**using subplot:**

```{r}

d<- highlight\_key(data\_filtered)

p1 <-ggplot(filter(data\_filtered,PA == "Bedok"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA)) + theme(legend.position = 'None')

p2 <- ggplot(filter(data\_filtered,PA == "Tampines"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA))+ theme(legend.position = 'None')

p3 <- ggplot(filter(data\_filtered,PA == "Jurong West"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

subplot(ggplotly(p1),ggplotly(p2),ggplotly(p3),shareY = T,margin = 0.025,nrows = 3)

```

**Output:**

Chart

Description automatically generated

**using ggplotly:**

```{r}

p <- ggplot(data\_filtered, aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA))

ggplotly(p)

```

**Output:**

Chart, bar chart

Description automatically generated

---

title: "Untitled"

description: |

A new article created using the Distill format.

author:

- name: Nora Jones

url: https://example.com/norajones

affiliation: Spacely Sprockets

affiliation\_url: https://example.com/spacelysprokets

date: "`r Sys.Date()`"

output: distill::distill\_article

---

```{r setup, include=FALSE}

knitr::opts\_chunk$set(echo = FALSE)

```

Distill is a publication format for scientific and technical writing, native to the web.

Learn more about using Distill for R Markdown at <https://rstudio.github.io/distill>.

# 1. Introduction

This exercise aims to plot interactive population pyramids on a planning area level from year 2000-2010.

## Import the packages needed.

```{r}

packages = c('tidyverse','readxl','dplyr','ggplot2','plotly','gganimate','ggiraph')

for(p in packages){

if(!require(p, character.only = T)){

install.packages(p)

}

library(p,character.only = T)

}

```

## 2. Data Exploration and preparation

Import both population data

The data is imported using read\_csv() function of \*\*readr\*\* package.

It can be observed that both datasets have same columns, hence, can be combined.

```{r}

data0010 <- read\_csv("respopagesextod2000to2010/respopagesextod2000to2010.csv")

data1120 <- read\_csv("respopagesextod2011to2020/respopagesextod2011to2020.csv")

```

As the analysis aims to analyse the change of population over the years from 2000 to 2020, the two datasets are combined using \*\*rbind()\*\*.

```{r}

data <- rbind(data0010, data1120)

```

Group population by planning area and plot a Pareto chart to inspect the population distribution over planning area.

It can be observed from the interactive bar chart that some planning areas are having 0 population.

From the Pareto chart, it can also be observed that more than 50% of the population comes from the top 9 planning areas ordered by total population count.

As there are a large number of planning areas, for ease of analysis,

the focus will be on these 9 planning areas.

```{r}

data\_grouped\_pa <- data %>%

group\_by(PA)%>%

summarise(

count\_by\_pa = sum(Pop, na.rm = TRUE))

#order the data and add the cumulative percentage column

data\_grouped\_pa <- data\_grouped\_pa[order(data\_grouped\_pa$count\_by\_pa, decreasing=TRUE),]

data\_grouped\_pa$cumulative <- cumsum(data\_grouped\_pa$count\_by\_pa)/sum(data\_grouped\_pa$count\_by\_pa) \* 100

#create the scale on secondary y-axis

scale <- tail(data\_grouped\_pa$cumulative, n=1)/head(data\_grouped\_pa$count\_by\_pa, n=1)

p <- ggplot(data\_grouped\_pa,

aes(x = reorder(PA, -count\_by\_pa),y = count\_by\_pa,

tooltip=count\_by\_pa)) +

geom\_bar\_interactive(stat = "identity", color="black", fill = 'light blue') +

geom\_path\_interactive(aes(y=cumulative/scale, group=1),colour="red", size=0.9) +

geom\_point\_interactive(aes(y=cumulative/scale, group=1,tooltip = cumulative),colour="red") +

scale\_y\_continuous(

# Features of the first y-axis

name = "Population Count",

# Add second y-axis and specify its features

sec.axis = sec\_axis(~ .\*scale, name="% of Total Running Sum of Population Count")

) +

theme(axis.text.x = element\_text(angle = 90)) +

labs(title="Pareto Chart", subtitle="Count of Population by planning area",

x="Planning Area", y=expression(count\_by\_pa(~mu~Ah/~mu~s)))

girafe(ggobj = p,width\_svg = 7,

height\_svg = 6)

```

Filter the data for the top 9 planning areas with the most population amount.

```{r}

filter\_list <- data\_grouped\_pa$PA[1:9]

print(filter\_list)

data\_filtered <- filter(data, PA %in% filter\_list)

data\_filtered$PopPercent <- data\_filtered$Pop/sum(data\_filtered$Pop) \* 100

```

## 3. Plot the population pyramid of the 9 planning areas

xbrks <- seq(-3500000, 3500000, 700000)

p <- ggplot(data\_filtered, aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

ggplotly

```{r}

?theme()

```

```{r}

p <- ggplot(data\_filtered, aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(x = "Age", y = "Population Count") +

coord\_flip() +

facet\_wrap(~PA)

ggplotly(p)%>%

layout(

title = "Population Pyramids",

yaxis = list(

title = "Age Group") ,

autosize = FALSE,width = 700, height = 700

)

```

d<- highlight\_key(data\_filtered)

p1 <-ggplot(filter(data\_filtered,PA == "Bedok"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA)) + theme(legend.position = 'None')

p2 <- ggplot(filter(data\_filtered,PA == "Tampines"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA))+ theme(legend.position = 'None')

p3 <- ggplot(filter(data\_filtered,PA == "Jurong West"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -Pop, no = Pop),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$Pop) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

subplot(ggplotly(p1),ggplotly(p2),ggplotly(p3),shareY = T,margin = 0.025,nrows = 3)

d<- highlight\_key(data\_filtered)

p1 <-ggplot(filter(data\_filtered,PA == "Bedok"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$PopPercent) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA)) + theme(legend.position = 'None')

p2 <- ggplot(filter(data\_filtered,PA == "Tampines"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$PopPercent) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA))+ theme(legend.position = 'None')

p3 <- ggplot(filter(data\_filtered,PA == "Jurong West"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$PopPercent) \* c(-1,1)) +

scale\_x\_discrete(breaks=data\_filtered$AG)+

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p4 <- ggplot(filter(data\_filtered,PA == "Woodlands"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$PopPercent) \* c(-1,1)) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p5 <- ggplot(filter(data\_filtered,PA == "Hougang"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$PopPercent)) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p6 <- ggplot(filter(data\_filtered,PA == "Yishun"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(labels = abs, limits = max(data\_filtered$PopPercent)) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

subplot(ggplotly(p1),ggplotly(p2),ggplotly(p3),shareY = T,margin = 0.025,nrows = 3)

#subplot(,ggplotly(p2),ggplotly(p3),ggplotly(p4),

# ggplotly(p5),ggplotly(p6),shareX = T,,margin = 0.025,shareY = TRUE,

# heights = c(0.8,0.2,0.1))

d<- highlight\_key(data\_filtered)

xbrks <- seq(-100, 100, 10)

p1 <- ggplot(filter(data\_filtered,PA == "Bedok"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

label=paste(round(abs(PopPercent)\*100, 0), "%", sep=""),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA))

p2 <- ggplot(filter(data\_filtered,PA == "Tampines"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA))

p3 <- ggplot(filter(data\_filtered,PA == "Jurong West"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p4 <- ggplot(filter(data\_filtered,PA == "Woodlands"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p5 <- ggplot(filter(data\_filtered,PA == "Hougang"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p6 <- ggplot(filter(data\_filtered,PA == "Yishun"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p7 <- ggplot(filter(data\_filtered,PA == "Ang Mo Kio"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p8 <- ggplot(filter(data\_filtered,PA == "Choa Chu Kang"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

p9 <- ggplot(filter(data\_filtered,PA == "Sengkang"), aes(x = AG, fill = Sex,

y = ifelse(test = Sex == "Males", yes = -PopPercent, no = PopPercent),

tooltip = Pop,frame = Time))+

geom\_col(position = "identity") +

scale\_y\_continuous(breaks = xbrks,labels = function(br) ifelse(abs(br)>=1000,paste0(abs(br)/1000, "k"), abs(br))) +

labs(title = "Population Pyramid", x = "Age", y = "Population Count") +

coord\_flip() +

facet\_grid(rows = vars(PA), scales = "free\_x", space = "free\_x")

subplot(ggplotly(p1),ggplotly(p2),ggplotly(p3),ggplotly(p4),ggplotly(p5),

ggplotly(p6),ggplotly(p7),ggplotly(p8),ggplotly(p9),shareX = T,nrows = 6)