Take-home Exercise 1 Part 2

Ng Jin Yao

2025-05-10

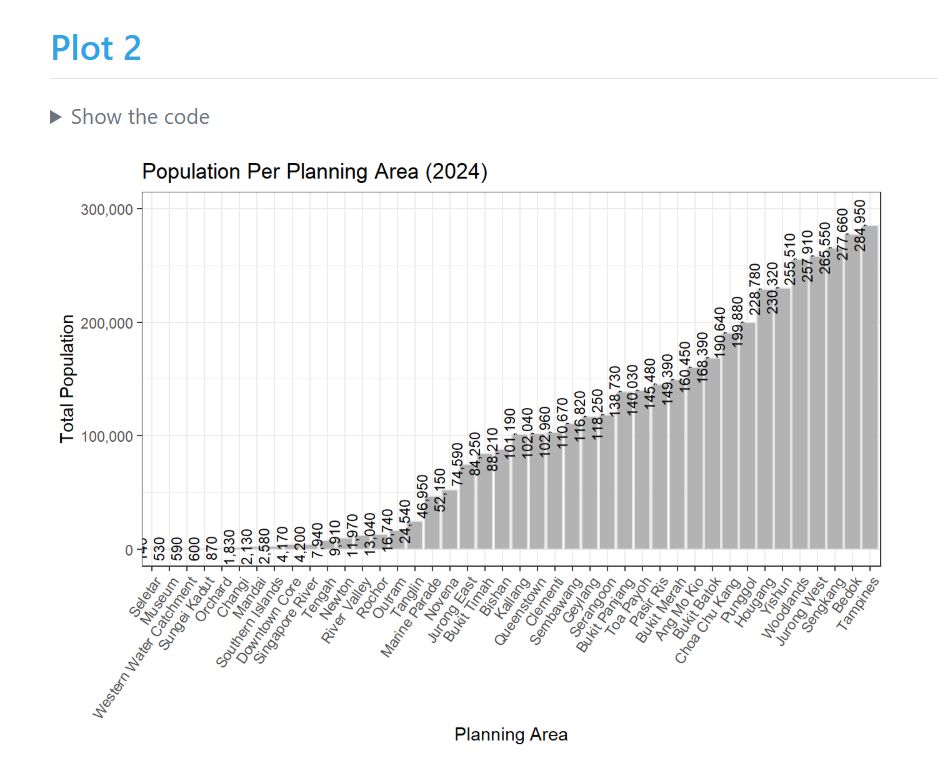
# Take Home Exercise 1 Part 2

## The Task

* Selecting one submission provided by your classmate, critic three good design principles and three areas for further improvement. With reference to the comment, prepare the makeover version of the data visualisation.

## Selected Visualisation and Feedback

Selected Visualisation: [Marga Thura](https://isss608-ay2024-2025.netlify.app/take-home_ex/take-home_ex01/take-home_ex01): Section 3, Plot 2



### 3 Good Design Principles

1. Usage of thousands separator in data labels and axis to improve readability for numbers
2. X-axis labels are tilted in an angle to help with readability for the Planning Area groups
3. Soft gridlines with a lighter shade than the barchart bar fill colors are used that do not distract from the main barchart data

### 3 Areas of Improvement:

1. Choice of color for the bars - a darker color can be used to further improve the contrast of the bar graph from the page
2. Alignment of Data labels with respective bars and data labels can be adjusted so that the labels do not exceed the plot area
3. Current graph has too many groups and the number difference is too wide, affecting the scaling and readability of the data labels - the graphs can be split to focus on top and bottom 10 populations to improve clarity as it allows graphs to show lesser groups and adjust scaling accordingly to improve the aesthetics

## Replicating Data Loading and Cleaning.

I replicated the data loading and cleaning of data using the code provided from Marga’s Take-home Exercise 1 page:

# Load necessary library  
library(readr)  
  
# Read the CSV file from the specified relative path  
respopagesex2024 <- read\_csv("data/respopagesex2024.csv")

pacman::p\_load(tidyverse, ggrepel, ggthemes,   
 hrbrthemes, patchwork,   
 haven, ggiraph, plotly, DT,   
 knitr, scales,  
 ggridges, ggpubr,   
 gganimate, gapminder, ggdist,   
 ggtext, ggalt,  
 cowplot)

library(readr)  
library(dplyr)  
  
# Load the dataset  
respop <- read\_csv("data/respopagesex2024.csv")  
  
# Convert types and clean  
respop\_clean <- respop %>%  
 mutate(  
 Pop = as.numeric(Pop),  
 Age = as.numeric(Age),  
 PA = as.factor(PA),  
 SZ = as.factor(SZ),  
 Sex = factor(Sex, levels = c("Males", "Females"))  
 ) %>%  
 filter(!is.na(Pop), Pop > 0)  
print(respop\_clean)

# A tibble: 37,243 × 6  
 PA SZ Age Sex Pop Time  
 <fct> <fct> <dbl> <fct> <dbl> <dbl>  
 1 Ang Mo Kio Ang Mo Kio Town Centre 0 Males 10 2024  
 2 Ang Mo Kio Ang Mo Kio Town Centre 0 Females 10 2024  
 3 Ang Mo Kio Ang Mo Kio Town Centre 1 Males 10 2024  
 4 Ang Mo Kio Ang Mo Kio Town Centre 1 Females 10 2024  
 5 Ang Mo Kio Ang Mo Kio Town Centre 2 Males 10 2024  
 6 Ang Mo Kio Ang Mo Kio Town Centre 2 Females 10 2024  
 7 Ang Mo Kio Ang Mo Kio Town Centre 3 Males 10 2024  
 8 Ang Mo Kio Ang Mo Kio Town Centre 3 Females 10 2024  
 9 Ang Mo Kio Ang Mo Kio Town Centre 4 Males 30 2024  
10 Ang Mo Kio Ang Mo Kio Town Centre 4 Females 10 2024  
# ℹ 37,233 more rows

|  |
| --- |
| Adjustment Made: |
| Made an adjustment to use the cleaned dataset which was cleaned above for the summary |

library(readr)  
 library(dplyr)  
 library(ggplot2)  
 library(scales)  
  
 # Load and summarize population data  
 # respopagesex2024 <- read\_csv("TakeHome\_01/respopagesex2024.csv")  
 #pop\_by\_pa <- respopagesex2024 %>%   
 pop\_by\_pa <- respop\_clean %>% #adjusted to use the cleaned dataset  
 mutate(Pop = as.numeric(Pop)) %>%  
 filter(!is.na(Pop) & Pop > 0) %>%  
 group\_by(PA) %>%  
 summarise(Total\_Pop = sum(Pop, na.rm = TRUE))  
  
 # Check population range  
 print(summary(pop\_by\_pa$Total\_Pop))

Min. 1st Qu. Median Mean 3rd Qu. Max.   
 140 8432 94700 99846 157685 284950

## Makeover Version of Visualisation

**Adjustments I have made to Marga’s original graph based on the 3 Areas of Improvement feedback:**

* Choice of color for the bars - a darker color can be used to further improve the contrast of the bar graph from the page:

|  |
| --- |
| Adjustment #1: |
| Adjusted fill color for geombar plot from “grey70” to a darker “grey40”. |

* Alignment of Data labels with respective bars and data labels can be adjusted so that the labels do not exceed the plot area:

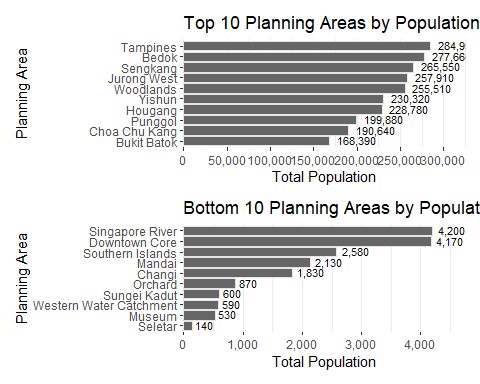
|  |
| --- |
| Adjustment #2: |
| Adjusted geom\_text hjust and vjust values to centre data labels to the respective bars in the chart |

* Current graph has too many groups and the number difference is too wide, affecting the scaling and readability of the data labels - the graphs can be split to focus on top and bottom 10 populations to improve clarity as it allows graphs to show lesser groups and adjust scaling accordingly to improve the aesthetics:

|  |
| --- |
| Adjustment #3: |
| 1. Flipped the axis for population and planning area for better clarity 2. Adjusted the visualisation to generate plot1 for top 10 population PAs and plot2 for bottom 10 population PAs. Then generated a composite figure of plot1 and plot2 together using the patchwork extension 3. Other miscellaneous adjustments to text size, axis limits to accommodate aesthetic adjustments for plot1, plot2 and the composite graph such as removing the horizontal axis and adjusting text size. |

## Composite Visualisation (using patchwork)

library(patchwork)  
plot1 / plot2

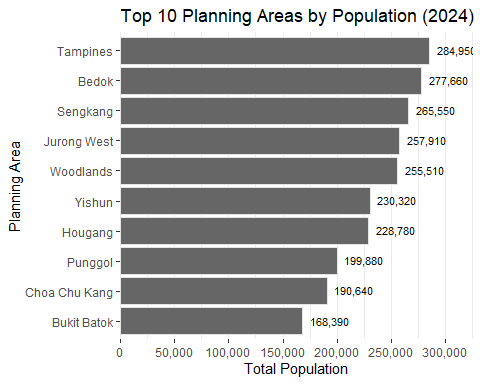


## Cleaning Code

# Disable scientific notation globally  
options(scipen = 999)  
  
# Load data  
respopagesex2024 <- read\_csv("data/respopagesex2024.csv")  
  
# Trim whitespace from PA and SZ  
respopagesex2024 <- respopagesex2024 %>%  
 mutate(  
 PA = trimws(PA),  
 SZ = trimws(SZ)  
 )  
  
# Clean and summarize population by PA, including subzone count  
pop\_by\_pa <- respopagesex2024 %>%  
 mutate(Pop = as.numeric(Pop)) %>%  
 filter(!is.na(Pop) & Pop > 0) %>%  
 group\_by(PA) %>%  
 summarise(  
 Total\_Pop = sum(Pop, na.rm = TRUE),  
 Num\_Subzones = n\_distinct(SZ) # Count unique subzones per PA  
 ) %>%  
 arrange(desc(Total\_Pop))

## Creating Bar Chart for Top 10 PAs

# Create bar chart for top 10  
top\_n <- 10   
top\_pa <- pop\_by\_pa %>%  
 slice\_max(order\_by = Total\_Pop, n = top\_n)  
  
plot1 <- ggplot(data = top\_pa, aes(x = reorder(PA, Total\_Pop), y = Total\_Pop)) +  
 geom\_bar(stat = "identity", fill = "grey40", color = "grey90") +  
 geom\_text(  
 aes(label = comma(Total\_Pop)),  
 #angle = 90,  
 hjust = -0.2,  
 vjust = 0.4,  
 size = 3  
 ) +  
 coord\_flip()+  
 theme\_bw() +  
 theme(panel.grid.major.y = element\_blank(),panel.border = element\_blank())+  
 ggtitle(paste("Top", top\_n, "Planning Areas by Population (2024)")) +  
 xlab("Planning Area") +  
 ylab("Total Population") +  
 #theme(axis.text.x = element\_text(angle = 55, hjust = 1)) +  
 scale\_y\_continuous(  
 labels = comma,  
 expand = c(0, 0),  
 breaks = seq(0, 325000, by = 50000),  
 limits = c(0, 325000)  
)  
  
print(plot1)



## Creating Bar Chart for Bottom 10 PAs

# Create bar chart for btm 10  
btm\_n <- 10 #   
btm\_pa <- pop\_by\_pa %>%  
 slice\_min(order\_by = Total\_Pop, n = btm\_n)  
  
plot2 <- ggplot(data = btm\_pa, aes(x = reorder(PA, Total\_Pop), y = Total\_Pop)) +  
 geom\_bar(stat = "identity", fill = "grey40", color = "grey90") +  
 geom\_text(  
 aes(label = comma(Total\_Pop)),  
 #angle = 90,  
 hjust = -0.2,  
 vjust = 0.4,  
 size = 3  
 ) +  
 coord\_flip()+  
 theme\_bw() +  
 theme(panel.grid.major.y = element\_blank(),panel.border = element\_blank())+  
 ggtitle(paste("Bottom", btm\_n, "Planning Areas by Population (2024)")) +  
 xlab("Planning Area") +  
 ylab("Total Population") +  
 #theme(axis.text.x = element\_text(angle = 55, hjust = 1)) +  
 scale\_y\_continuous(  
 labels = comma,  
 expand = c(0, 0),  
 breaks = seq(0, 4750, by = 1000),  
 limits = c(0, 4750)  
)  
  
print(plot2)

