df <- read.csv('../../期中/merged\_data.csv')

library(dplyr)

# 抓df的town.dist欄位1~3個字當成新欄位city的資料

df <- df %>%

mutate(city = substr(town.dist, 1, 3))

library(plyr)

# 將 gender 欄位的 0,1 改成男女

df$gender <- mapvalues(df$gender, from = c(0, 1), to = c('男', '女'))

library(ggplot2)

**# Q1 長條圖**

ggplot(df, aes(x = city, fill = gender)) +

geom\_bar(position = "dodge") +

labs(title = "縣市人口數分布",

x = "縣市",

y = "人數",

fill = "性別",

caption = paste('資料來源:資管大數據碩專一甲\n繪製日期:',

Sys.Date(), '\n繪製者:周政邦')) +

scale\_fill\_manual(values = c("男" = "#0072E3", "女" = "#FF79BC")) +

theme\_minimal() +

theme(plot.title = element\_text(color = '#007500', # 設定 title

face = 'bold', # 字體/粗體

size = 14, # 字體大小

hjust = 0.5), # 對齊方式，水平對齊 h

plot.caption = element\_text(color = '#003D79', # plot.caption 對右下角控制

size = 10),

axis.title.x = element\_text(color = '#00CACA', # X 軸 title 顏色

size = 14),

axis.title.y = element\_text(color = '#F75000', # Y 軸 title 顏色

size = 14),

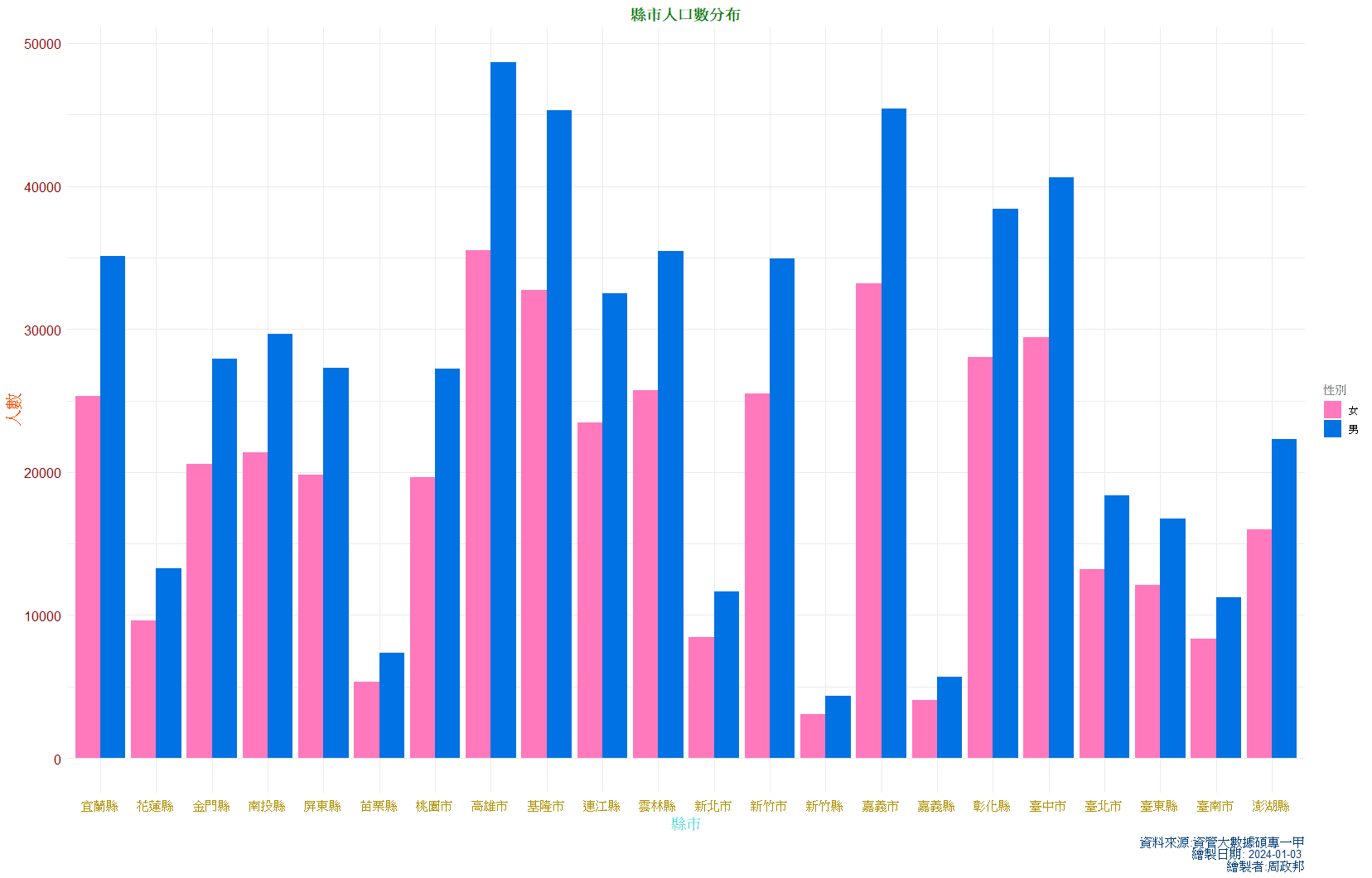
axis.text.x = element\_text(color = '#AE8F00', # X 軸 表格意義 顏色

size = 12),

axis.text.y = element\_text(color = '#930000', # Y 軸 表格意義 顏色

size = 12))

長條圖長下面的樣子



**# Q2 折線圖**

# 以 city 為依據抓 gender 的資料並加總，欄位名稱為 total\_count

df\_city\_sum <- df %>%

group\_by(city, gender) %>%

tally(name = "total\_count")

table(df\_city\_sum)

# 以 city 為依據，加總 total\_count 欄位的值

city\_sum <- df\_city\_sum %>%

group\_by(city) %>%

summarize(total\_sum = sum(total\_count))

library(scales)

# data.frame 為 df\_city\_sum，X軸為city資料，Y軸為total\_count資料

# 顏色以 gender 為依據

plot01 <- ggplot(df\_city\_sum, aes(x = city, y = total\_count, color = gender, group = gender)) +

geom\_point(color = 'red') +

geom\_line(aes(color = gender), linewidth = 0.5) +

scale\_y\_continuous(labels = comma) +

theme\_bw() +

labs(title = "縣市人口數分布",

x = "縣市",

y = "人數",

color = "性別",

caption = paste('資料來源:資管大數據碩專一甲\n繪製日期:',

Sys.Date(), '\n繪製者:周政邦')) +

theme(plot.title = element\_text(color = '#007500',

face = 'bold',

size = 14,

hjust = 0.5),

plot.caption = element\_text(color = '#003D79',

size = 10),

axis.title.x = element\_text(color = '#00CACA',

size = 14),

axis.title.y = element\_text(color = '#F75000',

size = 14),

axis.text.x = element\_text(color = '#AE8F00',

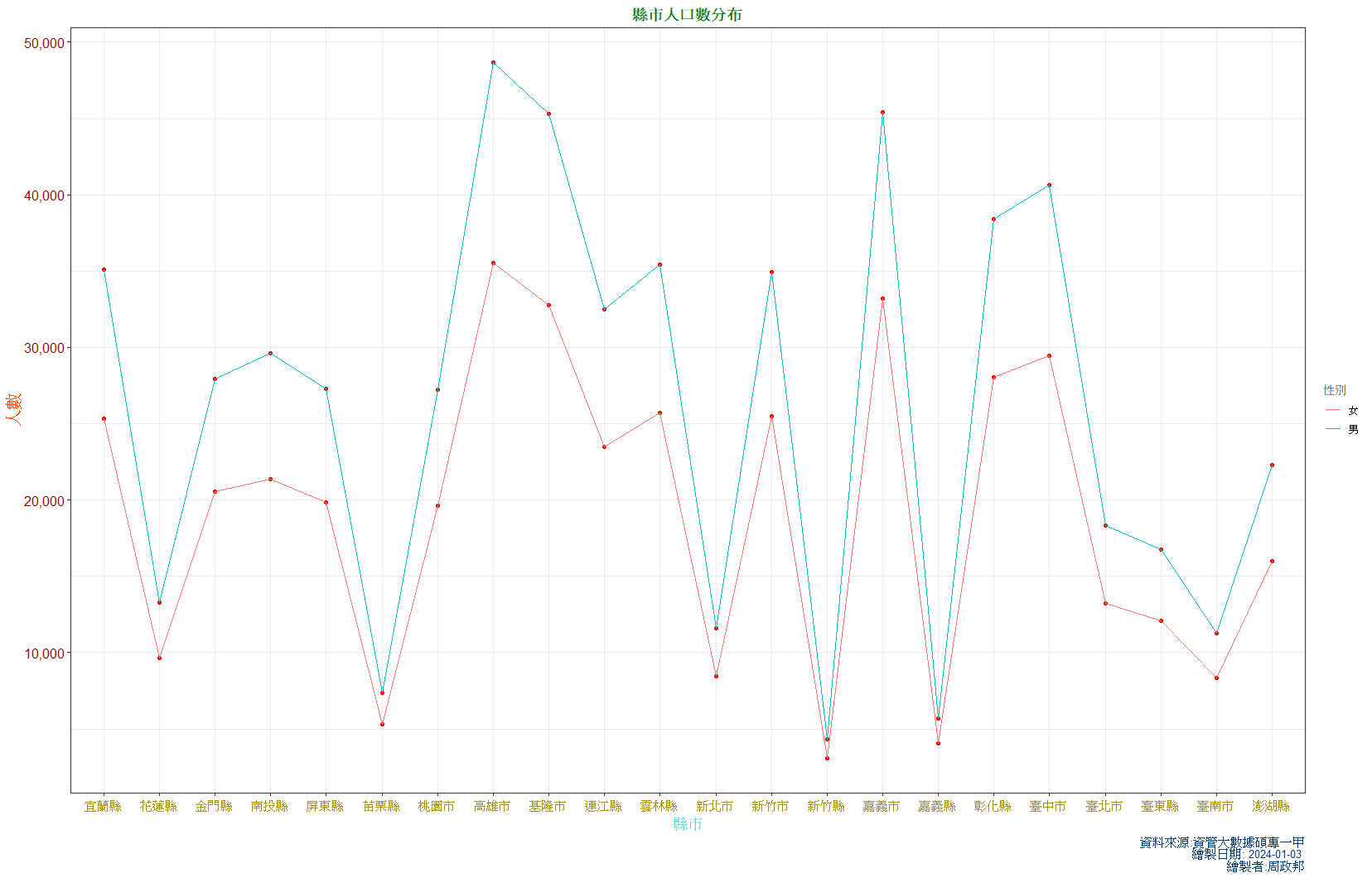
size = 12),

axis.text.y = element\_text(color = '#930000',

size = 12))

plot01

**折線圖**



**!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!**

**這邊後面是因為我上面的摺線圖XY軸用縣市跟人數，無法跑動畫效果  
所以用老師教得再寫一遍，看你們有沒有想寫**

library(lubridate)

q3 <- ymd(df$birthday)

which(is.na(q3)) # 抓出誤差值

# 新增誤差值

df[c(131653, 165646, 430975, 805547, 900521), 4]

df[131653, ]$birthday <- "1993/6/30"

df[165646, ]$birthday <- "1989/6/30"

df[430975, ]$birthday <- "1961/2/28"

df[805547, ]$birthday <- "1959/9/30"

df[900521, ]$birthday <- "1987/9/30"

# 用 year 函數抓年份，並新增 Year 欄位將值放入

df$Year <- year(df$birthday)

df.Consumption.Year <- aggregate(consumption~Year, mean, data = df)

library(scales)

plot1 <- ggplot(df.Consumption.Year, aes(x = Year, y = consumption)) +

geom\_point(shape = 10, color = 'red') +

geom\_line(color = 'blue') +

scale\_x\_continuous( # 設定X軸的刻度

breaks = seq(min(df.Consumption.Year$Year),

max(df.Consumption.Year$Year), 5), #設定5年為一個間距

labels = seq(min(df.Consumption.Year$Year),

max(df.Consumption.Year$Year), 5)) +

scale\_y\_continuous(labels = comma) +

theme\_bw() +

labs(title = '不同年代顧客累計消費金額直線圖',

subtitle = '大數據公司',

x = '年份',

y = '累計消費金額',

caption = paste("資料來源：銷售資訊系統\n製作日期：",

Sys.Date(), "\n製作者：周政邦")) +

theme(plot.title = element\_text(color = '#007500',

face = 'bold',

size = 14,

hjust = 0.5),

plot.subtitle = element\_text(color = '#9F0050',

face = "bold",

size = 14,

hjust = 0.5,

family = "Kaiu"),

plot.caption = element\_text(color = '#003D79',

size = 10),

axis.title.x = element\_text(color = '#00CACA',

size = 14),

axis.title.y = element\_text(color = '#F75000',

size = 14),

axis.text.x = element\_text(color = '#AE8F00',

size = 12),

axis.text.y = element\_text(color = '#930000',

size = 12))

plot1

# 動畫

library(gganimate)

plot1 +

geom\_point(aes(group = seq\_along(Year)), color = 'red') +

transition\_reveal(Year) + #transition\_reveal 動畫做轉移的效果以年為依據

labs(title = '不同年代顧客累計消費金額直線圖 年代：{round(frame\_along)}')

# round(frame\_along) 年份只顯示整數

# 存成新物件

plot1\_1 <-

plot1 +

geom\_point(aes(group = seq\_along(Year)), color = 'red') +

transition\_reveal(Year) + #transition\_reveal 動畫做轉移的效果以年為依據

labs(title = '不同年代顧客累計消費金額直線圖 年代：{round(frame\_along)}')

plot1\_1

library(gifski)

# Save as a gif

# 將效果存成物件

gif\_plot1\_1 <- animate(plot1\_1, fps = 20, duration = 30,

width = 800, height = 450,

renderer = gifski\_renderer(loop = FALSE))

gif\_plot1\_1

df.Consumption.Year

# 求出最高值年份及最低值年份

x1 <- rbind(df.Consumption.Year[which.max(df.Consumption.Year$consumption),],

df.Consumption.Year[which.min(df.Consumption.Year$consumption),])

adjust <- 10 ^ (floor(log(max(df.Consumption.Year$consumption), 10)) - 2)

adjust

max\_min <- cbind(x1, 金額 = c("最高", "最低"), offset = c(adjust, -adjust))

max\_min

library(ggplot2)

plot1 +

geom\_text(aes(x = Year, y = consumption + offset, label = 金額),

data = max\_min) # 顯示出最高值最低值

plot1 +

geom\_label(aes(x = Year, y = consumption + offset, label = 金額),

data = max\_min) # 將最高值最低值加邊框

library(ggrepel)

plot1 +

geom\_label\_repel(aes(x = Year, y = consumption + offset, label = 金額),

data = max\_min) # 避免標籤重疊影響閱讀

plot1 +

geom\_label\_repel(aes(x = Year, y = consumption, label = 金額),

data = max\_min, colour = '#02C874',

segment.colour= '#921AFF', segment.size = 1,#線的顏色跟粗細

label.r = 0.5, label.size = 0.5,

arrow = arrow(angle = 30,

length = unit(0.2, 'cm'), ends = 'last'),

box.padding = unit(1, 'cm'), label.padding = unit(0.2, 'cm'),

nudge\_x = 1, nudge\_y = 1)

library(dygraphs)

library(xts)

library(magrittr)

# 將Year(int)轉為Time Series(時間序列)

df.Consumption.Year$Year <- ts(df.Consumption.Year$Year)

str(df.Consumption.Year)

min(df.Consumption.Year$consumption)

# 轉成網頁顯示

dygraph(df.Consumption.Year, xlab = '年',

main = '大數據公司不同年代顧客累計消費金額直線圖') %>% # library(magrittr)這套件啟動%>%

dyAxis('y', label = '累計消費金額',

valueRange = c(min(df.Consumption.Year$consumption),

max(df.Consumption.Year$consumption)),

axisLabelColor = '#1911f7', axisLabelFontSize = 12) %>%

dyOptions(drawPoints = TRUE, pointSize = 5, pointShape = 'star',

colors = 'red') %>%

dyLegend(show = 'follow') %>% # 數值隨著滑鼠移動做顯示

dyHighlight(highlightCircleSize = 10, # 星號隨著滑鼠指到那個點作放大的效果

highlightSeriesBackgroundAlpha = 0.5,# 背景框線做透明

hideOnMouseOut = TRUE) %>%

dySeries('consumption', label = '累計消費金額') %>% # 註解的 Consumption 改成累計消費金額

dyOptions(labelsKMB = 'M') %>% # 將XY軸單位轉成千分位寫成K

dyRangeSelector(fillColor = 'rgb(255,100,200)',

strokeColor = 'rgb(150,250,150)',

height = 20) %>% # 畫面隨著範圍做比例縮放

dyCrosshair(direction = 'both') %>% # 滑鼠指定的點加十字標記

dyCSS('../../../CSS/dygraph.css') #CSS的語法

library(plotly)

ggplotly(plot1) # 滑鼠加註解，Export 可以以網頁開啟另存png

# 滑鼠顯示為1215R的 aes

ggplotly(plot1, tooltip = c('Year', 'consumption')) # 預設顯示

ggplotly(plot1, tooltip = c('Year')) # 只顯示年份

ggplotly(plot1, tooltip = c('consumption')) # 只顯示累積消費金額

library(scales) # 加載 comma 函數

plot2 <- ggplot(df.Consumption.Year, aes(x = Year, y = consumption)) +

geom\_point(aes(text = paste0('年：', Year, '<br>金額：', # geom\_point 設定點顯示的數據

comma(consumption, scale = 1/1e4, # 顯示千分位，1/1e4 <- 10的4次方

accuracy=0.01, # 小數點取到第二位

suffix='萬'))),

colour = "brown", size = 2, shape = 23) +

geom\_line(color = 'blue') +

scale\_x\_continuous( # 設定X軸的刻度

breaks = seq(min(df.Consumption.Year$Year),

max(df.Consumption.Year$Year), 5), #設定5年為一個間距

labels = seq(min(df.Consumption.Year$Year),

max(df.Consumption.Year$Year), 5)) +

scale\_y\_continuous(labels = comma) +

theme\_bw() +

labs(title = '不同年代顧客累計消費金額直線圖',

subtitle = '大數據公司',

x = '年份',

y = '累計消費金額',

caption = paste("資料來源：銷售資訊系統\n製作日期：",

Sys.Date(), "\n製作者：周政邦")) +

theme(plot.title = element\_text(color = '#007500',

face = 'bold',

size = 14,

hjust = 0.5),

plot.subtitle = element\_text(color = '#9F0050',

face = "bold",

size = 14,

hjust = 0.5,

family = "Kaiu"),

plot.caption = element\_text(color = '#003D79',

size = 10),

axis.title.x = element\_text(color = '#00CACA',

size = 14),

axis.title.y = element\_text(color = '#F75000',

size = 14),

axis.text.x = element\_text(color = '#AE8F00',

size = 12),

axis.text.y = element\_text(color = '#930000',

size = 12))

plot2

ggplotly(plot2, tooltip = 'text') # 動態圖，tooltip提示框

ggplotly(plot2, tooltip = 'text') %>%

layout(hoverlabel=list(bgcolor = '#97CBFF',

bordercolor = '#EAC100',

borderwidth = 2,

font = list(color = '#D94600',

size = 14)))

# tooltip 裡面的 layout

# hoverlabel 設定滑鼠過去標籤怎麼顯示

**# Q3 地圖**

library(sf)

taiwan\_shp\_sf <- read\_sf('../../../上課/project02/mapdata/COUNTY\_MOI\_1090820.shp')

head(taiwan\_shp\_sf) # 前幾筆資料

ggplot(taiwan\_shp\_sf) +

geom\_sf() # geom\_sf 這套件程式會自行判斷畫地圖

ggplot(taiwan\_shp\_sf) +

geom\_sf() +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) # 調整顯示的經緯度

st\_centroid(taiwan\_shp\_sf$geometry) # 抓資料裡面的經緯度的中心點

# 裡面有每個縣市的資料，故會抓取每個縣市的中心點

centroid <- st\_centroid(taiwan\_shp\_sf$geometry) %>%

st\_coordinates() %>%

as.data.frame %>%

setNames(c('Longitude', 'Latitude'))

# st\_centroid(taiwan\_shp\_sf$geometry) %>% st\_coordinates()

# 將中心點資料轉成矩陣資料

# as.data.frame

# 轉成資料型別

# setNames(c('Longitude', 'Latitude'))

# 將XT軸設定成別的名稱

taiwan\_shp\_sf <- cbind(taiwan\_shp\_sf, centroid)

# 將所需的資料設一個變數塞回去原本的資料內

ggplot(taiwan\_shp\_sf) +

geom\_sf(aes(fill=COUNTYNAME)) + # fill=COUNTYNAME 城市名稱填滿

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

geom\_point(aes(x = Longitude, y = Latitude), size = 2) +

annotate('text',

x = taiwan\_shp\_sf$Longitude, # 無法直接繼承，自行手動設定

y = taiwan\_shp\_sf$Latitude,

label = taiwan\_shp\_sf$COUNTYNAME,

color = 'blue',

size = 3,

vjust = -0.5) +

theme\_void() + # 取消背景

theme(legend.position = 'none') # 取消註解

#在 ggplot2 中，annotate 函數通常用於添加額外的標籤、文本或幾何形狀到繪圖中。

city.df <- merge(taiwan\_shp\_sf, city\_sum, # 兩個做合併，以前面為依據

by.x = 'COUNTYNAME', by.y = 'city',

all.x = TRUE) # 所有 X 軸資料作保留，保留地圖資料

head(city.df)

# 資料型態問題做轉換

city\_sum <- as.data.frame(city\_sum)

city\_levels <- factor(city\_sum$city,

levels = city\_sum[order(-city\_sum$total\_sum), 'city'],

ordered = TRUE)

city.df$COUNTYNAME <- factor(city.df$COUNTYNAME, levels = levels(city\_levels),

ordered = TRUE)

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5))

# fill = total\_sum，以消費金額多寡做顏色填充

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

# scale\_fill\_continuous(type = 'gradient') # 預設值

scale\_fill\_continuous(type = 'viridis', label = comma,

name = '各縣市人口數') +

theme\_void()

library(RColorBrewer)

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

scale\_fill\_gradientn(colours = brewer.pal(3,'BrBG'),

label = comma,

n.breaks = 10,

name = '各縣市人口數') +

theme\_void()

# 自行設定顏色

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

scale\_fill\_gradientn(colours = c('#ACD6FF', '#00E3E3', '#CE0000', '#BB3D00'),

label = comma,

n.breaks = 10,

name = '各縣市人口數') +

theme\_void()

library(showtext)

font\_add\_google('Lobster', 'Lobster')

font\_add\_google('Dancing Script', 'Dancing Script')

font\_add('Kaiu', 'C:/windows/fonts/kaiu.ttf')

showtext\_auto()

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

scale\_fill\_gradientn(colours = c('#ACD6FF', '#00E3E3', '#CE0000', '#BB3D00'),

label = comma,

n.breaks = 10,

name = '各縣市人口數') +

theme\_void() +

labs(title = '大數據公司各縣市平均累計消費金額分佈圖',

caption = paste("資料來源：大數據公司\n繪製日期：",

Sys.Date(),"\n繪製者：周政邦")) +

theme(plot.title = element\_text(colour = '#D200D2', face = 'bold',

size = 14, hjust = 0.5, family = 'Kaiu'),

plot.caption = element\_text(colour = '#3D7878', size = 10,

family = 'Kaiu'),

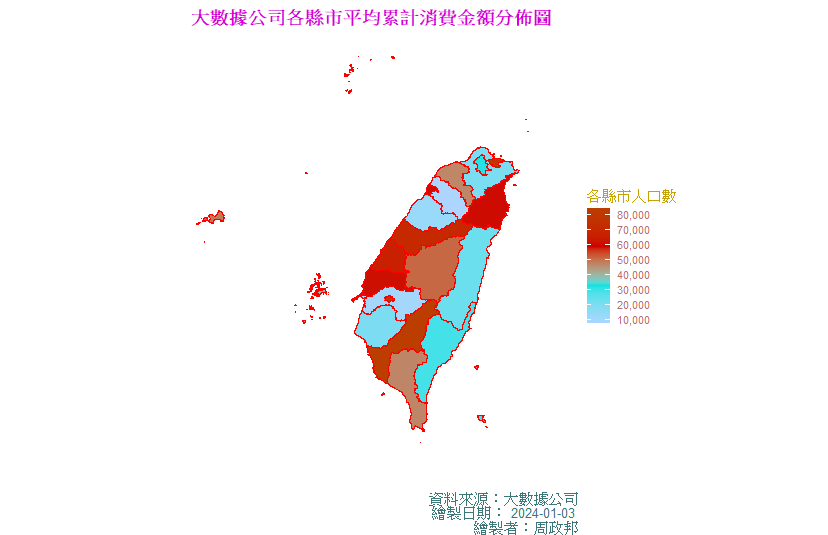
legend.title = element\_text(colour = '#C6A300', size = 10,

family = 'Kaiu'),

legend.text = element\_text(colour = '#B87070', size = 8,

family = 'Lobster'))

**做到這邊就是呈現下方的樣子，後面就是有動畫及各種深入的東西，一樣看各位有無想要寫**



# annotate

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

geom\_point(aes(x = Longitude, y = Latitude), size = 2) +

annotate('text',

x = city.df$Longitude,

y = city.df$Latitude,

label = city.df$COUNTYNAME,

color = 'black',

size = 3,

vjust = -0.5,

family = 'Kaiu') +

scale\_fill\_gradientn(colours = c('#ACD6FF', '#00E3E3', '#CE0000', '#BB3D00'),

label = comma,

n.breaks = 10,

name = '各縣市人口數') +

theme\_void() +

labs(title = '大數據公司各縣市人口數分佈圖',

caption = paste("資料來源：大數據公司\n繪製日期：",

Sys.Date(),"\n繪製者：周政邦")) +

theme(plot.title = element\_text(colour = '#D200D2', face = 'bold',

size = 14, hjust = 0.5, family = 'Kaiu'),

plot.caption = element\_text(colour = '#3D7878', size = 10,

family = 'Kaiu'),

legend.title = element\_text(colour = '#C6A300', size = 10,

family = 'Kaiu'),

legend.text = element\_text(colour = '#B87070', size = 8,

family = 'Lobster'))

# geom\_text

ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

scale\_fill\_gradientn(colours = c('#ACD6FF', '#00E3E3', '#CE0000', '#BB3D00'),

label = comma,

n.breaks = 10,

name = '各縣市人口數') +

theme\_void() +

labs(title = '大數據公司各縣市人口數分佈圖',

caption = paste("資料來源：大數據公司\n繪製日期：",

Sys.Date(),"\n繪製者：周政邦")) +

theme(plot.title = element\_text(colour = '#D200D2', face = 'bold',

size = 14, hjust = 0.5, family = 'Kaiu'),

plot.caption = element\_text(colour = '#3D7878', size = 10,

family = 'Kaiu'),

legend.title = element\_text(colour = '#C6A300', size = 10,

family = 'Kaiu'),

legend.text = element\_text(colour = '#B87070', size = 8,

family = 'Lobster')) +

geom\_point(aes(x = Longitude, y = Latitude), size = 2) +

geom\_text(aes(x = Longitude, y = Latitude, label = COUNTYNAME),

size = 3,

vjust = -0.5,

family = 'Kaiu')

tw\_plot1 <- ggplot(city.df) +

geom\_sf(aes(fill = total\_sum), color = 'red', size = 1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

scale\_fill\_gradientn(colours = c('#ACD6FF', '#00E3E3', '#CE0000', '#BB3D00'),

label = comma,

n.breaks = 10,

name = '各縣市人口數') +

theme\_void() +

labs(title = '大數據公司各縣市人口數分佈圖',

caption = paste("資料來源：大數據公司\n繪製日期：",

Sys.Date(),"\n繪製者：周政邦")) +

theme(plot.title = element\_text(colour = '#D200D2', face = 'bold',

size = 14, hjust = 0.5, family = 'Kaiu'),

plot.caption = element\_text(colour = '#3D7878', size = 10,

family = 'Kaiu'),

legend.title = element\_text(colour = '#C6A300', size = 10,

family = 'Kaiu'),

legend.text = element\_text(colour = '#B87070', size = 8,

family = 'Lobster'))

tw\_plot1 +

geom\_label(aes(x = Longitude, y = Latitude, label = COUNTYNAME),

data = city.df,

size = 3,

family = 'Kaiu')

library(ggrepel)

tw\_plot1 + geom\_label\_repel(aes(x = Longitude, y = Latitude, label = COUNTYNAME),

data = city.df, alpha = 1,

max.overlaps = 15) # 縣市加框加箭頭

tw\_plot1 + geom\_label\_repel(aes(x = Longitude, y = Latitude, label = COUNTYNAME),

data = city.df, colour = '#64A600',

segment.colour = '#FF5809', # 提示框連接地圖的線

segment.size = 1, label.r = 0.5, label.size = 1,

arrow = arrow(angle = 30, length = unit(0.2, 'cm'),

ends = 'last' ), # 箭頭

box.padding = unit(1, 'cm'),

label.padding = unit(0.2, 'cm'),

nudge\_x = 0.01, nudge\_y = 0.01,

max.overlaps = 40)

tw\_1 <- tw\_plot1 +

geom\_label\_repel(aes(x = Longitude, y = Latitude, label = COUNTYNAME),

data = city.df, colour = '#64A600',

segment.colour = '#FF5809', # 提示框連接地圖的線

segment.size = 1, label.r = 0.5, label.size = 1,

arrow = arrow(angle = 30, length = unit(0.2, 'cm'),

ends = 'last' ), # 箭頭

box.padding = unit(1, 'cm'),

label.padding = unit(0.2, 'cm'),

nudge\_x = 0.01, nudge\_y = 0.01,

max.overlaps = 40)

library(gganimate)

library(transformr)

# 動畫，從最大拚到最小

tw\_1 +

transition\_states(COUNTYNAME) +

shadow\_mark()

library(plotly)

library(scales)

ggplotly(tw\_plot1)

ggplotly(tw\_plot1, tooltip = c('total\_sum'))

library(scales)

tw\_plot2 <- ggplot(city.df) +

geom\_sf(aes(fill = total\_sum,

text =paste0(COUNTYNAME,'<br>','人口總數：',

number(total\_sum, scale = 1e-4, accuracy = 0.01,

suffix = '萬人',

decimal.mark = '.'))),

color = '#272727', size = 0.1) +

coord\_sf(xlim = c(118, 122.5),

ylim = c(21.5, 26.5)) +

scale\_fill\_gradientn(colours = c('#005AB5', '#02F78E', '#CE0000', '#5B00AE'),

label = label\_number(scale = 1e-4,

suffix = '萬人',

decimal.mark = '.'),

n.breaks = 8,

name = '縣市人口數') +

theme\_void() +

labs(title = '大數據公司各縣市人口數分佈圖') +

theme(plot.title = element\_text(colour = '#D200D2', face = 'bold',

size = 14, hjust = 0.5, family = 'Kaiu'),

plot.caption = element\_text(colour = '#3D7878', size = 10,

family = 'Kaiu'),

legend.title = element\_text(colour = '#C6A300', size = 10,

family = 'Kaiu'),

legend.text = element\_text(colour = '#B87070', size = 8,

family = 'Lobster'))

ggplotly(tw\_plot2, tooltip = c('text')) %>%

layout(xaxis = list(showline = FALSE),

yaxis = list(showline = FALSE))