Week 3 Assessment: Data Visualization and Exploration with ggplot2

Merhawi Kidane

2025-09-20

Exercise 1: Reproducing and Arranging ggplot2 Figures

In this exercise, we will work with a cleaned version of the NHANES dataset (clean_NHANES.csv). The goal is to recreate two figures using ggplot2, using the meaningful column names and category labels in the cleaned dataset.

These figures will demonstrate basic data visualization techniques using ggplot2, including mapping variables to aesthetics, grouping, and arranging multiple plots.

Plot 1:

- A histogram of age (in years).
- The histogram will be grouped by gender, so that male and female participants are distinguishable.
- Each bin will cover 5 years to match the original figure.

```r
f1 <- clean_NHANES %>%
  ggplot(aes(x = age, fill = gender)) +
  geom_histogram(
    binwidth = 5,
    position = "dodge",
    color = "black",
    boundary= 0) +
  labs(
    title = "Histogram of Age by Gender",
    x = "Age (Years)",
    y = "Number of Participants",
    fill = "Gender"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0, face = "bold")
  )
```

Plot 2:

- A bar graph of ethnicity.
- The bars will be grouped by gender to show the distribution of males and females across ethnic groups.

```
f2 <- clean_NHANES %>%
  ggplot(aes(reorder(ethnicity_2, ethnicity_2, FUN = length),     # Order by number of participants
             fill= gender)) +
  geom_bar(position = "stack", color="black") +
  labs(
    title= "Bar graph of Ethnicity by Gender",
    x= " Ethnicity",
    y= "Number of Participants",
    fill= "Gender",
  ) +
   theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0, face = "bold"),
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
```

Combine plots with cowplot (using the function plot_grid())

```
combined_cowpl <- plot_grid(
  f1, f2,                        # two plots
  labels = c("A", "B"),          # figure labels
  ncol = 2                     # two columns side by side
)

combined_cowpl
```

Combine plots with ggpubr (using the function ggarrange())

```
combined_ggarrange <- ggarrange(
  f1, f2,
  labels = c("A", "B"),     # same labels
  ncol = 2,                 # 2 columns
  common.legend = TRUE,     # merge legends
  legend = "bottom"         # move legend below
)

combined_ggarrange
```

Comparison of both methods

Both plot_grid() (cowplot) and ggarrange() (ggpubr) can combine multiple plots into rows and columns. The main difference is that ggarrange() provides built-in options for sharing legends and aligning axes across plots, while plot_grid() is more minimal but produces very clean, publication-ready layouts with precise control over labels and alignment. In this case, even with only two plots, the appearance of ggarrange() is slightly neater and more balanced,because the shared legend and automatic spacing help the plots align evenly and look cohesive.

Exercise 2 — Visualizing key characteristics (Age, Gender, Ethnicity)

In this section we create and save exploratory plots for **Age**, **Gender**, and both **ethnicity** variables from `clean_NHANES`. For each variable we will:

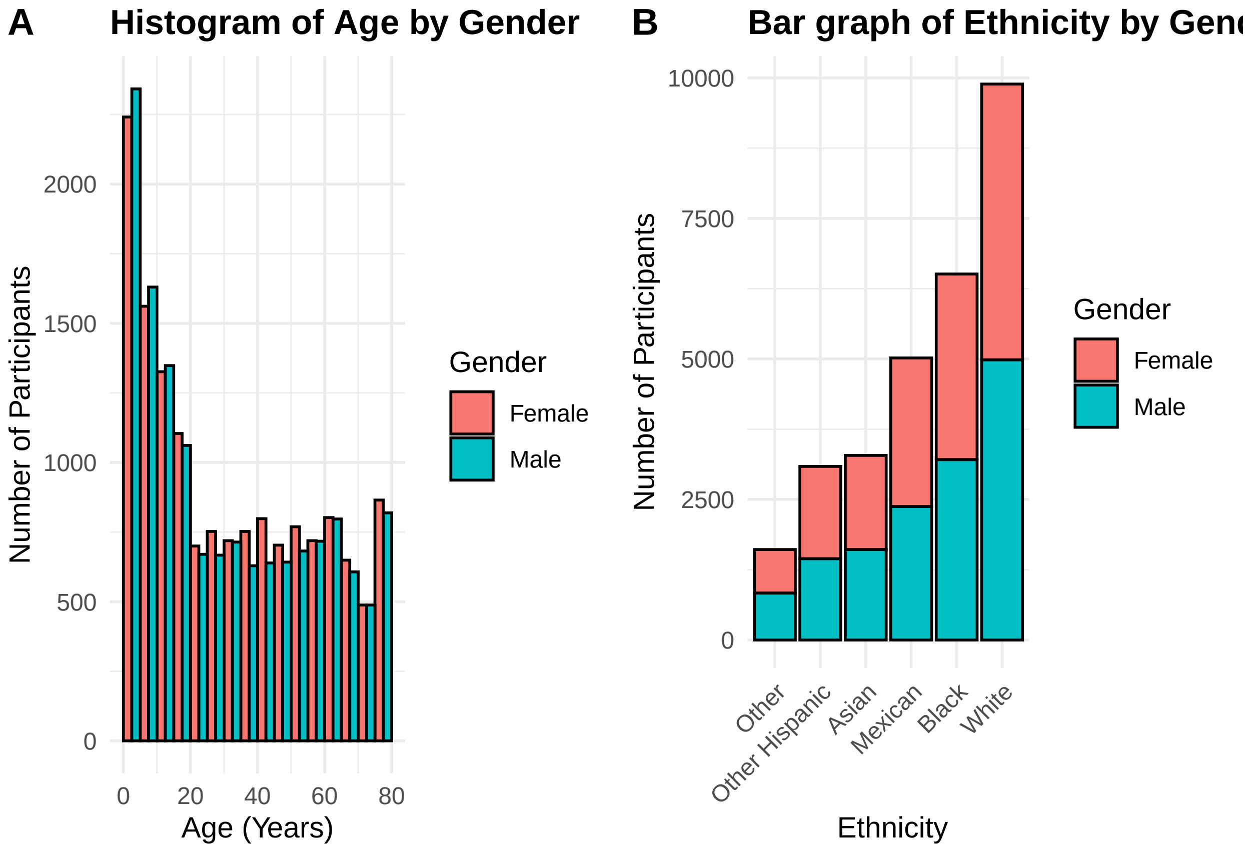


Figure 1: Combined display of Plot1 and Plot2, arranged into a single panel for comparison using plot grid.

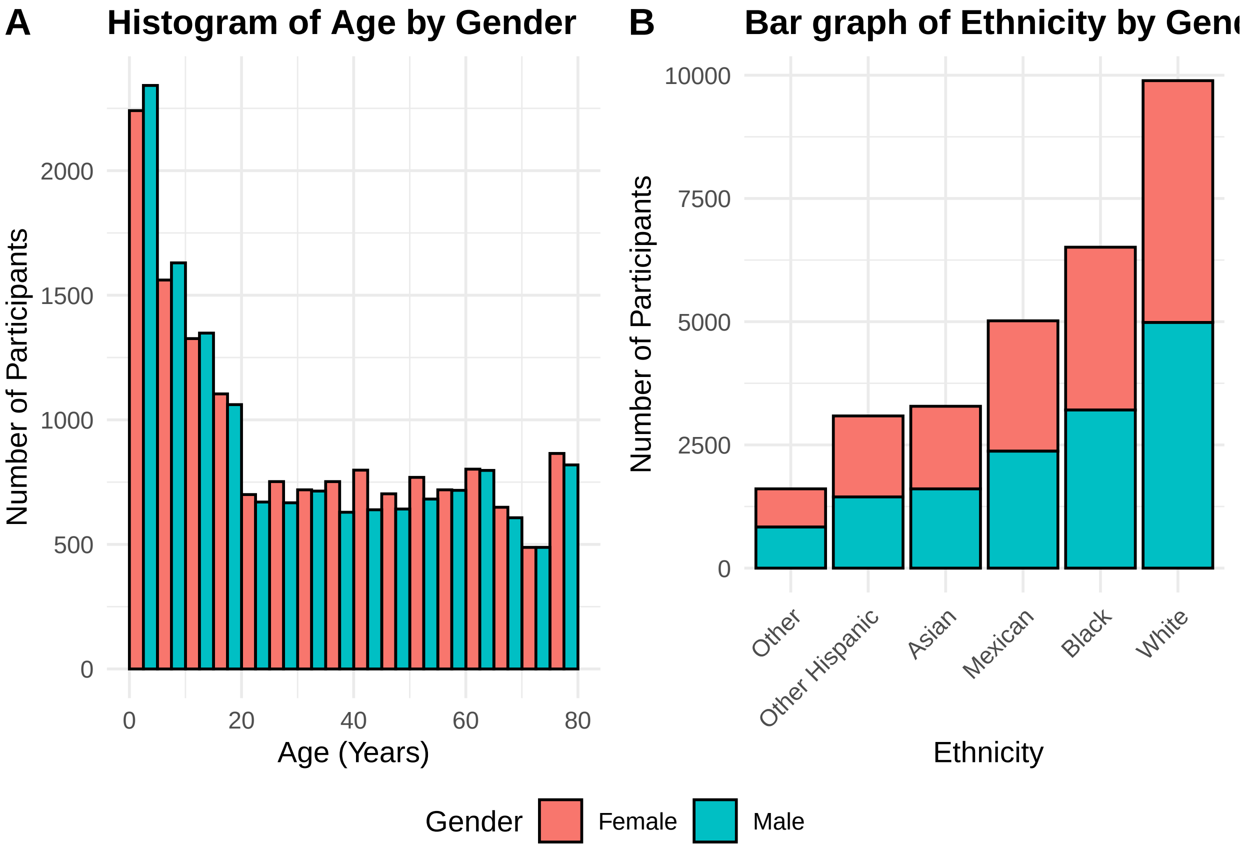


Figure 2: Combined display of Plot1 and Plot2, arranged into a single panel for comparison using ggarrange.

- Produce a clear plot (title, axis labels, legend where appropriate),
- Write a description of its distribution and the count of missing values,
- Save the plot with `ggsave()` for later use.

Age — histogram

```r
# Histogram of Age
age_histo <- ggplot(clean_NHANES, aes(x = age)) +
  geom_histogram(
    binwidth = 5,
    boundary = 0,            # so bins start at 0
    color = "black",
    fill = "lightblue"
  ) +
  labs(
    title = "Distribution of Age",
    x = "Age (years)",
    y = "Number of Participants"
  ) +
  theme_minimal()
# view plot
age_histo
```

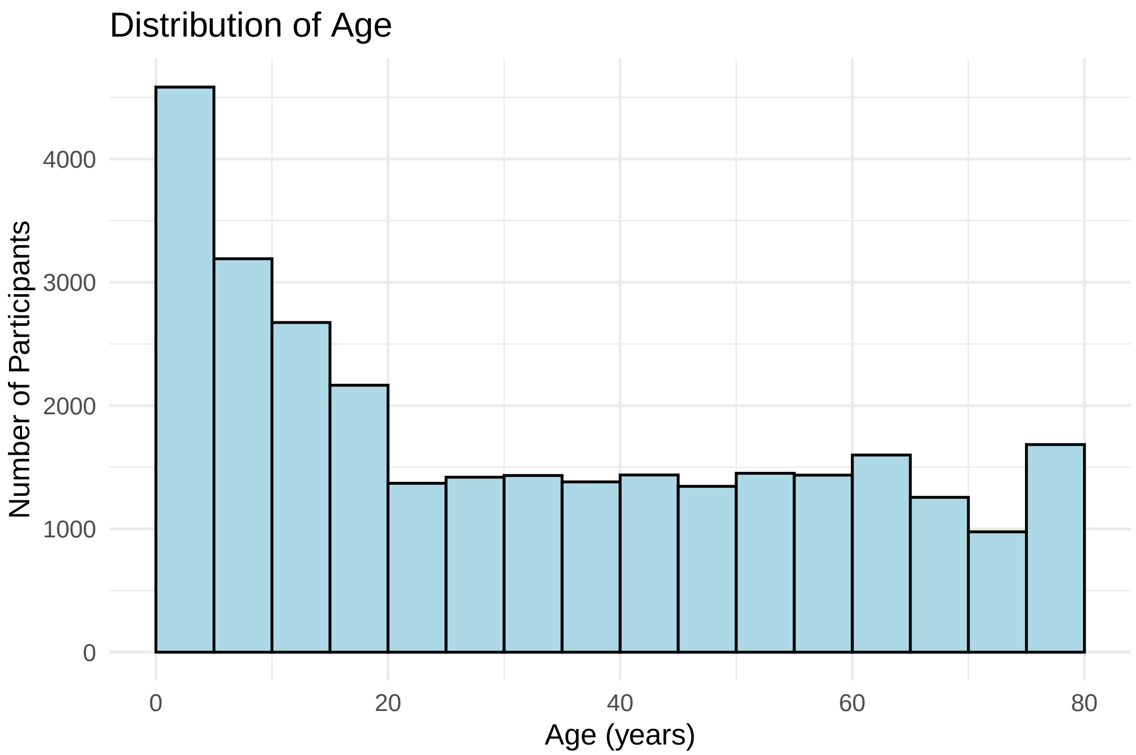


Figure 3: Age Histogram

```
# Check for missing values in Age
sum(is.na(clean_NHANES$age))
```

```
## [1] 0
```

```
# Save the plot for later reuse
ggsave(
  filename = "age\\_distribution.png",
  plot = age_histo,
  path = here("week_3"),  # my folder
  width =6, height =4
)
```

Age: Brief description of distribution and missing values

The histogram shows the distribution of participants' ages. The largest group is at the youngest ages (0–5 years), with frequencies decreasing toward adolescence and then evening out across adulthood up to about 80 years. The shape indicates a large number of children and a relatively even distribution of adults across age ranges in this sample. There are 0 missing values for age.

Gender — bar graph

```
#Gender Bar graph
gender_bar <- ggplot(clean_NHANES, aes( x=gender, fill= gender)) +
  geom_bar(color= "black") +
  labs(
    title = "Distribution of Gender",
    x= "Gender",
    y= "Number of Participants",
    fill= "Gneder"
  ) +
  theme_minimal()

# View plot
gender_bar
```

```
# Check for missing values
sum(is.na(clean_NHANES$gender))
```

```
## [1] 0
```

```
# Save the plot for later reuse
ggsave(
  filename = "gender distribution.png",
  plot = gender_bar,
  path= here("week_3"),
  width=6, height=4
)
```

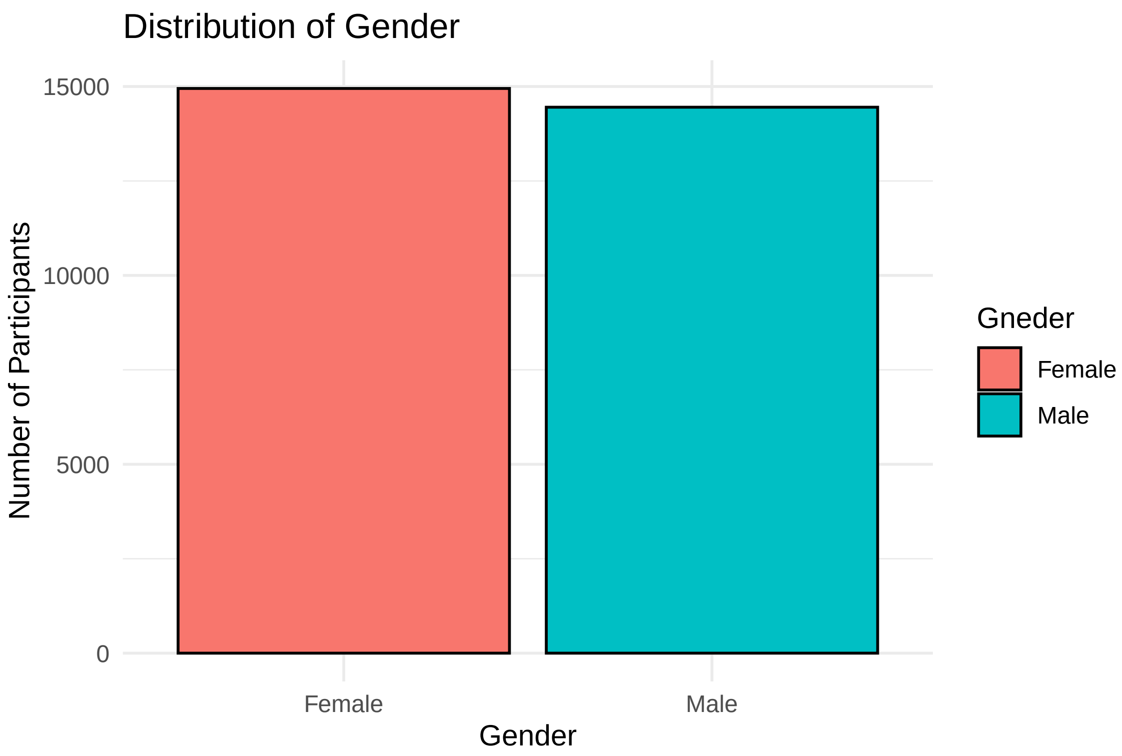


Figure 4: Gender Bar graph

Gender: Brief description of distribution and missing values

The bar plot shows the distribution of gender among participants. Females slightly outnumber males (around 15,000 vs. 14,000 participants), indicating a near-even gender split in the data-set. There are 0 missing values for gender.

Ethnicity_1 - Bar graph

```
ethnicity1_bar <- ggplot(
  clean_NHANES,
  aes(x= reorder(ethnicity_1, ethnicity_1, FUN= length),
      # Order by length of bars for better visibility.
  fill= reorder(ethnicity_1, ethnicity_1, FUN=length))
  # Order legend to match the order of bars.
  ) +
  geom_bar(color= "black") +
  labs(
    title = "Distribution of ethnicity1",
    x= "Ethnicity1",
    y= "Number of Participants",
    fill= "Ethnicity1"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )

# View plot
ethnicity1_bar
```

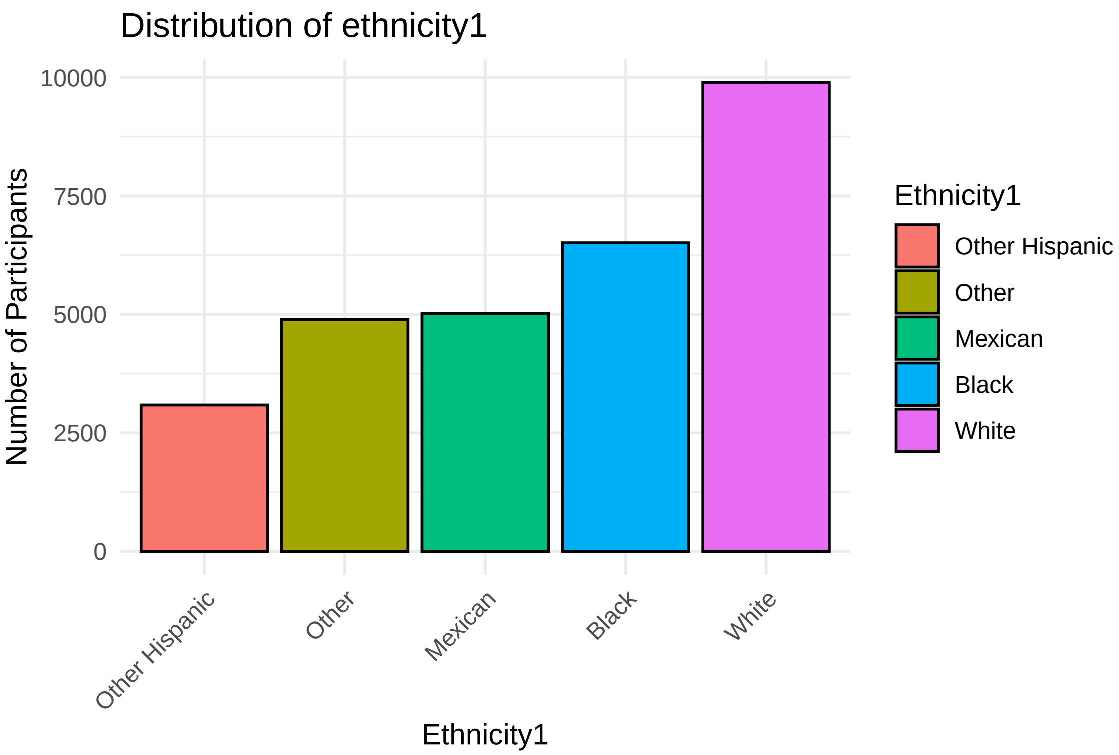


Figure 5: Ethnicity_1 Bar graph

```
# Check for missing values
sum(is.na(clean_NHANES$ethnicity_1))
```

```
## [1] 0
```

```
# Save for later reuse
ggsave(
  filename = "ethnicity1_distribution.png",
  plot = ethnicity1_bar,
  path = here("week_3"),
  width = 6, height = 4
)
```

Ethnicity_1: Brief description of distribution and missing values

The bar graph displays the number of participants from five different ethnic groups in this study. The White group has the largest number of participants, with just under 10,000. The Black group has approximately 6,500 participants. Both the Mexican and Other groups have similar representation, each with about 5,000 participants. The group with the fewest participants is Other Hispanic, with approximately 3,000. There are 0 missing values.

Ethnicity_2 — Bar graph

```
ethnicity2_bar <- ggplot(
  clean_NHANES,
  aes(reorder(ethnicity_2, ethnicity_2, FUN= length),   # Order by length of bars
  fill= reorder(ethnicity_2, ethnicity_2, FUN= length)) # Order legend to match the order of bars.
  ) +
  geom_bar(color= "black") +
  labs(title = "Distribution of ethnicity2",
       x= "Ethnicity2",
       y= "Number of Participants",
       fill= "Ethnicity2"
       ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust= 1)
  )

# View plot
ethnicity2_bar
```

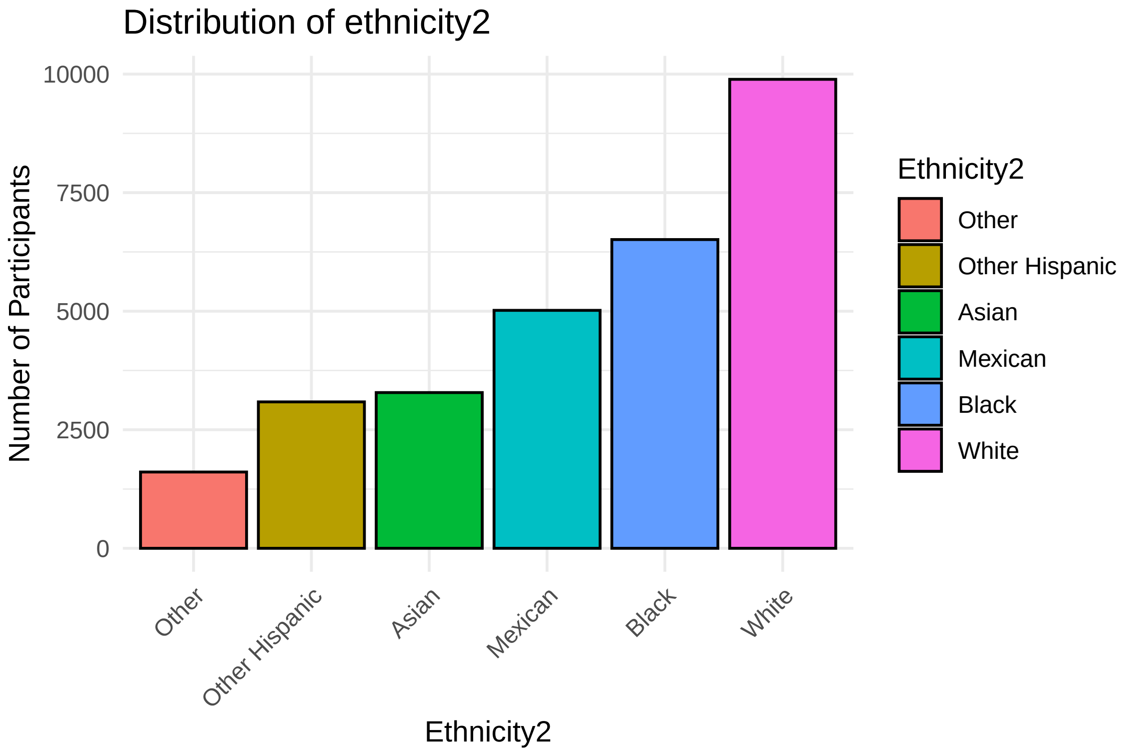


Figure 6: Ethnicity_2 Bar graph

```
# Check for missing values
sum(is.na(clean_NHANES$ethnicity_2))
```

```
## [1] 0
```

```r
# Save for later reuse
ggsave(
  filename = "ethnicity2_distribution.png",
  plot = ethnicity2_bar,
  path = here("week_3"),
  width = 6, height = 4
)
```

Ethnicity_1: Brief description of distribution and missing values

The bar graph presents the distribution of participants across six ethnic groups in a study. The largest group is White, with a participation count just under 10,000. The Black group follows, with approximately 6,500 participants. The Mexican group has a count of around 5,000. The Asian and Other Hispanic groups have similar representation, each with approximately 3,000 participants. The smallest group in the study is Other, with a count of around 1,800.

Comparative Description of bar charts of Ethnicity_1 and Ethnicity_2

The first bar graph displays participant counts across five ethnic groups, with White being the largest group (nearly 10,000 participants) and Other Hispanic being the smallest (around 3,000). The Other group is relatively large, with about 5,000 participants.

The second bar graph includes an additional Asian group. It shows a more granular breakdown, with the White, Black, and Mexican groups retaining the same counts as the first graph. The newly added Asian group and the Other Hispanic group each have around 3,000 participants, while the Other group has significantly decreased to about 1,800.

Better Variable Choice and Justification

The second variable is the better choice.

It is superior because it provides a more detailed and accurate representation of the study's participants by disaggregating the data. This allows for a more nuanced analysis of the Asian group, which was previously hidden within the broad and less informative "Other" category in the first graph. Using more specific categories aligns with modern research standards and enables a better understanding of potential differences and trends among diverse populations.

Remove varaible to be discarded

```r
newclean_NHANES <- clean_NHANES %>% select(-ethnicity_1)
# Check if the two data-sets are identical
identical(newclean_NHANES, clean_NHANES)
```

Exercise 3 - Improving ggplot figure

```r
# Import the new data-set of the diet study
diet_weight <- import(here("week_3", "diet.csv"))
```

Limitations of the Weight-vs-Week Plot and Proposed Improvements

In its current form the figure plots each participant's absolute weight over time. Although this shows the general trajectory of weight for each individual, it makes comparisons across participants difficult. Each person entered the study at a different baseline weight, so the lines are scattered across a wide y-axis range. As a result it is hard to see whether the diet produced a similar degree of weight loss for everyone, or to identify patterns such as the average change at a given week. In addition, plotting absolute values can visually emphasise heavier participants simply because their curves sit higher on the axis, not because they responded differently to the diet.

A more informative approach for this question is to express each person's weight relative to their own baseline (week 0). By creating a variable for "change from baseline" we normalise all participants to 0 at week 0. This allows direct visual comparison of the amount and trajectory of weight change, regardless of starting weight. Adding a summary line (such as the mean change across all participants) or using semi-transparent lines for individuals can further clarify the overall effect while still showing individual variation.

These changes are beneficial because they align the figure with the study's stated aim: to evaluate the benefit of the diet on reducing body weight "specifically in comparison to baseline weight at week 0." Expressing results as change from baseline improves interpretability, highlights differences in response over time, and matches how weight-loss interventions are typically reported in the literature.

```
# Create a new variable: change from baseline weight for each participant
diet_weight_change <- diet_weight %>%
  group_by(Participant) %>%
  mutate(
    Baseline = Weight[Week == 0],
    Weight_change = Weight - Baseline         # subtract baseline weight to find change
  ) %>%
  ungroup()

# Plot weekly weight change per participant with colors
ggplot(diet_weight_change, aes(
      x = Week,
      y = Weight_change,
      group = as.factor(Participant),         # group by participant
      color = as.factor(Participant)          # assign a different color
    )) +
  geom_line(linewidth = 1.2, alpha = 0.7) +           # individual participant lines
  #geom_point(size = 2, alpha = 0.4) +    # individual data points, if needed to check by removing #.
  scale_color_viridis_d(option = "turbo") + # visually distinct palette
  labs(
    title = "Weight Change Over Time by Participant",
    x = "Week",
    y = "Change from Baseline Weight (kg)",
    color = "Participant"                        # legend label
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "right"
  )
```

Exercise 4 - Exploring relationships through visualizations

```
# Step 1: Create a new column for average SBP
clean_NHANES <- clean_NHANES %>%
```

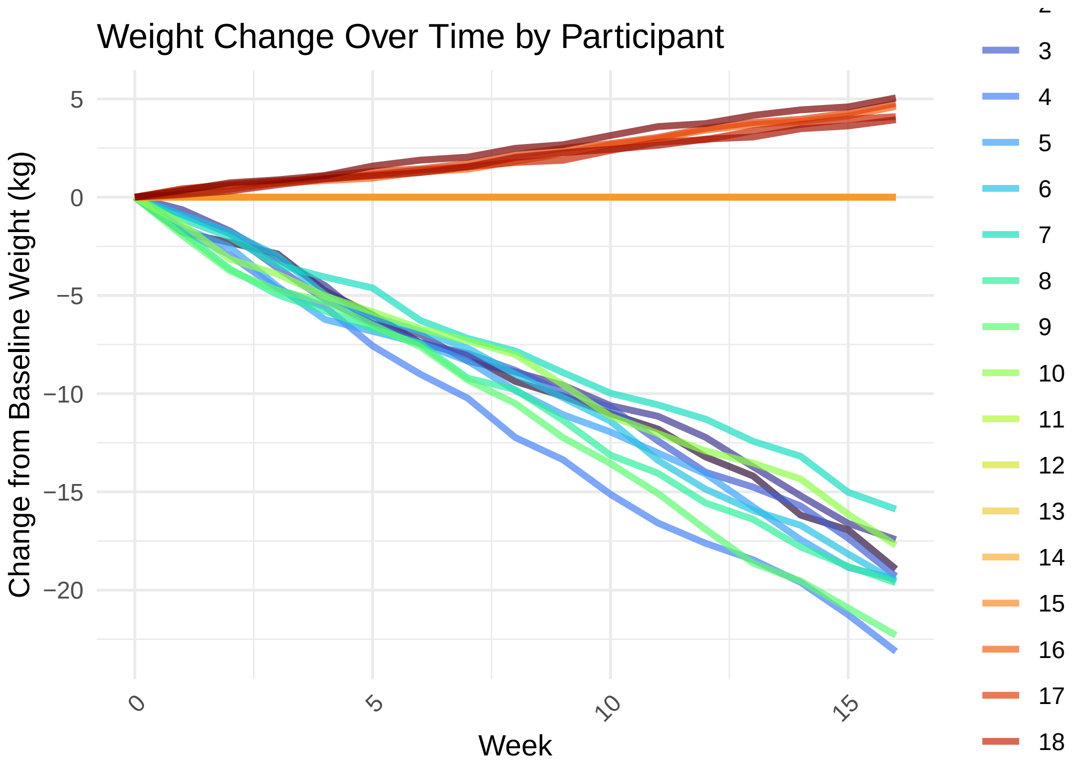


Figure 7: Weekly change in weight for each participant over the study period

```
  mutate(                       # compute the row-wise mean of the 4 SBP measurements
      avg_sbp = rowMeans(
      select(., systolic_bp_1, systolic_bp_2, systolic_bp_3, systolic_bp_4),
      na.rm = TRUE        # ignore NAs so mean is computed if at least one reading is present
    )
  )

# Step 2: Keep only individuals with non-missing Age, Gender, and Average SBP
nhanes_filtered <- clean_NHANES %>%
  filter(
    !is.na(age),        # keep rows where age is not missing
    !is.na(gender),     # keep rows where gender is not missing
    !is.na(avg_sbp)     # keep rows where avg_sbp is not missing
  )

# Step 3: Create and reorder hypertension categories in one mutate
nhanes_filtered <- nhanes_filtered %>%
  mutate(                            # Create new category
    hypertension_cat = factor(       # Reorder the hypertension categories in an ascending order
      case_when(
        avg_sbp < 120 ~ "Normal",
        avg_sbp >= 120 & avg_sbp <= 129 ~ "Elevated",
        avg_sbp >= 130 & avg_sbp <= 139 ~ "Stage 1 Hypertension",
        avg_sbp >= 140 ~ "Stage 2 Hypertension",
        TRUE ~ NA_character_
      ),
      levels = c("Normal", "Elevated", "Stage 1 Hypertension", "Stage 2 Hypertension")
    )
  )
```

Step-4: Descriptive plots

```
# Step 4a: Plot counts of individuals by hypertension category
ggplot(nhanes_filtered, aes(x = hypertension_cat, fill = hypertension_cat)) +
  geom_bar() +                                  # bar chart of counts
  scale_fill_brewer(palette = "Set2") +         # add color palette
  labs(                                         # Label
    title = "Distribution of Hypertension Categories in final sample",
    x = "Hypertension Category",
    y = "Number of Individuals",
    fill = "Category"
  ) +
  theme_minimal()+
 theme(
    axis.text.x = element_text(angle = 45, hjust = 1))      # clean minimal theme
```

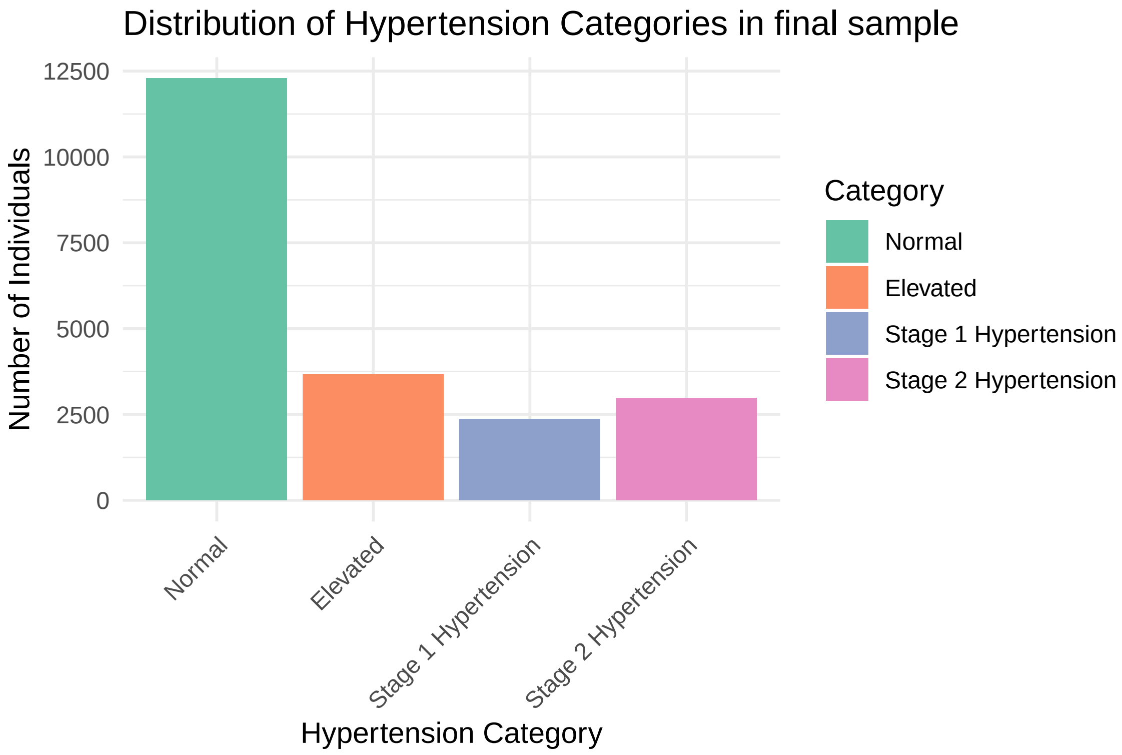


Figure 8: Counts of individuals by hypertension category

```
# Step 4b: Plot counts of individuals by gender
ggplot(nhanes_filtered, aes(x = gender, fill = gender)) +
  geom_bar() +                                  # bar chart of counts
  scale_fill_brewer(palette = "Pastel1") +      # soft color palette
  labs(
    title = "Gender Distribution in Final Sample",
    x = "Gender",
    y = "Number of Individuals",
    fill = "Gender"
  ) +
  theme_minimal()
```

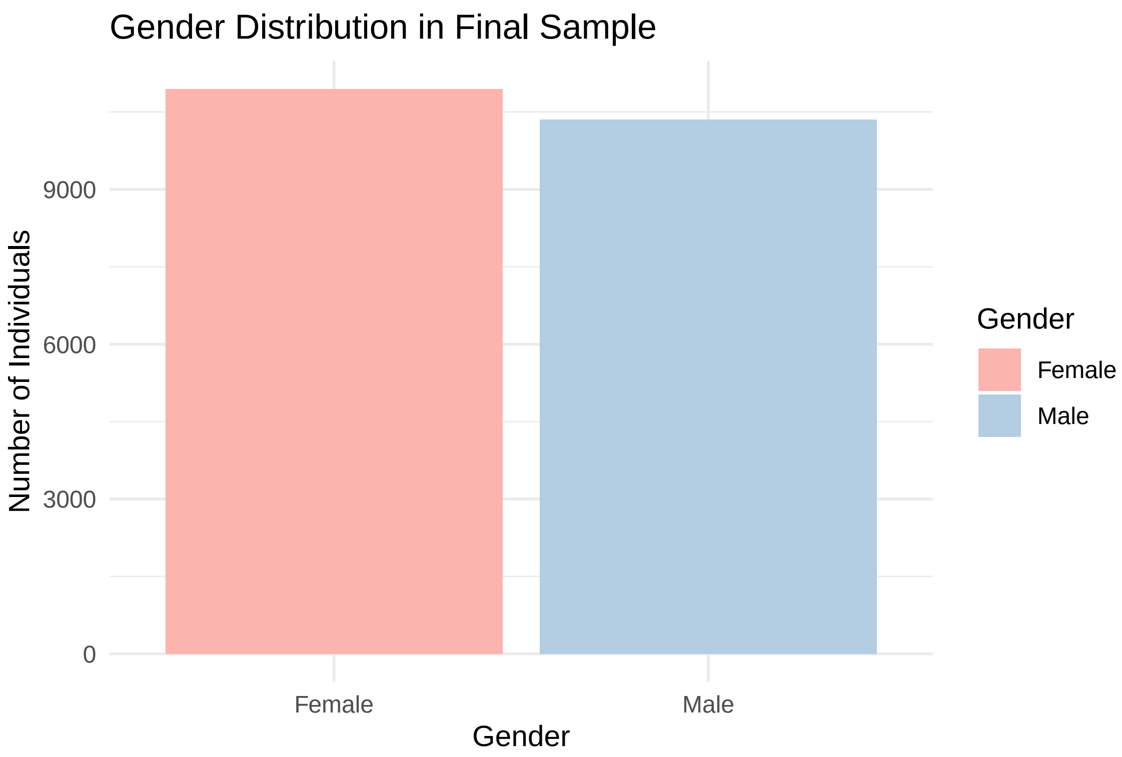
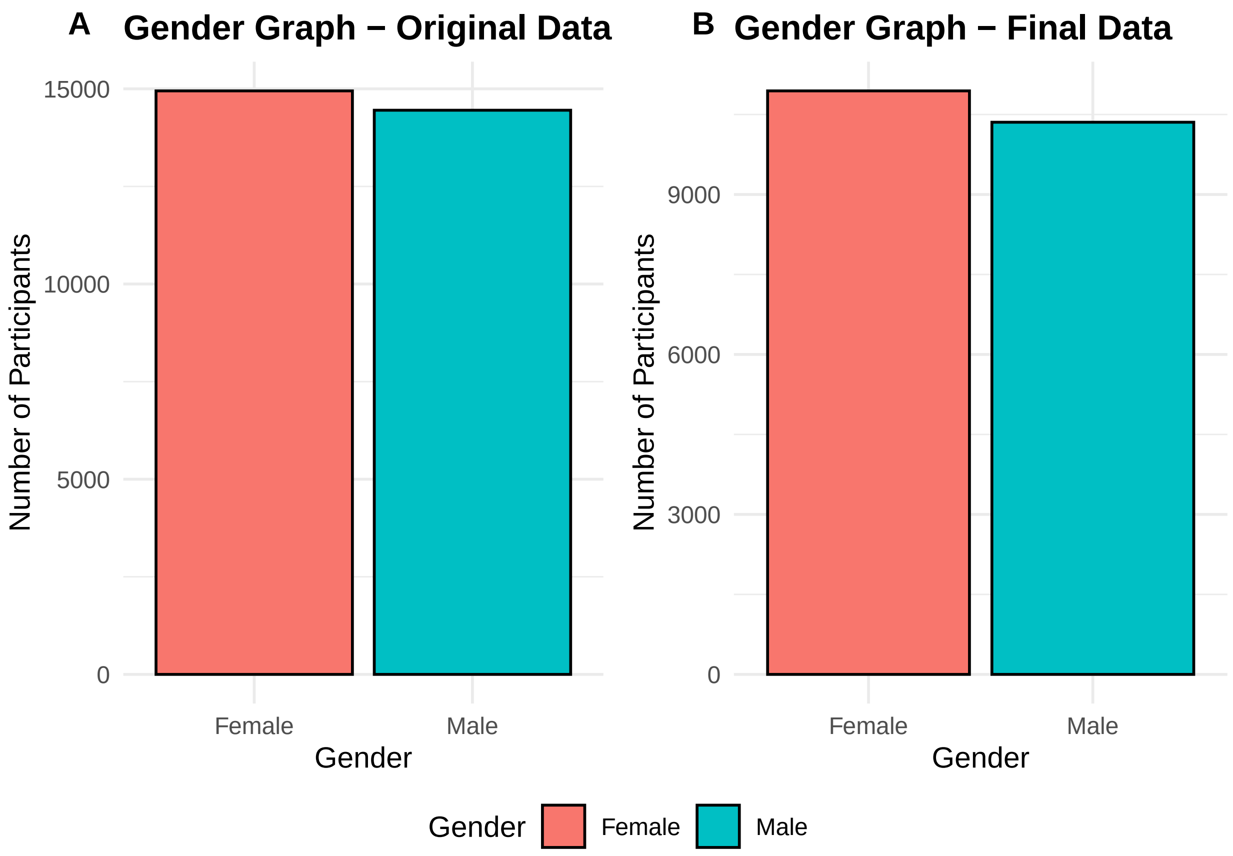


Figure 9: Gender distribution in the final sample

Comparison of the variable Gender before and after filtering the sample

```
# Original gender bar plot (from Exercise 2)
gender_bar <- ggplot(clean_NHANES, aes(x = gender, fill = gender)) +
  geom_bar(color = "black") +
  labs(
    title = "Gender Graph - Original Data",
    x = "Gender",
    y = "Number of Participants",
    fill = "Gender"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0, face = "bold")
)
# Same plot but using filtered data
gender_bar_filtered <- ggplot(nhanes_filtered, aes(x = gender, fill = gender)) +
  geom_bar(color = "black") +
  labs(
    title = "Gender Graph - Final Data",
    x = "Gender",
    y = "Number of Participants",
    fill = "Gender"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0, face = "bold")
)

# Combine side by side
ggarrange(gender_bar, gender_bar_filtered,
          ncol = 2, nrow = 1,
          labels = c("A", "B"),
          label.x = 0.1,
          common.legend= TRUE,
          legend= 'bottom',
          font.label = list(size = 12, face = "bold")
)
```



Panel A shows the gender distribution in the original dataset, while Panel B shows the distribution after restricting to individuals with complete data on age, gender, and SBP.
After filtering, the total number of females decreased from about ***15,000*** *to* ***10,800****, and the number of males decreased from about* ***14,500*** *to* ***10,500****.*
Although the absolute counts dropped, the gender proportions remained similar between the two samples.

Step-5: Changes of hypertension categories across age and gender using the facet_wrap()

```r
# Step 5: Explore relationship between Age, Hypertension Category, and Gender

# Boxplot of Age by Hypertension Category, faceted by Gender
ggplot(nhanes_filtered, aes(x = hypertension_cat, y = age, fill = hypertension_cat)) +
  geom_boxplot(outlier.alpha = 0.4) +   # display quartiles, median, and semi-transparent outliers
  scale_fill_brewer(palette = "Set2") +  # visually distinct color palette
  labs(
    title = "Age Distribution Across Hypertension Categories",
    x = "Hypertension Category",
    y = "Age (years)",
    fill = "Hypertension Category:"
  ) +
  facet_wrap(~ gender) +        # separate panels for male and female participants
  theme_minimal() +             # clean minimal theme for reporting
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),# tilt x-axis labels for readability
    legend.position = "bottom"           # place legend below the plot
  )
```

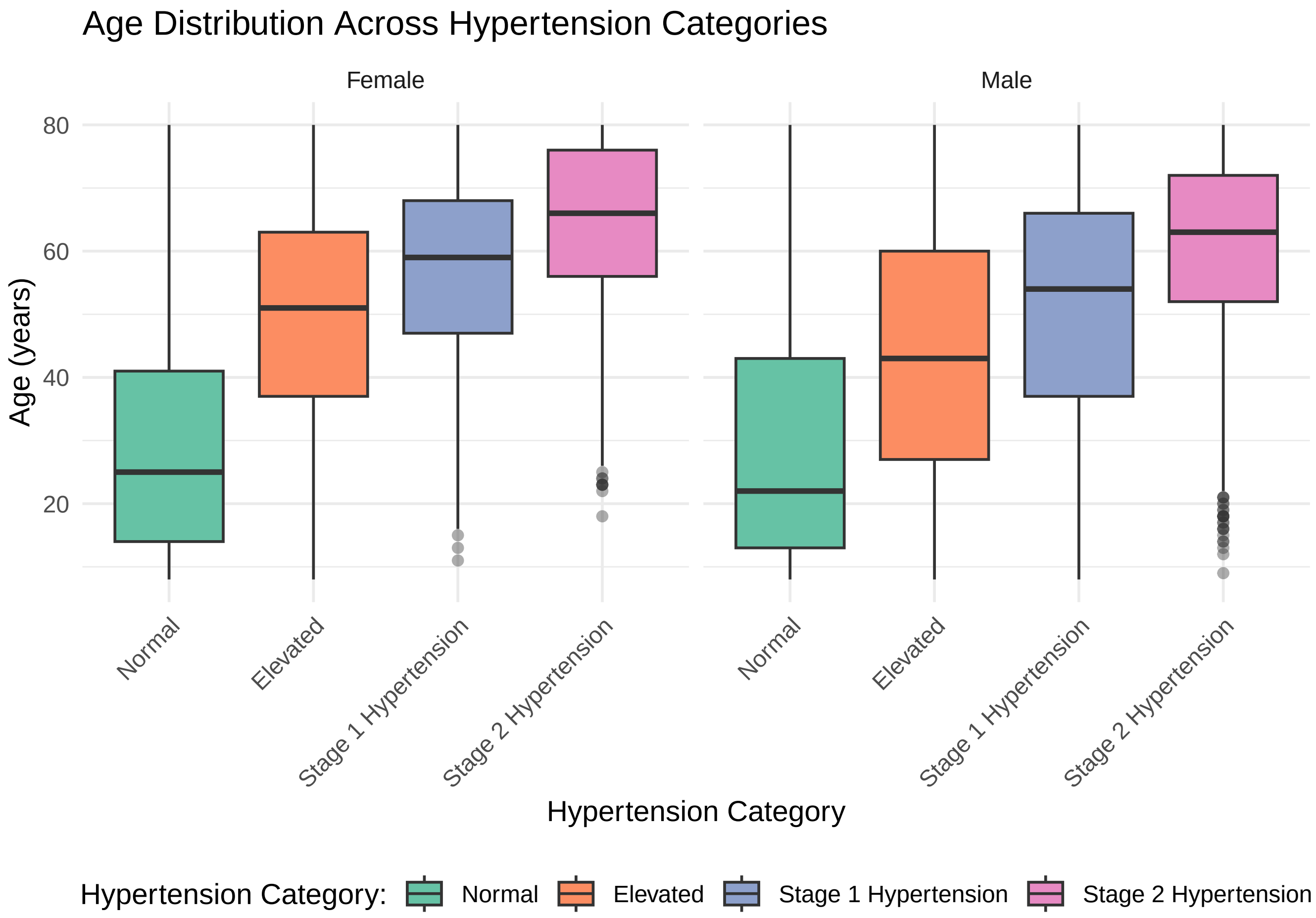


Figure 10: Age distribution across hypertension categories, faceted by gender

NB. I noticed the given hint later and worked on step 5 for the second time as shown below.

```r
# Step 5 extra: Categorize age and plot hypertension distribution

# Create age categories (decades)
nhanes_filtered <- nhanes_filtered %>%
  mutate(age_cat = cut(age,
                       breaks = c(0, 29, 39, 49, 59, 69, 79, 120),
                       labels = c("0-29","30-39","40-49","50-59","60-69","70-79","80+"),
                       right = TRUE))

# Stacked bar plot of hypertension categories across age groups, faceted by gender
ggplot(nhanes_filtered, aes(x = age_cat, fill = hypertension_cat)) +
  geom_bar(position = "stack", color = "black") +    # stacked bars showing counts
  scale_fill_brewer(palette = "Set2") +              # clean color palette
  labs(
    title = "Hypertension Category Distribution Across Age Groups",
    x = "Age Group",
    y = "Number of Participants",
    fill = "Hypertension Category:"
  ) +
  facet_wrap(~ gender) +                # separate panels for males and females
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "bottom"
  )
```

Narrative: Exploring Relationships Between Age, Gender, and Systolic Blood Pressure

The final dataset included 21,298 participants with complete information on age, gender, and average systolic blood pressure (SBP). After removing individuals with missing values, the number of females decreased to approximately 10,800 and males to around 10,500, maintaining a roughly balanced gender distribution.

Participants were categorized into four hypertension categories based on their average SBP: Normal (<120 mmHg), Elevated (120–129 mmHg), Stage 1 Hypertension (130–139 mmHg), and Stage 2 Hypertension (140 mmHg). The distribution of participants across these categories shows that the majority fall into the Normal category (~12,500 participants), followed by Elevated (~3,250), Stage 1 Hypertension (~2,500), and Stage 2 Hypertension (~3,000) categories. These counts correspond to approximate proportions of 59%, 15%, 12%, and 14%, respectively, highlighting that most participants have normal blood pressure but a substantial minority have elevated or high SBP.

The boxplot of age across hypertension categories, faceted by gender (Figure 1), illustrates that participants in higher hypertension categories tend to be older. Median age increases from the Normal group to Stage 2 Hypertension for both males and females. The boxplot also shows the spread and variability within each category, indicating that while older participants are more likely to have higher SBP, younger individuals can still fall into elevated categories.

The stacked bar plot of hypertension categories by age group, faceted by gender (Figure 2) further clarifies the relationship between age and blood pressure. The proportion of participants in higher hypertension categories increases with age in both sexes. For example, Stage 2 Hypertension is relatively rare among participants under 40 but becomes increasingly common in older age groups. This plot complements the boxplot by emphasizing changes in prevalence rather than individual-level distributions.

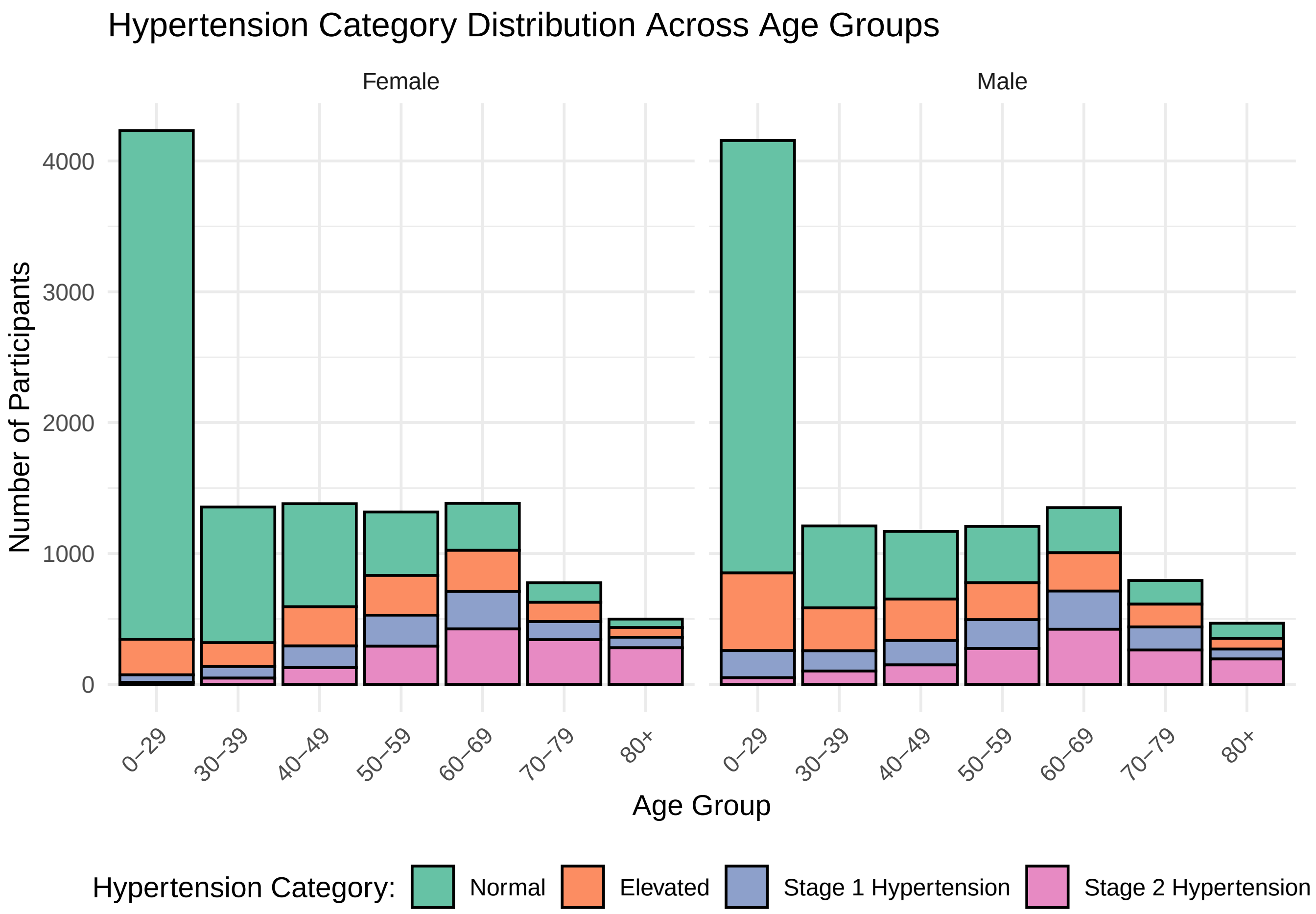


Figure 11: Hypertension category distribution across age groups, faceted by gender

Overall, the visualizations reveal a clear relationship between age, gender, and hypertension. Older participants are more likely to fall into elevated blood pressure categories, and this trend is consistent across males and females. These findings underscore the importance of regular blood pressure monitoring and suggest that targeted interventions may be particularly relevant for older adults.