Week 5-6 Data Preprocessing HW

B12801002 公衛三 張元貞

2025-10-19

1. Introduction & Setup

- Briefly describe the purpose of this analysis (NHANES data, 2021–2023).
- Load all required packages and datasets.
- Ensure reproducibility by including all code chunks in order.

This analysis utilizes data from the 2021-2023 National Health and Nutrition Examination Survey (NHANES) to perform a comprehensive data cleaning and exploratory analysis workflow. In the first phase (Week 5), the primary focus was on Body Mass Index (BMI) and Systolic Blood Pressure (SBP). We compared the data distribution and missingness rates of these variables before and after a rigorous outlier cleaning process, visualizing the results using boxplots. The second phase (Week 6) shifted to exploring sociodemographic disparities by analyzing the distribution of BMI across different race/ethnicity and educational attainment groups. Additionally, we conducted a detailed distribution analysis of repeated blood pressure (BP) measurements to examine trial-to-trial variability.

2. Week 5 Components (BMI & SBP Cleaning)

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 ? Steps for answering the question:

I. Install & load the related packages

```
# 1) Packages and folders ----------------------------------------------------
pkgs <- c("tidyverse","haven","janitor","stringr","scales","skimr","naniar") # 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))
```

```
## Warning: package 'tidyverse' was built under R version 4.4.3
```

```
## Warning: package 'ggplot2' was built under R version 4.4.2
```

```
## Warning: package 'tibble' was built under R version 4.4.2
```

```
## Warning: package 'tidyr' was built under R version 4.4.2
```

```
## Warning: package 'readr' was built under R version 4.4.3
```

```
## Warning: package 'purrr' was built under R version 4.4.2
```

```
## Warning: package 'dplyr' was built under R version 4.4.2
```

```
## Warning: package 'forcats' was built under R version 4.4.3
```

```
## Warning: package 'lubridate' was built under R version 4.4.3
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.5
## ✔ forcats   1.0.1     ✔ stringr   1.5.1
## ✔ ggplot2   3.5.1     ✔ tibble    3.2.1
## ✔ lubridate 1.9.4     ✔ tidyr     1.3.1
## ✔ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
## Warning: package 'haven' was built under R version 4.4.3
```

```
## Warning: package 'janitor' was built under R version 4.4.3
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
## Warning: package 'scales' was built under R version 4.4.2
```

```
##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
## Warning: package 'skimr' was built under R version 4.4.3
```

```
## Warning: package 'naniar' was built under R version 4.4.3
```

```
##
## 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 <- "C:\\Users\\USER\\Desktop\\健康大數據"                    # folder containing .XPT files

getwd()  # check working directory
```

```
## [1] "C:/Users/USER/Desktop/健康大數據/Big-Data-Hw"
```

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

II. Read the raw data files and quick view to the datasets

```
# 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	136344.00	3444.90	130378.00	133361.00	136344.00	139327.00	142310.0	▇▇▇▇▇
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	▇▁▁▁▇
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	▇▁▁▁▁
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	▁▁▇▁▃

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
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	▆▁▁▁▇
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	▇▁▁▁▇
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	3449.49	130378	133335	136382	139325	142310	▇▇▇▇▇
bpaocsz	190	0.98	3.52	0.67	2	3	4	4	5	▁▇▁▇▁
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	▁▅▇▁▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bpxopls1	284	0.96	72.34	12.72	35	63	71	80	158	▂▇▂▁▁
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	▂▇▂▁▁

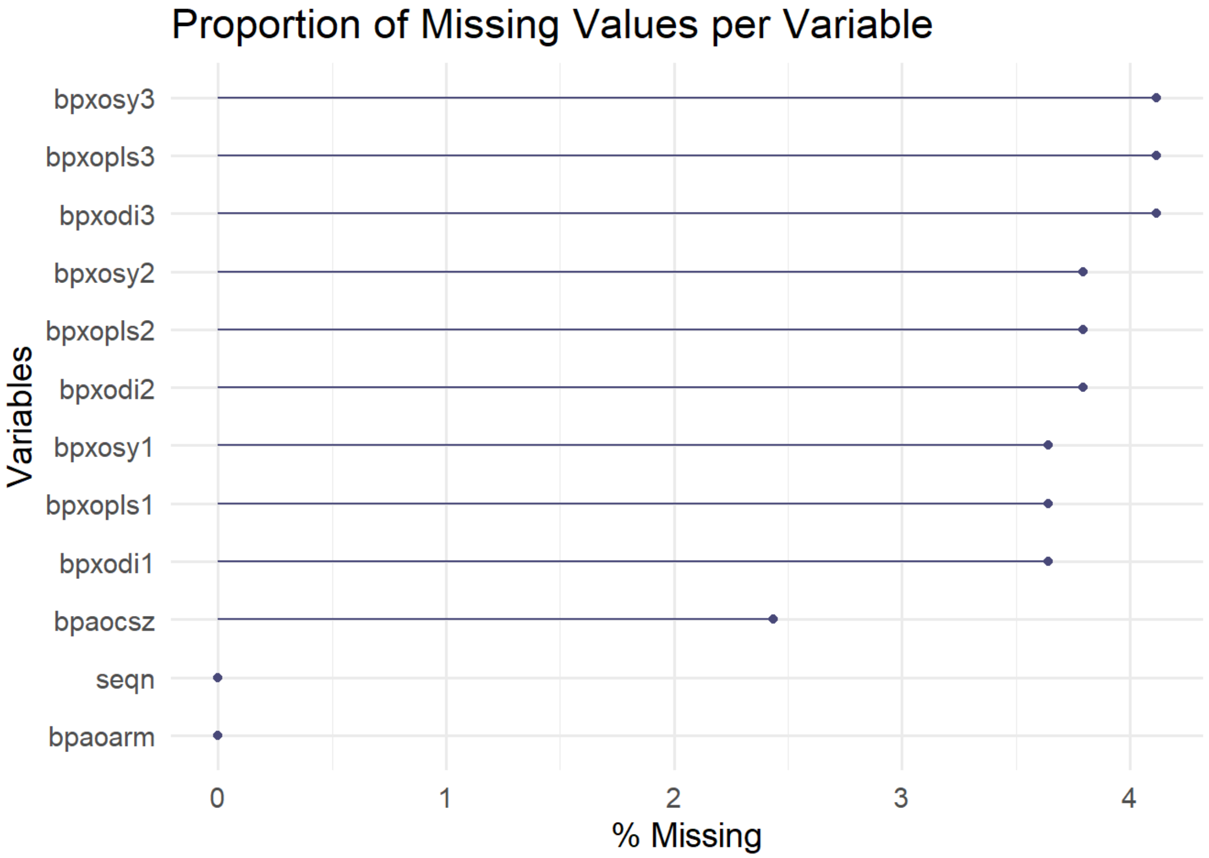
Name	bmx
Number of rows	8860
Number of columns	22
_______________________	
Column type frequency:	
numeric	22
________________________	
Group variables	None

Variable type: numeric

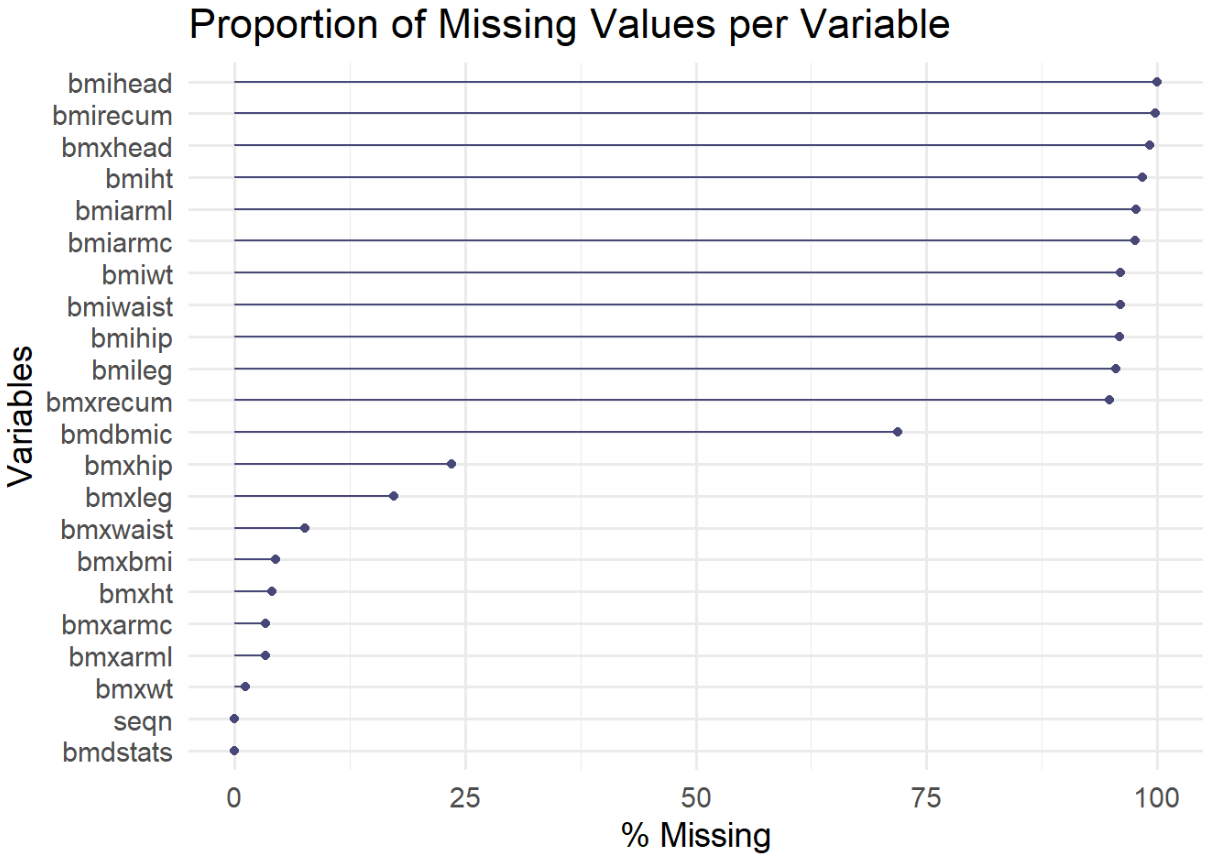
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
seqn	0	1.00	136345.83	3453.78	130378.0	133319.75	136377.5	139336.2	142310.0	▇▇▇▇▇
bmdstats	0	1.00	1.13	0.50	1.0	1.00	1.0	1.0	4.0	▇▁▁▁▁
bmxwt	106	0.99	70.55	30.39	2.7	54.20	71.7	89.1	248.2	▃▇▂▁▁
bmiwt	8515	0.04	2.88	0.62	1.0	3.00	3.0	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	8.14	11.1	21.60	26.4	31.7	74.8	▅▇▂▁▁
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	▁▁▇▁▁
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	▂▇▂▁▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bmihip	8499	0.04	1.00	0.00	1.0	1.00	1.0	1.0	1.0	▁▁▇▁▁

```
gg_miss_var(bpx, show_pct = TRUE) +
  theme_minimal(base_size = 14) +
  labs(title = "Proportion of Missing Values per Variable")
```



```
gg_miss_var(bmx, show_pct = TRUE) +
  theme_minimal(base_size = 14) +
  labs(title = "Proportion of Missing Values per Variable")
```



III. Find out the targeted column from datasets for this question and define them

```
sbp_cols <- names(bpx)[stringr::str_detect(names(bpx), "^bpxo?sy[1-3]$")]  # names() returns the column nam
es of a data frame.(character vector)
dbp_cols <- names(bpx)[stringr::str_detect(names(bpx), "^bpxo?di[1-3]$")]  # str_detect(x, pattern) returns
TRUE or FALSE for each element of x, depending on whether it matches the regex pattern.
```

IV. Build the original variables (including checking the coding correctness) and dataset for constructing plots before cleaning

```
bmi_raw <- bmx %>%
  transmute(seqn, bmi_raw = bmxbmi)

sbp_raw <- bpx %>%
  transmute(seqn, sbp_raw = rowMeans(select(., all_of(sbp_cols)), na.rm = TRUE))
dbp_raw <- bpx %>%
  transmute(seqn, dbp_raw = rowMeans(select(., all_of(dbp_cols)), na.rm=TRUE))

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

```
##
##    1    2
## 5575 6358
```

```
# 清理人口統計數據並創建 sex 因子變數
demo_sex <- demo %>%
  filter(is.na(riagendr) | riagendr %in% c(1, 2)) %>%
  transmute(
    seqn,
    age = ridageyr,
    sex = factor(riagendr, levels = c(1, 2), labels = c("Male", "Female"))
  )

## 數據整合與最終清理
dat_raw <- demo_sex %>%
  left_join(sbp_raw, by = "seqn") %>%  # 合併 SBP
  left_join(dbp_raw, by = "seqn") %>%  # 合併 DBP
  left_join(bmi_raw, by = "seqn") %>%  # 合併 BMI
  filter(age >= 20) %>% # 限制年齡 >= 20 歲
  mutate(
    # 將 rowMeans 產生的 NaN 轉換為標準 NA
    sbp_raw = ifelse(is.nan(sbp_raw), NA_real_, sbp_raw),
    dbp_raw = ifelse(is.nan(dbp_raw), NA_real_, dbp_raw),
    bmi_raw = ifelse(is.nan(bmi_raw), NA_real_, bmi_raw)
  )
```

V. Draw the raw data boxplots of BMI & mean SBP separately

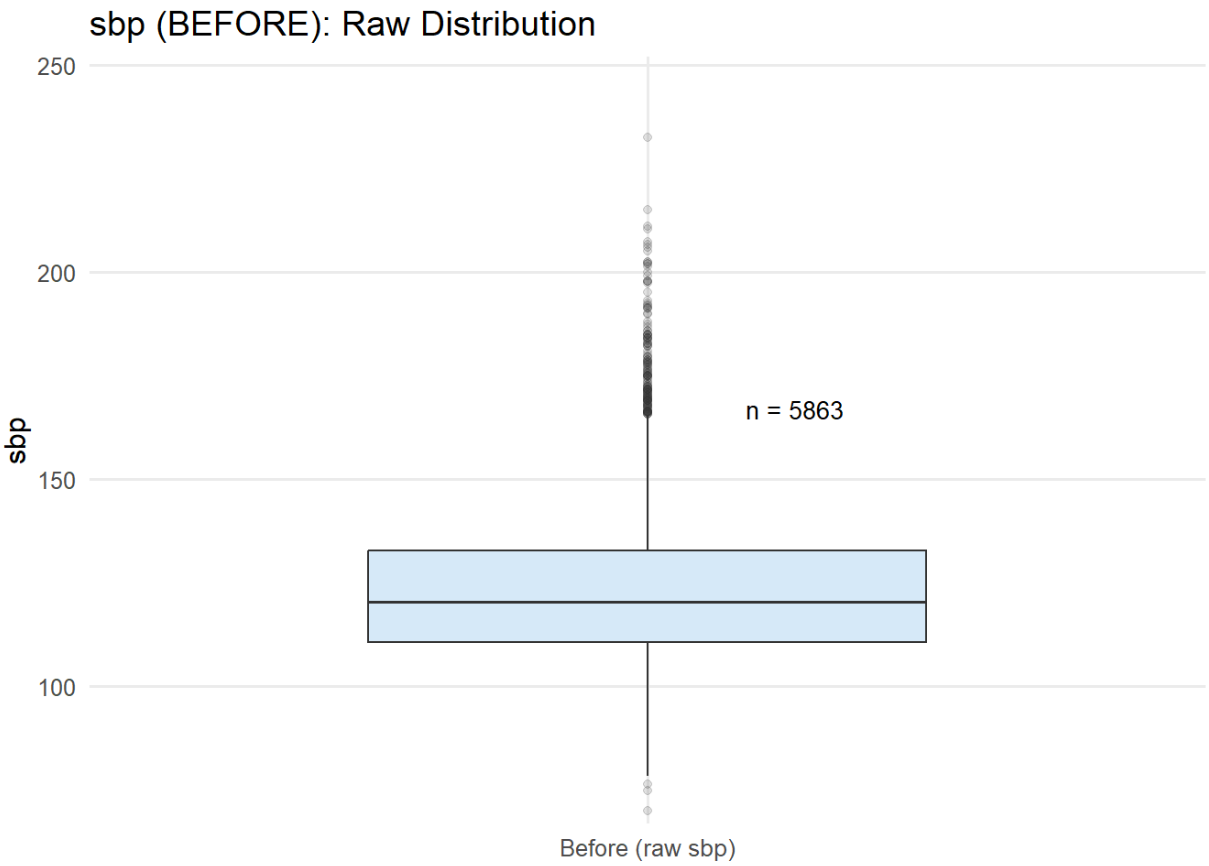
```
# ---- sbp boxplot (BEFORE) ----
sbp_before_df <- dat_raw %>% transmute(stage = "Before (raw sbp)", value = sbp_raw)
x_sbp <- sbp_before_df$value
qs_sbp <- quantile(x_sbp, c(.25,.75), na.rm = TRUE)  # na.rm=TRUE to ignore missing values
iqr_sbp <- qs_sbp[2]-qs_sbp[1]
upper_whisker <- min(max(x_sbp, na.rm = TRUE), qs_sbp[2] + 1.5*iqr_sbp)  # upper whisker position, Q3 + 1.5
×IQR, capped by max value.
sbp_before_label_y <- upper_whisker + 0.05*iqr_sbp
sbp_before_N <- sum(!is.na(x_sbp))   # count of non-missing values, !is.na() means "not NA"

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_label_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)" = "#D6E9F8")) +
  labs(title = "sbp (BEFORE): Raw Distribution", x = NULL, y = "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_before.png", p_sbp_before, bg = "white")
```

```
## Saving 7 x 5 in image
```

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

```
print(p_sbp_before)
```

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



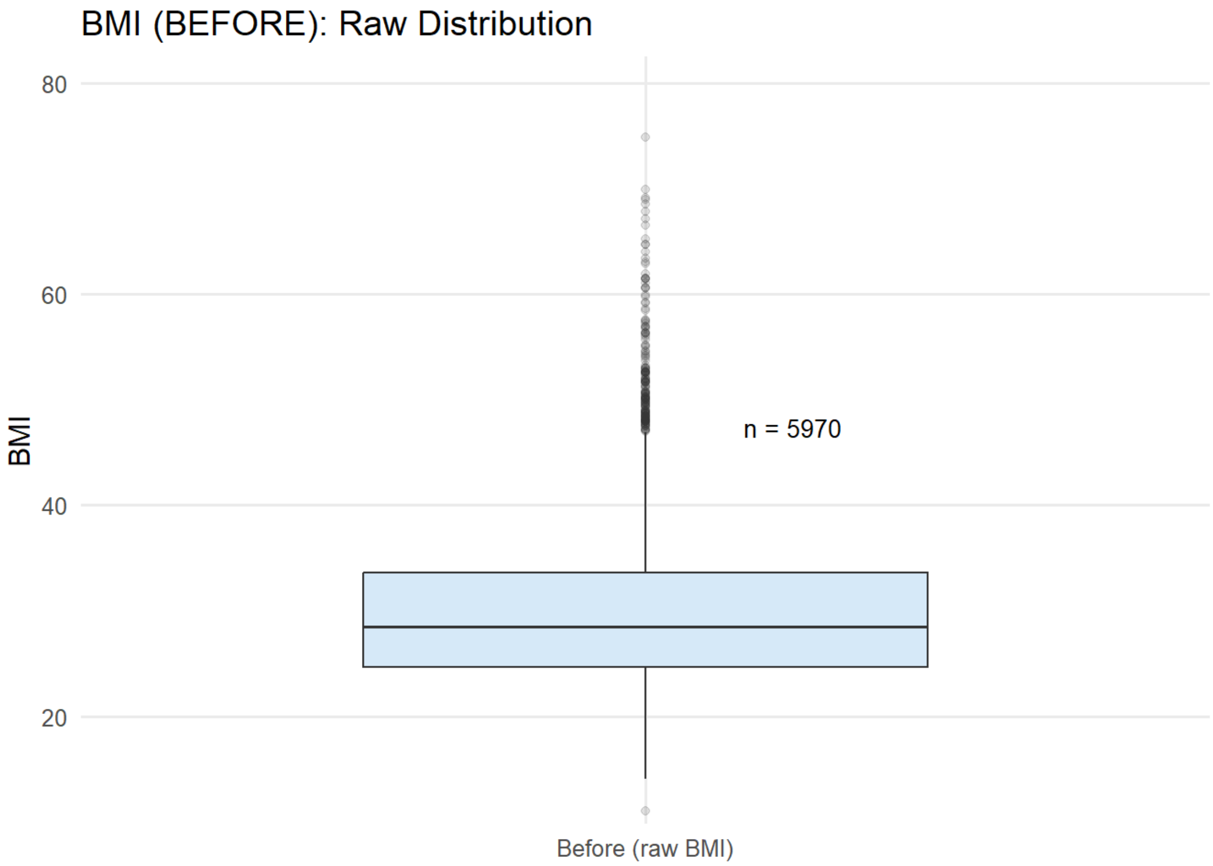
```
# ---- bmi boxplot (BEFORE) ----
bmi_before_df <- dat_raw %>% transmute(stage = "Before (raw BMI)", value = bmi_raw)
x_bmi <- bmi_before_df$value
qs_bmi <- quantile(x_bmi, c(.25,.75), na.rm = TRUE)  # na.rm=TRUE to ignore missing values
iqr_bmi <- qs_bmi[2]-qs_bmi[1]
upper_whisker <- min(max(x_bmi, na.rm = TRUE), qs_bmi[2] + 1.5*iqr_bmi)  # upper whisker position, Q3 + 1.5
×IQR, capped by max value.
bmi_before_label_y <- upper_whisker + 0.05*iqr_bmi
bmi_before_N <- sum(!is.na(x_bmi))   # 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) +
  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 7 x 5 in image
```

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

```
print(p_bmi_before)
```

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



VI. Outlier cleaning (Rule = physiologic bounds + IQR fences + MAD z-score)

```
sbp_LO <- 70; sbp_HI <- 260
sbp_clean <- sbp_raw %>%
  mutate(
    q1_sbp = quantile(sbp_raw, 0.25, na.rm=TRUE),
    q3_sbp = quantile(sbp_raw, 0.75, na.rm=TRUE),
    iqr_sbp = q3_sbp - q1_sbp,
    lo_iqr_sbp = q1_sbp - 1.5*iqr_sbp,
    hi_iqr_sbp = q3_sbp + 1.5*iqr_sbp,
    med_sbp = median(sbp_raw, na.rm=TRUE),
    madv_sbp = mad(sbp_raw, na.rm=TRUE),
    z = ifelse(madv_sbp > 0, (sbp_raw - med_sbp)/(madv_sbp*1.4826), 0),  # 1.4826 to make it comparable to
SD if normal
    flag = (sbp_raw < sbp_LO | sbp_raw > sbp_HI) | (sbp_raw < lo_iqr_sbp | sbp_raw > hi_iqr_sbp) | (abs(z)
> 3.5),  # flag outliers
    sbp_raw_clean = ifelse(flag, NA_real_, sbp_raw)
  ) %>% select(seqn, sbp_raw_clean)



BMI_LO <- 10; BMI_HI <- 80
bmi_clean <- bmx %>%
  transmute(seqn, bmxbmi) %>%
  mutate(
    q1_bmi = quantile(bmxbmi, 0.25, na.rm=TRUE),
    q3_bmi = quantile(bmxbmi, 0.75, na.rm=TRUE),
    iqr_bmi = q3_bmi - q1_bmi,
    lo_iqr_bmi = q1_bmi - 1.5*iqr_bmi,
    hi_iqr_bmi = q3_bmi + 1.5*iqr_bmi,
    med_bmi = median(bmxbmi, na.rm=TRUE),
    madv_bmi = mad(bmxbmi, na.rm=TRUE),
    z = ifelse(madv_bmi > 0, (bmxbmi - med_bmi)/(madv_bmi*1.4826), 0),  # 1.4826 to make it comparable to S
D if normal
    flag = (bmxbmi < BMI_LO | bmxbmi > BMI_HI) | (bmxbmi < lo_iqr_bmi | bmxbmi > hi_iqr_bmi) | (abs(z) > 3.
5),  # flag outliers
    bmxbmi_clean = ifelse(flag, NA_real_, bmxbmi)
  ) %>% select(seqn, bmxbmi_clean)
```

VII. Build AFTER cleaning datasets

```
dat_clean <- demo_sex %>%
  left_join(sbp_clean, by="seqn") %>%
  left_join(bmi_clean, by="seqn") %>%
  filter(age >= 20) %>%
  mutate(
    sbp_raw_clean = ifelse(is.nan(sbp_raw_clean), NA_real_, sbp_raw_clean),  # normalize NaN to names()
    bmxbmi_clean = ifelse(is.nan(bmxbmi_clean), NA_real_, bmxbmi_clean)  # normalize NaN to NA
  )
```

VIII. Draw the cleaned data boxplots of BMI & mean SBP

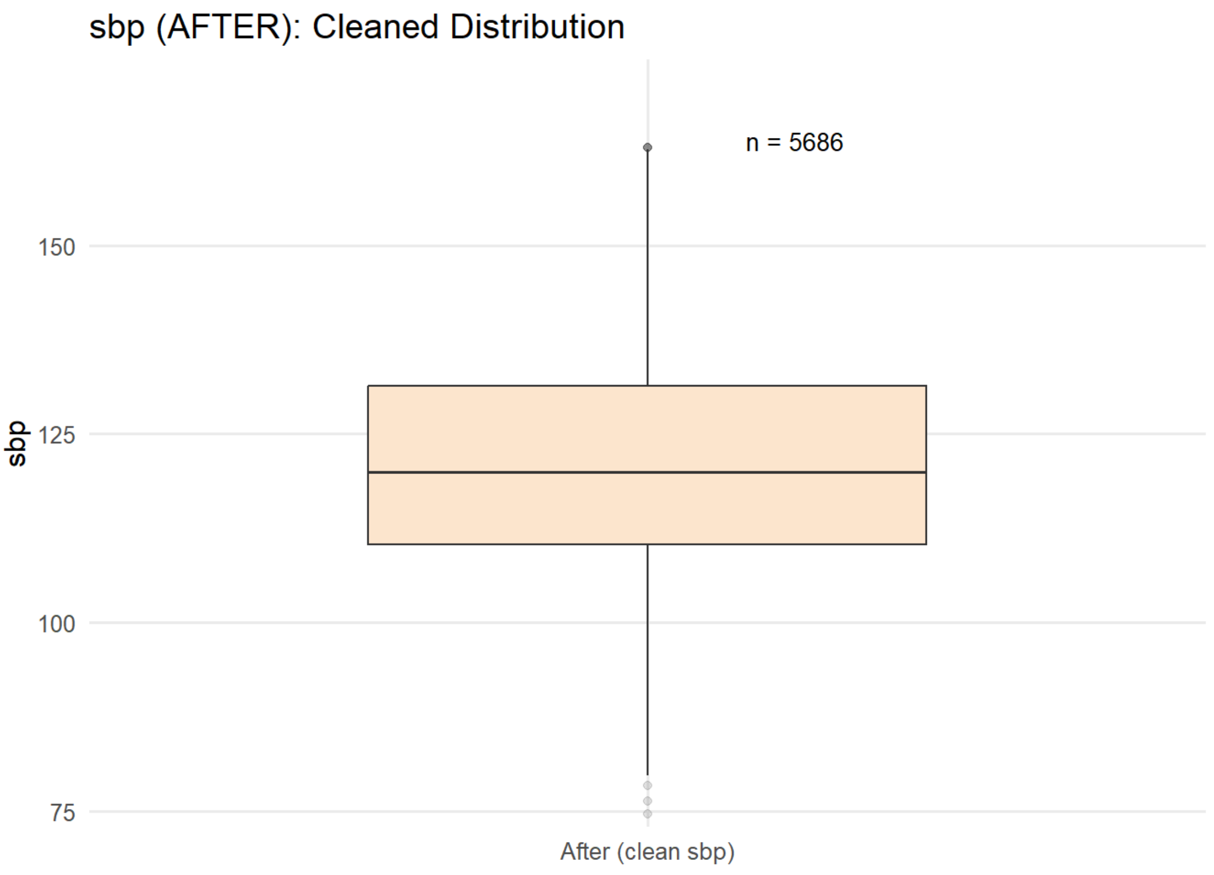
```
# ---- SBP boxplot (AFTER) ----
sbp_after_df <- dat_clean %>% transmute(stage = "After (clean sbp)", value = sbp_raw_clean)
x_sbp <- sbp_after_df$value
qs_sbp <- quantile(x_sbp, c(.25,.75), na.rm = TRUE);
iqr_sbp <- qs_sbp[2]-qs_sbp[1]
upper_whisker_sbp <- min(max(x_sbp, na.rm = TRUE), qs_sbp[2] + 1.5*iqr_sbp)
sbp_after_label_y <- upper_whisker_sbp + 0.05*iqr_sbp
sbp_after_N <- sum(!is.na(x_sbp))

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_label_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)" = "#FCE5CD")) +
  labs(title = "sbp (AFTER): Cleaned Distribution", x = NULL, y = "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 7 x 5 in image
```

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

```
print(p_sbp_after)
```

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



sbp (AFTER): Cleaned Distribution

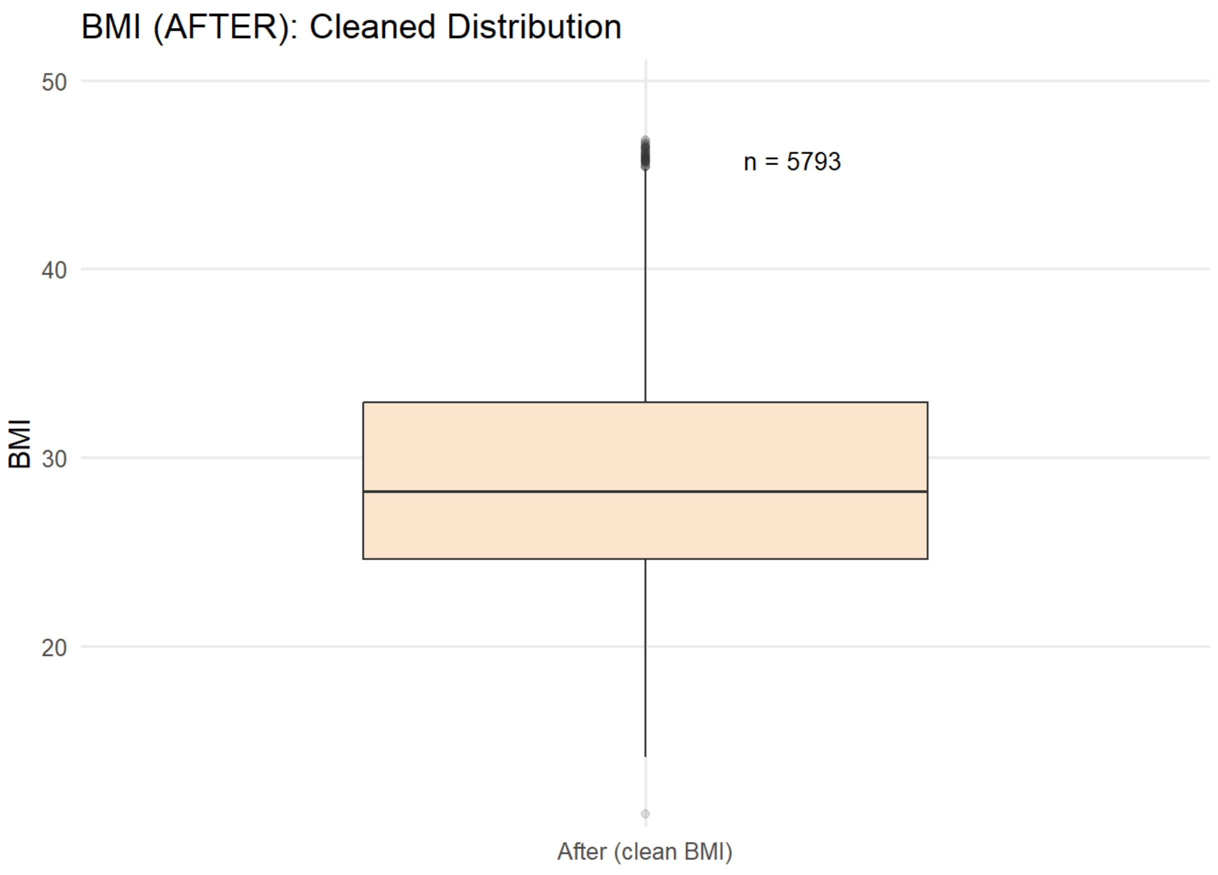
```
# ---- BMI boxplot (AFTER) ----
bmi_after_df <- dat_clean %>% transmute(stage = "After (clean BMI)", value = bmxbmi_clean)
x_bmi <- bmi_after_df$value
qs_bmi <- quantile(x_bmi, c(.25,.75), na.rm = TRUE);
iqr_bmi <- qs_bmi[2]-qs_bmi[1]
upper_whisker_bmi <- min(max(x_bmi, na.rm = TRUE), qs_bmi[2] + 1.5*iqr_bmi)
bmi_after_label_y <- upper_whisker_bmi + 0.05*iqr_bmi
bmi_after_N <- sum(!is.na(x_bmi))

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 7 x 5 in image
```

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

```
print(p_bmi_after)
```

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

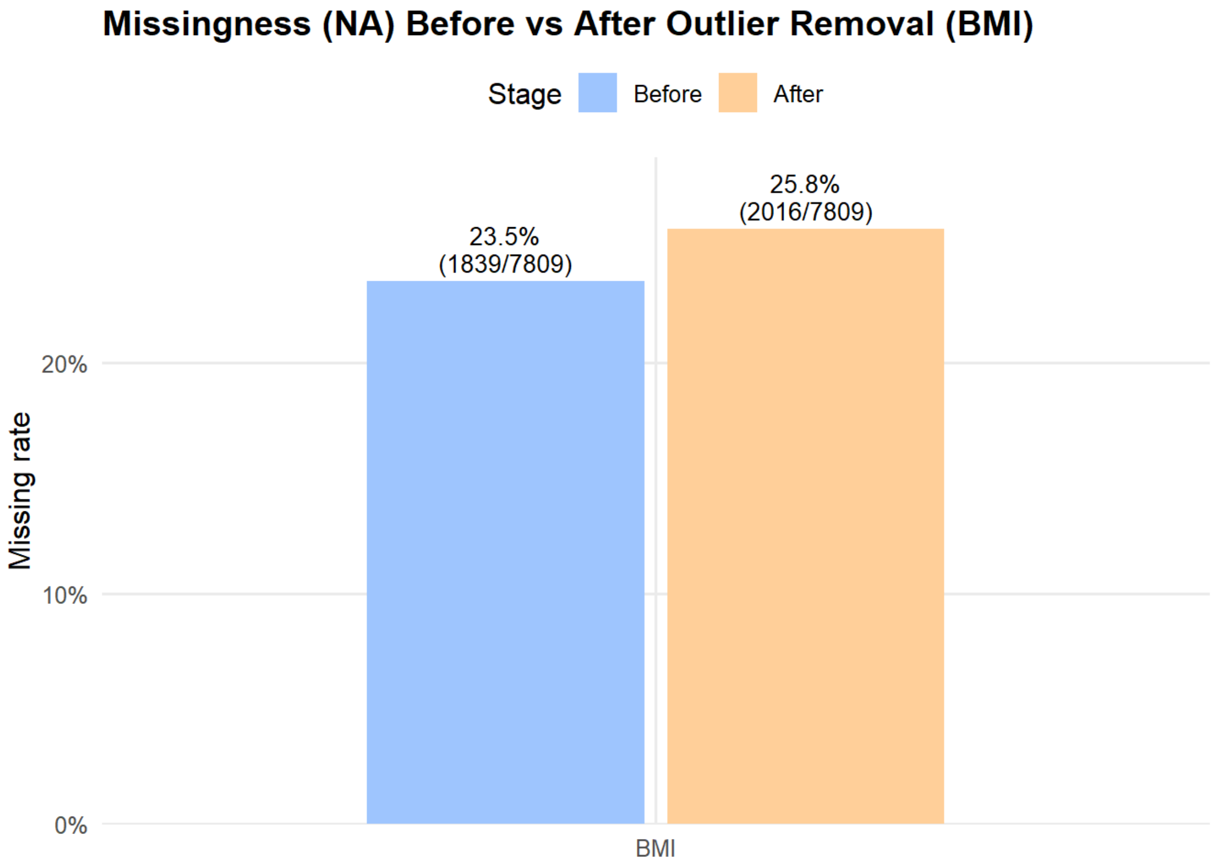


IX. Check the NA count change before and after cleaning (by bar plots)

```
miss_before_bmi <- 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_bmi <- 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)

miss_long_bmi <- bind_rows(miss_before_bmi, miss_after_bmi) %>%
  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_bmi, aes(variable, p_missing, fill = stage)) +
  geom_col(width=0.6, position="dodge") +                                                   # dodge to separ
ate bars
  geom_text(aes(label = paste0(scales::percent(p_missing, 0.1),
                               "\n(", n_missing, "/", n_total, ")")),                      # label on top o
f bars
            vjust=-0.2, size=3.5) +
  scale_y_continuous(labels=scales::percent) +
  labs(title = "SBP Missingness Before vs After Cleaning", x=NULL, y="Missing rate") +
  theme_minimal(base_size=12) + theme(legend.position="top")

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

p_na_bar_2 <- ggplot(miss_long_bmi, 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 7 x 5 in image
```

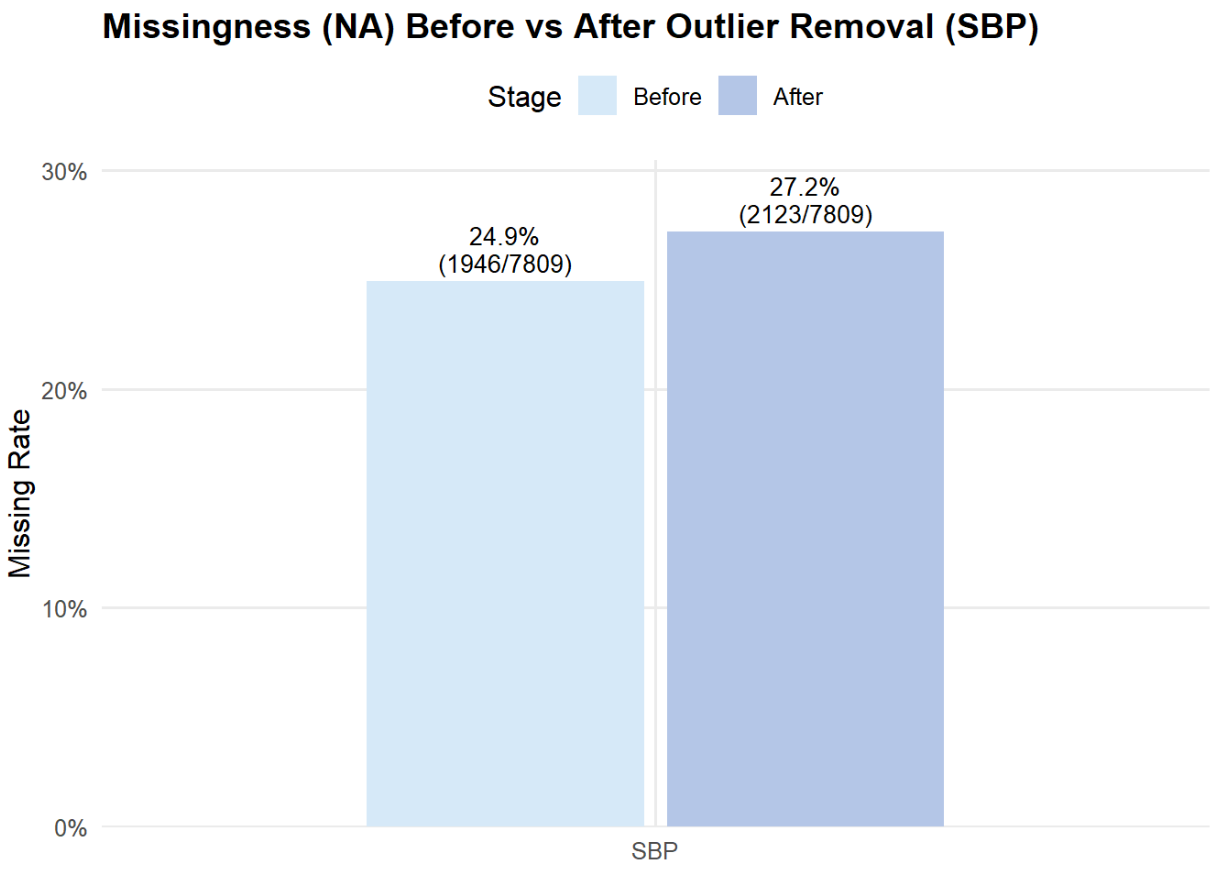
```
print(p_na_bar_2)
```



```
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_raw_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"))

pos <- position_dodge(width = 0.65)
p_na_bar_sbp <- ggplot(miss_long_sbp, 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" = "#D6E9F8", "After" = "#B4C6E7")) +
  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 7 x 5 in image
```

```
print(p_na_bar_sbp)
```



X. Scatter plot of cleaned BMI vs cleaned SBP by sex

```
p_scatter_bmi_sbp <- ggplot(dat_clean, aes(x = bmxbmi_clean, y = sbp_raw_clean, color = sex)) +

  # 1. 繪製散點圖 (geom_point)，設置 alpha 讓點重疊時能看出密度
  geom_point(alpha = 0.3, size = 1.5) +

  # 2. 添加性別分組的平滑/趨勢線 (geom_smooth)
  #    method="lm" 表示使用線性模型 (Linear Model) 來擬合趨勢
  geom_smooth(method = "lm", se = TRUE, linewidth = 1.2) +

  # 3. 設置圖表標籤和標題
  labs(
    title = "Association between Cleaned BMI and SBP by Sex",
    x = "Body Mass Index (BMI)",
    y = "Mean Systolic Blood Pressure (SBP, mmHg)",
    color = "Sex" # 圖例標題
  ) +

  scale_color_manual(values = c("Male" = "#0072B2", "Female" = "#D55E00")) + # 使用顏色友好的配色
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(face = "bold", hjust = 0.5), # 標題置中
    legend.position = "bottom",
    panel.grid.minor = element_blank()
  )

ggsave("outputs/q1_scatter_bmi_sbp_by_sex.png", p_scatter_bmi_sbp, bg = "white")
```

```
## Saving 7 x 5 in image
## `geom_smooth()` using formula = 'y ~ x'
```

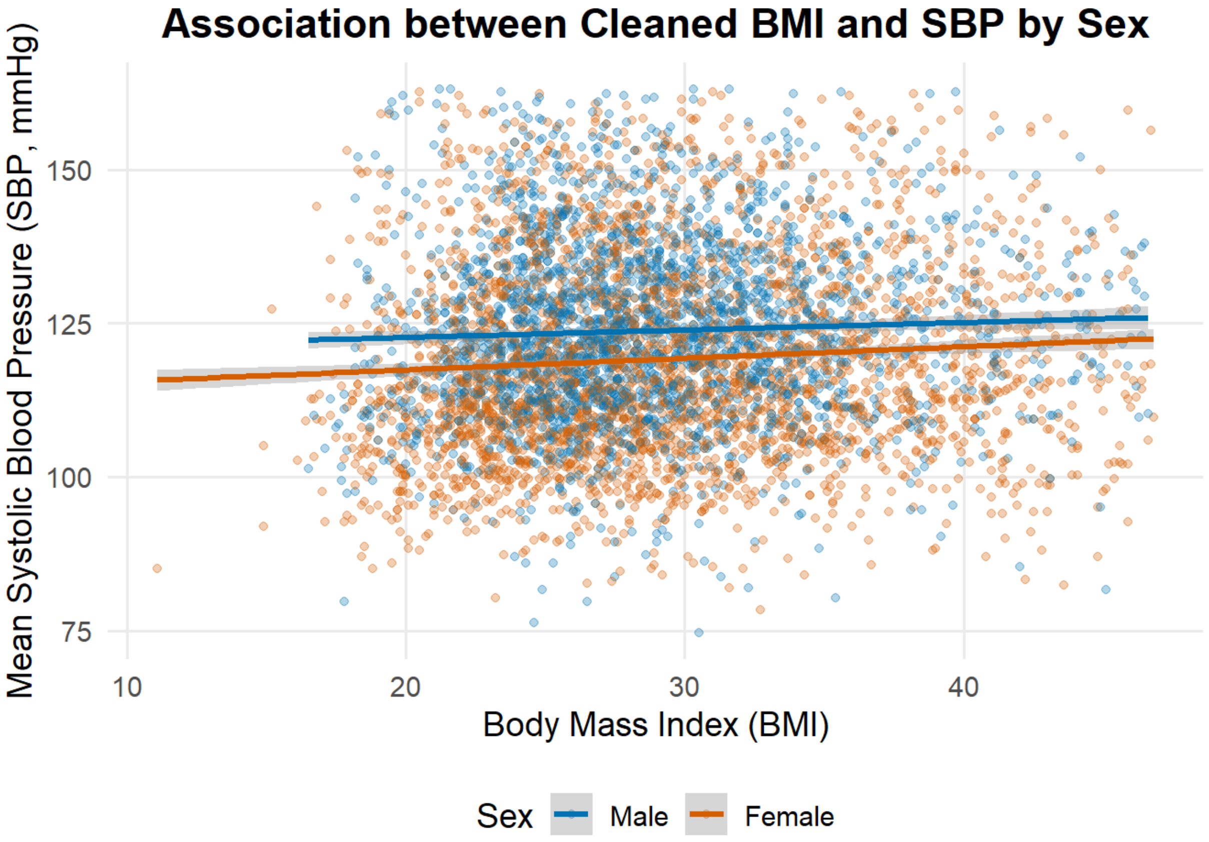
```
## Warning: Removed 2331 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 2331 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```
print(p_scatter_bmi_sbp)
```

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

```
## Warning: Removed 2331 rows containing non-finite outside the scale range
## (`stat_smooth()`).
## Removed 2331 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



The scatter plot clearly reveals a significant positive association between BMI and mean SBP (Systolic Blood Pressure). This indicates that as a subject's BMI increases, their mean SBP also tends to rise.

Overall Trend: Both trend lines exhibit an upward slope, confirming the positive correlation between BMI and mean SBP regardless of sex.

Sex Differences: Although both trend lines show a positive correlation, the SBP trend line for males (blue) generally lies above the SBP trend line for females (red) across most BMI ranges. This suggests that, at the same BMI level, males tend to have a higher SBP.

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

Q1. 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?

```
# 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
```

```r
# 2) Recode & relabel
dat_edu <- demo %>%
  transmute(
    seqn,
    age = ridageyr,
    EDU = case_when(                  # case_when() is like ifelse() but for multiple conditions
      dmdeduc2 %in% 1:5 ~ dmdeduc2,   # retain 1-5
      TRUE ~ NA_real_                 # 7/9 -> NA
    )
  ) %>%
  mutate(
    EDU = factor(EDU,                 # mutate() adds new variables or transforms existing ones
                 levels = 1:5,
                 labels = c("<9th grade", "9-11th grade", "High school/GED",
                            "Some college/AA", "College or above"))
  ) %>%
  left_join(dat_clean %>% select(seqn, bmxbmi_clean), by = "seqn") %>%
  drop_na(EDU, bmxbmi_clean)



dat_race <- demo %>%
  transmute(
    seqn,
    age = ridageyr,
    race = case_when(
      ridreth3 %in% 1:7 ~ ridreth3, # 保留 1-7 的有效編碼 （包括跳過的 5）
      TRUE ~ NA_real_              # 將其他編碼 （如缺失值 9） 設為 NA
    )
  ) %>%
  mutate(
    race = factor(race,
                  levels = 1:7,
                  labels = c("Mexican American",
                             "Other Hispanic",
                             "Non-Hispanic White",
                             "Non-Hispanic Black",
                             "UNUSED",                          # <<< 此處是編碼 5 的位置 ( unused )
                             "Non-Hispanic Asian",
                             "Other Race / Multi-Racial"
                             )
                  )
  ) %>%
  left_join(dat_clean %>% select(seqn, bmxbmi_clean), by = "seqn") %>%
  drop_na(race, bmxbmi_clean)



# 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 consistency

race_dist <- dat_race %>%
  count(race) %>%                 # count occurrences of each education level
  mutate(prop = n / sum(n),      # calculate proportions
         variable = "race") %>%   # add a variable column for clarity
  rename(category = race)         # rename race to category for consistency
```

```
# 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) #row.names=FALSE to avoid writing row numbers



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

Distribution of Educational Attainment (EDU)

category	n	prop	variable
<9th grade	278	0.048	EDU
9–11th grade	457	0.079	EDU
High school/GED	1227	0.212	EDU
Some college/AA	1749	0.302	EDU
College or above	2079	0.359	EDU

```
kable(race_dist, digits = 3, caption = "Distribution of race Attainment")
```

Distribution of race Attainment

category	n	prop	variable
Mexican American	390	0.067	race
Other Hispanic	593	0.102	race
Non-Hispanic White	3427	0.592	race
Non-Hispanic Black	689	0.119	race
Non-Hispanic Asian	330	0.057	race
Other Race / Multi-Racial	364	0.063	race

II. Please use boxplots to visualize the BMI distribution in different races and education levels.

```
p_bmi <- dat_edu %>%
  ggplot(aes(x = EDU, y = bmxbmi_clean)) +                               # aes() defines the aesthetic mapping: x-axis is EDU, y-axis is cleaned BMI
  geom_boxplot(position = position_dodge(0.8), outlier.alpha = 0.2) +    # position_dodge(0.8) separates boxplots for clarity; outlier.alpha adjusts outlier visibility
  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_1.png", p_bmi, width = 10, height = 6, bg = "white")


p_race <- dat_race %>%
  ggplot(aes(x = race, y = bmxbmi_clean)) +                               # aes() defines the aesthetic mapping: x-axis is race, y-axis is cleaned BMI
  geom_boxplot(position = position_dodge(0.8), outlier.alpha = 0.2) +    # position_dodge(0.8) separates boxplots for clarity; outlier.alpha adjusts outlier visibility
  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_1.png", p_race, width = 10, height = 6, bg = "white")
```

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

Educational Attainment (EDU)： The boxplots show a very wide distribution (large spread) of BMI values within almost all educational attainment groups. While the medians of most groups are quite similar, the College or above group's BMI median is noticeably, albeit slightly, lower than the others. However, a clear, overall correlation between educational attainment and the median BMI is not apparent in this visualization.

Race(race)： The BMI distributions are extremely wide for all race groups except for the Non-Hispanic Asian group. The median BMI for Non-Hispanic Asian individuals (<25) is distinctly lower than all other groups. The medians for the remaining groups are closely clustered, typically around 30. Similar to the education findings, a clear overall correlation between race/ethnicity and the median BMI among these higher-median groups is not strongly evident.

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:

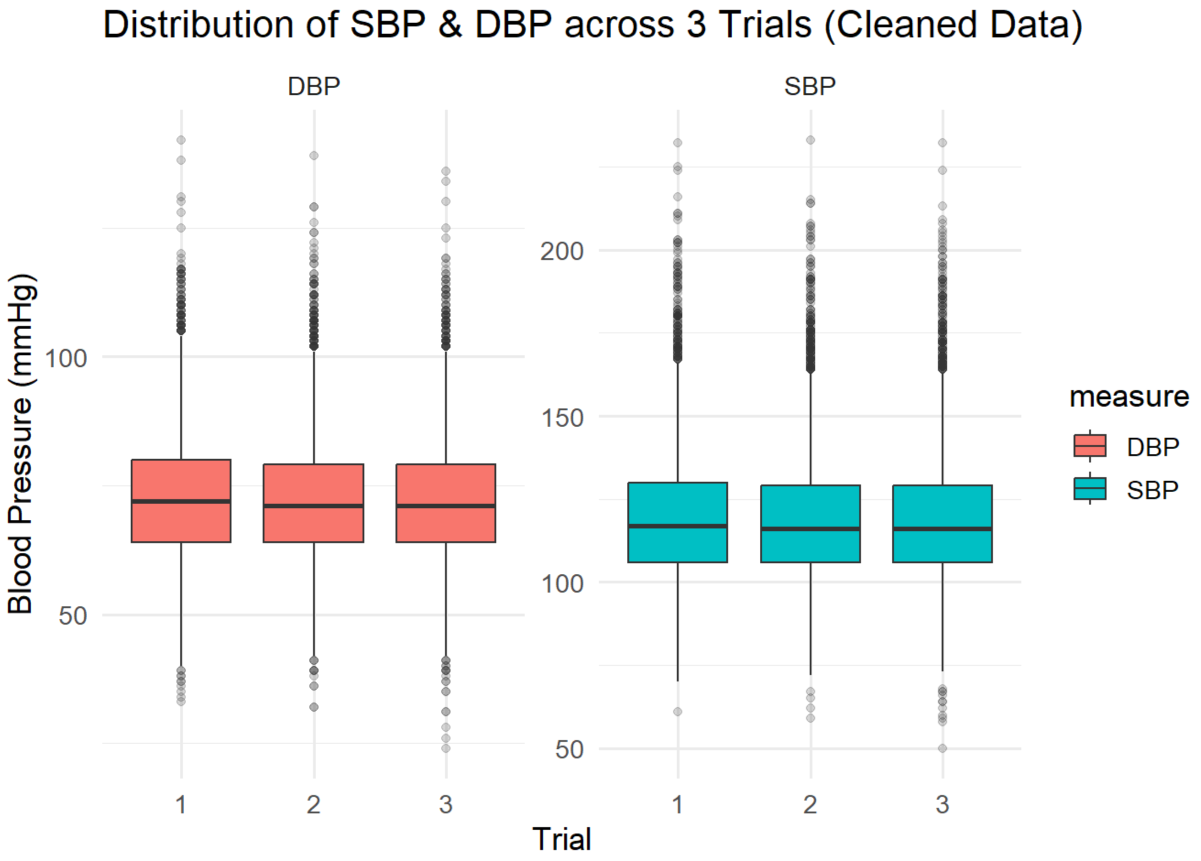
a. seqn: Participant ID
b. measure (new defined): Measurement type (SBP or DBP)
c. trial (new defined): Trial number (1, 2, or 3)
d. value (from each BP value): The recorded blood pressure value

```
bpx_long_clean <- bpx %>%
  select(seqn, all_of(c(sbp_cols, dbp_cols))) %>%
  # From the dataset bpx, you’re selecting: seqn: the participant ID. sbp_cols and dbp_cols: two vectors containing SBP and DBP measurement variable names
  # all_of() ensures all the columns listed in those vectors exist — otherwise 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” → diastolic; “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 stored in a new column named value.
  ) %>%
  mutate(
    measure = recode(measure,
                     "sy" = "SBP",
                     "di" = "DBP"),
    trial = as.integer(trial)
  )
```

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.

```
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)
```

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

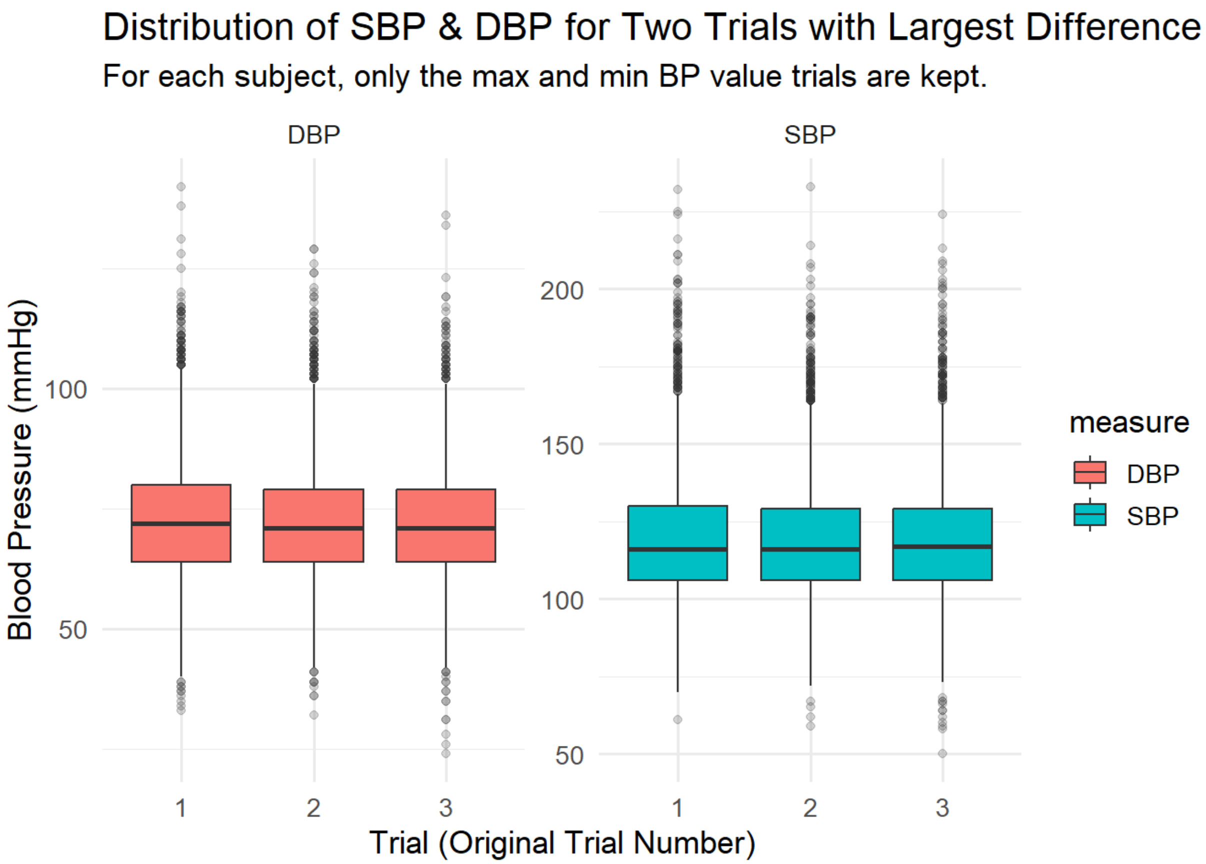


III. Now, suppose we are only interested in the two trials that show the largest difference for each subject. Please complete the tasks aboved.

```
# 1） 找出每個受試者/測量類型中，最大值和最小值所在的試驗 （即差異最大的兩次）
bpx_two_trials_only <- bpx_long_clean %>%
  # 按 seqn （受試者） 和 measure (SBP/DBP) 分組
  group_by(seqn, measure) %>%
  # 找出最大值和最小值的血壓值
  mutate(
    max_value = max(value, na.rm = TRUE),
    min_value = min(value, na.rm = TRUE)
  ) %>%
  filter(value == max_value | value == min_value) %>%
  distinct(value, .keep_all = TRUE) %>%
  slice(1:2) %>%
  select(-max_value, -min_value) %>%
  ungroup()
```

```
## Warning: There were 1132 warnings in `mutate()`.
## The first warning was:
## ℹ In argument: `max_value = max(value, na.rm = TRUE)`.
## ℹ In group 37: `seqn = 130401` and `measure = "DBP"`.
## Caused by warning in `max()`:
## ! no non-missing arguments to max; returning -Inf
## ℹ Run `dplyr::last_dplyr_warnings()` to see the 1131 remaining warnings.
```

```
# 2) 重新繪製 Boxplot
p_two_trials <- ggplot(bpx_two_trials_only, 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 for Two Trials with Largest Difference",
    subtitle = "For each subject, only the max and min BP value trials are kept.",
    x = "Trial (Original Trial Number)", y = "Blood Pressure (mmHg)"
  ) +
  theme_minimal(base_size = 13)

# 3) 儲存圖表
ggsave("outputs/BMI_by_BPX_Largest_Diff_Two_Trials.png", p_two_trials, width = 10, height = 6, bg = "white")
print(p_two_trials)
```



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

The blood pressure values were most likely measured consecutively on the same day to minimize error. In both the SBP and DBP, the median for Trial 1 is slightly higher than the medians for Trial 2 and Trial 3. The median values drop slightly from Trial 1 to Trial 2, and then remain consistent in Trial 3.

This pattern is characteristic of the “white-coat” effect or the habituation phenomenon, where a person’s first reading is elevated due to anxiety or a lack of habituation to the measurement process. Subsequent readings (Trial 2 and Trial 3) are typically lower and more representative of the person’s true resting blood pressure. This short-interval, repeated measurement method is a standard clinical practice to improve measurement accuracy.

4. Conclusion

BMI and SBP Association:

A significant positive correlation exists between BMI and mean SBP. Furthermore, this association varies by sex: the SBP trend line for males is generally higher than that for females across the BMI range.

Sociodemographic Disparities:

Race shows the most pronounced BMI difference, with Non-Hispanic Asian individuals having a distinctly lower median BMI (<25) compared to other groups, whose medians cluster around 30.

Educational Attainment showed a less distinct pattern, though the College or above group did exhibit a slightly lower median BMI compared to other levels.

Blood Pressure Measurement:

The distribution of repeated blood pressure trials (Trial 1 > Trial 2 ≈ Trial 3) strongly suggests that measurements were taken consecutively on the same day at short intervals. This methodology is essential for minimizing the white-coat effect and ensuring the accuracy of the final reported blood pressure value.

Learned about Reproducible Workflows

1. Transparency: Every step—from outlier removal (using combined physiologic, IQR, and MAD rules) to data reshaping (pivot_longer)—is explicitly coded, making the entire analytical process auditable and verifiable.
2. Efficiency: The pipe operator (%>%) allowed for the construction of clean, sequential, and highly readable code chunks, which is crucial for managing and updating complex scripts based on large survey datasets like NHANES.
3. Data Integrity: Defining and executing a multi-layered cleaning process ensured that the final conclusions were based on high-quality data, minimizing bias introduced by extreme outliers.

https://github.com/CYC14/Big-Data-HW.git (https://github.com/CYC14/Big-Data-HW.git)