Take-home\_Exercise 1

Demographic structures and distribution of Singapore in 2024

Audrey Tan

2025-05-04

# 1. Overview

## 1.1 Background and Task

Aero Media is a local online media company that publishes daily content on digital platforms is planning to release an article on **demographic structures and distribution of Singapore in 2024**. As a graphical editor of the media company (Aero Media), there is a need to prepare **at most** **three** non-interactive data visualisation for the daily article. This article aims to explore the distribution and structure of the population age, gender, and regions. Please refer to Section 2 to 7 for the rationalisation and preparation, and Section 8 for this Article. Please refer to Section 10 for the comments and edits for Phase 2 of the task.

## 1.2 The Data

To accomplish the task, *Singapore Residents by Planning Area / Subzone, Single Year of Age and Sex, June 2024* dataset shares by [Department of Statistics, Singapore (DOS)](https://www.singstat.gov.sg/) is used.

# 2. Loading Libraries and Packages

The code chunk below uses p\_load() of pacman package to check if packages are installed in the computer. If they are, then they will be launched into R.

pacman::p\_load(ggrepel, patchwork,  
 ggthemes, hrbrthemes,  
 tidyverse, readr, ggdist, ggExtra,  
 ggridges, colorspace, knitr, plotly, scales,  
 forcats, reshape2, png, grid, cowplot, ggplot2,  
 dplyr, magrittr, RColorBrewer, magick, rmarkdown)

| Library | Purpose |
| --- | --- |
| tidyverse | * Include dpylr for data manipulation with “mutate”, “group\_by”, “summarize”, etc. * Include ggplot2 for visualisation through histogram, boxplots, etc. * Include forcats for factoring and ordering of variables |
| ggdist | For visualizing distributions and uncertainty. It provides tools like stat\_halfeye(), stat\_pointinterval(), etc., for making raincloud plots |
| ggExtra | For adding functions and layers like ggMarginal() to ggplot2. |
| ggridges | For creating density plots for visualisation of continuous distribution |
| ggrepel | For repelling overlapping text labels in plot |
| ggthemes & hrbrthemes | For customisation of plot appearance |
| patchwork | For combination and alignment of multiple plots for each visualisation |
| knitr | For elegant, flexible and fast report generation of underlying dataframes |
| plotly | For plotting interactive statistical graphs |
| forcats | For working with categorical variables |
| reshape2 | For restructuring data between “wide” and “long” formats. |
| png | For reading and writing Portable Network Graphics (PNG) image files in R |
| grid | For powerful and flexible low-level graphics system in R |
| dplyr | For data manipulation. |
| magrittr | For introducing the pipe operator %>% into R. |
| RColorBrewer | For creating colour palettes |
| magick | For image processing to read, edit, compose, and convert images easily. |
| scales | For normalizing of a dataset using the mean value and standard deviation |
| rmarkdown | For converting R Markdown documents into different formats. |
| cowplot | For streamlining the process of creating and arranging plots. |
| ggplot2 | For creating visuals based on Grammar of Graphics. |

# 3. Importing Data

sg\_demo\_2024 <- read\_csv("respopagesex2024.csv",show\_col\_types = FALSE)

# 4. Data Understanding

## 4.1 Viewing Data Rows and Columns using glimpse

glimpse(sg\_demo\_2024)

Rows: 60,424  
Columns: 6  
$ PA <chr> "Ang Mo Kio", "Ang Mo Kio", "Ang Mo Kio", "Ang Mo Kio", "Ang Mo K…  
$ SZ <chr> "Ang Mo Kio Town Centre", "Ang Mo Kio Town Centre", "Ang Mo Kio T…  
$ Age <chr> "0", "0", "1", "1", "2", "2", "3", "3", "4", "4", "5", "5", "6", …  
$ Sex <chr> "Males", "Females", "Males", "Females", "Males", "Females", "Male…  
$ Pop <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 30, 10, 20, 10, 20, 30, 30, 10, 3…  
$ Time <dbl> 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024,…

* The data contains data on Singapore Residents by Planning Area / Subzone, Single Year of Age and Sex, June 2024.
* The dataframe has 6 columns and 60,424 rows. Four of the columns contains categorical data type and the other two are in double/real number data type.
  + The categorical attributes are: PA, SZ, Age and Sex
  + The numerical attributes are: Pop, and Time.

## 4.2 Data Dictionary for Variables

| Category | Column Headers | Type | Description |
| --- | --- | --- | --- |
| Planning Area | PA | CHAR | 55 area names |
| Subzone | SZ | CHAR | 332 area names |
| Single Year of Age | Age | CHAR | 0 to 90\_and\_over (in increments of 1) |
| Sex | Sex | CHAR | Males; Females |
| Resident Count | Pop | NUM | 0 to 1180 (in multiples of 10, with occasional breaks) |
| Time/Period | Time | NUM | 2024 |

|  |
| --- |
| Note. |
| 1. For June 2024, Planning Areas refer to areas demarcated in the Urban Redevelopment Authority’s Master Plan 2019. 2. Data from 2003 onward exclude residents who have been away from Singapore for a continuous period of 12 months or longer as at the reference period. 3. The figures have been rounded to the nearest 10. 4. The data may not add up due to rounding. 5. Data was generated on 24/9/2024. |

### 4.2.1 Categorical values

These are the categorical values within PA and SZ.

## The Data

Unique Categories in Dataset

|  | Values |
| --- | --- |
| PA | Ang Mo Kio, Bedok, Bishan, Boon Lay, Bukit Batok, Bukit Merah, Bukit Panjang, Bukit Timah, Central Water Catchment, Changi, Changi Bay, Choa Chu Kang, Clementi, Downtown Core, Geylang, Hougang, Jurong East, Jurong West, Kallang, Lim Chu Kang, Mandai, Marina East, Marina South, Marine Parade, Museum, Newton, North-Eastern Islands, Novena, Orchard, Outram, Pasir Ris, Paya Lebar, Pioneer, Punggol, Queenstown, River Valley, Rochor, Seletar, Sembawang, Sengkang, Serangoon, Simpang, Singapore River, Southern Islands, Straits View, Sungei Kadut, Tampines, Tanglin, Tengah, Toa Payoh, Tuas, Western Islands, Western Water Catchment, Woodlands, Yishun |
| SZ | Ang Mo Kio Town Centre, Cheng San, Chong Boon, Kebun Bahru, Sembawang Hills, Shangri-La, Tagore, Townsville, Yio Chu Kang, Yio Chu Kang East, Yio Chu Kang North, Yio Chu Kang West, Bayshore, Bedok North, Bedok Reservoir, Bedok South, Frankel, Kaki Bukit, Kembangan, Siglap, Bishan East, Marymount, Upper Thomson, Liu Fang, Samulun, Shipyard, Tukang, Brickworks, Bukit Batok Central, Bukit Batok East, Bukit Batok South, Bukit Batok West, Gombak, Guilin, Hillview, Hong Kah North, Alexandra Hill, Alexandra North, Bukit Ho Swee, Bukit Merah, City Terminals, Depot Road, Everton Park, Henderson Hill, Kampong Tiong Bahru, Maritime Square, Redhill, Singapore General Hospital, Telok Blangah Drive, Telok Blangah Rise, Telok Blangah Way, Tiong Bahru, Tiong Bahru Station, Bangkit, Dairy Farm, Fajar, Jelebu, Nature Reserve, Saujana, Senja, Anak Bukit, Coronation Road, Farrer Court, Hillcrest, Holland Road, Leedon Park, Swiss Club, Ulu Pandan, Central Water Catchment, Changi Airport, Changi Point, Changi West, Changi Bay, Choa Chu Kang Central, Choa Chu Kang North, Keat Hong, Peng Siang, Teck Whye, Yew Tee, Clementi Central, Clementi North, Clementi West, Clementi Woods, Faber, Pandan, Sunset Way, Toh Tuck, West Coast, Anson, Bayfront Subzone, Bugis, Cecil, Central Subzone, City Hall, Clifford Pier, Marina Centre, Maxwell, Nicoll, Phillip, Raffles Place, Tanjong Pagar, Aljunied, Geylang East, Kallang Way, Kampong Ubi, Macpherson, Defu Industrial Park, Hougang Central, Hougang East, Hougang West, Kangkar, Kovan, Lorong Ah Soo, Lorong Halus, Tai Seng, Trafalgar, International Business Park, Jurong Gateway, Jurong Port, Jurong River, Lakeside (Business), Lakeside (Leisure), Penjuru Crescent, Teban Gardens, Toh Guan, Yuhua East, Yuhua West, Boon Lay Place, Chin Bee, Hong Kah, Jurong West Central, Kian Teck, Safti, Taman Jurong, Wenya, Yunnan, Bendemeer, Boon Keng, Crawford, Geylang Bahru, Kallang Bahru, Kampong Bugis, Kampong Java, Lavender, Tanjong Rhu, Lim Chu Kang, Mandai East, Mandai Estate, Mandai West, Marina East, Marina South, East Coast, Katong, Marina East (Mp), Marine Parade, Mountbatten, Bras Basah, Dhoby Ghaut, Fort Canning, Cairnhill, Goodwood Park, Istana Negara, Monk’s Hill, Newton Circus, Orange Grove, North-Eastern Islands, Balestier, Dunearn, Malcolm, Moulmein, Mount Pleasant, Boulevard, Somerset, Tanglin, China Square, Chinatown, Pearl’s Hill, People’s Park, Flora Drive, Loyang East, Loyang West, Pasir Ris Central, Pasir Ris Drive, Pasir Ris Park, Pasir Ris Wafer Fab Park, Pasir Ris West, Airport Road, Paya Lebar East, Paya Lebar North, Paya Lebar West, Plab, Benoi Sector, Gul Basin, Gul Circle, Joo Koon, Pioneer Sector, Coney Island, Matilda, Northshore, Punggol Canal, Punggol Field, Punggol Town Centre, Waterway East, Commonwealth, Dover, Ghim Moh, Holland Drive, Kent Ridge, Margaret Drive, Mei Chin, National University Of S’pore, One North, Pasir Panjang 1, Pasir Panjang 2, Port, Queensway, Singapore Polytechnic, Tanglin Halt, Institution Hill, Leonie Hill, One Tree Hill, Oxley, Paterson, Bencoolen, Farrer Park, Kampong Glam, Little India, Mackenzie, Mount Emily, Rochor Canal, Selegie, Sungei Road, Victoria, Pulau Punggol Barat, Pulau Punggol Timor, Seletar, Seletar Aerospace Park, Admiralty, Sembawang Central, Sembawang East, Sembawang North, Sembawang Springs, Sembawang Straits, Senoko North, Senoko South, The Wharves, Anchorvale, Compassvale, Fernvale, Lorong Halus North, Rivervale, Sengkang Town Centre, Sengkang West, Lorong Chuan, Seletar Hills, Serangoon Central, Serangoon Garden, Serangoon North, Serangoon North Ind Estate, Upper Paya Lebar, Pulau Seletar, Simpang North, Simpang South, Tanjong Irau, Boat Quay, Clarke Quay, Robertson Quay, Sentosa, Southern Group, Straits View, Gali Batu, Kranji, Pang Sua, Reservoir View, Turf Club, Simei, Tampines East, Tampines North, Tampines West, Xilin, Chatsworth, Nassim, Ridout, Tyersall, Brickland, Forest Hill, Garden, Park, Plantation, Tengah Industrial Estate, Bidadari, Boon Teck, Braddell, Joo Seng, Kim Keat, Lorong 8 Toa Payoh, Pei Chun, Potong Pasir, Sennett, Toa Payoh Central, Toa Payoh West, Woodleigh, Tengeh, Tuas Bay, Tuas North, Tuas Promenade, Tuas View, Tuas View Extension, Jurong Island And Bukom, Semakau, Sudong, Bahar, Cleantech, Murai, Greenwood Park, Midview, North Coast, Senoko West, Woodgrove, Woodlands East, Woodlands Regional Centre, Woodlands South, Woodlands West, Khatib, Lower Seletar, Nee Soon, Northland, Springleaf, Yishun Central, Yishun East, Yishun South, Yishun West |
| Age | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90\_and\_Over |
| Sex | Males, Females |

## The Code

# Understanding the unique data values in each column that is categorical.  
  
get\_unique\_val <- function(df){lapply(df, unique)}  
sg\_demo\_2024\_cat <- sg\_demo\_2024[,c("PA", "SZ", "Age", "Sex")]  
unique\_val <- get\_unique\_val(sg\_demo\_2024\_cat)  
uni\_val\_df <- data.frame(  
 Values = sapply(unique\_val, paste, collapse = ", ")  
)  
  
kable(uni\_val\_df, caption = "Unique Categories in Dataset", row.names = NA, label = NULL)

### 4.2.2 Statistics of Numerical Variables

Notice that:

* Age variable is not a numerical value but a categorical value.
* The statistics for Pop variable are not very useful as the values are a summation for each combination of variables in a particular record.
* In Time variable, there is only one year of data at 2024.

## The Data

|  | Pop | Time |
| --- | --- | --- |
|  | Min. : 0.0 | Min. :2024 |
|  | 1st Qu.: 0.0 | 1st Qu.:2024 |
|  | Median : 20.0 | Median :2024 |
|  | Mean : 69.4 | Mean :2024 |
|  | 3rd Qu.: 90.0 | 3rd Qu.:2024 |
|  | Max. :1180.0 | Max. :2024 |

## The Code

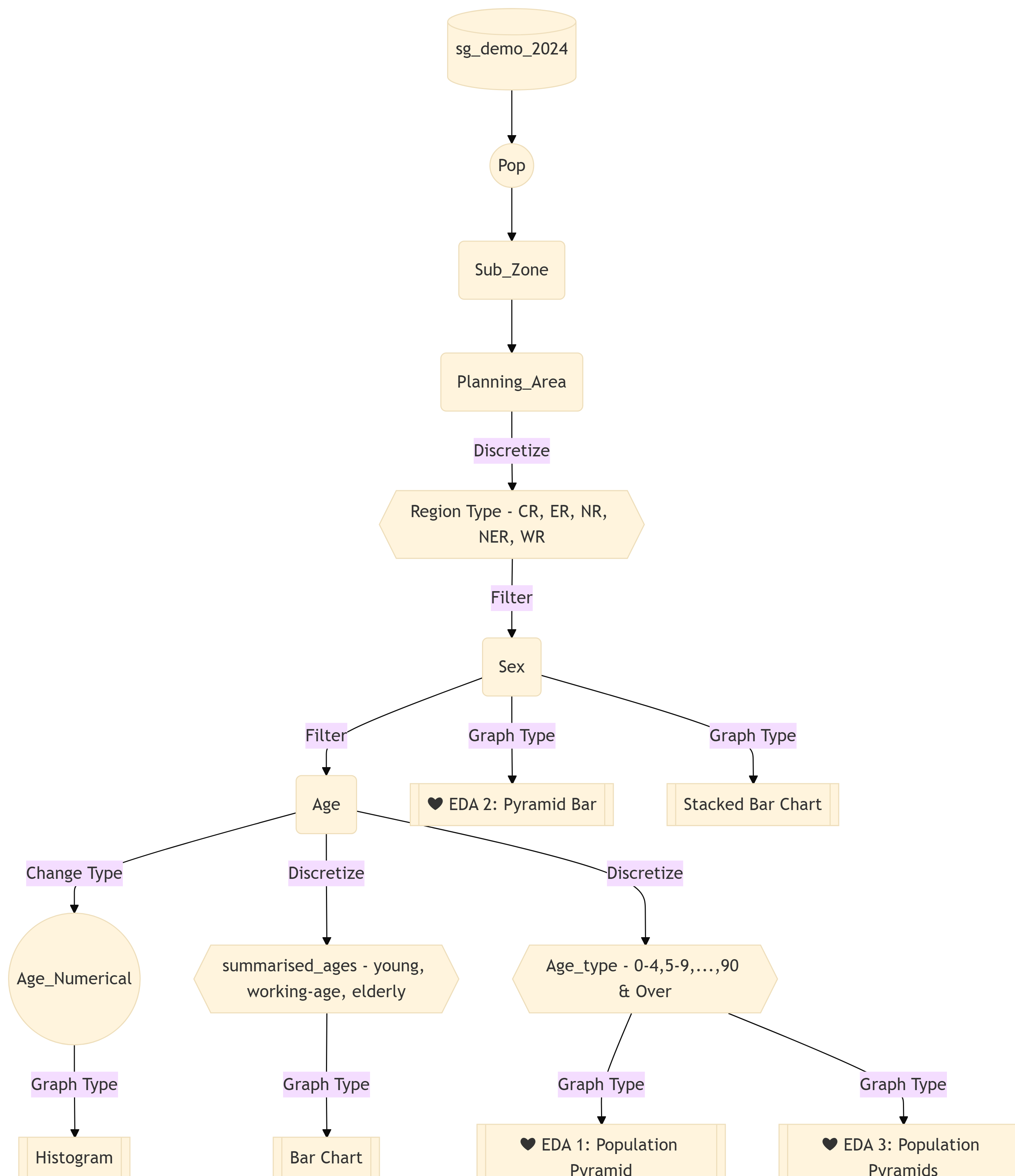
# Understanding the distribution of numerical values   
realis\_num <- sg\_demo\_2024[,c("Pop", "Time")]  
   
kable(summary(realis\_num))

## 4.3 Flowchart of data and graphs

The flowchart displays the course of action to be undertaken to pre-process the data to produce the eventual graphs.

* Starts from the data in cylindrical shape.
* Categorical variables are in rounded edge rectangles.
* Numerical variables are circles.
* Discretized data are in hexagons.
* Graphs are in a subroutine shape.

## The Flowchart



## The Code

%%{  
 init: {  
 'theme': 'base',  
 'themeVariables': {  
 }  
 }  
}%%  
  
flowchart TD  
 id1[(sg\_demo\_2024)]-->C((Pop))  
 C((Pop))-->G(Sub\_Zone)  
 G(Sub\_Zone)--> F(Planning\_Area)   
  
 F -->|Discretize| id2{{Region Type - CR, ER, NR, NER, WR}}  
 id2 -->|Filter| D(Sex)  
 D(Sex)--> |Filter| B(Age)  
 D(Sex)-->|Graph Type|K[[❤ EDA 2: Pyramid Bar]]   
  
 D(Sex)--> |Graph Type|N[[Stacked Bar Chart]]  
  
 B(Age) -->|Change Type| H((Age\_Numerical))  
 B(Age) -->|Discretize| id3{{summarised\_ages - young, working-age, elderly}}  
 B(Age) -->|Discretize| id4{{Age\_type - 0-4,5-9,...,90 & Over}}  
  
 id3--> |Graph Type|M[[Bar Chart]]  
 id4--> |Graph Type|A[[❤ EDA 1: Population Pyramid]]  
 id4--> |Graph Type|L[[❤ EDA 3: Population Pyramids]]  
  
 H((Age\_Numerical))-->|Graph Type|O[[Histogram]]

# 5. Data Pre-processing

There are 4 major tasks in data pre-processing: Data cleaning, Data Integration, Data Reduction, and Data Transformation. Here we focus on data cleaning, transformation and reduction.

## 5.1 Data Cleaning

Verifying data quality through checking for missing values and duplicates.

### 5.1.1 Missing Values

## The Results

# A tibble: 0 × 6  
# ℹ 6 variables: PA <chr>, SZ <chr>, Age <chr>, Sex <chr>, Pop <dbl>,  
# Time <dbl>

## The Code

sg\_demo\_2024[missing(sg\_demo\_2024),]

### 5.1.2 Duplicates

## The Results

# A tibble: 0 × 6  
# ℹ 6 variables: PA <chr>, SZ <chr>, Age <chr>, Sex <chr>, Pop <dbl>,  
# Time <dbl>

## The Code

sg\_demo\_2024[duplicated(sg\_demo\_2024),]

There were no missing values or duplicates.

## 5.2 Data Transformation/ Reduction

### 5.2.1 Data Label Encoding: Changing Age from categorical to numerical type

Age as a continuous variable will be useful for plotting distributions.

## The Code

sg\_demo\_2024 <- sg\_demo\_2024 %>%  
 mutate(Age\_numerical\_temp = ifelse(Age == "90\_and\_Over", "90", Age)) %>%  
 mutate(Age\_numerical = as.numeric(Age\_numerical\_temp)) %>%  
 select(-Age\_numerical\_temp) # remove the temporary column

### 5.2.2 Discretization: Creation of new Region\_type

The discretization method splits the continuous data into intervals which reduces the data size and for ease of visualising through smaller segments. There are 55 planning area (PA) and 332 subzone (SZ), referencing Urban Redevelopment Authority (URA), the PA or SZ can be grouped into 5 regions:

* Central Region (CR)
* East Region (ER)
* North Region (NR)
* North East Region (NER)
* West Region (WR)

Hence, a new column of Region\_type was created to reflect the 5 segmented regions using PA.

## The Code

# Encoding regions  
sg\_demo\_2024 <- sg\_demo\_2024 %>%   
 mutate(Region\_type = case\_when(  
 `PA` %in% c("Bishan", "Bukit Merah", "Bukit Timah", "Downtown Core", "Geylang",  
 "Kallang","Marina East","Marina South", "Marine Parade", "Museum",  
 "Newton", "Novena", "Orchard","Outram","Queenstown","River Valley",  
 "Rochor","Singapore River","Southern Islands", "Straits View",  
 "Tanglin","Toa Payoh") ~ "CR",  
 `PA` %in% c("Bedok","Changi","Changi Bay","Pasir Ris","Paya Lebar","Tampines") ~ "ER",  
 `PA` %in% c("Central Water Catchment","Lim Chu Kang","Mandai",  
 "Sembawang", "Simpang", "Sungei Kadut","Woodlands","Yishun") ~ "NR",  
 `PA` %in% c("Ang Mo Kio","Hougang", "North-Eastern Islands", "Punggol",   
 "Seletar", "Sengkang", "Serangoon") ~ "NER",  
 `PA` %in% c("Boon Lay", "Bukit Batok", "Bukit Panjang", "Choa Chu Kang",  
 "Clementi", "Jurong East", "Jurong West", "Pioneer", "Tengah",  
 "Tuas", "Western Islands","Western Water Catchment") ~ "WR"  
 ))

These are the summarised data.

## The Data

| Region\_type | Planning\_Area |
| --- | --- |
| CR | Bishan, Bukit Merah, Bukit Timah, Downtown Core, Geylang, Kallang, Marina East, Marina South, Marine Parade, Museum, Newton, Novena, Orchard, Outram, Queenstown, River Valley, Rochor, Singapore River, Southern Islands, Straits View, Tanglin, Toa Payoh |
| ER | Bedok, Changi, Changi Bay, Pasir Ris, Paya Lebar, Tampines |
| NER | Ang Mo Kio, Hougang, North-Eastern Islands, Punggol, Seletar, Sengkang, Serangoon |
| NR | Central Water Catchment, Lim Chu Kang, Mandai, Sembawang, Simpang, Sungei Kadut, Woodlands, Yishun |
| WR | Boon Lay, Bukit Batok, Bukit Panjang, Choa Chu Kang, Clementi, Jurong East, Jurong West, Pioneer, Tengah, Tuas, Western Islands, Western Water Catchment |

## The Code

pa\_mapping<- sg\_demo\_2024 %>%  
 group\_by(`Region\_type`) %>%  
 summarize(Planning\_Area = paste(unique(`PA`), collapse = ", "))  
  
kable(pa\_mapping, title = "Mapping of Planning Area to Region ")

### 5.2.3 Discretization: Creation of new Age\_Type

The ages were binned into Age\_type based on Singapore’s Department of Statistics classification.

## The Code

# Encoding ages  
sg\_demo\_2024 <- sg\_demo\_2024 %>%   
 mutate(Age\_type = case\_when(  
 Age %in% as.character(0:4) ~ "0-4",  
 Age %in% as.character(5:9) ~ "5-9",  
 Age %in% as.character(10:14) ~ "10-14",  
 Age %in% as.character(15:19) ~ "15-19",  
 Age %in% as.character(20:24) ~ "20-24",  
 Age %in% as.character(25:29) ~ "25-29",  
 Age %in% as.character(30:34) ~ "30-34",  
 Age %in% as.character(35:39) ~ "35-39",  
 Age %in% as.character(40:44) ~ "40-44",  
 Age %in% as.character(45:49) ~ "45-49",  
 Age %in% as.character(50:54) ~ "50-54",  
 Age %in% as.character(55:59) ~ "55-59",  
 Age %in% as.character(60:64) ~ "60-64",  
 Age %in% as.character(65:69) ~ "65-69",  
 Age %in% as.character(70:74) ~ "70-74",  
 Age %in% as.character(75:79) ~ "75-79",  
 Age %in% as.character(80:84) ~ "80-84",  
 Age %in% as.character(85:89) ~ "85-89",  
 Age == "90\_and\_Over" ~ "90 & Over",  
 TRUE ~ NA\_character\_  
 ))

### 5.2.4 Discretization: Creation of new summarised\_ages

Based on Singapore Department of Statistics’ classification, elderly are persons aged 65 and older. Based on the Ministry of Manpower the working-age population comprises of persons aged 15 and over. Thus, the remaining younger population are classified as young here.

## The Code

# Encoding summarised\_ages  
sg\_demo\_2024 <- sg\_demo\_2024 %>%   
 mutate(summarised\_ages = case\_when(  
 Age %in% as.character(0:14) ~ "young",  
 Age %in% as.character(15:64) ~ "working-age",  
 Age %in% as.character(65:89) ~ "eldery",  
 Age == "90\_and\_Over" ~ "eldery",  
 TRUE ~ NA\_character\_  
 ))

### 5.2.5 Discretization: Creation of new pop\_by\_gender\_age dataframe and new total\_pop variable

There is no need to keep Time, PA, SZ, and Age variables. This helps in data or dimension reduction and reduces computation power.

## The Code

# Using groupby() and summarise   
pop\_by\_gender\_age<- sg\_demo\_2024 %>%  
 group\_by(Sex, Age\_numerical, Age\_type, Region\_type)%>%  
 summarize(total\_pop = sum(Pop), .groups = "drop")

## 5.3 Preview of reduced dataframe

Dataframes *pop\_by\_gender\_age* and the original *sg\_demo\_2024* were used to plot graphs in Section 7 and 8.

### 5.3.1 Composition of *pop\_by\_gender\_age* dataframe

glimpse(pop\_by\_gender\_age)

Rows: 910  
Columns: 5  
$ Sex <chr> "Females", "Females", "Females", "Females", "Females", "…  
$ Age\_numerical <dbl> 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3,…  
$ Age\_type <chr> "0-4", "0-4", "0-4", "0-4", "0-4", "0-4", "0-4", "0-4", …  
$ Region\_type <chr> "CR", "ER", "NER", "NR", "WR", "CR", "ER", "NER", "NR", …  
$ total\_pop <dbl> 3040, 2400, 3520, 2260, 3150, 3430, 2650, 3780, 2350, 33…

* The dataframe has 5 columns and 910 rows. Three of the columns contains categorical data type and the other two are in double/real number data type.
* The categorical attributes are: Region\_type, Age\_type, and Sex
* The numerical attributes are: Age\_numerical, and total\_pop.

### 5.3.2 Understanding the distribution of numerical values

The median age here is computed by it’s Age\_numerical column and is not a representation of the entire population.

## The Data

|  | Age\_numerical | total\_pop |
| --- | --- | --- |
|  | Min. : 0 | Min. : 220 |
|  | 1st Qu.:22 | 1st Qu.:3340 |
|  | Median :45 | Median :4625 |
|  | Mean :45 | Mean :4608 |
|  | 3rd Qu.:68 | 3rd Qu.:6190 |
|  | Max. :90 | Max. :9100 |

## The Code

realis\_num2 <- pop\_by\_gender\_age[,c("Age\_numerical", "total\_pop")]  
   
kable(summary(realis\_num2))

# 6. Univariate/ Bivariate Analysis

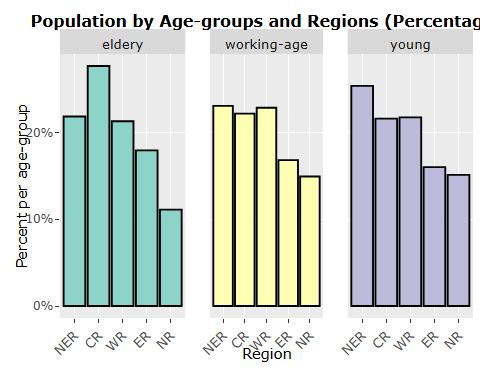
Here, we produce some univariate and bivariate analysis to obtain a general understanding of the data and also compare between and choose visualisations.

## 6.1 Bar Chart of Region by Different Population Age Groups

The bar chart hows the percentage and absolute figures of the different age group of residents by region

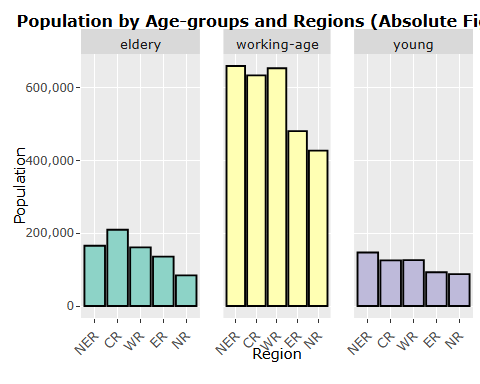
## The Code for Bar Chart on Percentage Figures

# Using groupby() and summarise   
pop\_by\_region<- sg\_demo\_2024 %>%  
 group\_by(summarised\_ages, Region\_type) %>%  
 summarize(pop\_region = sum(Pop), .groups = "drop")  
  
# Calculate percentages by summarised\_ages  
pop\_by\_region\_percent <- pop\_by\_region %>%  
 group\_by(summarised\_ages) %>%  
 mutate(percent = pop\_region / sum(pop\_region)) %>%  
 ungroup()  
  
p1 <- ggplot(pop\_by\_region\_percent,  
 aes(x = reorder(Region\_type, -pop\_region), y = percent,  
 fill = summarised\_ages, # Keep fill for visual distinction  
 text = paste("Age Group:", summarised\_ages, "<br>",  
 "Region:", Region\_type, "<br>",  
 "Percent:", scales::percent(percent, accuracy = 0.1), "<br>",  
 "Population:", scales::comma(pop\_region)))) +  
 geom\_bar(stat = "identity", color = "black") +  
 theme(plot.title = element\_text(hjust = 0.5, size = 12, face = "bold"),  
 axis.text.x = element\_text(angle = 45, hjust = 1, vjust = 1, size = 10),  
 axis.title.x = element\_text(vjust = -1),  
 strip.text = element\_text(size = 10),  
 panel.spacing.x = unit(1, "lines"),  
 legend.position = "none") +   
 labs(x = "Region",  
 y = "Percent per age-group",  
 title = "Population by Age-groups and Regions (Percentage)") +  
 scale\_y\_continuous(labels = scales::percent) +  
 facet\_wrap(vars(summarised\_ages), nrow = 1, scales = "free\_x") +  
 scale\_fill\_brewer(palette = "Set3") # Use a color palette for regions  
  
# Create an interactive plotly object with hover  
library(plotly)  
ggplotly(p1, tooltip = "text") %>%  
 style(hoverlabel = list(bgcolor = "blue"))



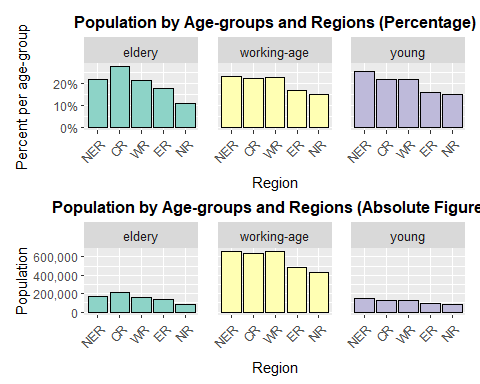
## The Code for Bar Chart on Absolute Figures

pop\_by\_region <- sg\_demo\_2024 %>%  
 group\_by(summarised\_ages, Region\_type) %>%  
 summarize(pop\_region = sum(Pop), .groups = "drop")  
  
# Plot the absolute population figures  
p2 <- ggplot(pop\_by\_region,  
 aes(x = reorder(Region\_type, -pop\_region), y = pop\_region,  
 fill = summarised\_ages, # Fill by region for distinction  
 text = paste("Age Group:", summarised\_ages, "<br>",  
 "Region:", Region\_type, "<br>",  
 "Population:", scales::comma(pop\_region)))) +  
 geom\_bar(stat = "identity", color = "black") +  
 theme(plot.title = element\_text(hjust = 0.5, size = 12, face = "bold"),  
 axis.text.x = element\_text(angle = 45, hjust = 1, vjust = 1, size = 10),  
 axis.title.x = element\_text(vjust = -1),  
 strip.text = element\_text(size = 10),  
 panel.spacing.x = unit(1, "lines"),  
 legend.position = "none") +  
 labs(x = "Region",  
 y = "Population",  
 title = "Population by Age-groups and Regions (Absolute Figures)",  
 fill = "Region") + # Update legend title  
 scale\_y\_continuous(labels = scales::comma) +  
 facet\_wrap(vars(summarised\_ages), nrow = 1, scales = "free\_x") +  
 scale\_fill\_brewer(palette = "Set3") # Use a color palette for regions  
  
# Create an interactive plotly object with hover  
ggplotly(p2, tooltip = "text") %>%  
 style(hoverlabel = list(bgcolor = "blue"))



## The Combined Bar Charts

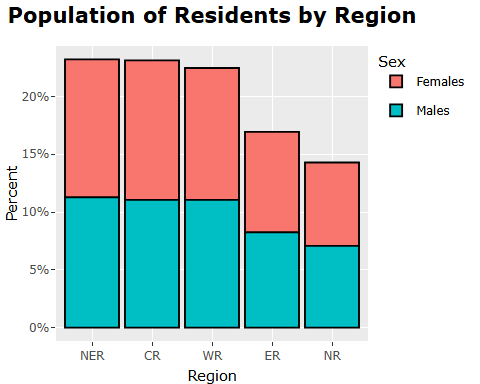
p1/p2



* There are less residents in the East and North regions, relative to other other regions.
* Most of the population are in the working-age group due to the larger age range (15-64).
* Among the elderly population, there are significantly more elderly in the central region, compared to other regions. There are less elderly in the northern region, relative to other regions.
* Among the younger population, there is a younger population in the north eastern region relative to other regions.
* The population in the working age group are close to evenly distributed in the 3 regions of North-Eastern, Central and Western Regions of Singapore.

## 6.2 Stacked Bar chart of Region by Population (with gender as hue)

## The Stacked Bar Chart



## The Code for Stacked bar chart

# Using groupby() and summarise   
pop\_by\_region<- sg\_demo\_2024 %>%  
 group\_by(Region\_type,Sex) %>%  
 summarize(pop\_region = sum(Pop), .groups = "drop")  
  
# plot the distribution as percentages  
p4<-ggplot(pop\_by\_region,   
 aes(x = reorder(Region\_type, -pop\_region), y = pop\_region/sum(pop\_region),  
 fill=Sex,  
 text = paste("Population:", pop\_region,  
 "\nPercent (%):", round(100 \*pop\_region / sum(pop\_region), digits = 2),  
 "\nGender:", Sex)))+   
 geom\_bar(stat = "identity", color = "black") +   
 theme(plot.title = element\_text(hjust = 0.5, size = 16, face = "bold")) +   
 labs(x = "Region",   
 y = "Percent",   
 title = "Population of Residents by Region") +  
 scale\_y\_continuous(labels = scales::percent)  
ggplotly(p4, tooltip = "text")%>%  
 style(  
 hoverlabel = list(bgcolor = "blue")  
 )

* There is a rather even distribution of gender in each region.
* ER and NR have lesser population compared to the other regions, as confirmed in section 6.1.

# 7. Data Preparation and Prototyping

## 7.1 Population Pyramid

Instead of the histogram produced in the earlier univariate/ bivariate analysis, a population pyramid can be produced for clearer visualisation of **Age and Gender Structure.**

#### Step 1: Preparatory work

## The Code to create pyramid- run later

pop\_by\_gender\_age <-  
 pop\_by\_gender\_age |>  
 mutate(total\_pop = if\_else(Sex == "Males", total\_pop, -total\_pop))

## The Code to calculate Total Population and Median Age

Check the total population and median ages:

* The total population is around 4,193,530.
* The median age of the population is around 42 years.

# Ensure total\_pop is positive for this calculation  
pop\_for\_median <- pop\_by\_gender\_age %>%  
 mutate(abs\_total\_pop = abs(total\_pop))  
  
# Calculate the cumulative population  
pop\_cumulative <- pop\_for\_median %>%  
 arrange(Age\_numerical) %>%  
 mutate(cumulative\_pop = cumsum(abs\_total\_pop))  
  
# Find the total population  
total\_population <- sum(pop\_cumulative$abs\_total\_pop)  
print(paste("Total Population:", total\_population))

[1] "Total Population: 4193530"

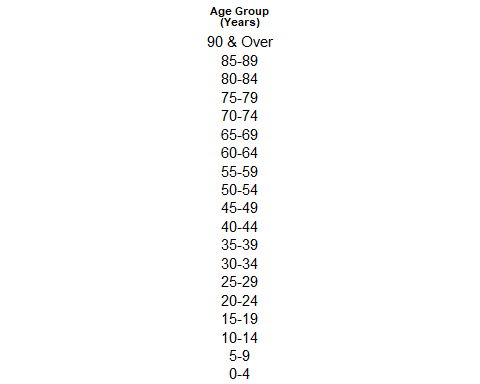
# Determine the median population (half of the total)  
median\_population\_point <- total\_population / 2  
  
# Find the age at which the cumulative population crosses the median point  
median\_age\_population <- pop\_cumulative %>%  
 filter(cumulative\_pop >= median\_population\_point) %>%  
 slice\_head(n = 1) %>%  
 pull(Age\_numerical)  
  
print(paste("Median Age of the Population:", median\_age\_population))

[1] "Median Age of the Population: 42"

## The Code to create Age labels and its plot

Age labels were created for ordered age.

# Creation of age labels  
age\_labels <-  
 tibble(  
 Age\_type = c(  
 "0-4",  
 "5-9",  
 "10-14",  
 "15-19",  
 "20-24",  
 "25-29",  
 "30-34",  
 "35-39",  
 "40-44",  
 "45-49",  
 "50-54",  
 "55-59",  
 "60-64",  
 "65-69",  
 "70-74",  
 "75-79",  
 "80-84",  
 "85-89",  
 "90 & Over"  
 )  
 ) |>  
 mutate(  
 Age\_type = fct\_inorder(Age\_type)  
 )  
  
# age\_labels plot only  
age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text() +  
 theme\_void()+  
 ggtitle("Age Group\n(Years)")+  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size=8,  
 face = "bold", color = "black"))



## The Code to create of new column Age\_Type\_Ordered

To ensure that the Age\_type column in pop\_by\_gender\_age is a factor with the levels taken directly from age\_labels$Age\_type, new column Age\_Type\_Ordered is created.

# Creation of ordered factor  
pop\_by\_gender\_age <- pop\_by\_gender\_age %>%  
 mutate(Age\_Type\_Ordered = factor(Age\_type, levels = age\_labels$Age\_type))  
print(levels(pop\_by\_gender\_age$Age\_Type\_Ordered))

[1] "0-4" "5-9" "10-14" "15-19" "20-24" "25-29"   
 [7] "30-34" "35-39" "40-44" "45-49" "50-54" "55-59"   
[13] "60-64" "65-69" "70-74" "75-79" "80-84" "85-89"   
[19] "90 & Over"

# Double checking factor for Age\_type matches for pop\_by\_gender\_age and age\_labels  
unique\_pop\_age <- unique(pop\_by\_gender\_age$Age\_type)  
print("Unique values in pop\_by\_gender\_age$Age\_type:")

[1] "Unique values in pop\_by\_gender\_age$Age\_type:"

print(unique\_pop\_age)

[1] "0-4" "5-9" "10-14" "15-19" "20-24" "25-29"   
 [7] "30-34" "35-39" "40-44" "45-49" "50-54" "55-59"   
[13] "60-64" "65-69" "70-74" "75-79" "80-84" "85-89"   
[19] "90 & Over"

unique\_labels\_age <- unique(age\_labels$Age\_type)  
print("Unique values in age\_labels$Age\_type:")

[1] "Unique values in age\_labels$Age\_type:"

print(unique\_labels\_age)

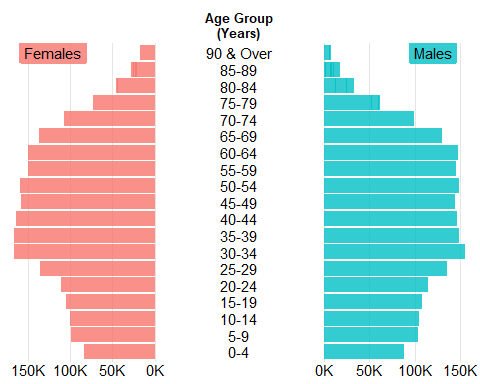
[1] 0-4 5-9 10-14 15-19 20-24 25-29 30-34   
 [8] 35-39 40-44 45-49 50-54 55-59 60-64 65-69   
[15] 70-74 75-79 80-84 85-89 90 & Over  
19 Levels: 0-4 5-9 10-14 15-19 20-24 25-29 30-34 35-39 40-44 45-49 ... 90 & Over

#### Step 2: Plotting of Male + Female + Age Label plots

### Group By For new dataframe

# Using groupby() and summarise   
#popfirt <- pop\_by\_gender\_age %>%  
# group\_by(Sex, Age\_Type\_Ordered) %>%  
# summarize(total\_pop = sum(total\_pop), .groups = "drop")

### Plot of Females and Males



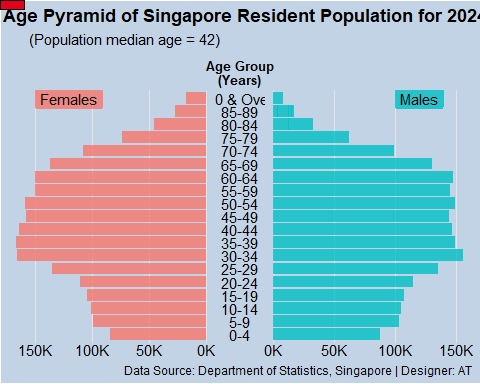
### Code of the plot of Females and Males

# Create Pyramid  
pop\_by\_gender\_age <-  
 pop\_by\_gender\_age |>  
 mutate(total\_pop = if\_else(Sex == "Males", total\_pop, -total\_pop))  
  
#one women plot  
population\_pyramid\_women <-  
 pop\_by\_gender\_age |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = total\_pop,  
 y = Age\_Type\_Ordered   
 )) +  
 geom\_col(fill = "#F8766D",alpha = 0.8) +  
 annotate(  
 geom = "label",  
 x = -120000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 alpha = 0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
)+  
 scale\_x\_continuous(breaks = breaks\_pretty(),   
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"))+  
 theme\_void()+  
 theme(  
 axis.text.x = element\_text(),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
  
# one men plot  
population\_pyramid\_men <-  
 pop\_by\_gender\_age |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = total\_pop,  
 y = Age\_Type\_Ordered   
 )) +  
 geom\_col(fill = "#00BFC4",alpha = 0.8)+  
 annotate(  
 geom = "label",  
 x = 120000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 alpha = 0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
) +  
 scale\_x\_continuous(breaks = breaks\_pretty(),   
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"))+  
 theme\_void()+  
 theme(  
 axis.text.x = element\_text(),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
# Plot age plot  
age\_labels\_plot <-  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text() +  
 theme\_void()+  
 ggtitle("Age Group\n(Years)")+  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size=10,  
 face = "bold", color = "black"))  
# Combine everything (age + tw0 gender plots)  
population\_pyramid\_women +  
 age\_labels\_plot +  
 population\_pyramid\_men

#### Final Step: View the Population Pyramid Plot

After layout adjustments, the population pyramid was plotted.

## Population Pyramid- Adjustments made



## Code for adjustments

p6<-population\_pyramid\_women +  
 age\_labels\_plot +  
 population\_pyramid\_men +  
 plot\_layout(  
 widths = c(4.3, 1, 4.3))+ # both sides are 4.3 times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Singapore Resident Population for 2024",   
 subtitle="(Population median age = 42)",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(face = "bold", hjust = 0.20),   
 plot.subtitle = element\_text( hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"))  
)  
plot(p6)  
  
grid.rect(  
 x = unit(0, "npc"),   
 y = unit(1, "npc"),   
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)

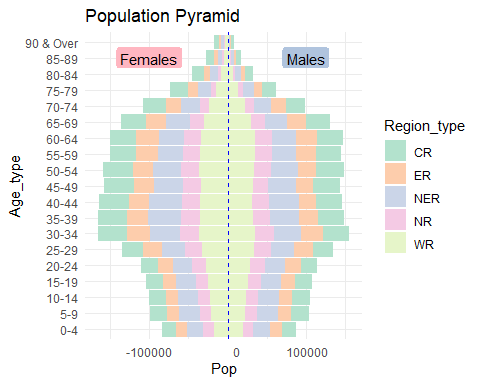
The population pyramid visualises the population’s age and gender distribution. The population size is shown on the x-axis while the age-groups are represented on the middle y-axis. The population numbers are depicted within each age-group bar by gender on the left (Females) and right (Males). The bottom bar represents the youngest age group (0-4) and is incremental in age-group towards the top (90 & Over).

It is possible to visualise the regions within each age group as what [**Michal Palenik**](https://www.iz.sk/en/projects/eu-regions/AT) produced for three employment statuses. However, this method faces the issue of complexity and difficulty reading in our case due to having 5 regions, with the regions close in region size per age-group.

See an example below which visualises the complexity.

### 7.1.1 Comparative Distribution and Structure of Age, Gender, and Population

## Population Pyramid with Region



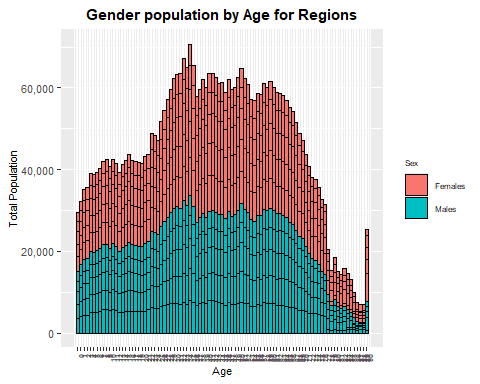
## Code for Population Pyramid with Region

# Order the data by Region\_type within each Age and Gender group.   
sg\_demo\_2024\_reshaped <- sg\_demo\_2024 %>%  
 arrange(Sex, Age\_type, Region\_type)  
  
# ordered factor  
sg\_demo\_2024\_reshaped <- sg\_demo\_2024\_reshaped %>%  
 mutate(Age\_Type\_Ordered2 = factor(Age\_type, levels = age\_labels$Age\_type))  
  
# Create a basic bar chart for one gender  
basic\_plot <- ggplot(sg\_demo\_2024\_reshaped,   
 aes(x = Age\_Type\_Ordered2,   
 y= ifelse(Sex=='Females', -Pop, Pop),   
 fill = Region\_type)) +   
geom\_bar(stat = "identity")   
  
# Create population pyramids for both genders and combine them  
population\_pyramid <- basic\_plot +  
 coord\_flip() +  
 theme\_minimal() +  
 scale\_fill\_brewer(palette = "Pastel2") +  
 labs(  
 x = "Age\_type",  
 y = "Pop",   
 fill = "Region\_type",   
 title = "Population Pyramid") +  
 scale\_y\_continuous(breaks = breaks\_pretty(),  
 labels = function(y) format(y, scientific = FALSE))+  
 geom\_hline(yintercept = 0, color = "blue", linetype = "dashed", size = 0.5) + # Males and Females separation  
 annotate(geom = "label",x = 18, y = -100000,  
 label = "Females", fill = "lightpink",  
 color = "black",label.size = 0)+  
 annotate(geom = "label",x = 18, y = 100000,  
 label = "Males", fill = "lightsteelblue",  
 color = "black",label.size = 0)  
print(population\_pyramid) # Display the plot

## 7.2 Histogram

* A histogram or density plot is usually used to visualise distributions. However, density plots assumes that each row of data is an independent observation of that variable. In our case, the Pop column was a summation instead of a single observation.
* Here, distribution of **Gender population by Age for Regions** is visualised.

## The Histogram

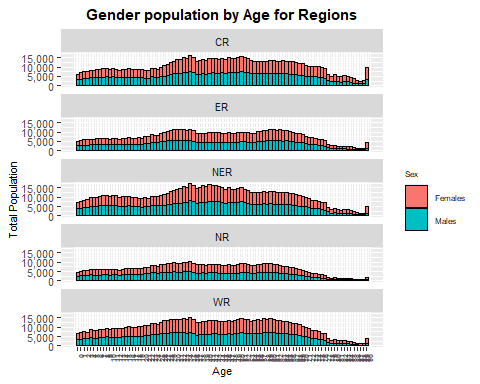


## The Code for Histogram

# Using groupby() and summarise   
pop\_by\_age\_sex\_region<- sg\_demo\_2024 %>%  
 group\_by(Age\_numerical, Sex, Region\_type) %>%  
 summarize(pop\_age = sum(Pop), .groups = "drop")  
  
# Sort ages  
sorted\_ages2 <- sort(unique(pop\_by\_age\_sex\_region$Age\_numerical))  
  
# Plot the age distribution using a histogram  
p7<-ggplot(pop\_by\_age\_sex\_region, aes(x = Age\_numerical,   
 y = pop\_age, fill = Sex,   
 text = paste("Age:", Age\_numerical, "<br>",  
 "Gender:", Sex, "<br>",  
 "Population:", scales::comma(pop\_age)))) +  
 geom\_bar(stat = "identity", color = "black", width = 1) +  
 scale\_x\_continuous(breaks = sorted\_ages2,  
 labels = sorted\_ages2) +  
 labs(title = "Gender population by Age for Regions",  
 x = "Age",  
 y = "Total Population") +  
 scale\_y\_continuous(labels = scales::comma) +   
 theme(axis.text.x = element\_text(angle = 90, hjust = 1, size=5, face = "bold"),  
 plot.title = element\_text(hjust = 0.5, size = 11, face = "bold"),  
 axis.title.x = element\_text(size = 8),  
 axis.title.y = element\_text(size = 8),  
 axis.text.y = element\_text(size = 8),  
 legend.title = element\_text(size = 6),  
 legend.text = element\_text(size = 6),  
 strip.text = element\_text(size = 8))  
  
p7

## Multiple Histograms

p7+ facet\_wrap(vars(Region\_type), nrow=5)



The histogram/ stacked bar chart shows the regional population distribution and regional gender distribution/ structure. It is harder to read and discarded.

# 8. Daily Article



#### Singapore

## **Demographic structures and distribution of Singapore in 2024**

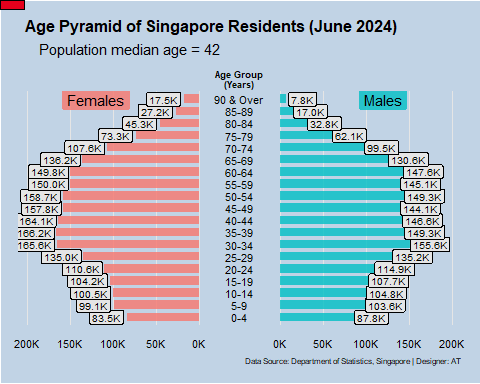
Author/ Graphical Editor: Audrey Tan

Published Date: 03 May 2025

##### In this article, we will cover population pyramids and pyramid graphs. These are useful to visualise the demographic structures and distributions of a country. Data is as at end June.

## 8.1 Population Pyramid

### Population Pyramid



### The Code for Population Pyramid

# Using groupby() and summarise   
popfirt <- pop\_by\_gender\_age %>%  
 group\_by(Sex, Age\_Type\_Ordered) %>%  
 summarize(total\_pop = sum(total\_pop), .groups = "drop")  
  
# --- Plot for Full Region (popfirt) ---  
# Initial plot  
popfirt <-  
 popfirt |>  
 mutate(pop\_sex = if\_else(Sex == "Males", total\_pop, -total\_pop))  
  
#one women plot  
population\_pyramid\_women <-  
 popfirt |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = -pop\_sex ,  
 y = Age\_Type\_Ordered   
 )) +  
 geom\_col(fill = "#F8766D",alpha = 0.8, width=0.7) +  
 geom\_label(  
 aes(x = -pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 3, trim = TRUE), "K")),  
 hjust = 0.7, nudge\_x = -0.1 \* max(abs(popfirt$pop\_sex )),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = -120000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 alpha = 0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(-200000, 0, by = 50000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(-200000,0) # Extend to the left for females  
 ) +  
 theme\_void()+  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
  
# one men plot  
population\_pyramid\_men <-  
 popfirt |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = pop\_sex,  
 y = Age\_Type\_Ordered   
 )) +  
 geom\_col(fill = "#00BFC4",alpha = 0.8, width=0.7)+  
 geom\_label(  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 nudge\_x = ifelse(subset(popfirt, Sex == "Males")$Age\_Type\_Ordered %in% c("90 & Over", "85-89"),  
 0.11 \* max(abs(popfirt$pop\_sex)),  
 0.11 \* max(abs(popfirt$pop\_sex))),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = 120000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 alpha = 0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(0, 200000, by = 50000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(0, 200000) # Extend to the right for males  
 ) +  
 scale\_y\_discrete() +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
   
# Plot age plot  
age\_labels\_plot <-  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text(size=3) +  
 theme\_void()+  
 ggtitle("Age Group\n(Years)")+  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size=7,  
 face = "bold", color = "black"))  
  
  
# Combine everything (age + tw0 gender plots)  
p\_full\_pyr <-population\_pyramid\_women +  
 age\_labels\_plot +  
 population\_pyramid\_men +  
 plot\_layout(  
 widths = c(3, 1, 3)) + # both sides are x times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Singapore Residents (June 2024)",  
 subtitle = "Population median age = 42",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(size = 12, face = "bold", hjust = 0.1),  
 plot.caption = element\_text(size = 6),  
 plot.subtitle = element\_text(hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm")))  
  
plot(p\_full\_pyr)  
  
grid.rect(  
 x = unit(0, "npc"),  
 y = unit(1, "npc"),  
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)

**What is a population pyramid?**

Population pyramids show the **age and gender population distribution**. It is useful for studying the effects of armed conflicts, mortality, birth policies and migration.

**What are the types of population pyramid?**

There are three main types: Triangular shape (Expansive), Bell shape (Stationary), Bulb shape (Constrictive).

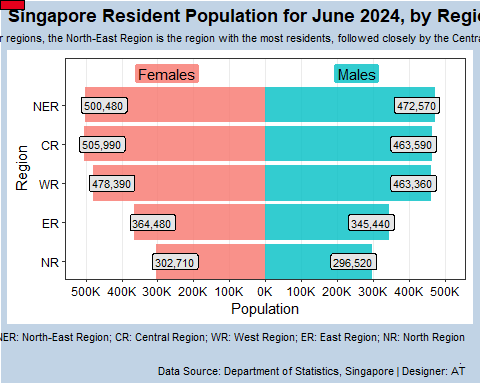
**Insights**

#### **How about Singapore’s population pyramid?**

* Singapore has a Constrictive type in year 2024. It is wider in the middle due to larger representation by the middle-age and elderly, but fewer young people. This is corroborated with the median age at 42.
* The top of the pyramid is wider than other two pyramid types, suggesting longer life expectancy.
* Females appears to outlive males as they aged, as seen from the longer bars. This difference is more prevalent for older age-groups (75s to more than 90s).
* This suggests an ageing population and low fertility.

## 8.2 Regional Distribution

## Regional Population



## The Code for Regional Population

# Using groupby() and summarise   
pop\_by\_sex<- sg\_demo\_2024 %>%  
 group\_by(Sex, Region\_type) %>%  
 summarize(pop\_sex = sum(Pop), .groups = "drop")  
  
# Compute the position of labels  
data <- pop\_by\_sex %>%  
 group\_by(Region\_type) %>% # Group by Region\_type first  
 arrange(desc(Sex)) %>%  
 mutate(prop = pop\_sex / sum(pop\_sex) \* 100) %>% # percentage within each region  
 mutate(ypos = cumsum(prop) - 0.5 \* prop) #y-positions for labels within each region  
   
# Calculate total population per region for ordering  
region\_order <- data %>%  
 group\_by(Region\_type) %>%  
 summarise(total\_region\_pop = sum(abs(pop\_sex))) %>%  
 arrange(total\_region\_pop) %>%  
 pull(Region\_type)  
  
# Order the Region\_type factor  
data$Region\_type <- factor(data$Region\_type, levels = region\_order)  
  
# Plot  
ggplot(data, aes(y = Region\_type, fill = Sex)) +  
 geom\_bar(data = subset(data, Sex == "Males"),  
 aes(x = pop\_sex),  
 stat = "identity", fill = "#00BFC4",alpha = 0.8) +  
 geom\_label(data = subset(data, Sex == "Males"),  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex,   
 scientific = FALSE, big.mark = ","))),  
 hjust = 0.5, nudge\_x = -0.1 \* max(abs(data$pop\_sex)),   
 size = 3, color = "black", fill = "grey90") +  
 geom\_bar(data = subset(data, Sex == "Females"),  
 aes(x = -pop\_sex),  
 stat = "identity", fill = "#F8766D",alpha = 0.8) +  
 geom\_label(data = subset(data, Sex == "Females"),  
 aes(x = -pop\_sex,  
 label = paste0(format(abs(pop\_sex),   
 scientific = FALSE, big.mark = ","))),  
 hjust = 0.5, nudge\_x = 0.1 \* max(abs(data$pop\_sex)),   
 size = 3, color = "black", fill = "grey90") +  
scale\_x\_continuous(  
 breaks = breaks\_pretty(10),  
 labels = function(x) scales::comma(abs(x) / 1000, suffix = "K")  
)+  
 annotate(  
 geom = "label",  
 x = -max(abs(data$pop\_sex)) \* 1,  
 y = length(unique(data$Region\_type)) + 0.8,  
 label = "Females",  
 fill = "#F8766D",  
 alpha = 0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.2, "lines"),  
 hjust=-0.8  
 ) +  
 annotate(  
 geom = "label",  
 x = max(abs(data$pop\_sex)) \* 1,  
 y = length(unique(data$Region\_type)) + 0.8,  
 label = "Males",  
 fill = "#00BFC4",  
 alpha = 0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.2, "lines"),  
 hjust=2.5  
 ) +  
 scale\_y\_discrete(expand = expansion(mult = c(0, 0.3))) +  
 theme\_bw() + # Using a clean theme  
 theme(axis.text = element\_text(colour = "black"),  
 plot.title = element\_text(lineheight = 0.8),  
 panel.grid.major.y = element\_blank(), # Remove horizontal grid lines  
 panel.grid.minor = element\_blank(),  
 legend.position = "none") +  
 labs(y = "Region", x = "Population", fill = "Sex")+  
 plot\_annotation(title = "Singapore Resident Population for June 2024, by Region",  
 subtitle="Relative to other regions, the North-East Region is the region with the most residents, followed closely by the Central Region.",  
 caption = "NER: North-East Region; CR: Central Region; WR: West Region; ER: East Region; NR: North Region  
  
. \n Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(face = "bold", hjust = 0.16),   
 plot.subtitle = element\_text(size=8, hjust = 0.6),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.caption=element\_text(size=8)))  
  
grid.rect(  
 x = unit(0, "npc"),   
 y = unit(1, "npc"),   
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)

**Insights**

#### **What is the regional-gender population distribution?**

* Most residents are clustered in NER, CR, and WR. A smaller segment in ER and NR.
* Closely even distribution of genders per region seen by colours and values.

*Abbreviations-> NER: North-East Region; CR: Central Region; WR: West Region; ER: East Region; NR: North Region*

## 8.3 Region-Age-Gender Distribution

## Code for the creation of age labels

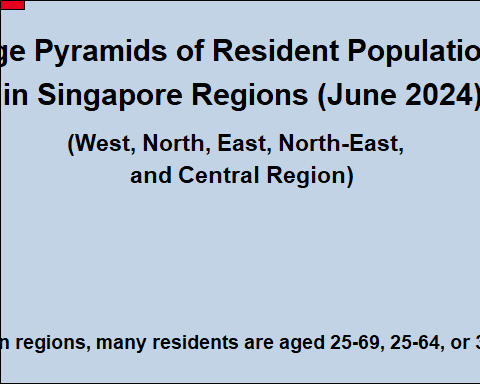
# Creation of age labels  
age\_labels <-  
 tibble(  
 Age\_type = c(  
 "0-4",  
 "5-9",  
 "10-14",  
 "15-19",  
 "20-24",  
 "25-29",  
 "30-34",  
 "35-39",  
 "40-44",  
 "45-49",  
 "50-54",  
 "55-59",  
 "60-64",  
 "65-69",  
 "70-74",  
 "75-79",  
 "80-84",  
 "85-89",  
 "90 & Over"  
 )  
 ) |>  
 mutate(  
 Age\_type = fct\_inorder(Age\_type)  
 )

## Code for creation of blank canvas

# Create a blank plot with a blue background  
p <- ggplot() +  
 theme\_void() + # Clean  
 theme(plot.background = element\_rect(fill = "#C1D3E5"),  
 ) # Set the background   
  
# Add text annotation  
p <- p +  
 annotate(  
 "text",  
 x = 0.5, # x-coordinate of the text (normalized)  
 y = 0.7, # y-coordinate of the text (normalized)  
 label = "Age Pyramids of Resident Population \n in Singapore Regions (June 2024)",  
 color = "black", # Color of the text  
 size = 8, # Size of the text  
 fontface = "bold", # Style of the text  
 hjust = 0.5, # Horizontal justification (0.5 = center)  
 vjust = 1.3 # Vertical justification (0.5 = center)  
 )+  
 annotate(  
 "text",  
 x = 0.5, # x-coordinate of the second text (normalized)  
 y = 0.4, # y-coordinate of the second text (normalized)  
 label = "(West, North, East, North-East, \n and Central Region)",   
 color = "black", # Color of the second text  
 size = 6, # Size of the second text  
 fontface = "bold", # Style of the second text  
 hjust = 0.5, # Horizontal justification (0.5 = center)  
 vjust = -1.9 # Vertical justification (0.5 = center)  
 )+  
 annotate(  
 "text",  
 x = 0.5, # x-coordinate of the second text (normalized)  
 y = 0.3, # y-coordinate of the second text (normalized)  
 label = "Within regions, many residents are aged 25-69, 25-64, or 30-69",  
 color = "black", # Color of the second text  
 size = 5, # Size of the second text  
 fontface = "bold", # Style of the second text  
 hjust = 0.5, # Horizontal justification (0.5 = center)  
 vjust = -1.2 # Vertical justification (0.5 = center)  
 )  
  
# Create the grid.rect grob  
rect\_grob <- grid.rect(  
 x = unit(0, "npc"),  
 y = unit(1, "npc"),  
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)



# Add the grid.rect grob to the plot's gtable  
p <- p +   
 annotation\_custom(  
 grob = rect\_grob,  
 xmin = -Inf, # Extend to the edges of the plot panel  
 xmax = Inf,  
 ymin = -Inf,  
 ymax = Inf  
 )  
  
# Print the plot  
plot(p)

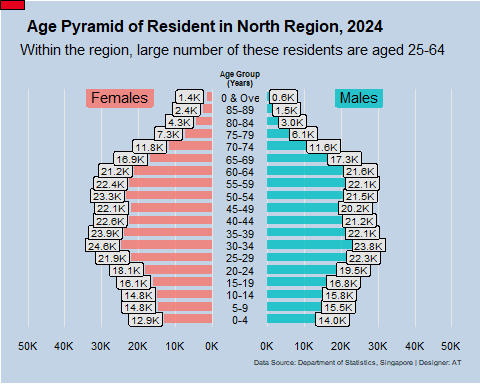


## Code for the creation of each region’s dataframe

# Using groupby() and summarise   
pop\_ner <- pop\_by\_gender\_age %>%  
 filter(Region\_type == "NER") %>%  
 group\_by(Sex, Age\_Type\_Ordered) %>%  
 summarize(pop\_sex = sum(total\_pop), .groups = "drop")  
  
# Using groupby() and summarise   
pop\_er <- pop\_by\_gender\_age %>%  
 filter(Region\_type == "ER") %>%  
 group\_by(Sex, Age\_Type\_Ordered) %>%  
 summarize(pop\_sex = sum(total\_pop), .groups = "drop")  
  
# Using groupby() and summarise   
pop\_wr <- pop\_by\_gender\_age %>%  
 filter(Region\_type == "WR") %>%  
 group\_by(Sex, Age\_Type\_Ordered) %>%  
 summarize(pop\_sex = sum(total\_pop), .groups = "drop")  
  
# Using groupby() and summarise   
pop\_cr <- pop\_by\_gender\_age %>%  
 filter(Region\_type == "CR") %>%  
 group\_by(Sex, Age\_Type\_Ordered) %>%  
 summarize(pop\_sex = sum(total\_pop), .groups = "drop")  
  
# Using groupby() and summarise   
pop\_nr <- pop\_by\_gender\_age %>%  
 filter(Region\_type == "NR") %>%  
 group\_by(Sex, Age\_Type\_Ordered) %>%  
 summarize(pop\_sex = sum(total\_pop), .groups = "drop")

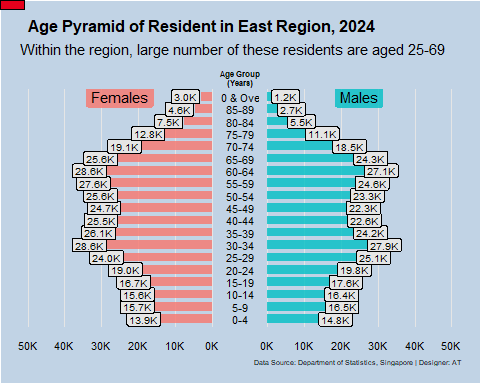
## North Region

# Initial plot  
pop\_nr <-  
 pop\_nr |>  
 mutate(pop\_sex = if\_else(Sex == "Males", pop\_sex, -pop\_sex))  
  
# One women plot  
population\_pyramid\_women <-  
 pop\_nr |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = -pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#F8766D",alpha=0.8, width=0.7) +  
# Add Labelling   
 geom\_label(  
 aes(x = -pop\_sex,  
 label = paste0(format(pop\_sex / 1000,   
 scientific = FALSE, digits = 2, trim = TRUE), "K")  
 ),  
 hjust = 0.8, nudge\_x = -0.1 \* max(abs(pop\_nr$pop\_sex)),   
 size = 2.6, color = "black", fill = "grey90") +  
# Annotations  
 annotate(  
 geom = "label",  
 x = -25000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 color = "black",  
 alpha=0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(-50000, 0, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(-50000,0) # Extend to the left for females  
 ) +  
   
 theme\_void()+  
 theme(  
 axis.text.x = element\_text(size=8,margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
  
# One men plot  
population\_pyramid\_men <-  
 pop\_nr |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#00BFC4",alpha=0.8, width=0.7)+  
 # Add Labelling   
 geom\_label(  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE,  
 digits = 2,  
 trim = TRUE), "K")),  
 nudge\_x = ifelse(subset(pop\_nr, Sex == "Males")$Age\_Type\_Ordered %in% c("90 & Over", "85-89"),  
 0.15 \* max(abs(pop\_nr$pop\_sex)), # those less than 1K   
 0.15 \* max(abs(pop\_nr$pop\_sex))), # others  
 size = 2.6, color = "black", fill = "grey90") +  
 #Annotations  
 annotate(  
 geom = "label",  
 x = 25000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 alpha=0.8,  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(0, 50000, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(0, 50000) # Extend to the right for males  
 ) +  
scale\_y\_discrete()+  
 theme\_void()+  
 theme(  
 axis.text.x = element\_text(size=8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
   
  
# Plot age plot  
age\_labels\_plot <-  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text(size=3) +  
 theme\_void()+  
 ggtitle("Age Group\n(Years)")+  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size=6,  
 face = "bold", color = "black"))  
  
  
p15<-population\_pyramid\_women +  
 age\_labels\_plot +  
 population\_pyramid\_men +  
 plot\_layout(  
 widths = c(10, 1.8, 10))+ # both sides are x times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Resident in North Region, 2024",   
 subtitle="Within the region, large number of these residents are aged 25-64",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(size=12, face = "bold", hjust = 0.1),   
 plot.caption=element\_text(size=5),  
 plot.subtitle = element\_text(hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm")))  
  
plot(p15)  
  
grid.rect(  
 x = unit(0, "npc"),   
 y = unit(1, "npc"),   
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)



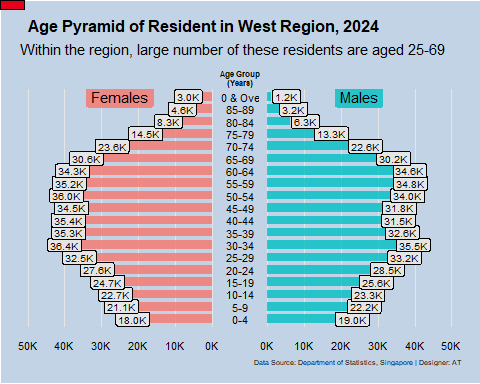
## East Region

library(tidyverse)  
library(scales)  
library(patchwork)  
library(grid)  
  
# --- Plot for East Region (pop\_er) ---  
# Initial plot  
pop\_er <-  
 pop\_er |>  
 mutate(pop\_sex = if\_else(Sex == "Males", pop\_sex, -pop\_sex))  
  
population\_pyramid\_women\_er <-  
 pop\_er |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = -pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#F8766D", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = -pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 hjust = 0.7, nudge\_x = -0.1 \* max(abs(pop\_er$pop\_sex)),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = -25000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(-50000, 0, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(-50000,0) # Extend to the left for females  
 ) +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
population\_pyramid\_men\_er <-  
 pop\_er |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#00BFC4", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 nudge\_x = ifelse(subset(pop\_er, Sex == "Males")$Age\_Type\_Ordered %in% c("90 & Over", "85-89"),  
 0.13 \* max(abs(pop\_er$pop\_sex)),  
 0.13 \* max(abs(pop\_er$pop\_sex))),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = 25000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(0, 50000, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(0, 50000) # Extend to the right for males  
 ) +  
 scale\_y\_discrete() +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
age\_labels\_plot <- # Assuming this is already defined  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text(size = 3) +  
 theme\_void() +  
 ggtitle("Age Group\n(Years)") +  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size = 6,  
 face = "bold", color = "black"))  
  
p\_er <- population\_pyramid\_women\_er +  
 age\_labels\_plot +  
 population\_pyramid\_men\_er +  
 plot\_layout(  
 widths = c(10, 1.8, 10)) + # both sides are x times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Resident in East Region, 2024",  
 subtitle = "Within the region, large number of these residents are aged 25-69",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(size = 12, face = "bold", hjust = 0.1),  
 plot.caption = element\_text(size = 5),  
 plot.subtitle = element\_text(hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm")))  
  
plot(p\_er)  
  
grid.rect(  
 x = unit(0, "npc"),  
 y = unit(1, "npc"),  
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)



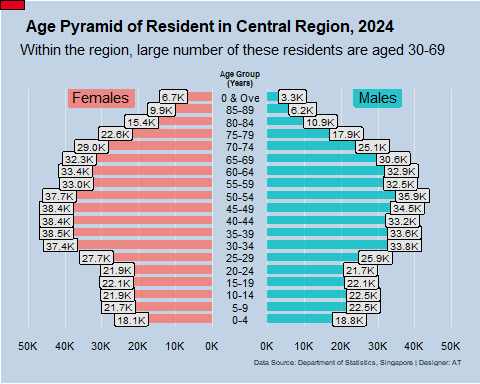
## West Region

# --- Plot for West Region (pop\_wr) ---  
# Initial plot  
pop\_wr <-  
 pop\_wr |>  
 mutate(pop\_sex = if\_else(Sex == "Males", pop\_sex, -pop\_sex))  
  
population\_pyramid\_women\_wr <-  
 pop\_wr |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = -pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#F8766D", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = -pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 hjust = 0.5, nudge\_x = -0.1 \* max(abs(pop\_wr$pop\_sex)),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = -25000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(-50000, 0, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(-50000,0) # Extend to the left for females  
 ) +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
population\_pyramid\_men\_wr <-  
 pop\_wr |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#00BFC4", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 nudge\_x = ifelse(subset(pop\_wr, Sex == "Males")$Age\_Type\_Ordered %in% c("90 & Over", "85-89"),  
 0.11 \* max(abs(pop\_wr$pop\_sex)),  
 0.11 \* max(abs(pop\_wr$pop\_sex))),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = 25000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(0, 50000, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(0, 50000) # Extend to the right for males  
 ) +  
 scale\_y\_discrete() +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
age\_labels\_plot <- # Assuming this is already defined  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text(size = 3) +  
 theme\_void() +  
 ggtitle("Age Group\n(Years)") +  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size = 6,  
 face = "bold", color = "black"))  
  
p\_wr <- population\_pyramid\_women\_wr +  
 age\_labels\_plot +  
 population\_pyramid\_men\_wr +  
 plot\_layout(  
 widths = c(10, 1.8, 10)) + # both sides are x times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Resident in West Region, 2024",  
 subtitle = "Within the region, large number of these residents are aged 25-69",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(size = 12, face = "bold", hjust = 0.1),  
 plot.caption = element\_text(size = 5),  
 plot.subtitle = element\_text(hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm")))  
  
plot(p\_wr)  
  
grid.rect(  
 x = unit(0, "npc"),  
 y = unit(1, "npc"),  
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)



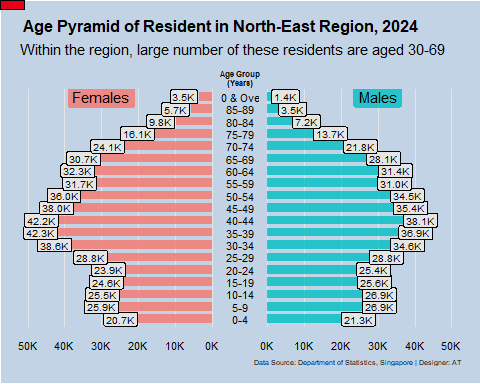
## Central Region

# --- Plot for Central Region (pop\_cr) ---  
# Initial plot  
pop\_cr <-  
 pop\_cr |>  
 mutate(pop\_sex = if\_else(Sex == "Males", pop\_sex, -pop\_sex))  
  
  
population\_pyramid\_women\_cr <-  
 pop\_cr |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = -pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#F8766D", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = -pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 hjust = 0.5, nudge\_x = -0.1 \* max(abs(pop\_cr$pop\_sex)),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = -30000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = breaks\_pretty(4),  
 labels = function(x) scales::comma(abs(x) / 1000, suffix = "K"),  
 limits = c(-max(abs(pop\_cr$pop\_sex)) \* 1.3, 0)) +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
population\_pyramid\_men\_cr <-  
 pop\_cr |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#00BFC4", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 nudge\_x = ifelse(subset(pop\_cr, Sex == "Males")$Age\_Type\_Ordered %in% c("90 & Over", "85-89"),  
 0.09 \* max(abs(pop\_cr$pop\_sex)),  
 0.09 \* max(abs(pop\_cr$pop\_sex))),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = 30000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = breaks\_pretty(4),  
 labels = function(x) scales::comma(abs(x) / 1000, suffix = "K"),  
 limits = c(0, max(abs(pop\_cr$pop\_sex)) \* 1.3)) +  
 scale\_y\_discrete() +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
age\_labels\_plot <- # Assuming this is already defined  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text(size = 3) +  
 theme\_void() +  
 ggtitle("Age Group\n(Years)") +  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size = 6,  
 face = "bold", color = "black"))  
  
p\_cr <- population\_pyramid\_women\_cr +  
 age\_labels\_plot +  
 population\_pyramid\_men\_cr +  
 plot\_layout(  
 widths = c(10, 1.8, 10)) + # both sides are x times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Resident in Central Region, 2024",  
 subtitle = "Within the region, large number of these residents are aged 30-69",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(size = 12, face = "bold", hjust = 0.1),  
 plot.caption = element\_text(size = 5),  
 plot.subtitle = element\_text(hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm")))  
  
plot(p\_cr)  
  
grid.rect(  
 x = unit(0, "npc"),  
 y = unit(1, "npc"),  
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)



## North East Region

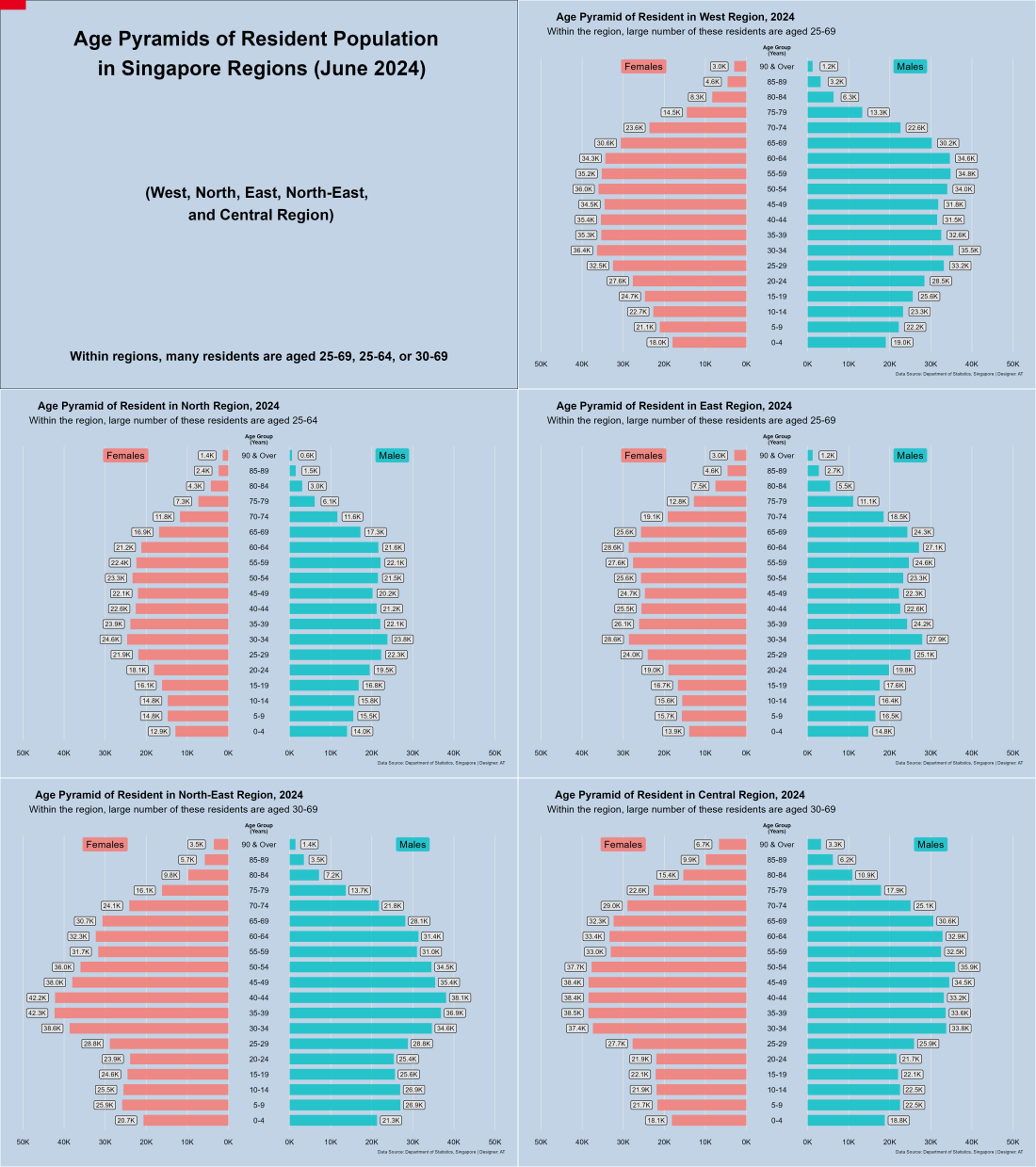
# --- Plot for North East Region (pop\_ner) ---  
# Initial plot  
pop\_ner <-  
 pop\_ner |>  
 mutate(pop\_sex = if\_else(Sex == "Males", pop\_sex, -pop\_sex))  
  
  
population\_pyramid\_women\_ner <-  
 pop\_ner |>  
 filter(Sex == "Females") |>  
 ggplot(aes(  
 x = -pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#F8766D", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = -pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 hjust = 0.5, nudge\_x = -0.1 \* max(abs(pop\_ner$pop\_sex)),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = -30000,  
 y = 19,  
 label = "Females",  
 fill = "#F8766D",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(-50000, 0, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(-50000,0) # Extend to the left for females  
 ) +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
population\_pyramid\_men\_ner <-  
 pop\_ner |>  
 filter(Sex == "Males") |>  
 ggplot(aes(  
 x = pop\_sex,  
 y = Age\_Type\_Ordered  
 )) +  
 geom\_col(fill = "#00BFC4", alpha = 0.8, width = 0.7) +  
 geom\_label(  
 aes(x = pop\_sex,  
 label = paste0(format(pop\_sex / 1000,  
 scientific = FALSE, digits = 2, trim = TRUE), "K")),  
 nudge\_x = ifelse(subset(pop\_ner, Sex == "Males")$Age\_Type\_Ordered %in% c("90 & Over", "85-89"),  
 0.08 \* max(abs(pop\_ner$pop\_sex)),  
 0.08 \* max(abs(pop\_ner$pop\_sex))),  
 size = 2.6, color = "black", fill = "grey90"  
 ) +  
 annotate(  
 geom = "label",  
 x = 30000,  
 y = 19,  
 label = "Males",  
 fill = "#00BFC4",  
 color = "black",  
 alpha = 0.8,  
 label.size = 0,  
 label.padding = unit(0.3, "lines")  
 ) +  
 scale\_x\_continuous(breaks = seq(0, 50000, by = 10000),  
 labels=function(x) scales::comma(abs(x)/ 1000, suffix = "K"),  
 limits = c(0, 50000) # Extend to the right for males  
 ) +  
 scale\_y\_discrete() +  
 theme\_void() +  
 theme(  
 axis.text.x = element\_text(size = 8, margin = margin(t = 10)),  
 panel.grid.major.x = element\_line(color = "grey90")  
 )  
  
age\_labels\_plot <- # Assuming this is already defined  
 age\_labels |>  
 ggplot(  
 aes(  
 x = 1,  
 y = Age\_type,  
 label = Age\_type  
 )  
 ) +  
 geom\_text(size = 3) +  
 theme\_void() +  
 ggtitle("Age Group\n(Years)") +  
 theme(plot.title = element\_text(hjust = 0.5, vjust = 2, size = 6,  
 face = "bold", color = "black"))  
  
p\_ner <- population\_pyramid\_women\_ner +  
 age\_labels\_plot +  
 population\_pyramid\_men\_ner +  
 plot\_layout(  
 widths = c(10, 1.8, 10)) + # both sides are x times the age\_label\_plot  
 plot\_annotation(  
 title = "Age Pyramid of Resident in North-East Region, 2024",  
 subtitle = "Within the region, large number of these residents are aged 30-69",  
 caption = "Data Source: Department of Statistics, Singapore | Designer: AT",  
 theme = theme(plot.title = element\_text(size = 12, face = "bold", hjust = 0.1),  
 plot.caption = element\_text(size = 5),  
 plot.subtitle = element\_text(hjust = 0.08),  
 plot.background = element\_rect(fill = "#C1D3E5"),  
 plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm")))  
  
plot(p\_ner)  
  
grid.rect(  
 x = unit(0, "npc"),  
 y = unit(1, "npc"),  
 width = unit(0.05, "npc"),  
 height = unit(0.025, "npc"),  
 just = c("left", "top"),  
 gp = gpar(fill = "#e5001c", lwd = 0)  
)



## The Final Combined Plot

# 2. Save the individual plots as image files  
ggsave("p\_title.png", p, width = 8, height = 6, units = "in", dpi = 300)  
ggsave("p\_nr.png", p15, width = 8, height = 6, units = "in", dpi = 300)  
ggsave("p\_er.png", p\_er, width = 8, height = 6, units = "in", dpi = 300)  
ggsave("p\_wr.png", p\_wr, width = 8, height = 6, units = "in", dpi = 300)  
ggsave("p\_ner.png", p\_ner, width = 8, height = 6, units = "in", dpi = 300)  
ggsave("p\_cr.png", p\_cr, width = 8, height = 6, units = "in", dpi = 300)  
  
# 3. Read the image files using magick  
image\_p\_title <- image\_read("p\_title.png")  
image\_nr <- image\_read("p\_nr.png")  
image\_er <- image\_read("p\_er.png")  
image\_wr <- image\_read("p\_wr.png")  
image\_ner <- image\_read("p\_ner.png")  
image\_cr <- image\_read("p\_cr.png")  
  
# 4. Scale by width  
target\_width = 550  
image\_p\_title <- image\_scale(image\_p\_title, geometry = paste0(target\_width, "x"))  
image\_nr <- image\_scale(image\_nr, geometry = paste0(target\_width, "x"))  
image\_er <- image\_scale(image\_er, geometry = paste0(target\_width, "x"))  
image\_wr <- image\_scale(image\_wr, geometry = paste0(target\_width, "x"))  
image\_ner <- image\_scale(image\_ner, geometry = paste0(target\_width, "x"))  
image\_cr <- image\_scale(image\_cr, geometry = paste0(target\_width, "x"))  
  
# 5. Create the rows  
row1 <- image\_append(c(image\_p\_title, image\_wr), stack = FALSE)  
row2 <- image\_append(c(image\_nr, image\_er), stack = FALSE)  
row3 <- image\_append(c(image\_ner, image\_cr), stack = FALSE)  
  
# 6. Stack the rows vertically  
combined\_image <- image\_append(c(row1, row2, row3), stack = TRUE)  
  
# 7. Save the combined image  
image\_write(combined\_image, "combined\_pyramids\_2x3.png")  
  
# 8. Display the combined image in R  
print(combined\_image)

# A tibble: 1 × 7  
 format width height colorspace matte filesize density  
 <chr> <int> <int> <chr> <lgl> <int> <chr>   
1 PNG 1100 1239 sRGB TRUE 0 118x118



**Insights**

#### **Gender, Age and Regional Distributions**

* Wider distribution suggests greater diversity in the population per gender/ region:
  + Females have significantly wider distributions than Males for NER, CR, WR.
  + Regions NER, CR, WR have wider distributions than ER and NR.
* Across ages, wider gap between both genders’ population count observed for NER, CR, and WR, compared to ER and NR.
* Rising numbers from age 0- 35, plateauing around age 35- 60, then dips.
* Around the ages of 0- 30, the population values between genders are close.
* A large population around age 30- 69 suggests an older population.
* After age 75, a wider population gap is observed between the higher females compared to males. This is stark for “over-90-years” (especially for CR), suggesting longevity for older females.
* Distribution shape similar within (NER and CR), and (NR, ER, WR).

# 9. Summary and Future Research

## 9.1 Summary

Based on the EDA above, we can conclude the following about the demographic distribution and structure of Singapore in 2024:

1. **Age and Gender Distribution:** The population pyramid displays age-group variation in population. There is an ageing population with residents at a median age of 42. There are lesser younger and elderly from low birth rates and decline due to old-age mortality. There were increasingly more Females compared to Males as the age-group increases, suggesting Females outliving Males which corroborates with general biological and behavioural reasons.
2. **Age and Gender Distribution, for regions**: There is a rather even distribution of genders per region. Also, most residents are clustered in NER, CR and WR. Across different ages, a wider gap between the population count for both genders was observed for NER, CR, and WR as compared to ER and NR. Distribution shape similar within (NER and CR), and (NR, ER, WR).

## 9.2 Future Research

* Further research can be done on the regions where the labour force (age 15 and over) reside in. This might be useful for urban planning of offices.
* With more data, labour force participation rate per region can be obtained by zooming into the working age groups (15-24, 25-54, 55-64, 65 and over).

# 10. Comments and Edits

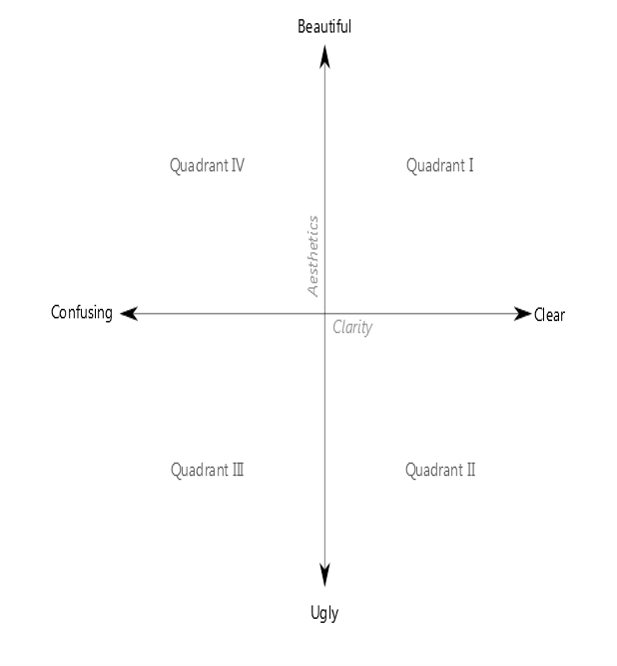
## 10.1 Background and Task

In Phase 2 of the take home exercise, one submission by a fellow classmate is selected. Three good design principles and three areas for further improvement are provided. The makeover version of the data visualisation is prepared based on the suggestions from the quadrants of achieving both clarity and aesthetics, the components of a graph, having a well-worded title and the right visualisation.

## Clarity or Aesthetics?

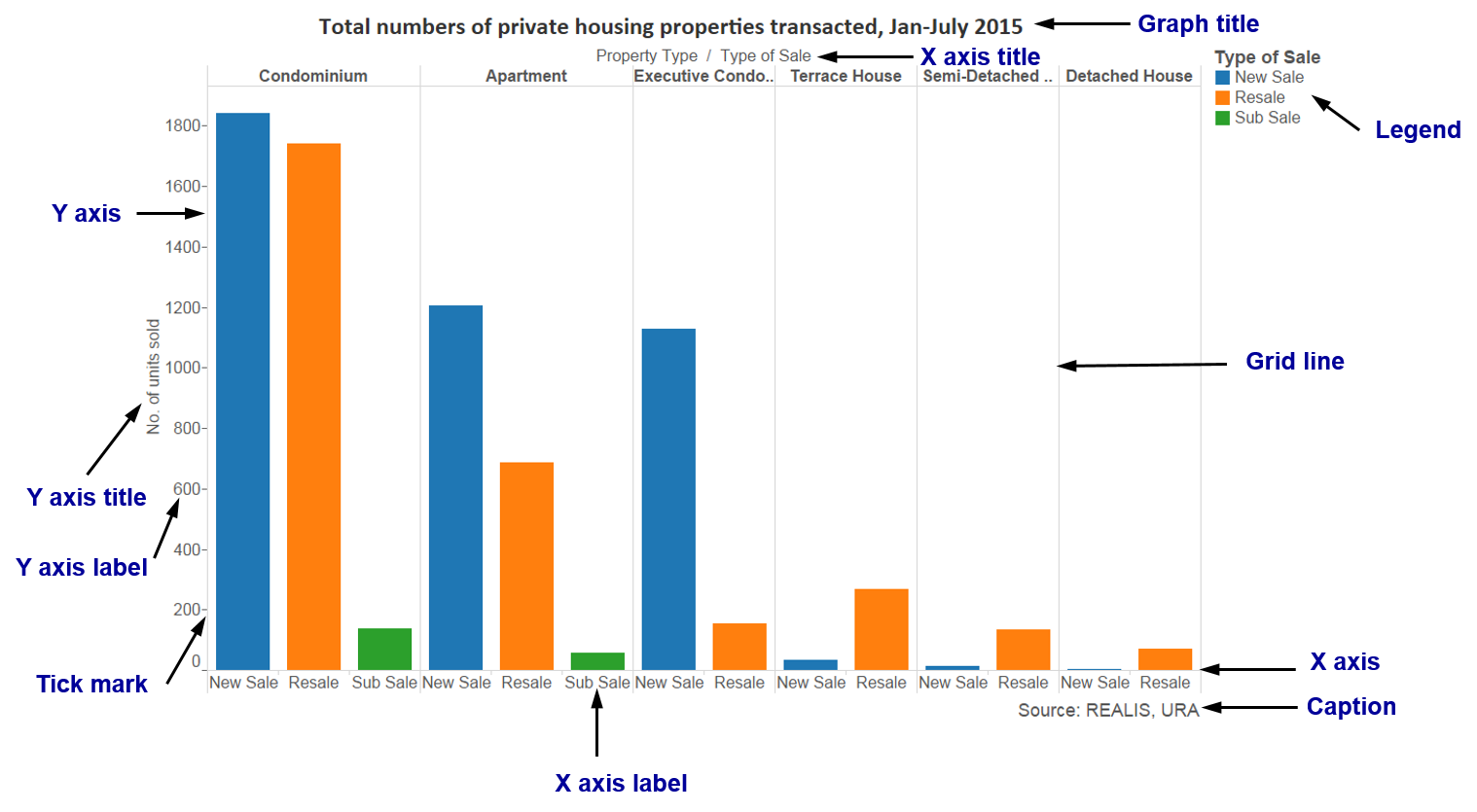
These are what the 4 quadrants measure:

* Quadrant 1: Beautiful and Clear
* Quadrant 2: Clear and Ugly
* Quadrant 3: Ugly and Confusing
* Quadrant 4: Beautiful and Confusing



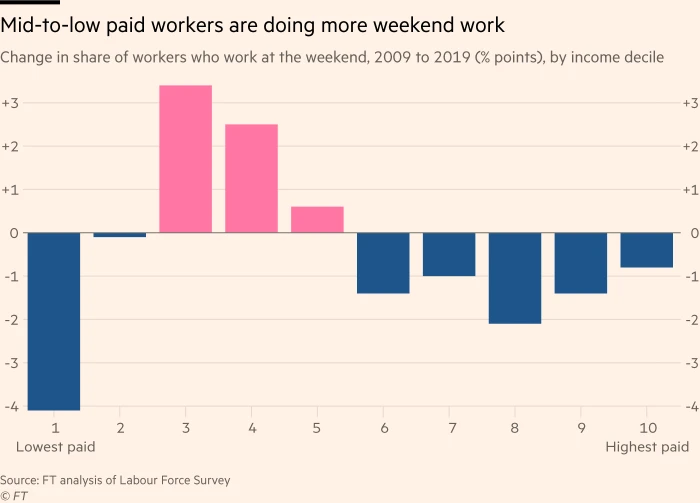
## Components of a Graph

These are the different components of a graph pointed out by the blue annotations.



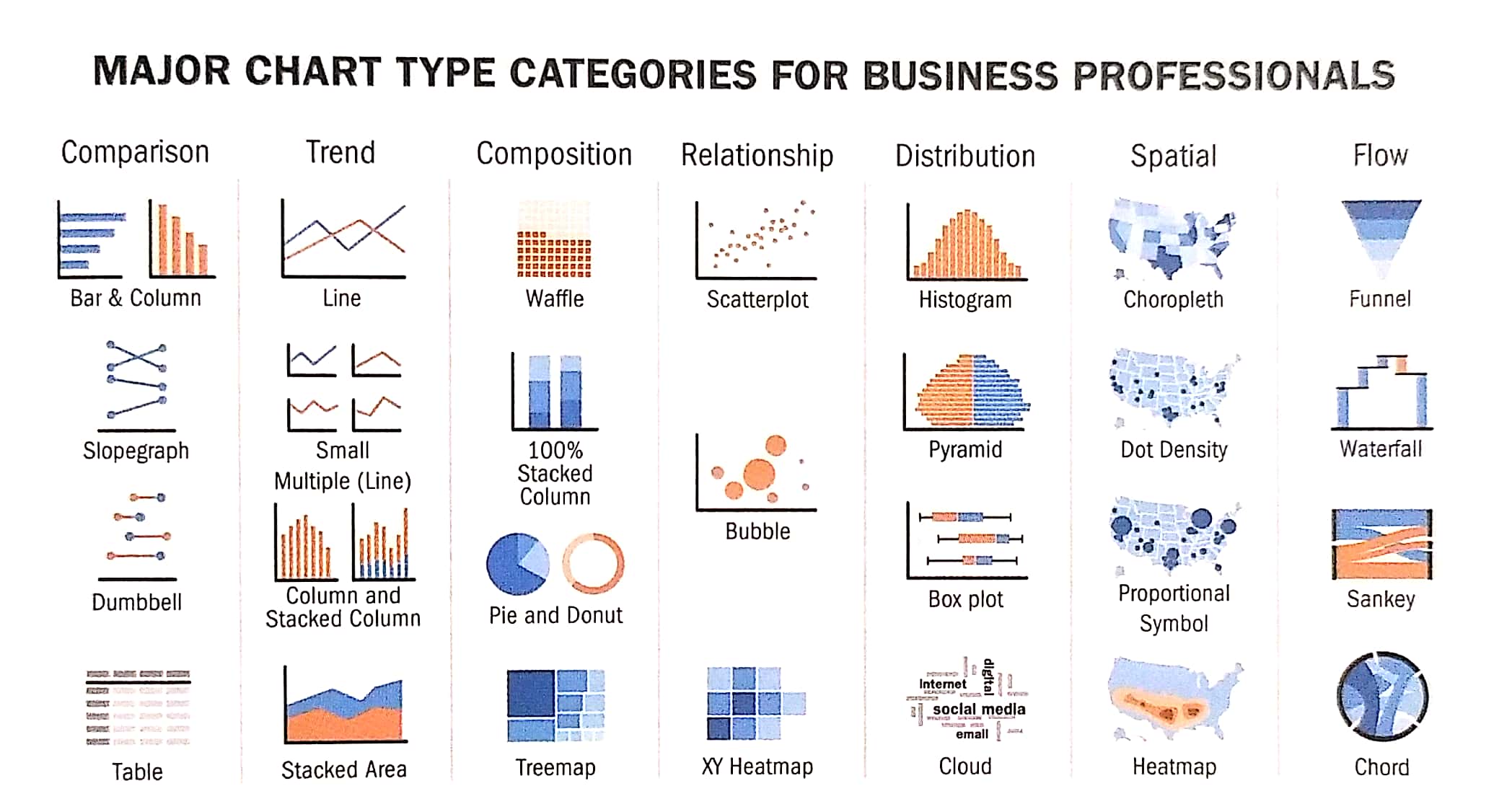
## Well-worded title

There is a title in bold followed by subtitles. They provide the background to the chart and assist to tell a data story. There is also a caption to cite the data source.

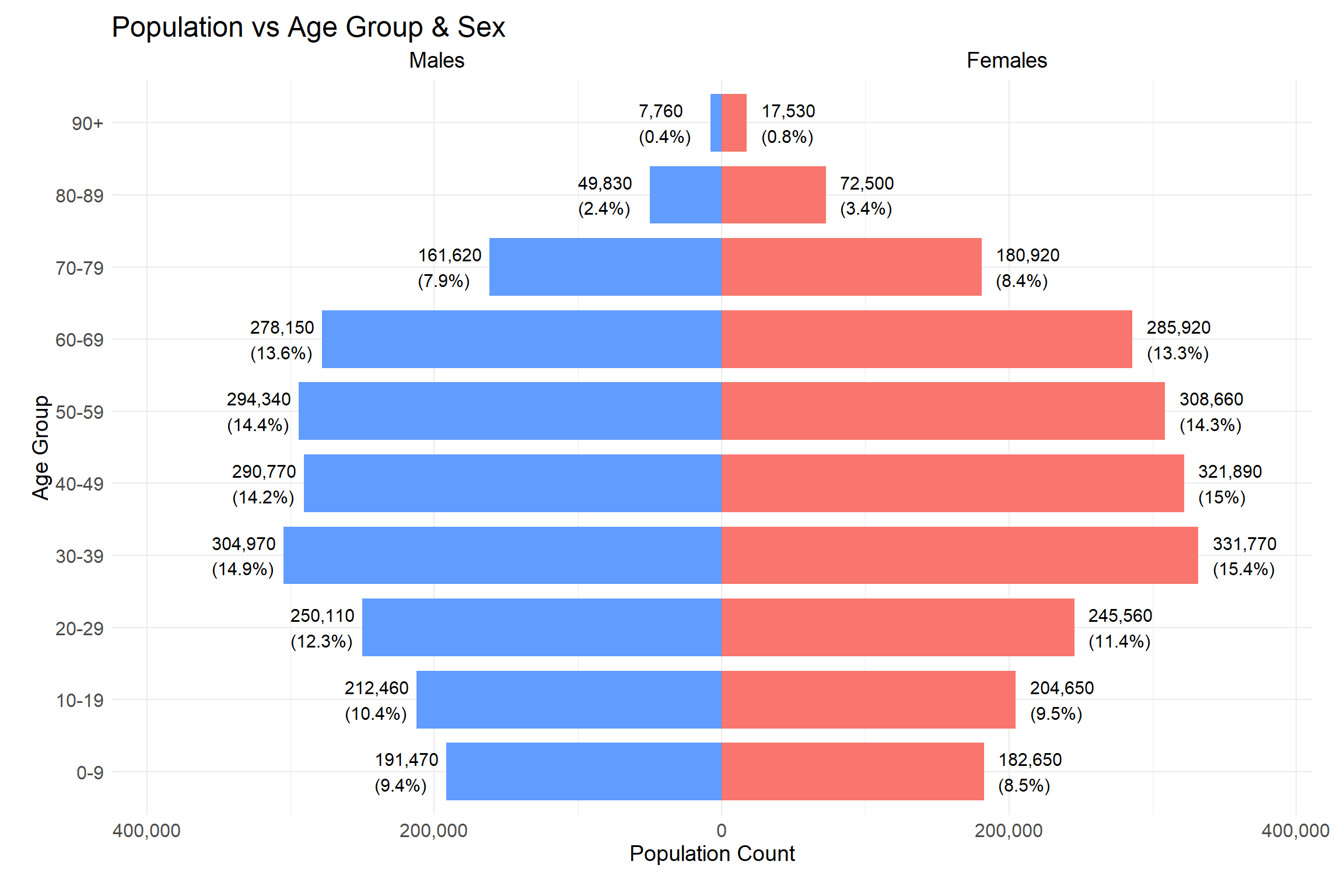


## Choosing the right visualisation

Here, the task was to create at most three graphs to display the demographic structures and distributions of a country. The designer may like to look into chart types such as histograms, population pyramids, and boxplots.



## 10.2 Selected Graph and Feedback



**Constructive Feedback**

**3 Good Design Principles:**

Currently the selected graph is both beautiful (aesthetic) and clear (clarity) in Quadrant 1.

1. **Beautiful:** Overall, the font and palette throughout the chart is easy on the eyes for visual perception. This is done by the designer’s choice of contrasting colours between Males (blue) and Females (red). The colours for both genders also contrast well against the white background colour.
2. **Clear:** The targeted audience of the media may not have a statistical background. Thus the population pyramid provides clear graphics, which may be the right choice of visualisation for population distribution intended for the layman. This was also the choice of graph provided by the Department of Statistics Singapore in their [**Age Pyramid of Resident Population**](https://www.singstat.gov.sg/find-data/search-by-theme/population/population-and-population-structure/visualising-data/population-dashboard). It can be seen that there were deliberations made by the designer of the selected graph through the clever use of derived values such as totals and percentages to reveal more interesting patterns than values at granular levels. In this case, the designer self-initiated to derive categorical bins of the age groups using total counts and percentages of population for different genders. The population pyramid may then be replicated into regions. Depending on the problem statement, these may provide sufficient levels of analysis for broader information, without drilling down to introduce complexity. This communicates to a wider audience at higher levels compared to navigating selected or numerous areas.
3. **Beautiful:** The soft gridlines and elements are well spaced, and do not distract. The Y-axis follows the conventional pyramid shape of younger to older ages as it rises up the pyramid. This is as the population pyramid derived its name from having a heavy base from a growing population (more babies born than mortality), leading to an upright pyramid or triangular shape.

**3 Suggested Areas for Future Improvements:**

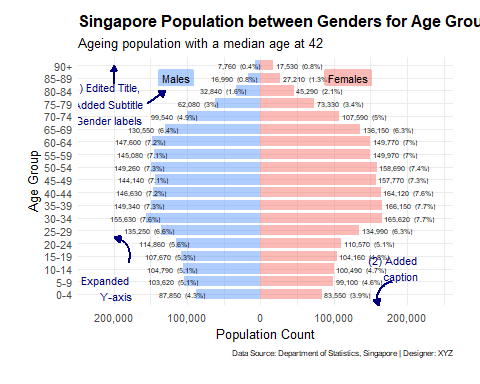
Minor adjustments and suggestions to move higher up Quadrant 1.

1. **Clear:** Further well-worded subtitle, annotations and coloured labels to provide details or tell the data story clearer.
2. **Clear:** Caption on data source for credibility and/or credits to the designer may be added.
3. **Clear:** Y-Axis may be further improved to expand the age-groups to match common 5-year age-groups standards and Singapore’s census data. The breakdown provides clear details of subtle shifts in trends and patterns of the population structure.

## 10.3 Makeover of the Graph

The makeover attempts to implement the earlier 3 suggested areas which are pointed out by the arrows and corresponding numbers.

## The Plot



## The Code

library(ggplot2)  
library(dplyr)  
library(scales)  
  
#--------- Age Groups ----------------  
sg\_demo\_2024 <- sg\_demo\_2024 %>%  
 mutate(  
 # Convert age to numeric, handle "90\_and\_Over"  
 Age\_num = case\_when(  
 Age == "90\_and\_Over" ~ 90,  
 TRUE ~ suppressWarnings(as.numeric(Age)) # Avoid warnings from "90\_and\_Over"  
 ),  
   
 # Group into age bands  
 Age\_Group = case\_when(  
 Age\_num >= 0 & Age\_num <= 4 ~ "0-4",  
 Age\_num >= 5 & Age\_num <= 9 ~ "5-9",  
 Age\_num >= 10 & Age\_num <= 14 ~ "10-14",  
 Age\_num >= 15 & Age\_num <= 19 ~ "15-19",  
 Age\_num >= 20 & Age\_num <= 24 ~ "20-24",  
 Age\_num >= 25 & Age\_num <= 29 ~ "25-29",  
 Age\_num >= 30 & Age\_num <= 34 ~ "30-34",  
 Age\_num >= 35 & Age\_num <= 39 ~ "35-39",  
 Age\_num >= 40 & Age\_num <= 44 ~ "40-44",  
 Age\_num >= 45 & Age\_num <= 49 ~ "45-49",  
 Age\_num >= 50 & Age\_num <= 54 ~ "50-54",  
 Age\_num >= 55 & Age\_num <= 59 ~ "55-59",  
 Age\_num >= 60 & Age\_num <= 64 ~ "60-64",  
 Age\_num >= 65 & Age\_num <= 69 ~ "65-69",  
 Age\_num >= 70 & Age\_num <= 74 ~ "70-74",  
 Age\_num >= 75 & Age\_num <= 79 ~ "75-79",  
 Age\_num >= 80 & Age\_num <= 84 ~ "80-84",  
 Age\_num >= 85 & Age\_num <= 89 ~ "85-89",  
 Age\_num >= 90 ~ "90+",  
 TRUE ~ NA\_character\_  
 )  
 )  
  
#------------------Pop Pyramid--------------------  
# Summarise and compute plot values  
pop\_pyramid <- sg\_demo\_2024 %>%  
 group\_by(Age\_Group, Sex) %>%  
 summarise(Pop = sum(Pop), .groups = "drop") %>%  
 group\_by(Sex) %>%  
 mutate(Percent = Pop / sum(Pop) \* 100) %>%  
 ungroup()  
  
# Order Age\_group for proper vertical alignment  
age\_levels <- c("0-4",  
 "5-9",  
 "10-14",  
 "15-19",  
 "20-24",  
 "25-29",  
 "30-34",  
 "35-39",  
 "40-44",  
 "45-49",  
 "50-54",  
 "55-59",  
 "60-64",  
 "65-69",  
 "70-74",  
 "75-79",  
 "80-84",  
 "85-89",   
 "90+")  
pop\_pyramid$Age\_Group <- factor(pop\_pyramid$Age\_Group, levels = age\_levels)  
  
# Flip male values to negative  
pop\_pyramid <- pop\_pyramid %>%  
 mutate(Pop\_plot = ifelse(Sex == "Males", -Pop, Pop),  
 Label = paste0(comma(abs(Pop)), " (", round(abs(Percent), 1), "%)"))  
  
# Plot  
ggplot(pop\_pyramid, aes(x = Age\_Group, y = Pop\_plot, fill = Sex)) +  
 geom\_bar(stat = "identity", width = 0.8, position = "identity") +  
 coord\_flip() +  
 geom\_text(aes(label = Label),  
 position = position\_nudge(y = ifelse(pop\_pyramid$Sex == "Males", -50000, 10000)),  
 hjust = ifelse(pop\_pyramid$Sex == "Males", 0, 0.1),  
 size = 1.8, color = 'black') +  
 scale\_y\_continuous(labels = function(x) comma(abs(x)), expand = expansion(mult = c(0.1, 0.1))) +  
 scale\_fill\_manual(values = c("Males" = "#619CFF80", "Females" = "#F8766D80")) +  
 labs(title = "Singapore Population between Genders for Age Groups (2024)",  
 subtitle = "Ageing population with a median age at 42",  
 x = "Age Group",   
 y = "Population Count") +  
 annotate(  
 geom = "label",  
 x = 18,  
 y = 120000,  
 label = "Females",  
 fill = "#F8766D80",  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines"),  
 size=3  
 ) +  
 annotate(  
 geom = "label",  
 x = 18,  
 y = -115000,  
 label = "Males",  
 fill = "#619CFF80",  
 color = "black",  
 label.size = 0,  
 label.padding = unit(0.3, "lines"),  
 size=3  
 ) +   
 theme\_minimal()+  
 labs(caption = "Data Source: Department of Statistics, Singapore | Designer: XYZ")+  
 theme(  
 plot.margin = margin(10, 20, 20, 20), # top, right, bottom, left  
 legend.position = "none",  
 axis.text.x = element\_text(size = 8),  
 axis.text.y = element\_text(size = 8),  
 axis.title.x = element\_text(size = 10),  
 axis.title.y = element\_text(size = 10),  
 plot.title = element\_text(size = 12, face='bold'),  
 plot.subtitle = element\_text(size = 10, hjust=0),  
 plot.caption= element\_text(size = 6)  
 )+   
#-------------------------Labelling------------------------------  
# Curve 1  
 geom\_curve(  
 aes(x = 17.5, y = -200000, xend = 19, yend = -200000),  
 arrow = arrow(  
 length = unit(0.03, "npc"),   
 type="closed"   
 ),  
 colour = "navyblue",  
 size = 0.8,  
 angle = 180)+   
 # Curve 2  
 geom\_curve(  
 aes(x = 16, y = -155000, xend = 17, yend = -130000),  
 arrow = arrow(  
 length = unit(0.03, "npc"),   
 type="closed"   
 ),  
 colour = "navyblue",  
 size = 0.8,  
 angle = 180)+   
 # Text 1  
 annotate(  
 "text",  
 x = 16, # Same x-coordinate as the arrow's start  
 y = -150000, # Same y-coordinate as the arrow's start  
 label = "(1) Edited Title, \n Added Subtitle\nand Gender labels", # The text added  
 hjust = 1.1, # Adjust horizontal alignment to position the text relative to the point  
 vjust = 0.5, # Adjust vertical alignment  
 size = 3,  
 colour="navyblue")+   
 # Curve 3  
 geom\_curve(  
 aes(x = 3.5, y =-180000, xend = 5.5, yend = -200000),  
 arrow = arrow(  
 length = unit(0.03, "npc"),   
 type="closed"   
 ),  
 colour = "navyblue",  
 size = 0.8,  
 angle = 90)+   
 # Text 3  
 annotate(  
 "text",  
 x = 1.5, # Same x-coordinate as the arrow's start  
 y = -170000, # Same y-coordinate as the arrow's start  
 label = "(3) Expanded\nY-axis", # The text added  
 hjust = 1.1, # Adjust horizontal alignment to position the text relative to the point  
 vjust = 0.5, # Adjust vertical alignment  
 size = 3,  
 colour="navyblue")+   
 # Curve 4  
 geom\_curve(  
 aes(x = 2, y = 180000, xend = 0.1, yend = 160000),  
 arrow = arrow(  
 length = unit(0.03, "npc"),   
 type="closed"   
 ),  
 colour = "navyblue",  
 size = 0.8,  
 angle = 90)+  
 # Text 4  
 annotate(  
 "text",  
 x = 3.1, # Same x-coordinate as the arrow's start  
 y = 220000, # Same y-coordinate as the arrow's start  
 label = "(2) Added\ncaption", # The text added  
 hjust = 1.1, # Adjust horizontal alignment to position the text relative to the point  
 vjust = 0.5, # Adjust vertical alignment  
 size = 3,  
 colour="navyblue")

# 11. References and Inspiration

* 2travelacrosstime (2023) [**Types of Population Pyramids**](https://2travellingacrosstime.com/2023/08/19/types-of-population-pyramids/)
* David Keyes (2024) [**How to make polished population pyramids in ggplot: part 1**](https://rfortherestofus.com/2024/07/population-pyramid-part-1)
* Dr. Kam Tin Seong (2025) [**Lesson 2: Designing Graphs to Enlighten**](https://isss608-ay2024-25apr.netlify.app/lesson/lesson02/lesson02-designing%20graphs%20to%20enlighten)
* Michal Palenik (2024) [**Deomographic Pyramid by Economic Activity Austria 2023**](https://www.iz.sk/en/projects/eu-regions/AT)
* Singapore Department of Statistics (2020) [**National Statistical Standards**](https://www.singstat.gov.sg/-/media/files/standards_and_classifications/nsa.ashx)
* Singstat (2024) [**Population Trends**](https://www.singstat.gov.sg/find-data/search-by-theme/population/geographic-distribution/latest-data)
* Steven P. Sanderson II, MPH (2023) [**Creating Population Pyramid Plots in R with ggplot2**](https://www.r-bloggers.com/2023/09/creating-population-pyramid-plots-in-r-with-ggplot2)
* National Geographic (2023) [**Population Pyramid**](https://education.nationalgeographic.org/resource/population-pyramid/)
* URA (2025) [**Regional Highlights**](https://www.ura.gov.sg/Corporate/Planning/Master-Plan/Master-Plan-2019/Regional-Highlights)