Take-Home\_Ex01\_Part2

Andre Ong Jia Kang

2025-05-10

# 1. Overview

## 1.1 Background

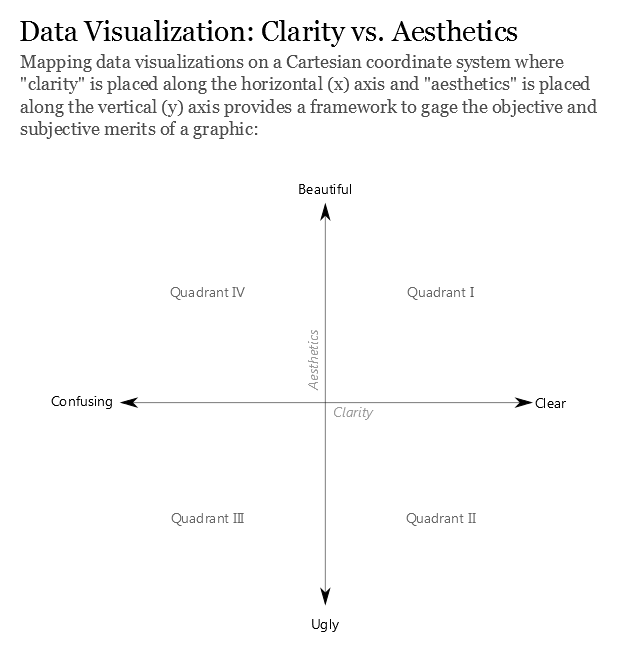
*We are tasked with the below instructions for ‘Take-Home\_Ex01\_Part2’:*

|  |
| --- |
| Part 2 Instructions |
| Part 2: 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 visualization. |

## 1.2 Framework to Critic Visualizations & Classmates links

During lecture, an article published by [Ben Jones](https://dataremixed.com/2012/05/data-visualization-clarity-or-aesthetics/) was shared as a framework to judge data visualization plots. I will be using this as a baseline to praise or critic the plots in this exercise.

### 1.2.1 The 4 Quadrants



*The Four Quadrants:*

* Quadrant I is for the winning visualizations that are both clear AND beautiful.
* Quadrant II, where those unfortunate clear but ugly visualizations live.
* Quadrant III, the saddest plot of land reserved for those ugly AND confusing works that should never have been.
* Quadrant IV – the insidious land of the visualizations that stun you with their graphical beauty, but don’t impart a lick of understanding of the real world, at least not of the accurate sort. Even worse, they tend to mislead the reader, sometimes with malicious intent.

## 1.3 Classmate Links

I will be using the following classmate’s submission **(Click the link below)** for this activity. His data cleaning will be replicated in the initial steps to get the original plot.

[Plot 3](https://cabbage-wonderland.netlify.app/takehome_ex/takehome_ex01/takehome_ex01)

# 2. Getting Started

## 2.1 Loading of libraries and dataset

**Libraries:**

pacman::p\_load(ggrepel, patchwork,   
 ggthemes, hrbrthemes,  
 tidyverse, ggiraph, plotly,   
 patchwork, DT, dplyr, scales, forcats, dplyr, grid)

**Dataset:**

df <- read\_csv("data/respopagesex2024.csv")

## 2.2 Loading original skeleton and codes

|  |
| --- |
| Original Author’s Theme |
| We will keep to the original author’s code to replicate his/her plot. |

**Theme:**

Using the original author’s theme.

common\_theme <- theme\_minimal(base\_size = 16) +  
 theme(  
 axis.text = element\_text(size = 18),  
 axis.title = element\_text(size = 20),  
 plot.title = element\_text(size = 18, face = "bold"),  
 legend.text = element\_text(size = 17),  
 legend.title = element\_text(size = 16)  
 )

**Creating a Numeric Age Column & Age Grouping:**

Column type created are integers to be used for plotting.

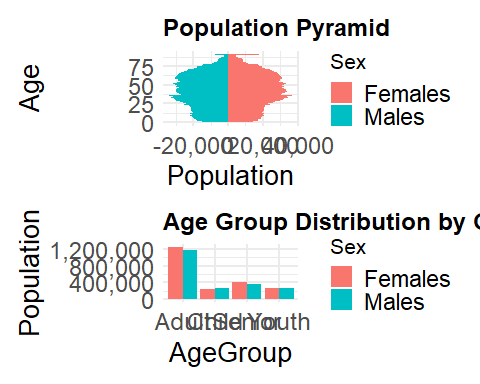
df <- df %>%  
 mutate(  
 AgeNum = suppressWarnings(  
 ifelse(Age == "90\_and\_Over", 90, as.numeric(Age))  
 )  
 )  
  
df <- df %>%  
 mutate(  
 AgeGroup = case\_when(  
 AgeNum <= 12 ~ "Child",  
 AgeNum <= 24 ~ "Youth",  
 AgeNum <= 64 ~ "Adult",  
 TRUE ~ "Senior"  
 )  
 )

# 3. Original Plot: Population Structure by Age / Gender

## 3.1 Orginal Plot

|  |
| --- |
| Replicating the Original Plot |
| Below is the original plot (Plot 3) from the chosen classmate’s link. It might look smaller from the original link but the overall aesthetic remains the same. |

#### Plot



#### Code

df\_pyramid <- df %>%  
 filter(AgeNum <= 90) %>%  
 mutate(Pop = ifelse(Sex == "Males", -Pop, Pop))  
  
p6 <- ggplot(df\_pyramid, aes(x = AgeNum, y = Pop, fill = Sex)) +  
 geom\_col(width = 1) +  
 coord\_flip() +  
 labs(title = "Population Pyramid", x = "Age", y = "Population") +  
 scale\_y\_continuous(labels = label\_comma()) +  
 common\_theme  
  
p7 <- df %>%  
 group\_by(Sex, AgeGroup) %>%  
 summarise(Pop = sum(Pop)) %>%  
 ggplot(aes(x = AgeGroup, y = Pop, fill = Sex)) +  
 geom\_bar(stat = "identity", position = "dodge") +  
 labs(title = "Age Group Distribution by Gender", y = "Population") +  
 scale\_y\_continuous(labels = label\_comma()) +  
 common\_theme  
  
(p6 / p7) + plot\_layout(heights = c(1.3, 1))

### **3.1.1 Good Design Principles**

1. **Clean Plots with Gridlines**

The gridlines for both the 1) Pyramid plot and 2) Bar charts help to guide the user on the axis references.

1. **Colour Theme**

The colour code of males and females were consistent and easily distinguishable for both plots. I am instantly able to identify the gender from the plots.

1. **Clarity of Secondary Bar Chart Plot**

The bar chart helps to provide a simple but direct comparison of the ranked categorical data. It describes the total population of each ‘Age groups’ where majority of the population are ‘Adults’.

### **3.1.2 Areas of Improvements**

1. **Axis Scale**

The x-axis of the pyramid plot has a negative number for the population and the scaling is too large to identify the population number of each age.

Suggestion: **This requires changing the x-axis to only positive values and changing the scale into thousands.**

1. **Binning of Age Groups**

The bars in the pyramid plot is too clustered, where I am unable to distinguish the age groups clearly as all the ages are represented in the plot. Secondly, the Y-axis scale of only 4 digits does not clearly identify all the age groups for identification.

Suggestion: **This can tidied by grouping the ages into ranges for better representation and also reduce the amount of the bars on the pyramid plot**

1. **Editing the Age Grouping & Sorting it via age**

For the bar chart plot, the ‘Adult’ group might be too bias as majority of the population falls under the range between ‘25 - 65 years old’. It would be better to increases the amount of groups from the initial 4 for a more accurate representation of the population groups. Secondly, the groups are not sorted properly making it confusing to identify which has the least population.

Suggestion: **This can be adjusted by splitting the ‘Adult’ groups into ‘Young Adults’, ‘Middle-Aged Adults’ & ‘Older Adults’. Ensuring the groups are categorized as equal/fair as possible. The new groups are sorting according to increasing age groups for easier identification.**

# 4 Rework: Improvements to the Plot

## 4.1 Additional Code Chunks to the plot

### 4.1.1 Editing the Age column and Binning

The Age column is not entirely numeric due to character values “90\_and\_Over”. I will directly address this by creating a new column called ‘AgeNum2’ and re-code those character values to “90” using string replacement. This will make the binning of new age ranges for the edited pyramid plot easier.

**Binning into ranges:**

I will bin the ages into incremental ranges of 5 using AgeNum column into ‘AgeNum2’.

age\_breaks <- seq(0, 90, by = 5)  
age\_labels <- c(paste(seq(0, 80, 5), seq(4, 84, 5), sep = "-"), "85-89", "90+")  
  
df <- df %>%  
 mutate(  
 AgeNum = ifelse(AgeNum == "90+", 90, AgeNum),  
 AgeNum = as.numeric(AgeNum),  
 AgeNum2 = cut(  
 AgeNum,  
 breaks = c(seq(0, 90, 5), Inf),  
 labels = age\_labels,  
 right = FALSE  
 )  
 )

**Expanding Age Groupings:**

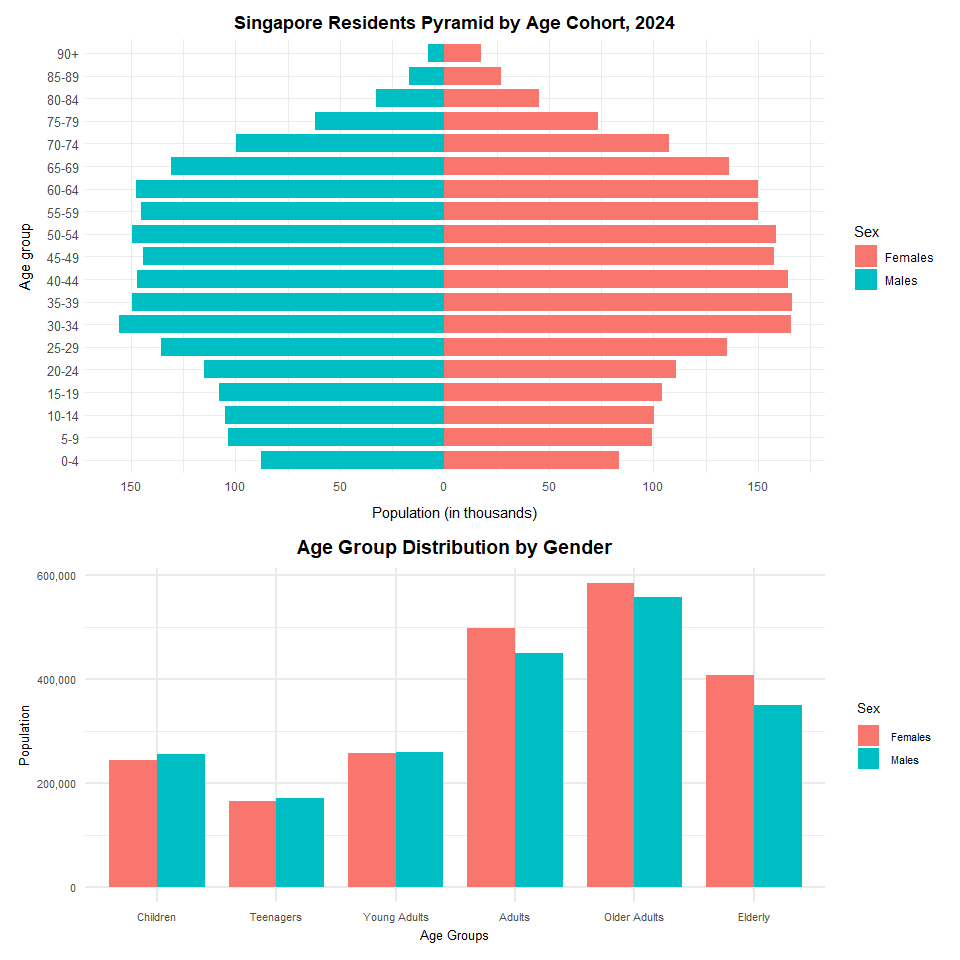
Adding Another column ‘AgeGroup2’ by editing the Age ranges

df <- df %>%  
 mutate(  
 AgeGroup2 = case\_when(  
 AgeNum <= 12 ~ "Children",  
 AgeNum <= 20 ~ "Teenagers",  
 AgeNum <= 30 ~ "Young Adults",  
 AgeNum <= 45 ~ "Adults",  
 AgeNum <= 64 ~ "Older Adults",  
 TRUE ~ "Elderly"  
 )  
 )

## 4.2 Reworking the Actual Plot

Combining the additional code blocks above together with optimizing the original code. The below plot is derived.

#### Plot



#### Code

df\_pyramid <- df %>%  
 mutate(Pop = ifelse(Sex == "Males", -Pop, Pop))  
  
p6 <- ggplot(df\_pyramid, aes(x = AgeNum2, y = Pop, fill = Sex)) +  
 geom\_bar(stat = "identity", width = 0.8) +  
 coord\_flip() +  
 scale\_y\_continuous(  
 breaks = seq(-150000, 150000, 50000),  
 labels = abs(seq(-150, 150, 50))  
 ) +  
 labs(  
 title = "Singapore Residents Pyramid by Age Cohort, 2024",  
 x = "Age group",  
 y = "Population (in thousands)"  
 ) +  
 theme\_minimal() +  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold"),  
 axis.text.y = element\_text(size = 10),  
 axis.title.x = element\_text(margin = margin(t = 10))  
 )  
  
p7 <- df %>%  
 group\_by(Sex, AgeGroup2) %>%  
 summarise(Pop = sum(Pop)) %>%  
 ggplot(aes(x = AgeGroup2, y = Pop, fill = Sex)) +  
 geom\_bar(stat = "identity", position = "dodge", width = 0.8) +  
 scale\_x\_discrete(limits = c(  
 "Children","Teenagers","Young Adults","Adults","Older Adults","Elderly"  
 )) +  
 labs(title = "Age Group Distribution by Gender", y = "Population", x = "Age Groups") +  
 scale\_y\_continuous(labels = label\_comma()) +  
 common\_theme +  
 theme(  
 plot.title = element\_text(size = 14, hjust = 0.5, face = "bold"),  
 axis.title.x = element\_text(size = 10, margin = margin(t = 4)),  
 axis.title.y = element\_text(size = 10),  
 axis.text.y = element\_text(size = 8),  
 axis.text.x = element\_text(size = 8),  
 legend.title = element\_text(size = 10),  
 legend.text = element\_text(size = 8)  
 )  
  
(p6 / p7) + plot\_layout(heights = c(1.3, 1))

|  |
| --- |
| Conclusion |
| With these changes to the original plots, we can visualize the Age and Population distribution much clearer. |