes = FALSE) +
  scale_fill_manual(values = c("Before (raw SBP)" = "#F9D5E5")) +
  labs(title = "SBP (BEFORE): Raw Distribution", x = NULL, y = "Mean SB
P") +
  scale_y_continuous(expand = expansion(mult = c(0.02, 0.12))) +
  theme_minimal(base_size = 12) + theme(legend.position = "none", panel.
grid.minor = element_blank())
ggsave("outputs/q1_box_sbp_before.png", p_sbp_before, bg = "white")

## Saving 5 x 4 in image
```

```
## Warning: Removed 1946 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```

```
p_sbp_before
```

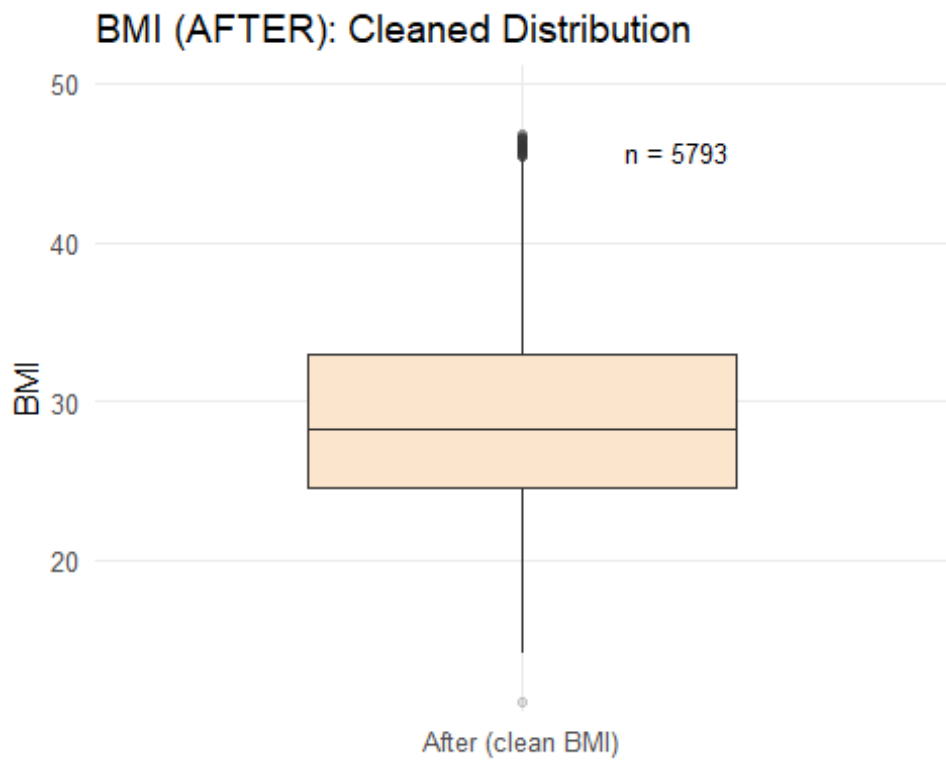
```
## Warning: Removed 1946 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```




```
# 6) OUTLIER CLEANING (then compute cleaned means) ----------------------
-----------------
#    Rule = physiologic bounds + IQR fences + MAD z-score; after remova
l we create "clean" vars.
BMI_LO <- 10; BMI_HI <- 80
bmi_clean <- bmx %>%
  transmute(seqn, bmxbmi) %>%
  mutate(
    q1 = quantile(bmxbmi, 0.25, na.rm=TRUE),
    q3 = quantile(bmxbmi, 0.75, na.rm=TRUE),
    iqr = q3 - q1,
    lo_iqr = q1 - 1.5*iqr,
    hi_iqr = q3 + 1.5*iqr,
    med = median(bmxbmi, na.rm=TRUE),
    madv = mad(bmxbmi, na.rm=TRUE),
    z = ifelse(madv > 0, (bmxbmi - med)/(madv*1.4826), 0),  # 1.4826 to
 make it comparable to SD if normal
```

```r
    flag = (bmxbmi < BMI_LO | bmxbmi > BMI_HI) | (bmxbmi < lo_iqr | bmx
bmi > hi_iqr) | (abs(z) > 3.5),  # flag outliers
    bmxbmi_clean = ifelse(flag, NA_real_, bmxbmi)
  ) %>% select(seqn, bmxbmi_clean)

# 6b) OUTLIER CLEANING for SBP/DBP ------------------------------------
-----------------
SBP_LO <- 70; SBP_HI <- 260
DBP_LO <- 40; DBP_HI <- 150

sbpdbp_clean <- bpx %>%
  transmute(seqn,
            sbp = rowMeans(select(., all_of(sbp_cols)), na.rm = TRUE),
            dbp = rowMeans(select(., all_of(dbp_cols)), na.rm = TRUE))
%>%
  mutate(
    # SBP
    sbp_q1 = quantile(sbp, 0.25, na.rm = TRUE),
    sbp_q3 = quantile(sbp, 0.75, na.rm = TRUE),
    sbp_iqr = sbp_q3 - sbp_q1,
    sbp_lo_iqr = sbp_q1 - 1.5 * sbp_iqr,
    sbp_hi_iqr = sbp_q3 + 1.5 * sbp_iqr,
    sbp_med = median(sbp, na.rm = TRUE),
    sbp_madv = mad(sbp, na.rm = TRUE),
    sbp_z = ifelse(sbp_madv > 0, (sbp - sbp_med) / (sbp_madv * 1.4826),
 0),
    sbp_flag = (sbp < SBP_LO | sbp > SBP_HI) | (sbp < sbp_lo_iqr | sbp
> sbp_hi_iqr) | (abs(sbp_z) > 3.5),
    sbp_clean = ifelse(sbp_flag, NA_real_, sbp),
    # DBP
    dbp_q1 = quantile(dbp, 0.25, na.rm = TRUE),
    dbp_q3 = quantile(dbp, 0.75, na.rm = TRUE),
    dbp_iqr = dbp_q3 - dbp_q1,
    dbp_lo_iqr = dbp_q1 - 1.5 * dbp_iqr,
    dbp_hi_iqr = dbp_q3 + 1.5 * dbp_iqr,
    dbp_med = median(dbp, na.rm = TRUE),
    dbp_madv = mad(dbp, na.rm = TRUE),
    dbp_z = ifelse(dbp_madv > 0, (dbp - dbp_med) / (dbp_madv * 1.4826),
 0),
    dbp_flag = (dbp < DBP_LO | dbp > DBP_HI) | (dbp < dbp_lo_iqr | dbp
> dbp_hi_iqr) | (abs(dbp_z) > 3.5),
    dbp_clean = ifelse(dbp_flag, NA_real_, dbp)
  ) %>%
  select(seqn, sbp_clean, dbp_clean)

# 7) Build AFTER (clean) dataset ---------------------------------------
------------------
dat_clean <- demo_sex %>%
  left_join(bmi_clean, by="seqn") %>%
```

```
  filter(age >= 20) %>%
  mutate(
    bmxbmi_clean = ifelse(is.nan(bmxbmi_clean), NA_real_, bmxbmi_clean)
  # normalize NaN to NA
  )
  # 7b) Merge cleaned SBP into dat_clean ------------------------------
-------------------
dat_clean <- dat_clean %>%
  left_join(sbpdbp_clean, by = "seqn") %>%
  mutate(
    sbp_clean = ifelse(is.nan(sbp_clean), NA_real_, sbp_clean)
  )

# 8) AFTER plots ----------------------------------------------------
----------------
# ---- BMI boxplot (AFTER) ----
bmi_after_df <- dat_clean %>% transmute(stage = "After (clean BMI)", va
lue = bmxbmi_clean)
x <- bmi_after_df$value
qs <- quantile(x, c(.25,.75), na.rm = TRUE);
iqr <- qs[2]-qs[1]
upper_whisker <- min(max(x, na.rm = TRUE), qs[2] + 1.5*iqr)
bmi_after_label_y <- upper_whisker + 0.05*iqr
bmi_after_N <- sum(!is.na(x))

p_bmi_after <- ggplot(bmi_after_df, aes(stage, value, fill = stage)) +
  geom_boxplot(width = 0.6, outlier.alpha = 0.15, fatten = 1.2) +
  geom_text(data = tibble(stage="After (clean BMI)", y=bmi_after_label_
y, N=bmi_after_N),
            aes(stage, y, label=paste0("n = ", N)), hjust = -1, size =
3.5, inherit.aes = FALSE) +
  scale_fill_manual(values = c("After (clean BMI)" = "#FCE5CD")) +
  labs(title = "BMI (AFTER): Cleaned Distribution", x = NULL, y = "BMI")
 +
  scale_y_continuous(expand = expansion(mult = c(0.02, 0.12))) +
  theme_minimal(base_size = 12) + theme(legend.position = "none", panel.
grid.minor = element_blank())
ggsave("outputs/q1_box_bmi_after.png", p_bmi_after, bg = "white")

## Saving 5 x 4 in image

## Warning: Removed 2016 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).

p_bmi_after

## Warning: Removed 2016 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```

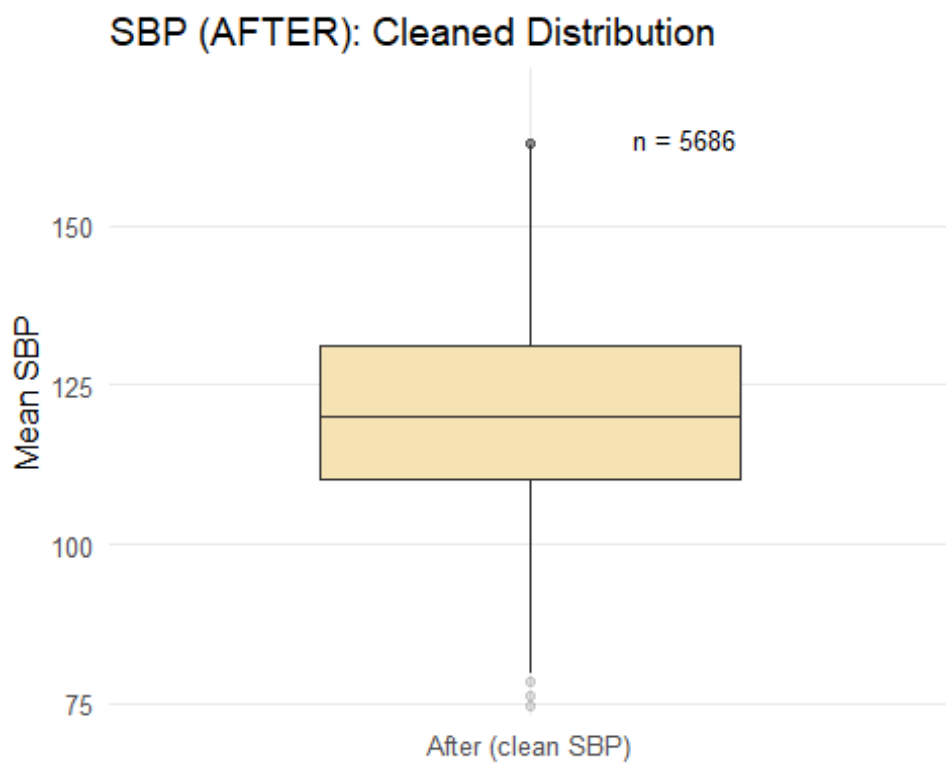



```
# 8b) AFTER SBP boxplot ---------------------------------------------
----------------
sbp_after_df <- dat_clean %>% transmute(stage = "After (clean SBP)", va
lue = sbp_clean)
x <- sbp_after_df$value
qs <- quantile(x, c(.25, .75), na.rm = TRUE)
iqr <- qs[2] - qs[1]
upper_whisker <- min(max(x, na.rm = TRUE), qs[2] + 1.5 * iqr)
sbp_after_label_y <- upper_whisker + 0.05 * iqr
sbp_after_N <- sum(!is.na(x))

p_sbp_after <- ggplot(sbp_after_df, aes(stage, value, fill = stage)) +
  geom_boxplot(width = 0.6, outlier.alpha = 0.15, fatten = 1.2) +
  geom_text(data = tibble(stage = "After (clean SBP)", y = sbp_after_la
bel_y, N = sbp_after_N),
            aes(stage, y, label = paste0("n = ", N)), hjust = -1, size
= 3.5, inherit.aes = FALSE) +
  scale_fill_manual(values = c("After (clean SBP)" = "#F6E3B4")) +
  labs(title = "SBP (AFTER): Cleaned Distribution", x = NULL, y = "Mean
 SBP") +
  scale_y_continuous(expand = expansion(mult = c(0.02, 0.12))) +
  theme_minimal(base_size = 12) + theme(legend.position = "none", panel.
grid.minor = element_blank())
ggsave("outputs/q1_box_sbp_after.png", p_sbp_after, bg = "white")

## Saving 5 x 4 in image
```

```
## Warning: Removed 2123 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```

```
p_sbp_after
```

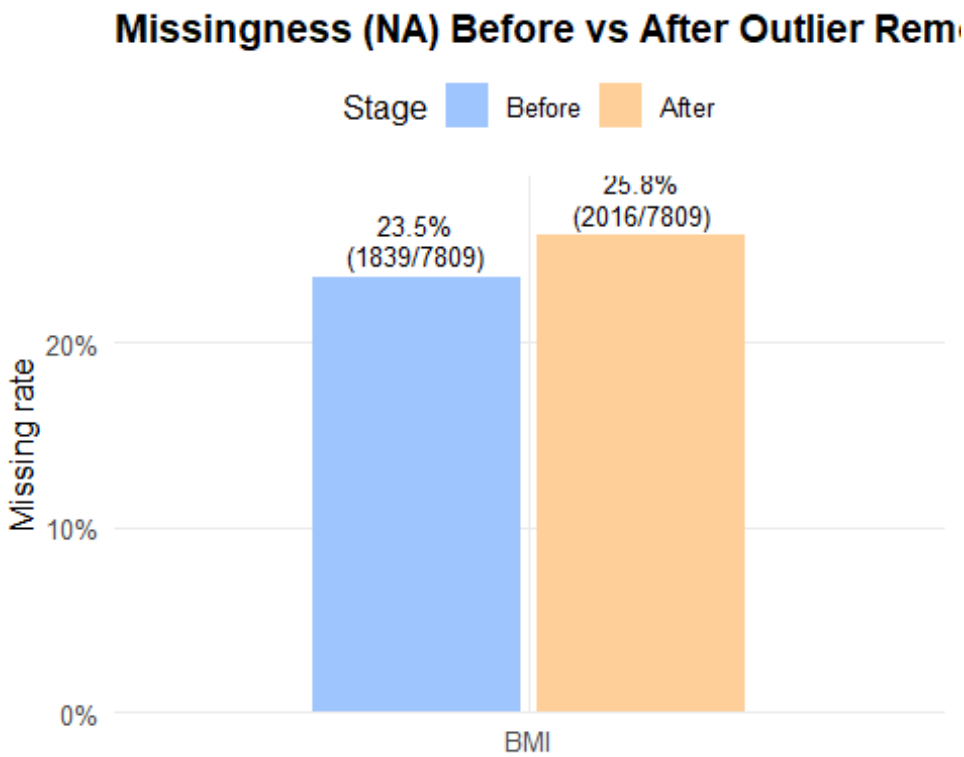
```
## Warning: Removed 2123 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```



- Missingness barplot (Before vs After).

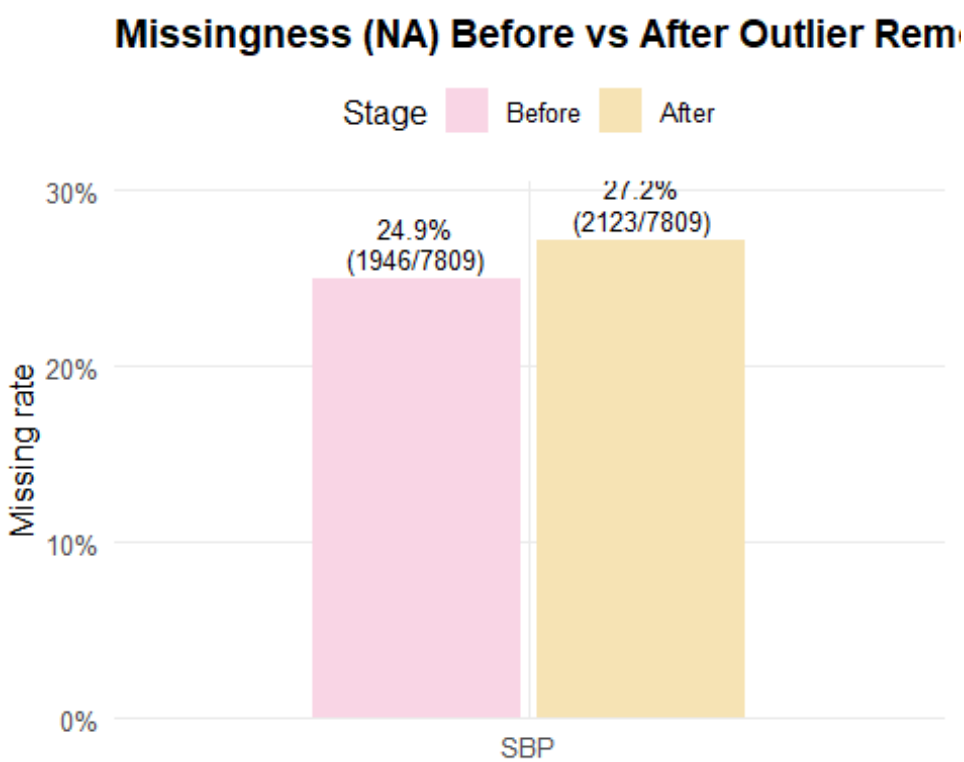
```
# 9) Missing value comparison ----------------------------------------
----------------------------
miss_before <- tibble(
  stage     = "Before",
  variable  = "BMI",
  n_missing = sum(is.na(dat_raw$bmi_raw)),
  n_total   = nrow(dat_raw)
) %>% mutate(p_missing = n_missing / n_total)

miss_after <- tibble(
  stage     = "After",
  variable  = "BMI",
  n_missing = sum(is.na(dat_clean$bmxbmi_clean)),
  n_total   = nrow(dat_clean)
) %>% mutate(p_missing = n_missing / n_total)
```

```r
miss_long <- bind_rows(miss_before, miss_after) %>%
  mutate(stage = factor(stage, levels = c("Before","After")),  # ensure
 order in plot legend
         variable = factor(variable, levels = "BMI"))          # ensure
 order in x-axis

p_na_bar_1 <- ggplot(miss_long, aes(variable, p_missing, fill = stage))
 +
  geom_col(width=0.6, position="dodge") +
                    # dodge to separate bars
  geom_text(aes(label = paste0(scales::percent(p_missing, 0.1),
                               "\n(", n_missing, "/", n_total, ")")),
                    # label on top of bars
            vjust=-0.2, size=3.5) +
  scale_y_continuous(labels=scales::percent) +
  labs(title = "SBP Missingness Before vs After Cleaning", x=NULL, y="M
issing rate") +
  theme_minimal(base_size=12) + theme(legend.position="top")

pos <- position_dodge(width = 0.65) # to align text labels with bars wh
en using dodge

p_na_bar_2 <- ggplot(miss_long, aes(variable, p_missing, fill = stage))
 +
  geom_col(width = 0.6, position = pos) +
  geom_text(aes(label = paste0(scales::percent(p_missing, 0.1),
                               "\n(", n_missing, "/", n_total, ")")),
            position = pos, vjust = -0.2, size = 3.5, lineheight = 0.95)
 +
  scale_y_continuous(labels = scales::percent, expand = expansion(mult
= c(0, 0.12))) +
  scale_fill_manual(values = c("Before" = "#9EC5FE", "After" = "#FFCF99
")) +
  labs(title = "Missingness (NA) Before vs After Outlier Removal (BMI)",
       x = NULL, y = "Missing rate", fill = "Stage") +
  theme_minimal(base_size = 12) +
  theme(panel.grid.minor = element_blank(),
        plot.title = element_text(face = "bold"),
        legend.position = "top")
ggsave("outputs/q1_na_bmi_before_after.png", p_na_bar_2, bg = "white")

## Saving 5 x 4 in image

p_na_bar_2
```




```
# 9b) Missing value comparison for SBP ------------------------------
----------------
miss_before_sbp <- tibble(
  stage     = "Before",
  variable  = "SBP",
  n_missing = sum(is.na(dat_raw$sbp_raw)),
  n_total   = nrow(dat_raw)
) %>% mutate(p_missing = n_missing / n_total)

miss_after_sbp <- tibble(
  stage     = "After",
  variable  = "SBP",
  n_missing = sum(is.na(dat_clean$sbp_clean)),
  n_total   = nrow(dat_clean)
) %>% mutate(p_missing = n_missing / n_total)

miss_long_sbp <- bind_rows(miss_before_sbp, miss_after_sbp) %>%
  mutate(stage = factor(stage, levels = c("Before", "After")),
         variable = factor(variable, levels = "SBP"))

p_na_bar_sbp <- ggplot(miss_long_sbp, aes(variable, p_missing, fill = s
tage)) +
  geom_col(width = 0.6, position = pos) +
  geom_text(aes(label = paste0(scales::percent(p_missing, 0.1),
                               "\n(", n_missing, "/", n_total, ")")),
            position = pos, vjust = -0.2, size = 3.5, lineheight = 0.95)
```

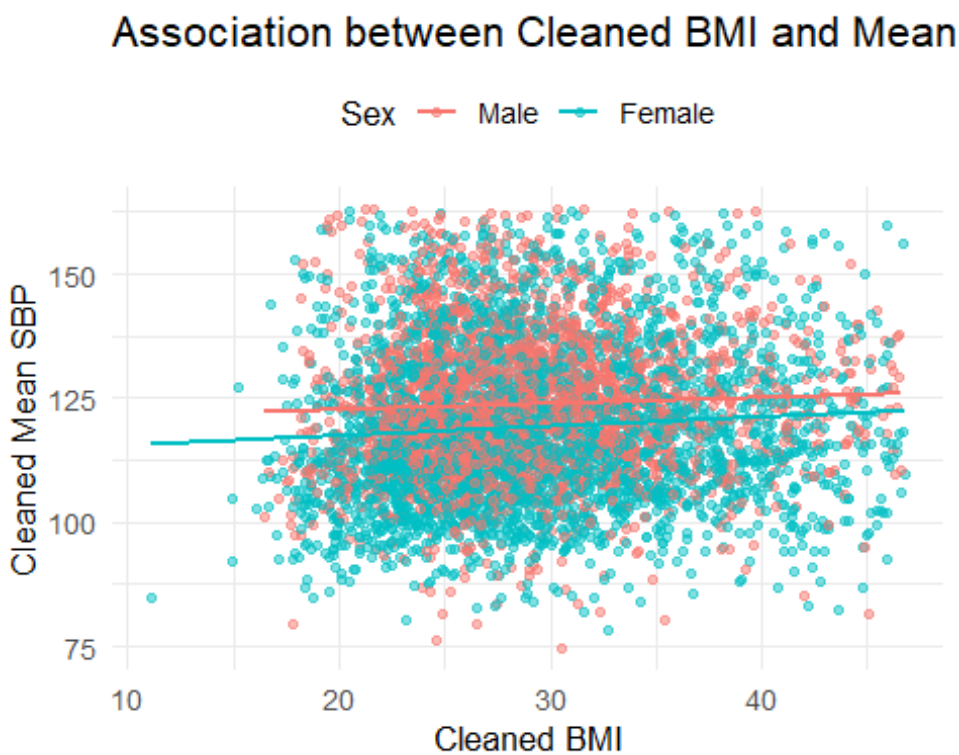
```
 +
  scale_y_continuous(labels = scales::percent, expand = expansion(mult
= c(0, 0.12))) +
  scale_fill_manual(values = c("Before" = "#F9D5E5", "After" = "#F6E3B4
")) +
  labs(title = "Missingness (NA) Before vs After Outlier Removal (SBP)",
       x = NULL, y = "Missing rate", fill = "Stage") +
  theme_minimal(base_size = 12) +
  theme(panel.grid.minor = element_blank(),
        plot.title = element_text(face = "bold"),
        legend.position = "top")
ggsave("outputs/q1_na_sbp_before_after.png", p_na_bar_sbp, bg = "white")

## Saving 5 x 4 in image

p_na_bar_sbp
```



- Scatter plot: BMI vs SBP by sex.

```
# 10) Scatter plot: Cleaned BMI vs Cleaned SBP by sex ----------------
-----------------
scatter_df <- dat_clean %>%
  filter(!is.na(bmxbmi_clean), !is.na(sbp_clean), !is.na(sex))

p_scatter <- ggplot(scatter_df, aes(x = bmxbmi_clean, y = sbp_clean, co
lor = sex)) +
  geom_point(alpha = 0.5, size = 1.5) +
```

```
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Association between Cleaned BMI and Mean SBP by Sex",
       x = "Cleaned BMI", y = "Cleaned Mean SBP", color = "Sex") +
  theme_minimal(base_size = 13) +
  theme(legend.position = "top")
ggsave("outputs/q1_scatter_bmi_sbp_by_sex.png", p_scatter, bg = "white")

## Saving 5 x 4 in image
## `geom_smooth()` using formula = 'y ~ x'

p_scatter

## `geom_smooth()` using formula = 'y ~ x'
```



男女皆呈相似的正相關。

3. Week 6 Components (EDU, Race, and BP Trials)

- Recode and relabel variables (EDU & Race).

```
# 1) Check the original coding distribution
demo %>% count(dmdeduc2)

## # A tibble: 7 × 2
##   dmdeduc2     n
##      <dbl> <int>
## 1        1   373
```

```
## 2         2   666
## 3         3  1749
## 4         4  2370
## 5         5  2625
## 6         9    11
## 7        NA  4139
```

```
demo %>% count(ridreth3)
```

```
## # A tibble: 6 × 2
##   ridreth3     n
##      <dbl> <int>
## 1         1  1117
## 2         2  1373
## 3         3  6217
## 4         4  1597
## 5         6   681
## 6         7   948
```

```
# 2) Recode & relabel
dat_edu <- demo %>%
  transmute(
    seqn,
    age = ridageyr,
    EDU = case_when(                  # case_when() is like ifelse() bu
t for multiple conditions
      dmdeduc2 %in% 1:5 ~ dmdeduc2,   # retain 1–5
      TRUE ~ NA_real_                 # 7/9 -> NA
    ),
    RACE = case_when(
      ridreth3 %in% 1:5 ~ ridreth3,
      TRUE ~ NA_real_
    )
  ) %>%
 mutate(
    EDU = factor(EDU,
                 levels = 1:5,
                 labels = c("<9th grade", "9–11th grade", "High school/
GED",
                            "Some college/AA", "College or above")),
    RACE = factor(RACE,
                  levels = 1:5,
                  labels = c("Mexican American", "Other Hispanic", "Non
-Hispanic White",
                             "Non-Hispanic Black", "Other Race"))
  ) %>%
  left_join(dat_clean %>% select(seqn, bmxbmi_clean), by = "seqn") %>%
  drop_na(EDU, RACE, bmxbmi_clean)
```

- Distribution tables and plots (EDU, Race) & Export CSV + Quarto/Markdown table.

```
# 3) distribution table
edu_dist <- dat_edu %>%
  count(EDU) %>%                  # count occurrences of each education
level
  mutate(prop = n / sum(n),       # calculate proportions
         variable = "EDU") %>%    # add a variable column for clarity
  rename(category = EDU)          # rename EDU to category for consisten
cy

race_dist <- dat_edu %>%
  count(RACE) %>%
  mutate(prop = n / sum(n),
         variable = "RACE") %>%
  rename(category = RACE)

# 4) output table & csv
write.csv(edu_dist, file = "outputs/EDU_distribution.csv", row.names =
FALSE) #row.names=FALSE to avoid writing row numbers
write.csv(race_dist, file = "outputs/RACE_distribution.csv", row.names
= FALSE)

library(knitr)
kable(edu_dist, digits = 3, caption = "Distribution of Educational Atta
inment (EDU)")
```

Distribution of Educational Attainment (EDU)

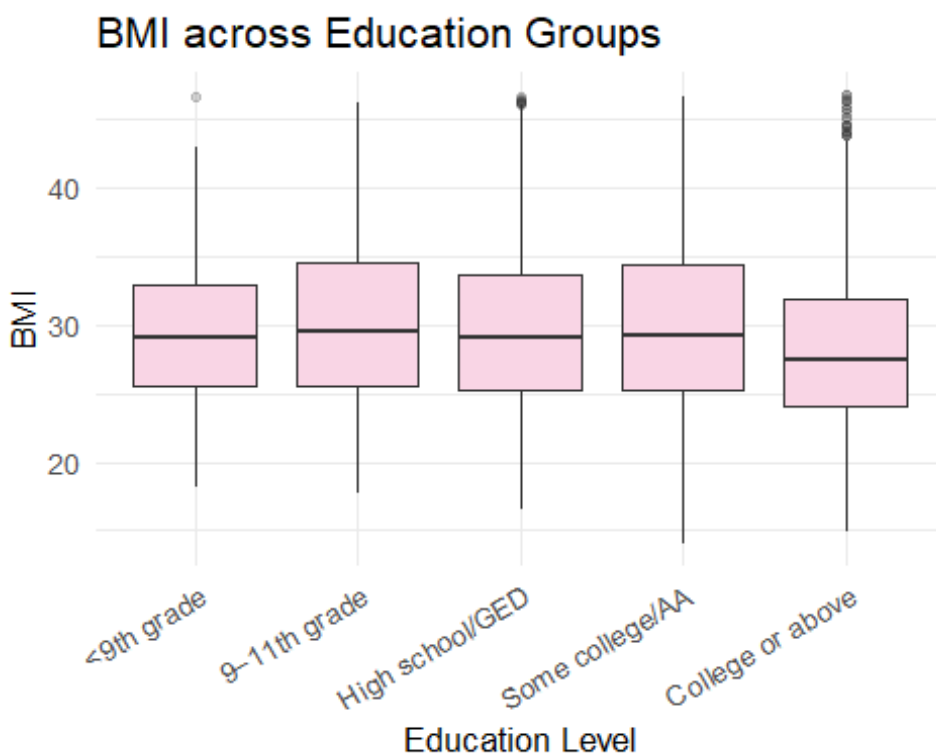
category	n	prop	variable
<9th grade	247	0.048	EDU
9–11th grade	393	0.077	EDU
High school/GED	1112	0.218	EDU
Some college/AA	1545	0.303	EDU
College or above	1799	0.353	EDU

```
kable(race_dist, digits = 3, caption = "Distribution of Race (RACE)")
```

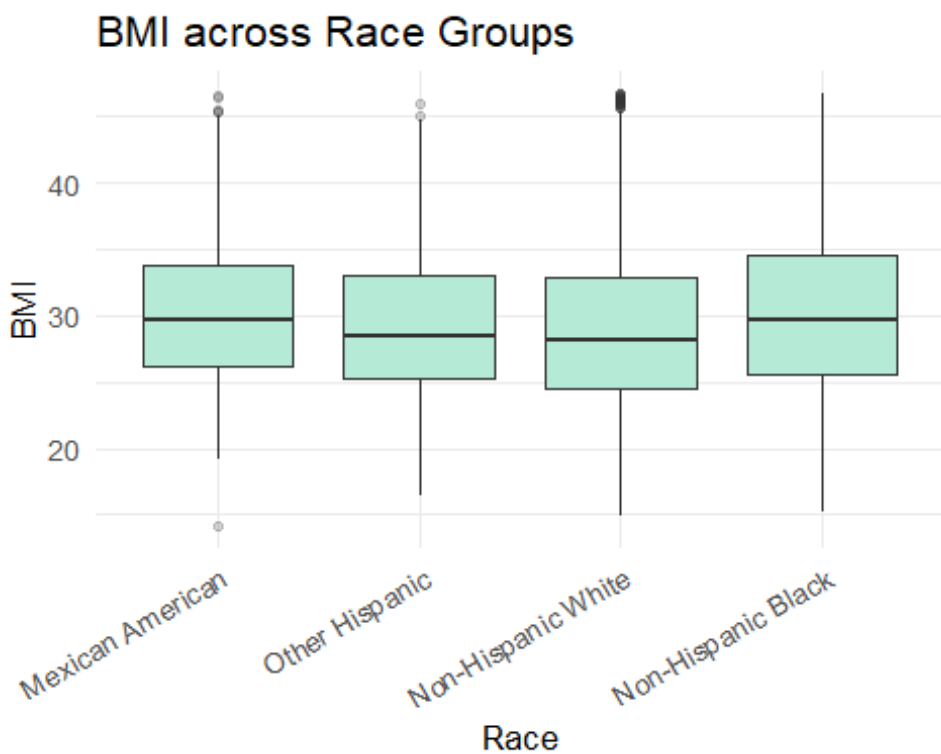
Distribution of Race (RACE)

category	n	prop	variable
Mexican American	390	0.077	RACE
Other Hispanic	593	0.116	RACE
Non-Hispanic White	3425	0.672	RACE
Non-Hispanic Black	688	0.135	RACE

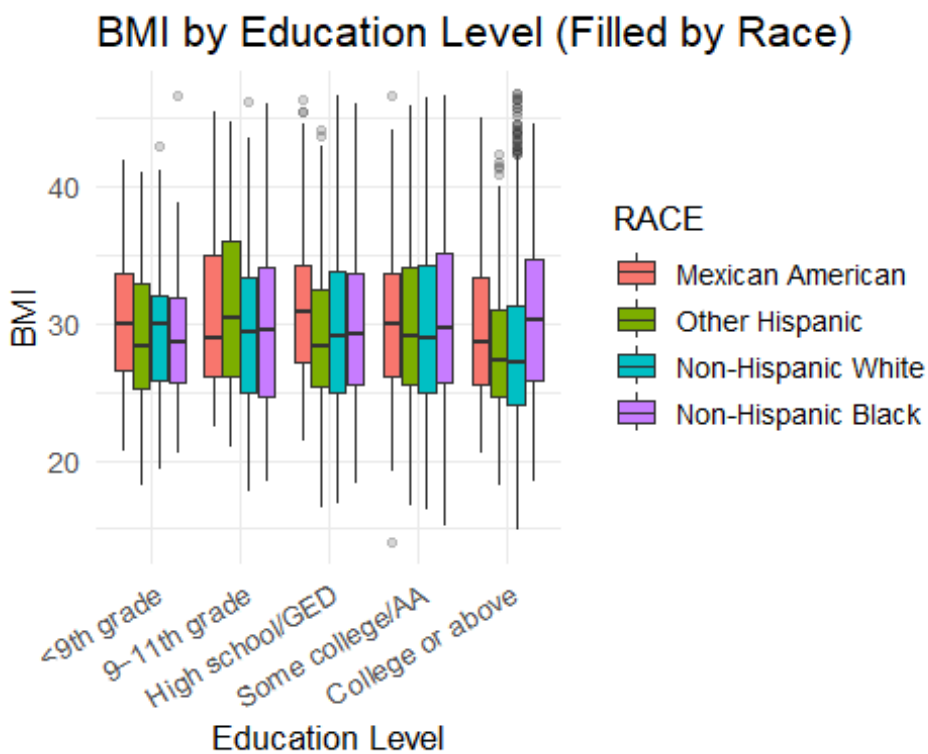
```r
# 5) Boxplot for visualization

# (a) 單純 BMI ~ EDU
p_bmi_edu <- ggplot(dat_edu, aes(x = EDU, y = bmxbmi_clean)) +
  geom_boxplot(outlier.alpha = 0.2, fill = "#F9D5E5") +
  labs(title = "BMI across Education Groups", x = "Education Level", y
= "BMI") +
  theme_minimal(base_size = 13) +
  theme(axis.text.x = element_text(angle = 30, hjust = 1))
ggsave("outputs/BMI_by_EDU.png", p_bmi_edu, width = 10, height = 6, bg
= "white")
p_bmi_edu
```



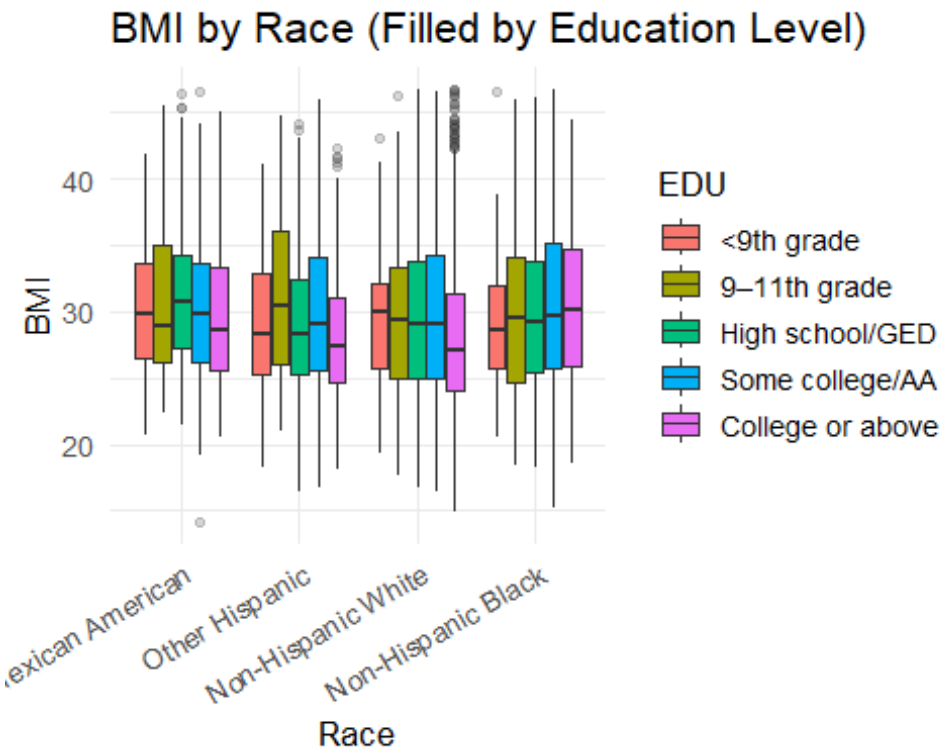

```r
# (b) 單純 BMI ~ RACE
p_bmi_race <- ggplot(dat_edu, aes(x = RACE, y = bmxbmi_clean)) +
  geom_boxplot(outlier.alpha = 0.2, fill = "#B5EAD7") +
  labs(title = "BMI across Race Groups", x = "Race", y = "BMI") +
  theme_minimal(base_size = 13) +
  theme(axis.text.x = element_text(angle = 30, hjust = 1))
ggsave("outputs/BMI_by_RACE.png", p_bmi_race, width = 10, height = 6, b
g = "white")
p_bmi_race
```




```
# (c) BMI ~ EDU, fill by RACE
p_bmi_edu_fill_race <- ggplot(dat_edu, aes(x = EDU, y = bmxbmi_clean, f
ill = RACE)) +
  geom_boxplot(position = position_dodge(0.8), outlier.alpha = 0.2) +
  labs(title = "BMI by Education Level (Filled by Race)", x = "Educatio
n Level", y = "BMI") +
  theme_minimal(base_size = 13) +
  theme(axis.text.x = element_text(angle = 30, hjust = 1))
ggsave("outputs/BMI_by_EDU_filled_by_RACE.png", p_bmi_edu_fill_race, wi
dth = 10, height = 6, bg = "white")

p_bmi_edu_fill_race
```




```
# (d) BMI ~ RACE, fill by EDU
p_bmi_race_fill_edu <- ggplot(dat_edu, aes(x = RACE, y = bmxbmi_clean,
fill = EDU)) +
  geom_boxplot(position = position_dodge(0.8), outlier.alpha = 0.2) +
  labs(title = "BMI by Race (Filled by Education Level)", x = "Race", y
 = "BMI") +
  theme_minimal(base_size = 13) +
  theme(axis.text.x = element_text(angle = 30, hjust = 1))
ggsave("outputs/BMI_by_RACE_filled_by_EDU.png", p_bmi_race_fill_edu, wi
dth = 10, height = 6, bg = "white")
p_bmi_race_fill_edu
```



・EDU / RACE 分布：已輸出的表格顯示各教育程度與族群在樣本中有不同的樣本數與比例。

・教育程度 vs BMI：不同教育程度之間 BMI 的中位數與變異明顯不一；視覺上較低教育程度群組常出現較高的中位數與較寬的 IQR／較多極端值，而高教育程度群組 BMI 較集中且偏低。

・族群 vs BMI：不同族群的 BMI 中位數與分布也有差異，某些族群中位數偏高且散布較廣，極端值數量亦不同。

・交互觀察（EDU × RACE）：在相同教育層級內，不同族群仍呈現不同的 BMI 分布，表示教育與族群對 BMI 的視覺差異是同時存在的。

・總結建議：圖形顯示教育與族群均與 BMI 有可見差異（描述性），若需定量比較或檢定差異大小，建議做回歸或多群組檢定以提供統計支持。

・Reshape BP trials (wide → long).

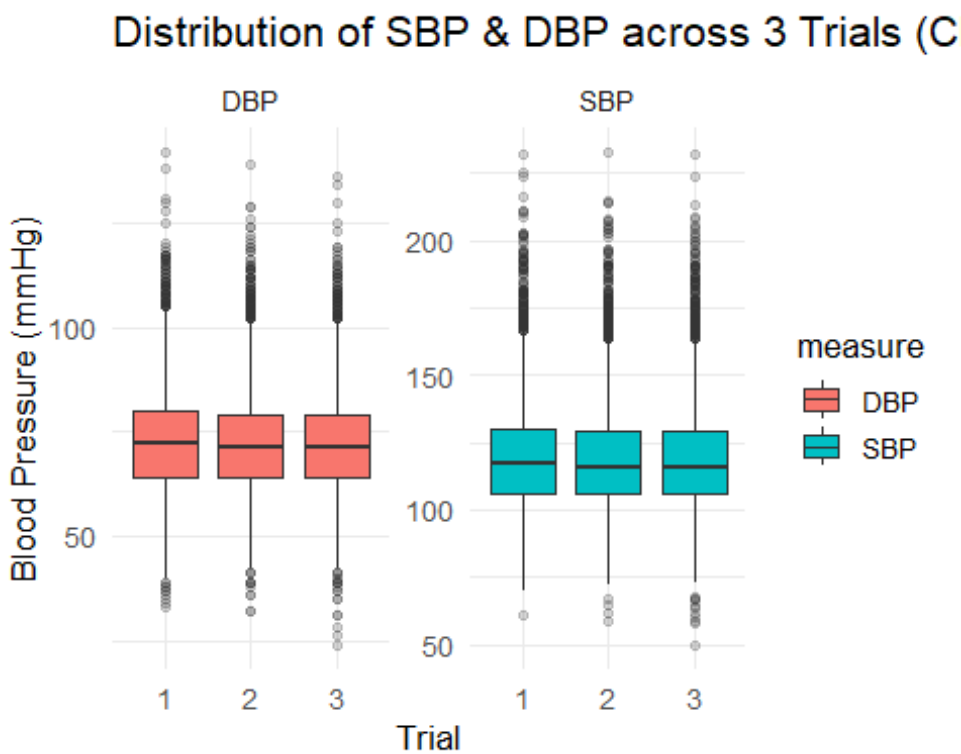
```
library(tidyverse)

# 1) Capture only the necessary columns for SBP & DBP and transform to
long format ------------------------- # nolint
#    Support both naming patterns (bpxosy1 or bpxsy1); the 'o' is optio
nal.
bpx_long_clean <- bpx %>%
  select(seqn, all_of(c(sbp_cols, dbp_cols))) %>%
```

```r
  # From the dataset bpx, you're selecting: seqn: the participant ID. s
bp_cols and dbp_cols: two vectors containing SBP and DBP measurement va
riable names
  # all_of() ensures all the columns listed in those vectors exist — ot
herwise R will throw an error
  pivot_longer(
    cols = -seqn, #take every column except seqn and pivot them.
    names_to = c("measure", "trial"), # Split the original column names
 into two new variables
    names_pattern = "^bpxo([sd]i|sy)([1-3])$",
    # This regular expression defines how column names are split:
    # ^bpxo means names start with "bpxo".
    # ([sd]i|sy) captures the part indicating pressure type: "di" → dia
stolic; "sy" → systolic
    # ([1-3]) captures the measurement number (1, 2, or 3).
    # $ means "end of the string."
    values_to = "value" # The actual blood pressure readings will be st
ored in a new column named value.
  ) %>%
  mutate(
    measure = recode(measure,
                     "sy" = "SBP",
                     "di" = "DBP"),
    trial = as.integer(trial)
  )
```

- Boxplots for SBP & DBP across trials.

```r
bpx_long_clean_PLOT <- ggplot(bpx_long_clean, aes(x = factor(trial), y
= value, fill = measure)) +
  geom_boxplot(outlier.alpha = 0.2) +
  facet_wrap(~ measure, scales = "free_y") +
  labs(title = "Distribution of SBP & DBP across 3 Trials (Cleaned Data)
",
       x = "Trial", y = "Blood Pressure (mmHg)") +
  theme_minimal(base_size = 13)
bpx_long_clean_PLOT
```

```
## Warning: Removed 1802 rows containing non-finite outside the scale r
ange
## (`stat_boxplot()`).
```



4. Homework Extensions (if applicable)

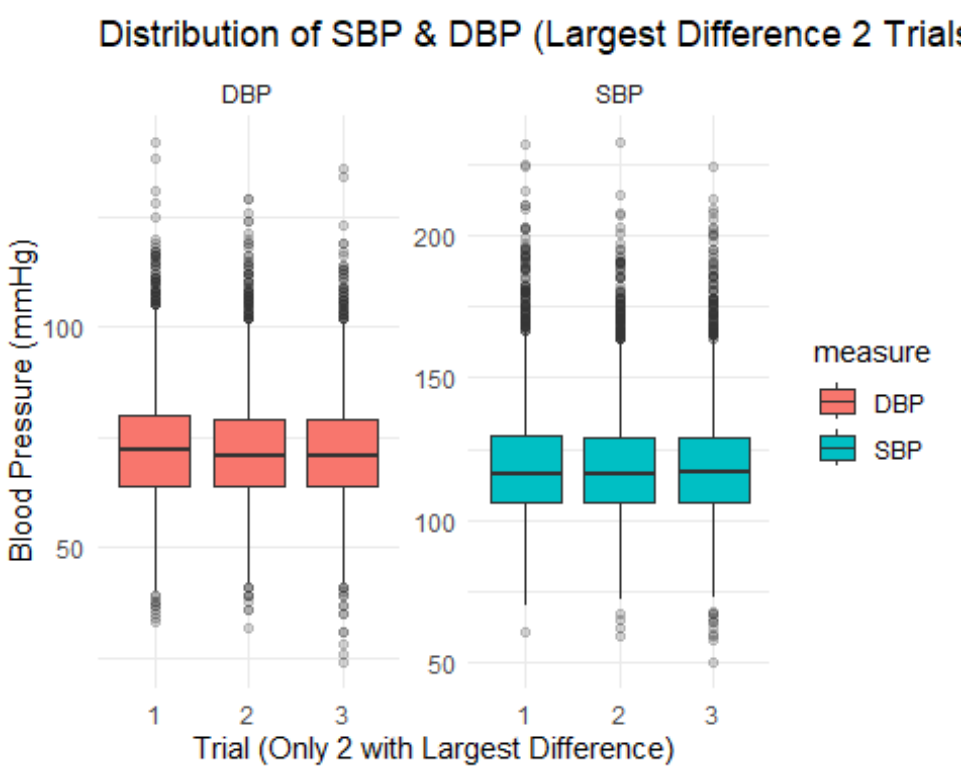
- Race distribution (homework Q2).

As mentioned at “Distribution tables and plots (EDU, Race) & Export CSV + Quarto/Markdown table.”

- Select two trials with the largest difference (homework Q3).

```
# 1. 找出每個人每種血壓（SBP/DBP）三次量測的最大差異組合
bpx_long_diff <- bpx_long_clean %>%
  group_by(seqn, measure) %>%
  filter(sum(!is.na(value)) >= 2) %>% # 至少有兩次量測
  mutate(
    # 計算三次量測的所有兩兩差異
    diff12 = abs(value[trial == 1] - value[trial == 2]),
    diff13 = abs(value[trial == 1] - value[trial == 3]),
    diff23 = abs(value[trial == 2] - value[trial == 3])
  ) %>%
  ungroup()

# 2. 對每個人每種血壓，找出差異最大的那兩次
bpx_long_maxdiff <- bpx_long_diff %>%
  group_by(seqn, measure) %>%
  # 計算三組差異，找最大
```

```r
  mutate(
    maxdiff = max(diff12, diff13, diff23, na.rm = TRUE),
    keep_trials = case_when(
      maxdiff == diff12 ~ list(c(1,2)),
      maxdiff == diff13 ~ list(c(1,3)),
      maxdiff == diff23 ~ list(c(2,3)),
      TRUE ~ list(NA)
    )
  ) %>%
  # 保留最大差異的那兩次
  filter(trial %in% unlist(keep_trials[1])) %>%
  ungroup()

# 3. 繪圖
bpx_long_maxdiff_PLOT <- ggplot(bpx_long_maxdiff, aes(x = factor(trial),
 y = value, fill = measure)) +
  geom_boxplot(outlier.alpha = 0.2) +
  facet_wrap(~ measure, scales = "free_y") +
  labs(title = "Distribution of SBP & DBP (Largest Difference 2 Trials
per Subject)",
       x = "Trial (Only 2 with Largest Difference)", y = "Blood Pressur
e (mmHg)") +
  theme_minimal()
bpx_long_maxdiff_PLOT
```



視覺上三次量測（Trial 1 – 3）在 SBP 與 DBP 的 boxplot 中，中位數與 IQR 大量重疊、分布類似，且沒有明顯的系統性上升或下降趨勢。 對每位保留「差異最大兩次」後的比較亦顯示整體分布差異並不大。 因此可推論：這些血壓量測很可能是在同一天、短時間內（數分鐘內）完成，而非長時間間隔量測。

5. Conclusion

・Summary

在清理後的資料中，BMI 在不同教育程度與族群間分布有可見差異：較低教育層級與某些族群的 BMI 中位數及變異較高；交互觀察顯示同一教育層級內不同族群仍有差異。BMI 與平均 SBP 視覺上呈明顯正相關，男女皆為正向趨勢且大致相似（散點與分性別回歸線顯示僅有輕微位置/斜率差異）。三次血壓量測的箱型圖與「保留最大差異兩次」的結果均未顯示系統性趨勢，整體變異合理，推論量測應為同一次訪談內、短時間內完成（非長間隔測量）。

・What I learned about reproducible workflows

・明確分段： 資料載入 → 變數識別 → 前處理/重編碼 → 離群值清理 → 建立清理後資料集 → 繪圖與輸出；每一步保留可執行的 script。

・可重現性實作：使用 tidyverse 一致語法、固定檔案目錄（outputs/data_raw）、將中間結果寫入檔案（CSV/圖檔），並在程式中註記清理規則與假設。

・模組化與紀錄：把重複邏輯封成函式或區塊、加入註解與版本資訊（sessionInfo 或 git commit），便於追蹤與重複執行。

・可追溯輸出：所有表格與圖都存檔（outputs），使報告與原始分析可對應；避免硬編路徑、保留原始與清理後變數以利檢查。