

EDA with R

Data Visualization with **ggplot & dplyr**

Chung Fu,Yang
2016-06-22

Speaker

About me

- Alger Yang, M.S, 政大統計所碩二
- 角色：
 - 資料分析屍
 - 資料視覺化愛好者
 - 健保資料庫的掃地工人(資料清理...)
- 常用工具：
 - R、SQL、SPSS
- 目前工作：
 - DSP講師
 - 健保資料庫資料分析
 - 政大風管所法律組量化資料分析研究助理



課程綱要

Agenda

- Data Visualization
- ggplot in R
 - 介紹
 - 架構
 - 起手式(基本語法)
 - 應用(各種圖形的呈現)
 - Line
 - Bar
 - Histogram
 - point
 - 進階技巧傳承篇
- 互動視覺化plotly

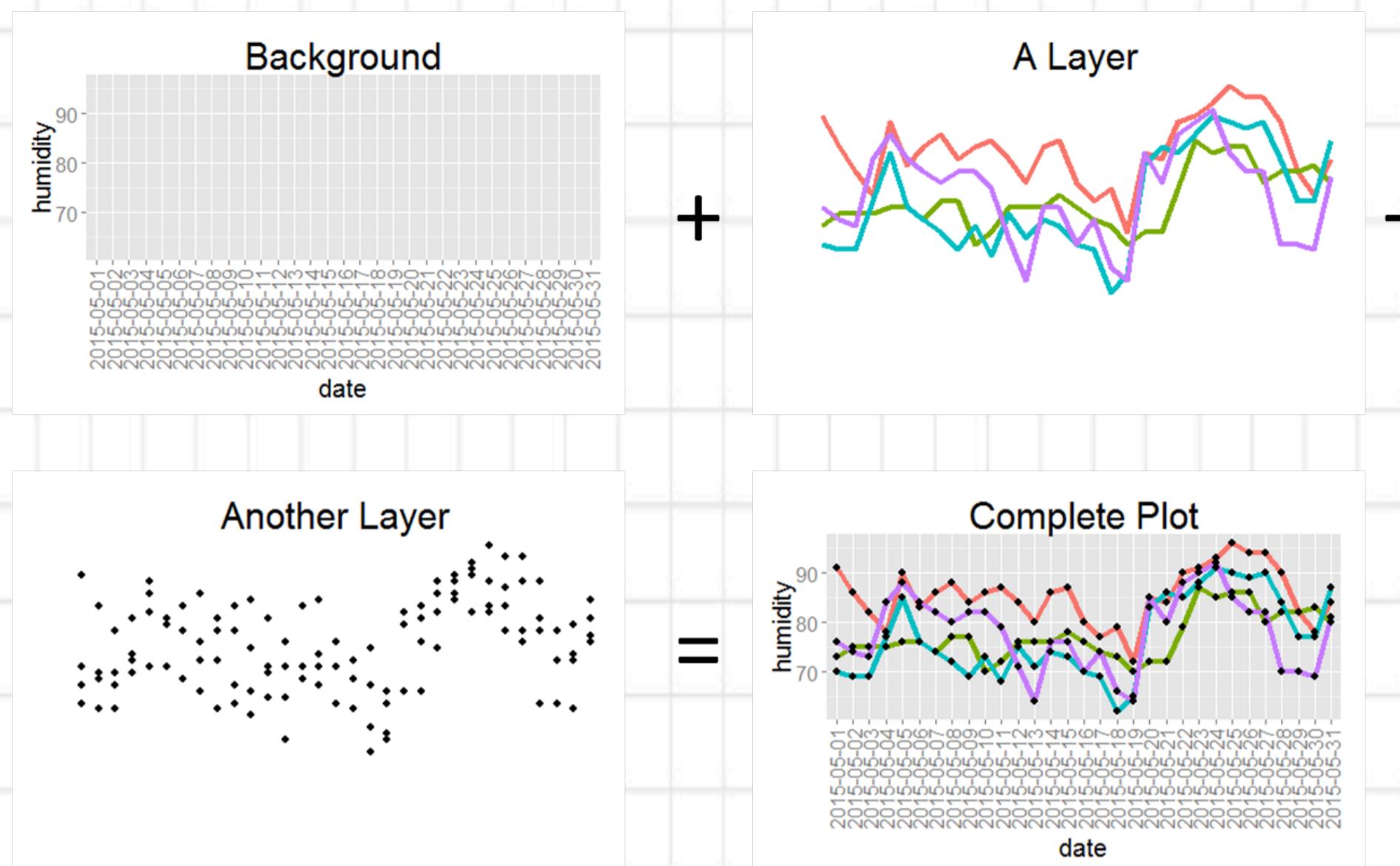
Data Visualization

- 清晰有效地傳達與溝通訊息
- 教學、研究、宣傳、
- 美學、功能兼顧
- 統計圖形、訊息可視化
- 參考Johnson於DSHC meetp的DataViz 介紹
- Bonus from AlgerHighchart Introduction

ggplot2簡介

- 2015年，最受歡迎的R套件之一
- R環境下的繪圖套件
- 取自 “The Grammar of Graphics” (Leland Wilkinson, 2005)
- 設計理念
 - 採用圖層系統
 - 用抽象的概念來控制圖形，避免細節繁瑣
 - 圖形美觀
- 最受歡迎的視覺視套件

The Anatomy of a Plot



ggplot2基本架構

- 資料 (data) 和映射 (mapping)
- 幾何對象 (geometric)
- 座標尺度 (scale)
- 統計轉換 (statistics)
- 座標系統 (coordinate)
- 圖層 (layer)
- 刻面 (facet)
- 主題 (theme)

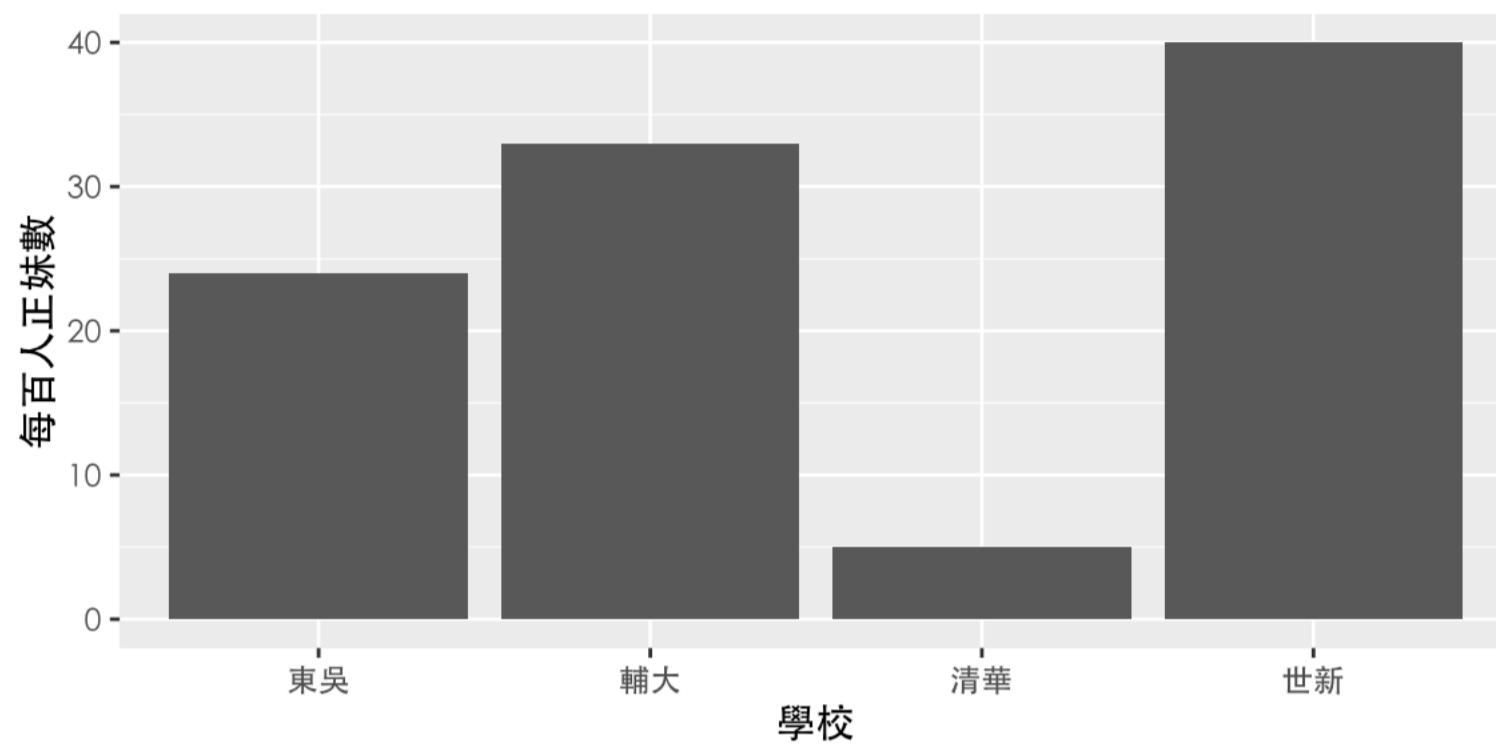
ggplot2 基本語法

```
ggplot(data=..., aes(x=..., y=...)) + geom_xxx(...) +  
stat_xxx(...) + facet_xxx(...) + ...
```

- **ggplot** 描述 data 從哪來
- **aes** 描述圖上的元素跟 data 之類的對應關係
- **geom_xxx** 描述要畫圖的類型及相關調整的參數
- 常用的類型諸如：**geom_bar, geom_points, geom_line, geom_polygon**

馬上來個範例

```
data <- data.frame(學校 = c("輔大", "東吳", "世新", "清華"), 每百人正妹數 = c(33, 24, 40, 5))  
#隨便定義一個資料表  
ggplot(data = data, aes(x = 學校, y = 每百人正妹數)) + geom_bar(stat = "identity") +  
  theme_grey(base_family = "STHeiti")
```



首先請先安裝以下套件

```
#install.packages(c("ggplot2", "data.table", "dplyr", "reshape2"), repos="http://cran.csie.ntu.edu.tw/")  
library(ggplot2)  
library(dplyr)  
library(reshape2)  
library(data.table)
```



：巧虎，為什麼還要安裝dplyr還有reshape2呢？



：因為在進行資料視覺化的時候，一定會經過資料得清理與萃取，得到一個dataframe後，才會進行畫圖，所以其實資料整理是很重要的！！



：原來如此呀，難怪你自稱健保資料庫掃地工人～

進行資料分析前，總要先有資料吧！？

```
# 瞭解現在我們所處在的路徑  
getwd()  
# 設定我們檔案存放的路徑  
setwd()  
# 讀檔起手式  
temp <- read.csv("歷年護理人員資料.csv") #請輸入正確的檔案路徑  
# 若讀入的是亂碼，試試以下  
temp <- read.csv("歷年護理人員資料.csv",fileEncoding = 'big5') #如果你是mac  
# 或是  
temp <- read.csv("歷年護理人員資料.csv",fileEncoding = 'utf-8') #如果你是windows
```



以為開始了嗎？

- 進行分析前，我們必須去了解資料的型態與特性

```
str(temp)
```

```
'data.frame': 25634 obs. of 6 variables:  
 $ 性別    : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...  
 $ 醫療層級: Factor w/ 4 levels "地區醫院","基層院所",...: 4 4 4 4 4 4 4 4 4 4 ...  
 $ 縣市    : Factor w/ 23 levels "", "高雄市", "花蓮縣", ...: 2 2 2 2 2 2 2 2 2 2 ...  
 $ 年齡    : int 20 20 20 20 20 20 20 20 21 21 ...  
 $ 年度    : int 2005 2006 2007 2008 2009 2010 2011 2012 2005 2006 ...  
 $ 人數    : int 56 17 11 1 5 5 7 7 71 75 ...
```

身為資料分析師，一定要有的好習慣！

- 瞭解基本的個變數統計量值

summary(temp)

性別	醫療層級	縣市	年齡	年度
F:21468	地區醫院:7774	台北市 : 1865	Min. :18	Min. :2005
M: 4166	基層院所:7053	台中市 : 1841	1st Qu.:28	1st Qu.:2007
	區域醫院:7409	新北市 : 1735	Median :37	Median :2009
	醫學中心:3398	高雄市 : 1685	Mean :38	Mean :2009
		臺南市 : 1606	3rd Qu.:47	3rd Qu.:2011
		彰化縣 : 1588	Max. :65	Max. :2012
		(Other):15314		
人數				
Min. : 1				
1st Qu.: 2				
Median : 9				
Mean : 30				
3rd Qu.: 31				
Max. : 729				

資料介紹

	性別	醫療層級	縣市	年齡	年度	人數
11988	F	地區醫院	臺南市	26	2005	93
14728	F	地區醫院	彰化縣	53	2012	6
15286	F	基層院所	花蓮縣	37	2011	6
15225	F	基層院所	花蓮縣	30	2006	11
5936	F	區域醫院	台中市	32	2009	260
19682	F	基層院所	新北市	42	2011	53
21969	M	醫學中心	台中市	27	2005	3

欄位說明

- 性別：M：男生，F：女生
- 醫療層級：1：醫學中心，2：區域醫院，3：地區醫院，4：基層院所
- 縣市：台灣22個縣市
- 年齡：台灣護理人員的年齡
- 年度：第幾年的資料
- 人數：護理人員的人數
- 敘述：
 - 這個資料表由健保資料庫的歷年醫事人員基本資料檔轉換而得，並非常見的原始資料檔。

量化 v.s. 量化 : Line Chart

Line chart

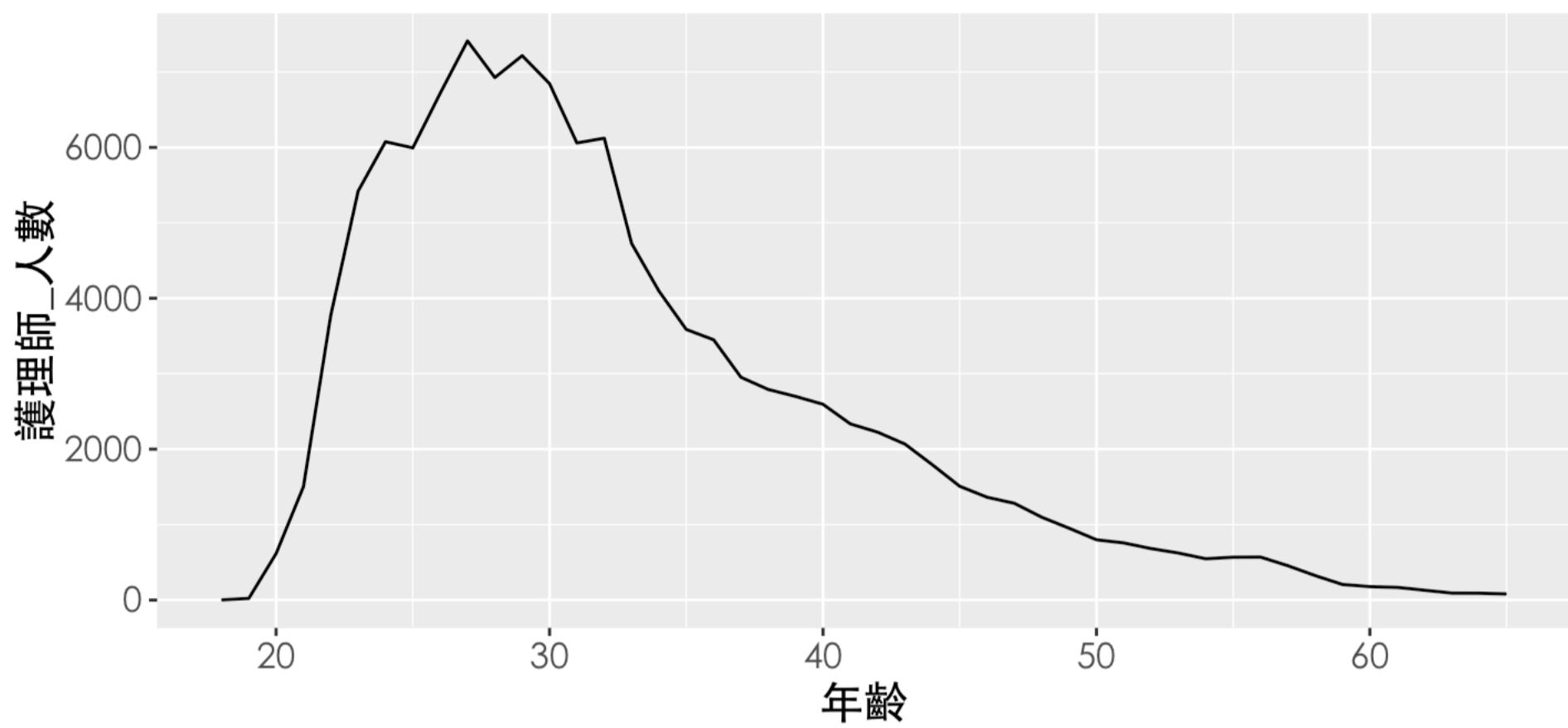
- **geom_line**
- 首先來看看2008年各個年齡層的護理師人數吧！
- 是不是發現護理師主要都集中在某個年齡層呢？？

```
temp_age <- temp %>%
  filter(年度 == "2008") %>% #挑出年度為2008年
  group_by(年齡) %>% #根據年齡分群
  summarise(護理師_人數 = sum(人數)) #計算總人數
# 資料整理可以直接複製貼上
thm <- function() theme(text=element_text(size = 15, family = "STHeiti")) # 控制字體與大小
# STHeiti是只有Mac才有的字體，用來解決Mac系統中文顯示錯誤的問題
# Windows系統的使用者請忽略 `+ thm()` 指令
## Line chart
```

Line chart

可以與第10頁做簡單的對照

```
ggplot(temp_age,aes(x = 年齡, y = 護理師_人數))+geom_line()+thm()
```



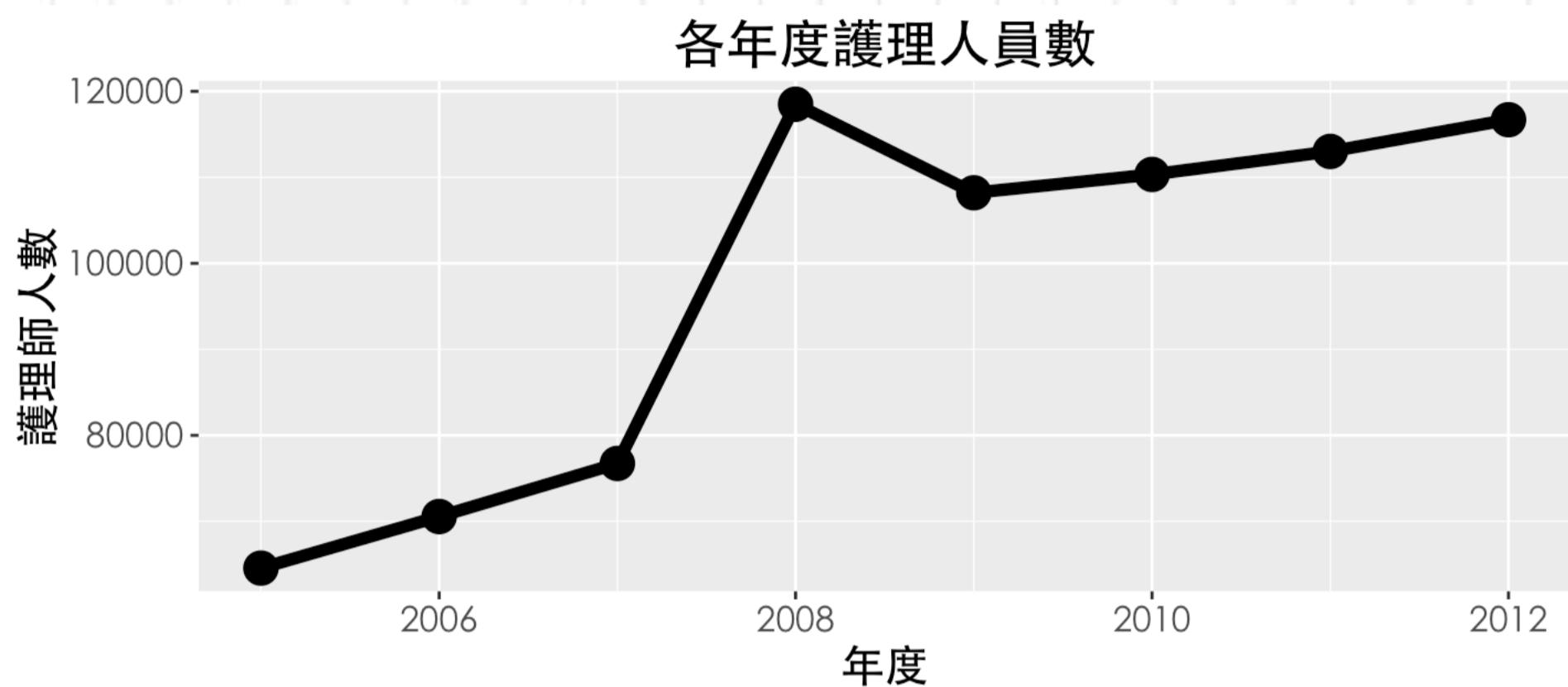
Line chart and Scatter plot

- 歷年的醫護人員是不是逐年的再增加呢？

```
temp_year <- temp %>%
  group_by(年度) %>% #根據年度分群
  summarise(護理師_人數 = sum(人數)) #計算總人數
# 資料整理可以直接複製貼上
```

Line chart and Scatter plot

```
ggplot(temp_year,aes(x = 年度 , y = 護理師_人數)) +  
  geom_line(size = 2) + geom_point(size = 5) + # 控制點與線的大小粗細  
  thm() + labs(title = "各年度護理人員數", x = "年度", y = "護理師人數") # lab用來幫圖形的標  
題、x軸與y軸做命名
```



How to plot multiple lines?

- 比較高雄市、台中市、台南市，中南部的直轄市，歷年的護理人員數是否有差異？
- 寬資料轉長資料(Wide format to long format)

```
temp_t <- read.csv("我是寬資料.csv", header=T)
temp_long <- temp_t %>% melt(id='year') #id是將保留的欄位名稱
colnames(temp_long) <- c('年度', '地區', '人數')
```

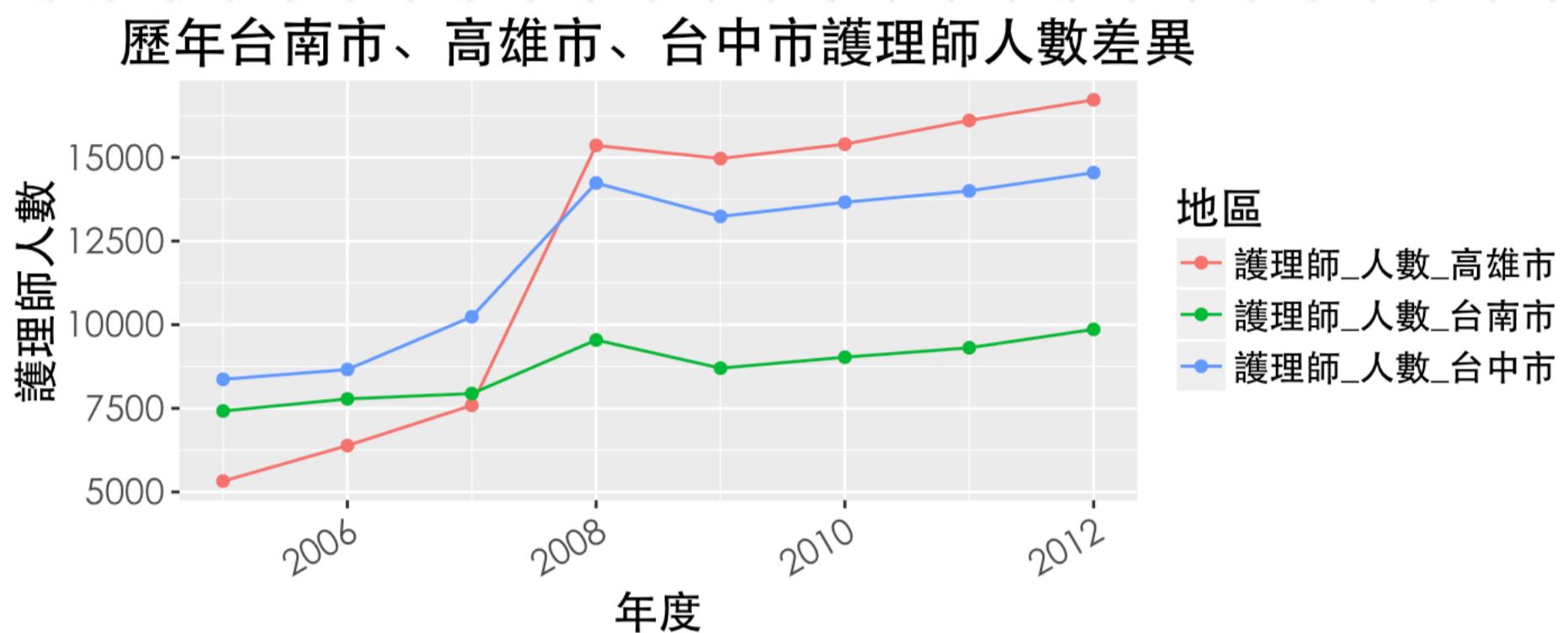
轉換結果

Wide format					Long format
	year	護理師_人數_高雄市	護理師_人數_台南市	護理師_人數_台中市	
1	2005	5324	7420	8367	
2	2006	6385	7781	8660	
3	2007	7589	7941	10236	
4	2008	15358	9543	14236	
5	2009	14967	8700	13236	
6	2010	15397	9028	13662	
7	2011	16107	9310	14000	
8	2012	16722	9861	14545	

	年度	地區	人數
1	2005	護理師_人數_高雄市	5324
2	2006	護理師_人數_高雄市	6385
3	2007	護理師_人數_高雄市	7589
4	2008	護理師_人數_高雄市	15358
5	2009	護理師_人數_高雄市	14967
6	2010	護理師_人數_高雄市	15397
7	2011	護理師_人數_高雄市	16107
8	2012	護理師_人數_高雄市	16722
9	2005	護理師_人數_臺南市	7420
10	2006	護理師_人數_臺南市	7781
11	2007	護理師_人數_臺南市	7941
12	2008	護理師_人數_臺南市	9543
13	2009	護理師_人數_臺南市	8700
14	2010	護理師_人數_臺南市	9028
15	2011	護理師_人數_臺南市	9310
16	2012	護理師_人數_臺南市	9861
17	2005	護理師_人數_台中市	8367
18	2006	護理師_人數_台中市	8660

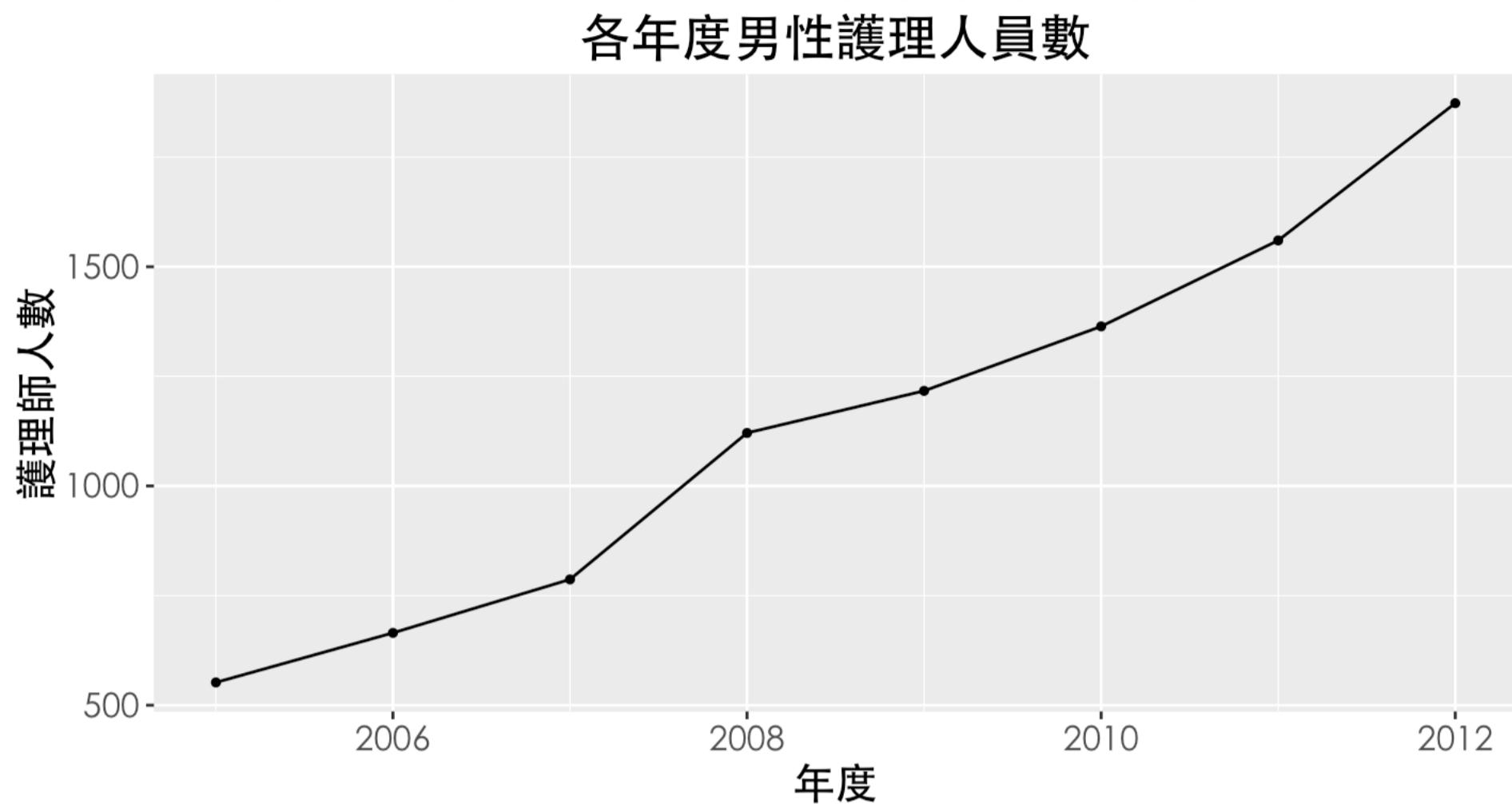
Multiple line 出來吧！！！

```
# group按照不同區域劃線  
ggplot(temp_long, aes(x = 年度, y = 人數 , group = 地區, color = 地區)) +  
  geom_line() +  
  geom_point() + theme() +  
  labs(title="歷年臺南市、高雄市、台中市護理師人數差異",x="年度",y="護理師人數") +  
  theme(axis.text.x = element_text(angle = 30, hjust = 1)) # 控制x軸字體傾斜程度
```



小挑戰

- 試著畫出歷年男性護理師的人數趨勢？
- 如下圖



解答來囉！！

- 良好的註解與排列習慣，會讓你編輯程式更有效率！

```
## 資料整理
temp_t <- temp %>%
  filter(性別=="M") %>% #挑出男性
  group_by(年度) %>% #根據年度分群
  summarise(護理師_人數 = sum(人數)) #計算總人數

## 繪圖
ggplot(temp_t,aes(x = 年度 , y = 護理師_人數)) +
  geom_line(size = 0.5) + geom_point(size = 1)+ # 控制點與線的大小粗細
  thm() + labs(title = "各年度男性護理人員數" , x = "年度" , y = "護理師人數")
```

質化 v.s. 量化：Bar Chart

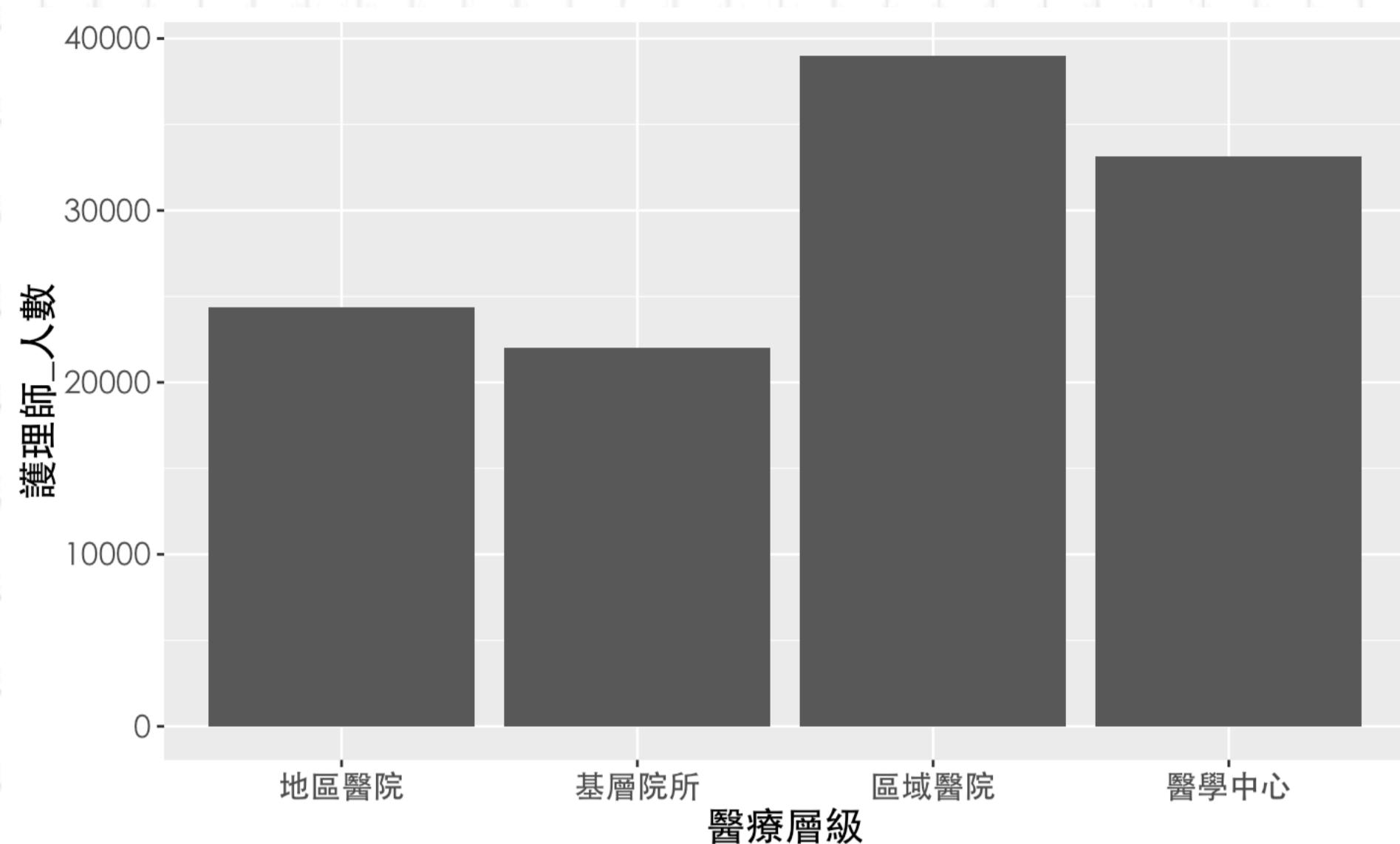
- 以2008年來說，不同的醫療層級，護理人員的分佈狀況如何呢？

#資料整理

```
temp_hosp <- temp %>%
  filter(年度 == "2008") %>% #選擇年度2008
  group_by(年度, 醫療層級) %>% #對年度與醫療層級分群
  summarise(護理師_人數 = sum(人數))

#畫圖
ggplot(temp_hosp, aes(x = 醫療層級, y = 護理師_人數)) +
  geom_bar(stat = 'identity') + thm()# stat='identity'以表格的值做為bar的高度
```

Bar Chart



Bar Chart

- 如何控制長條圖的排序呢？有兩種比較簡單的方式。
- 手動修改levels

```
class(temp_hosp$醫療層級) #看看這個變數的屬性類型
```

```
[1] "factor"
```

```
# 發現他是integer的屬性，所以我們要將其轉成factor來改變他的順序
```

```
temp_hosp$醫療層級 <- factor(temp_hosp$醫療層級, levels = c("醫學中心", "區域醫院", "地區醫院", "基層院所"))
```

- 利用**arrange**事先依照y軸對應的數值的大小來排序

```
# 我們也可以在一開始先對資料排序完在畫圖
```

```
temp_hosp <- temp_hosp %>% arrange(護理師_人數)
```

```
# 此時在接上畫圖的code就會自動排序囉！
```

小挑戰！？請將資料整理如下：

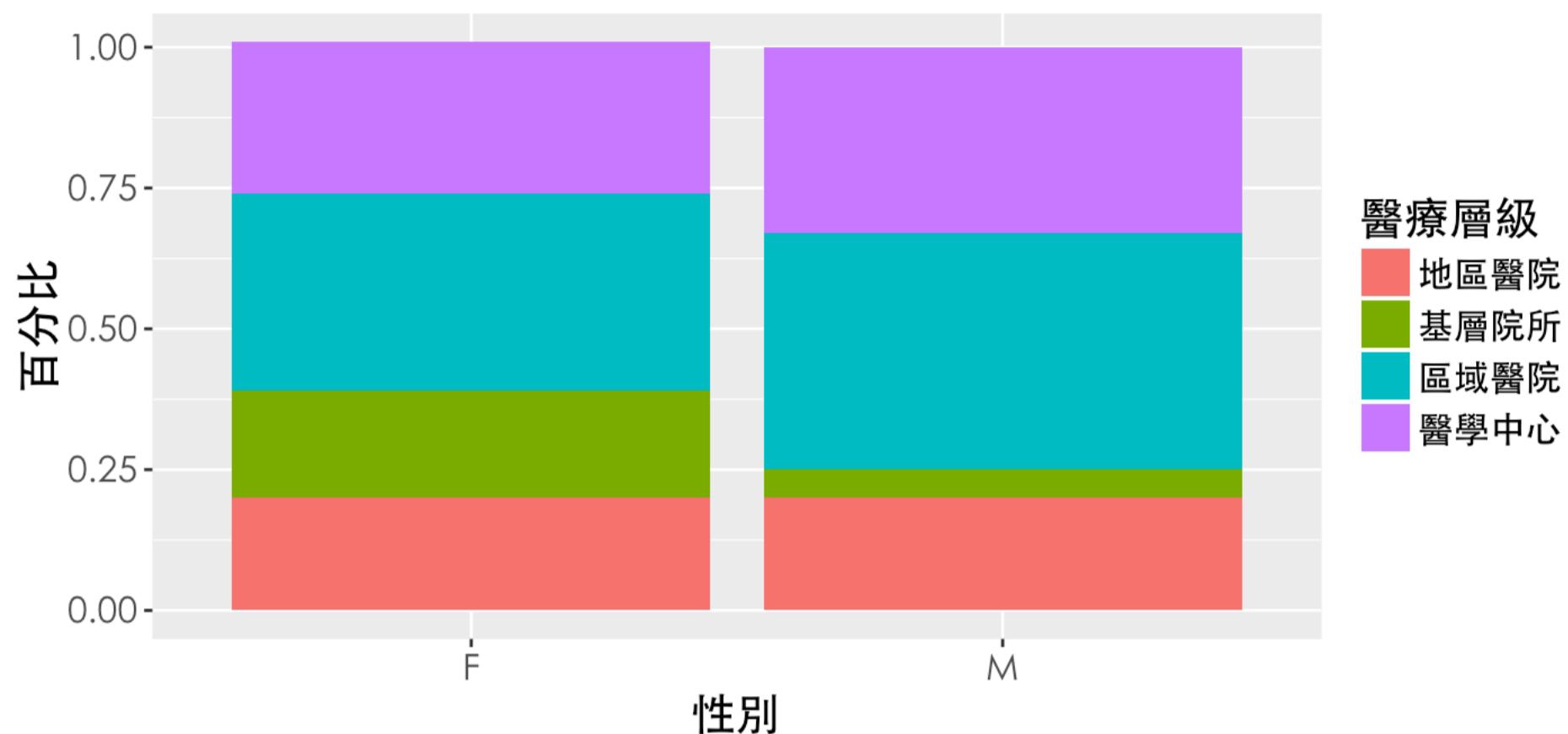
	性別	醫療層級	護理師_人數	比例
1	F	地區醫院	24106	0.20
2	F	基層院所	21945	0.19
3	F	區域醫院	38561	0.33
4	F	醫學中心	32764	0.28
5	M	地區醫院	235	0.21
6	M	基層院所	54	0.05
7	M	區域醫院	457	0.41
8	M	醫學中心	375	0.34

解答code

```
# 資料整理大挑戰~~~  
temp_MF <- temp %>%  
  group_by(性別,醫療層級) %>%  
  summarise(護理師人數 = sum(人數)) %>%  
  mutate(百分比 = round(護理師人數/sum(護理師人數),2))
```

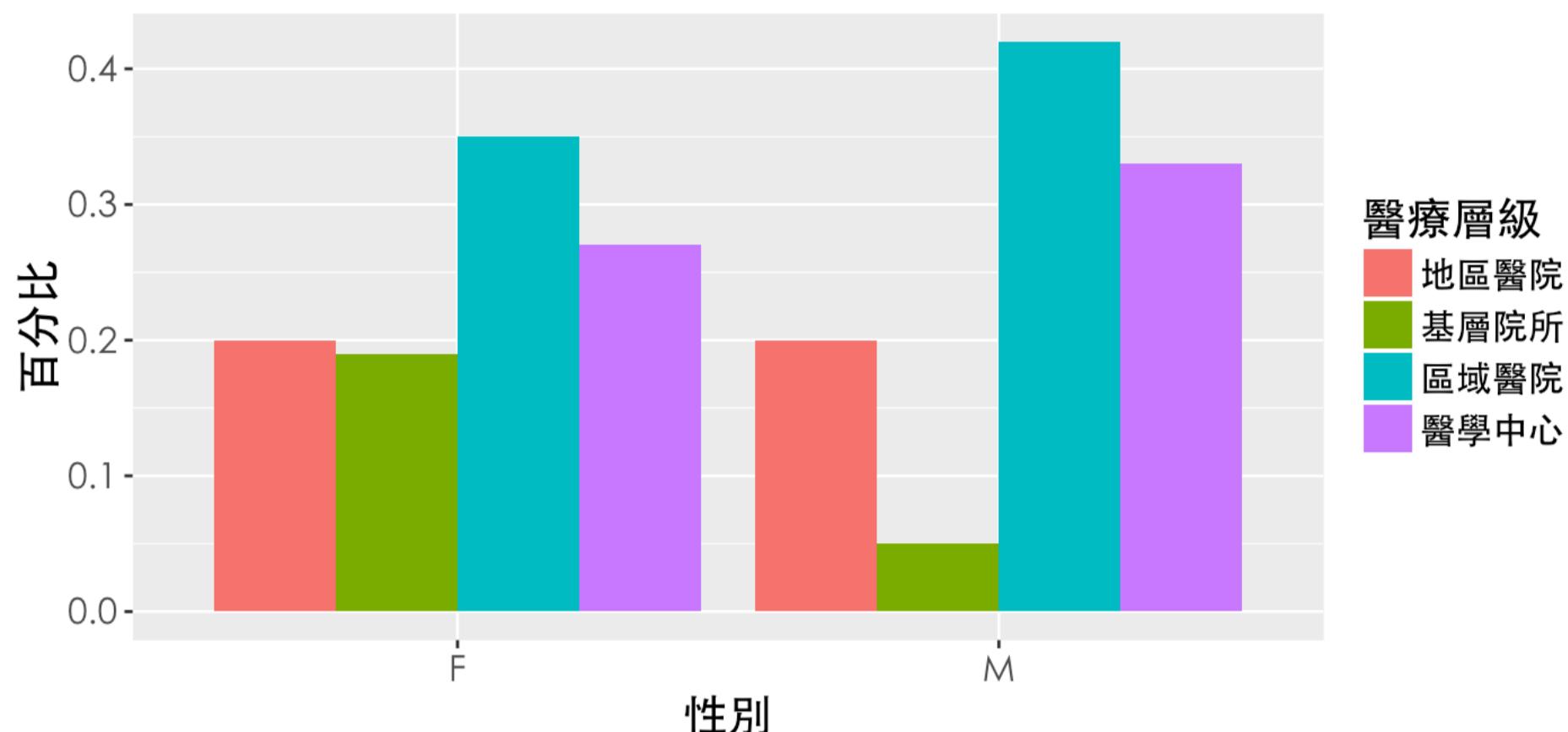
兩種類別.stack

```
ggplot(temp_MF,aes(x = 性別, y = 百分比 , fill = 醫療層級)) +  
  geom_bar(stat = 'identity' , position = 'stack') + thm()
```



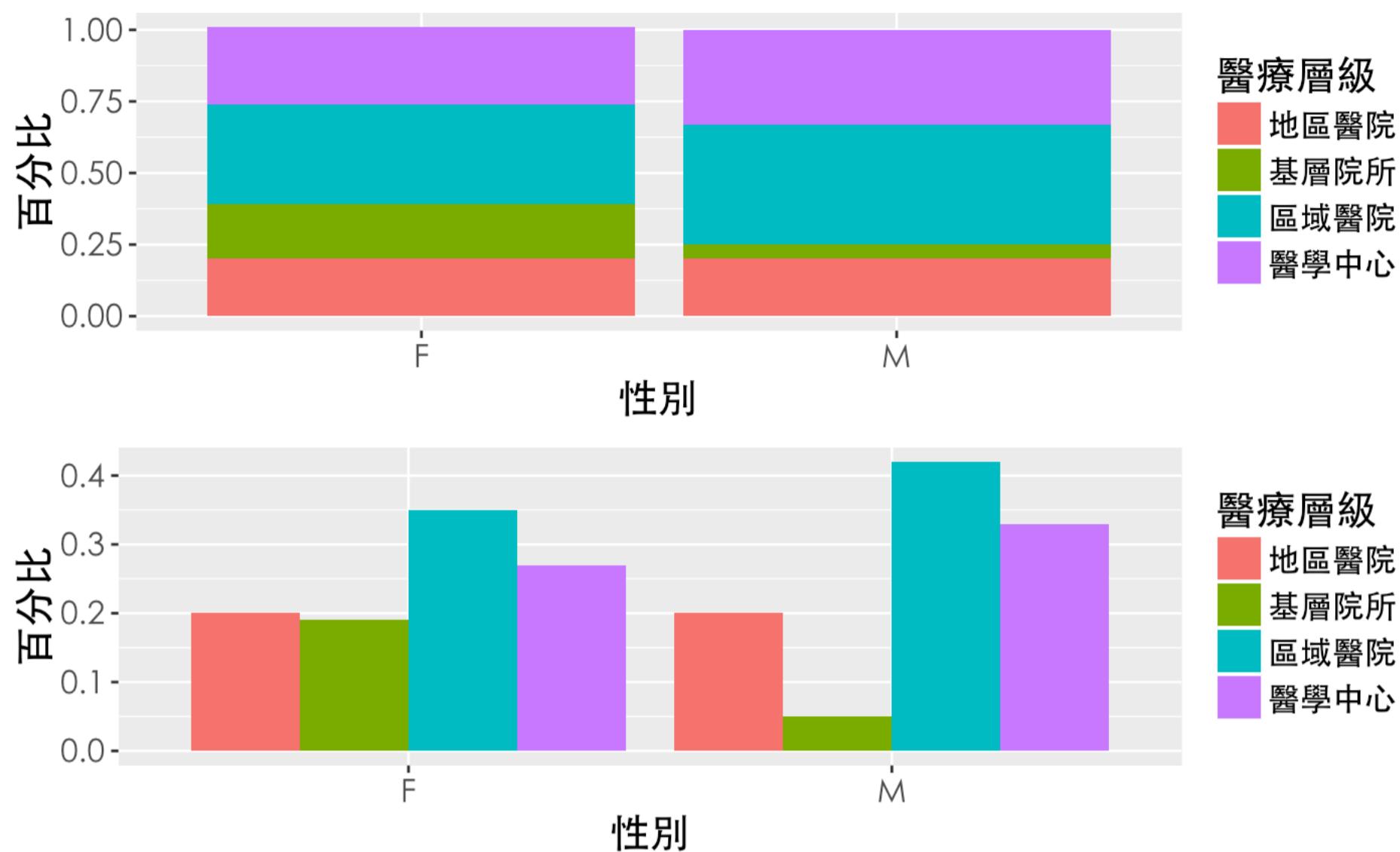
兩種類別.dodge

```
ggplot(temp_MF,aes(x = 性別, y = 百分比 , fill = 醫療層級 )) +  
  geom_bar(stat = 'identity' , position = 'dodge') + thm()
```



圖形思考？

stack與dodge之間的差異？



資料讀檔與下載

YouBike-Weather Data

```
# 讀檔起手式  
ubike <- read.csv('ubikebyhourutf8/ubike-hour-201502-utf8.csv') #請輸入正確的檔案路徑  
# 讀檔進階招式  
#ubike <- read.csv('檔案路徑',  
#                   colClasses = c("factor", "integer", "integer", "factor", "factor",  
#                                         "numeric", "numeric", "integer", "numeric", "integer",  
#                                         "integer", "numeric", "numeric", "integer", "integer",  
#                                         "numeric", "numeric", "numeric", "numeric", "numeric",  
#                                         "numeric"))  
# 讀檔大絕招  
#ubike <- fread('檔案路徑',  
#                 data.table = FALSE,  
#                 colClasses = c("factor", "integer", "integer", "factor",  
#                               "factor", "numeric", "numeric", "integer",  
#                               "numeric", "integer", "integer", "numeric",  
#                               "numeric", "integer", "integer", "numeric",  
#                               "numeric", "numeric", "numeric", "numeric",  
#                               "numeric"))
```

展示資料

	date	hour	sno	sarea		sna	lat	lng	tot	avg.sbi
1	2015-02-01	0	1	信義區	捷運市政府站(3號出口)	25	122	180	4.15	
2	2015-02-01	0	2	大安區	捷運國父紀念館站(2號出口)	25	122	48	1.32	
3	2015-02-01	0	3	信義區	台北市政府	25	122	40	0.41	
4	2015-02-01	0	5	信義區	興雅國中	25	122	60	1.22	
5	2015-02-01	0	7	信義區	信義廣場(台北101)	25	122	80	2.95	
6	2015-02-01	0	8	信義區	世貿三館	25	122	60	1.98	
	max.sbi	min.sbi	std.sbi	avg.bemp	max.bemp	min.bemp	std.bemp	temp		
1	11	0	3.64	176	180	169	3.64	14		
2	5	0	1.38	47	48	43	1.38	14		
3	1	0	0.49	40	40	39	0.49	14		
4	3	0	0.97	58	59	56	0.97	14		
5	9	0	3.35	76	79	69	3.48	14		
6	7	0	2.54	57	59	52	2.54	14		
	humidity	pressure	max.anemo	rainfall						
1	87	1026	1.78	0						
2	87	1025	0.85	0						
3	87	1026	1.61	0						
4	88	1026	1.91	0						
5	87	1026	1.91	0						
6	87	1026	1.62	0						

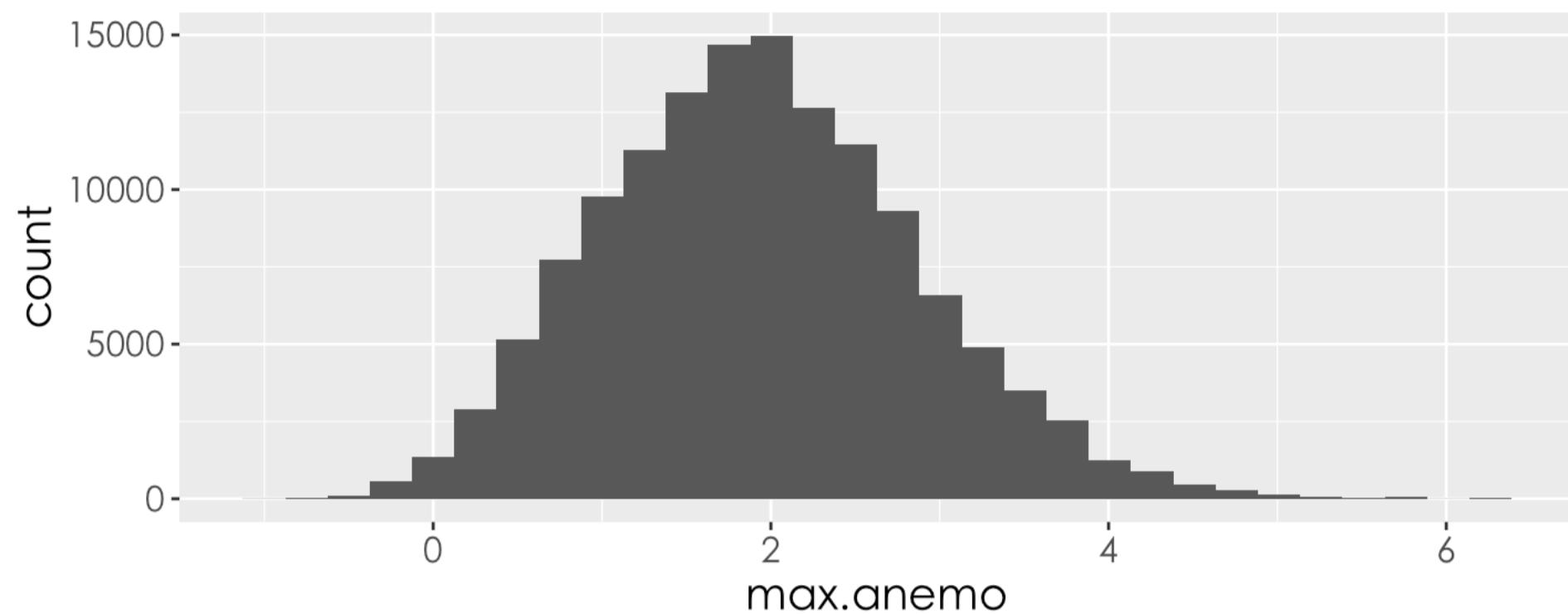
欄位說明

	name.eng	name.cht		name.eng	name.cht
1	date	日期	12	std.sbi	車輛數標準差
2	hour	時間	13	avg.bemp	平均空位數
3	sno	場站代號	14	max.bemp	最大空位數
4	sarea	場站區域	15	min.bemp	最小空位數
5	sna	場站名稱	16	std.bemp	空位數標準差
6	lat	緯度	17	temp	平均氣溫
7	lng	經度	18	humidity	溼度
8	tot	總停車格	19	pressure	氣壓
9	avg.sbi	平均車輛數	20	max.anemo	最大風速
10	max.sbi	最大車輛數	21	rainfall	降雨量
11	min.sbi	最小車輛數			

單一數值：Histogram

- `geom_histogram`

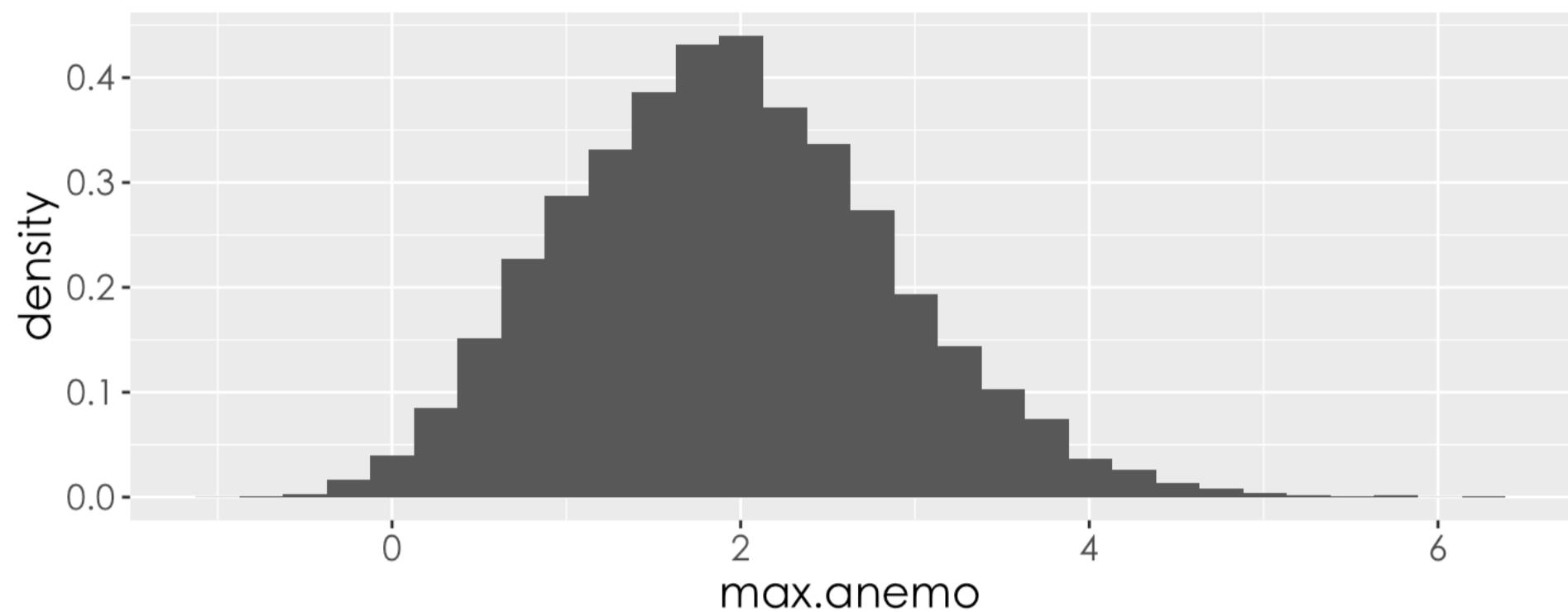
```
ggplot(ubike) +  
  geom_histogram(aes(x = max.anemo, y =..count..)) + thm()
```



Histogram

- `aes(y=..count..)` vs. `aes(y=..density..)`

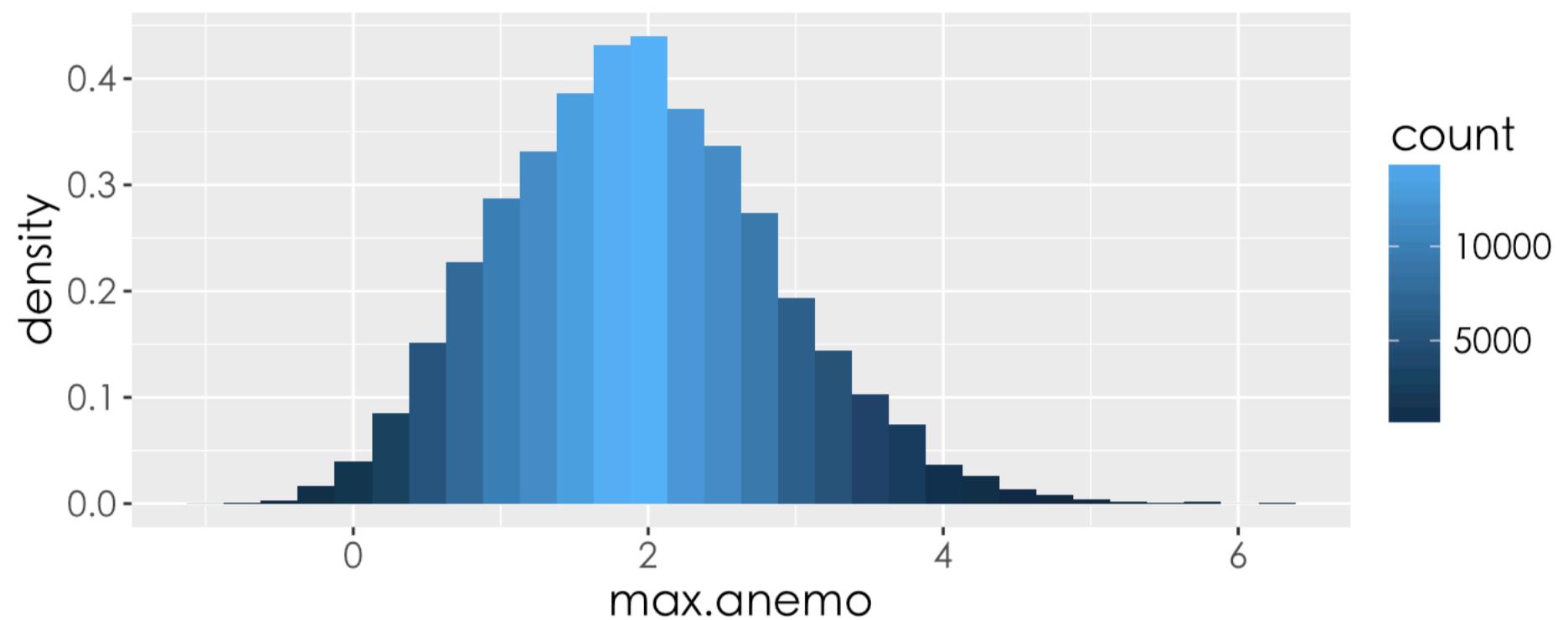
```
ggplot(ubike) +  
  geom_histogram(aes(x = max.anemo , y = ..density..)) + thm()
```



Histogram

- `aes(x=..., y=..., fill=...)`

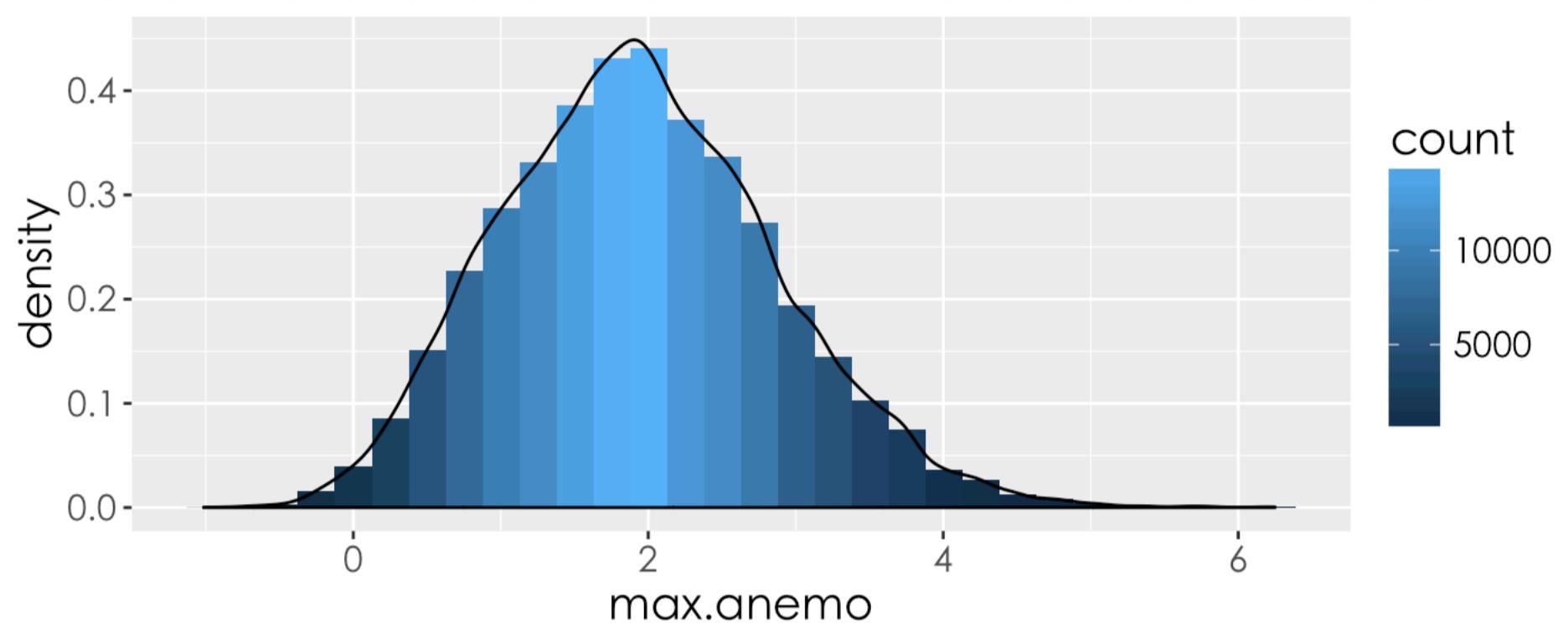
```
ggplot(ubike) +  
  geom_histogram(aes(x = max.anemo, y = ..density.., fill = ..count..)) + thm()
```



Histogram + Density

- `geom_histogram() + geom_density()`

```
ggplot(ubike, aes(x = max.anemo)) +  
  geom_histogram(aes(y = ..density.., fill = ..count..)) +  
  geom_density() + thm()
```



量化 v.s. 量化 : Scatter Plot

繪圖之前的整理資料

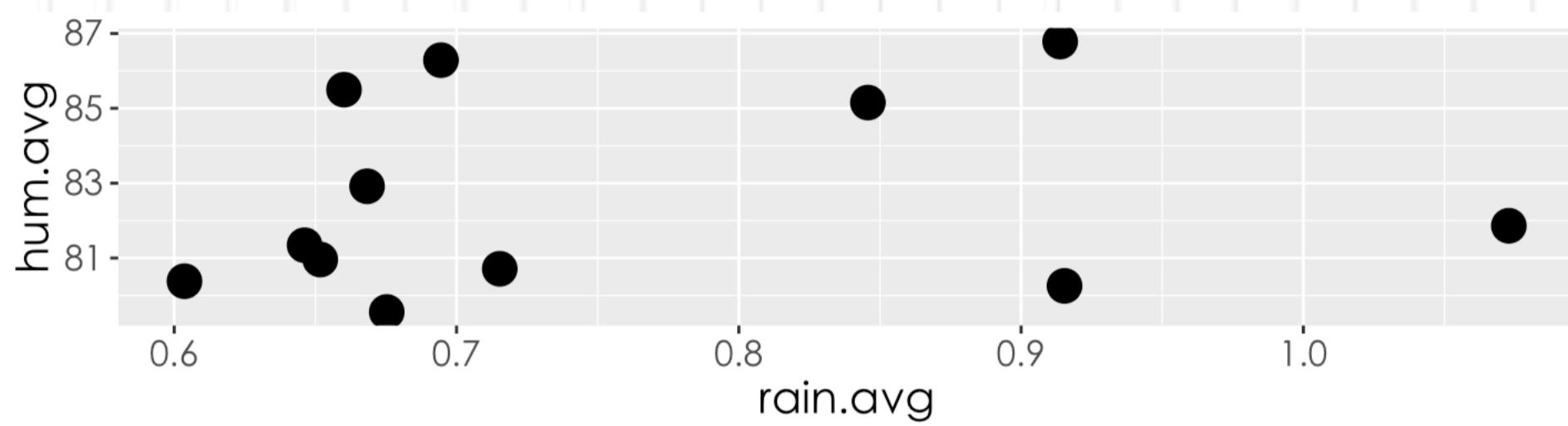
- 文山區各站點在"2015-02"的平均溼度 vs. 平均雨量

	sna	rain.avg	hum.avg
1	國立政治大學	0.91	86.78
2	捷運動物園站(2號出口)	0.69	86.29
3	捷運景美站	0.68	79.56
4	捷運木柵站	0.66	85.50
5	考試院	0.92	80.26
6	羅斯福景隆街口	0.60	80.38
7	師範大學公館校區	0.72	80.71
8	台北花木批發市場	0.65	80.96
9	臺北市立景美女中	1.07	81.86
10	文山行政中心	0.85	85.15
11	文山運動中心	0.67	82.92
12	興豐公園	0.65	81.34

Scatter plot

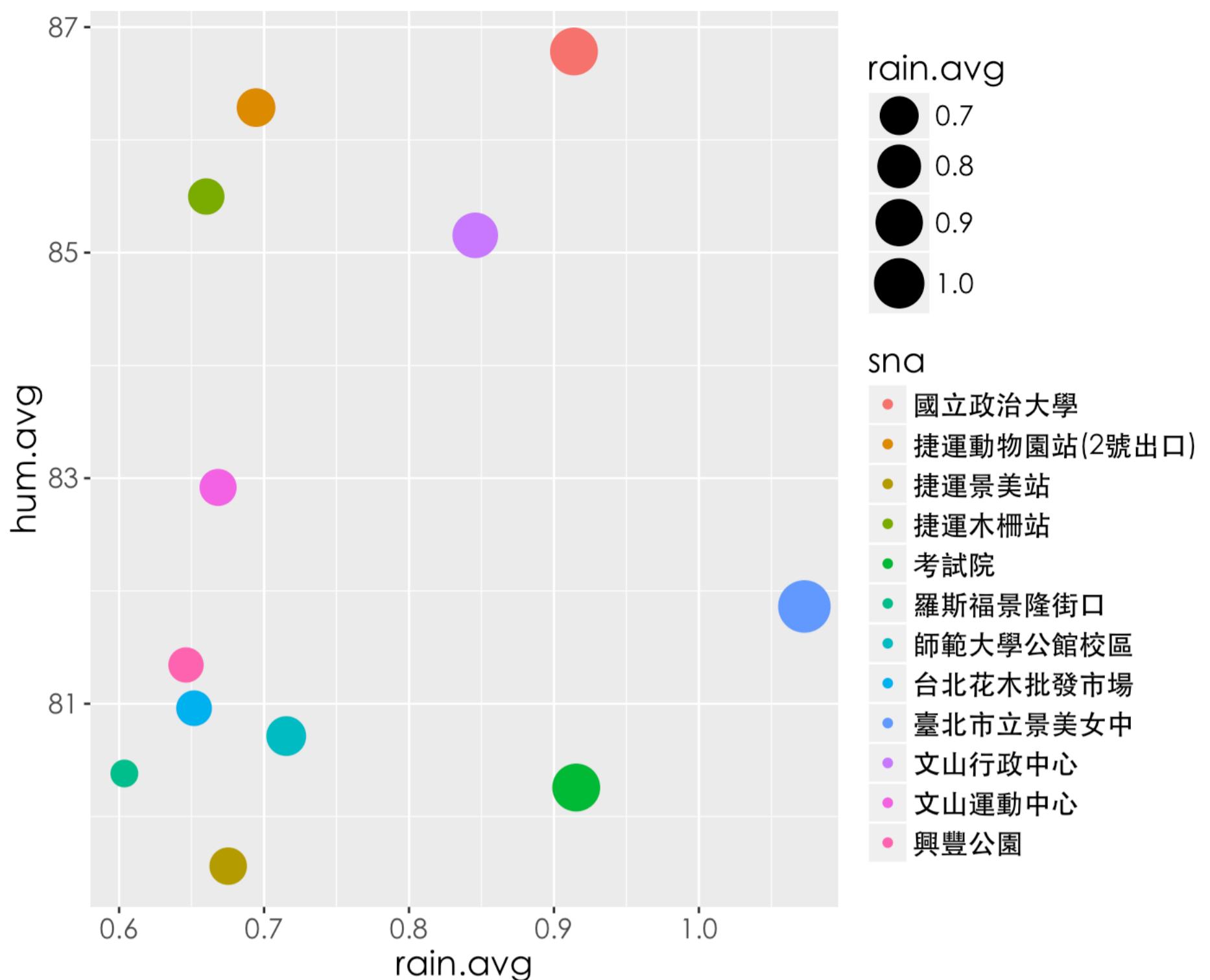
- `geom_point()`
- 參數 `size` 放在 `aes()` 函數外面，表示所有的資料固定用相同的 `size`

```
ggplot(x3) +  
  geom_point(aes(x = rain.avg, y = hum.avg), size=5) + #size控制點的大小  
  theme()
```



Grouped scatter plot

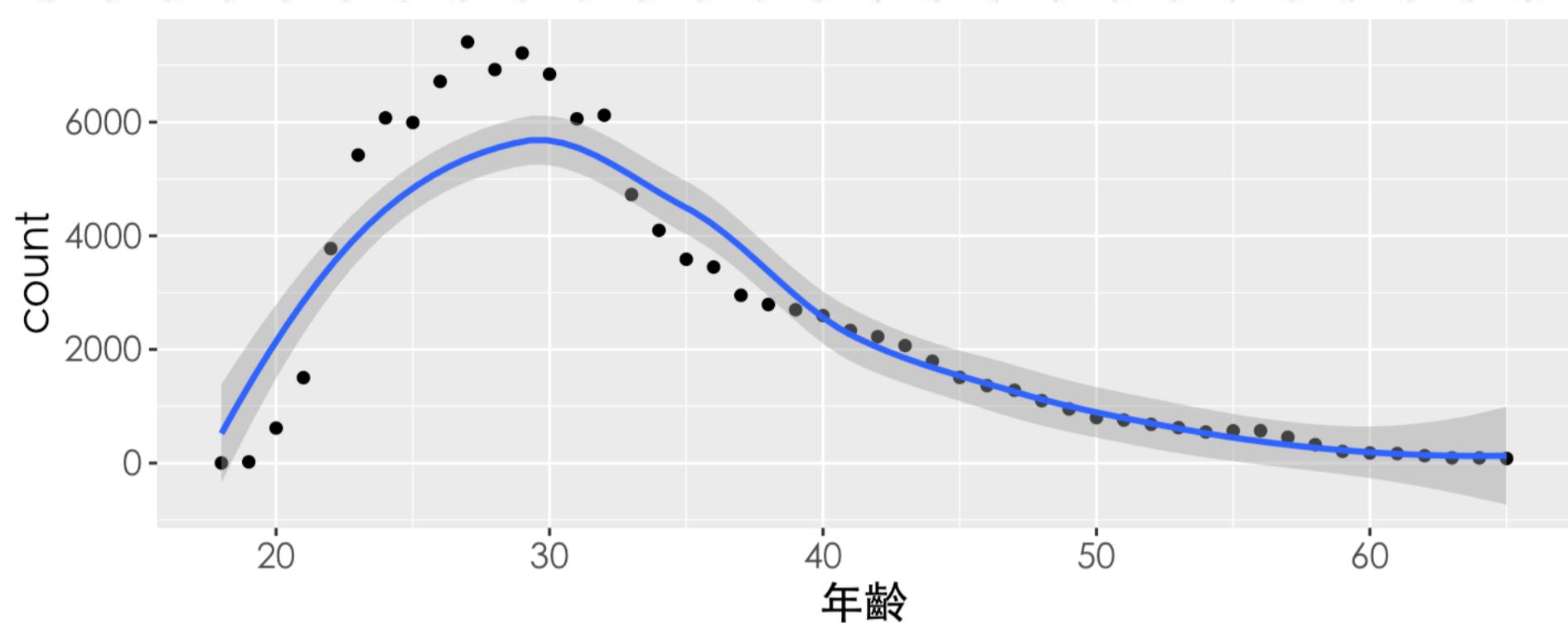
```
ggplot(x3) +  
# aes裡的 colour 和 size 可依資料調整顏色和大小，使用 scale_size() 控制資料  
geom_point(aes(x = rain.avg, y = hum.avg, colour=sna, size=rain.avg)) +  
scale_size(range=c(5,10)) +  
thm()
```



進階技巧傳承篇

statistics

```
temp_age <- temp %>%
  filter(年度 == "2008") %>% # 挑出2008年度
  group_by(年齡) %>% # 根據年齡分群
  summarise(count = sum(人數)) # 計算人數
ggplot(temp_age,aes(x = 年齡, y = count)) +
  geom_point() + stat_smooth() + theme()
```



facet

- 資料整理

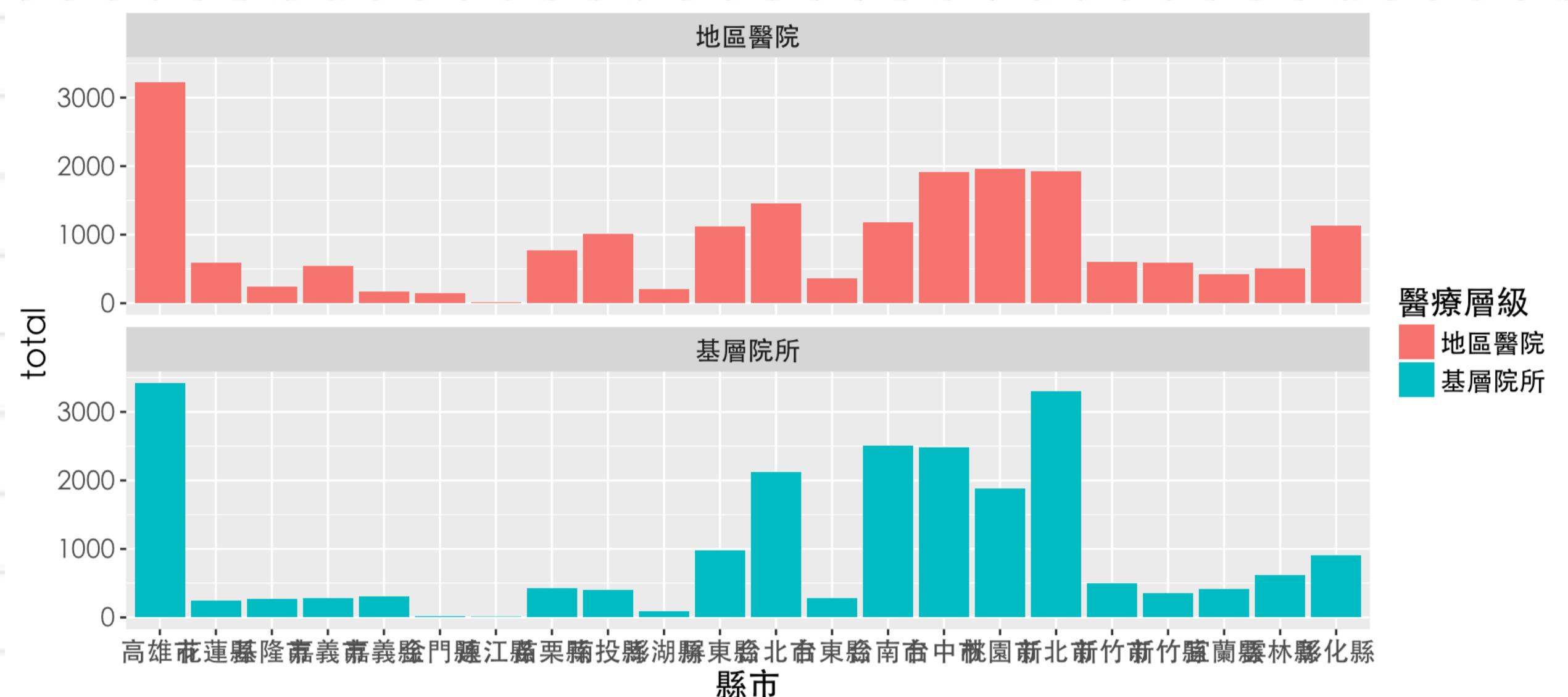
```
temp_facet <- temp %>%
  filter(年度 == "2012") %>%
  filter(醫療層級 == "地區醫院" | 醫療層級=="基層院所") %>%
  group_by(醫療層級, 縣市) %>% #選擇年度、醫療層級、縣市作為分群
  summarise(total = sum(人數)) # 計算分群下的總人數
```

facet

- theme(axis.text.x = element_text(angle = 90, hjust = 1)) #控制字的方向
- coord_flip()

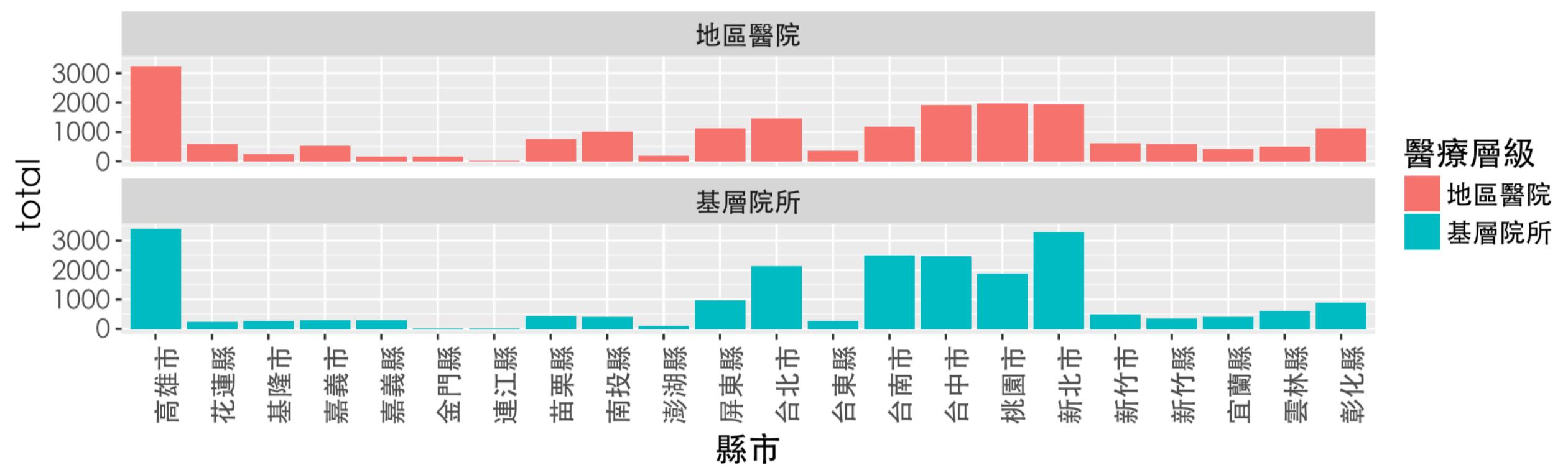
```
ggplot(temp_facet,aes(x = 縣市, y = total,fill = 醫療層級 ))+  
  geom_bar(stat='identity') +  
  thm() +  
  facet_wrap( ~ 醫療層級, nrow = 2)
```

有沒有發現怪怪der？？？



更動字的方向

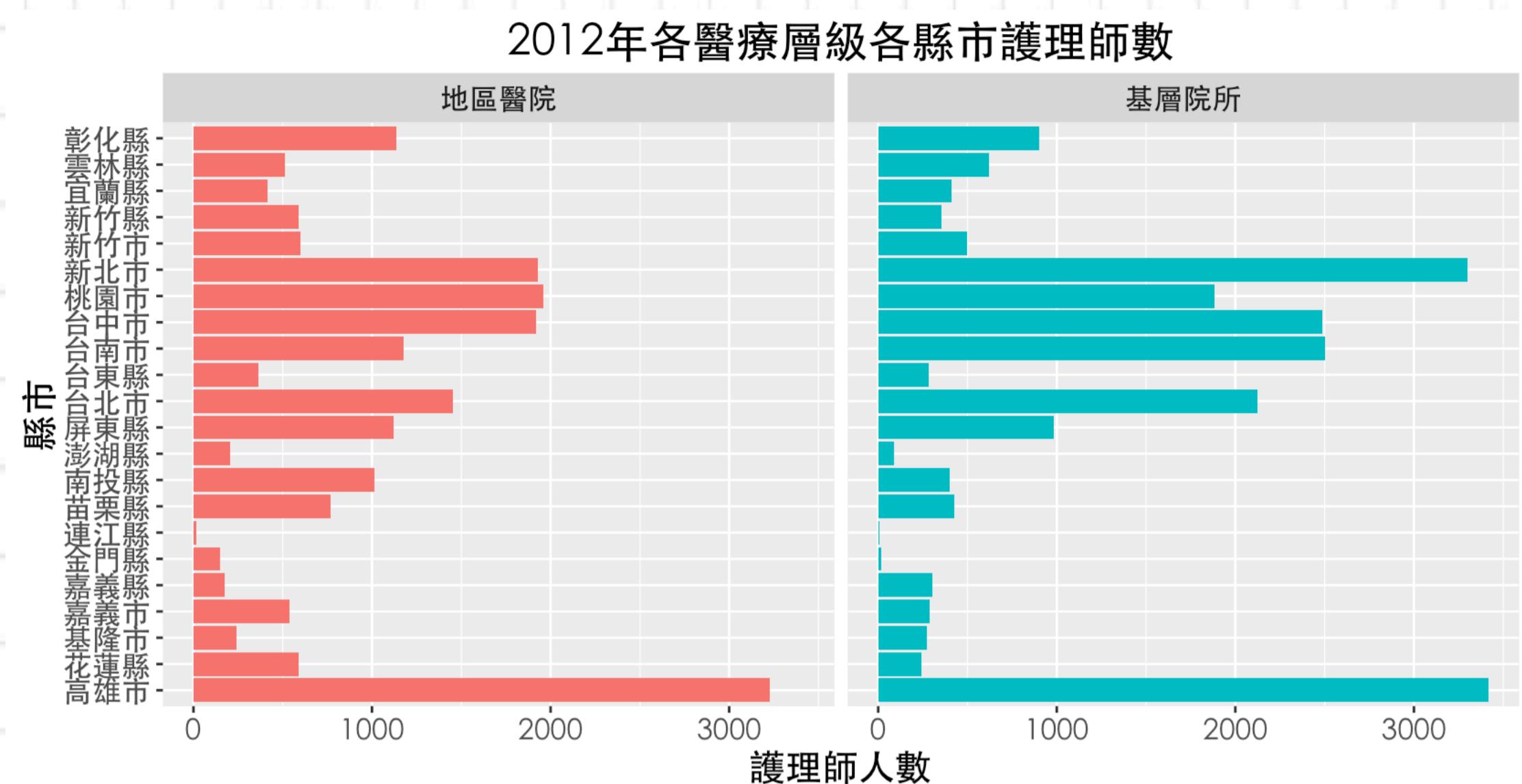
```
ggplot(temp_facet,aes(x = 縣市, y = total,fill = 醫療層級))+
  geom_bar(stat='identity') +
  thm() +
  facet_wrap(~ 醫療層級, nrow = 2) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) #控制字的方向
```



coord_flip()

```
ggplot(temp_facet,aes(x = 縣市, y = total, fill = 醫療層級)) +  
  geom_bar(stat = 'identity') +  
  thm() +  
  facet_wrap(~醫療層級, ncol = 2) + coord_flip() +  
  labs(title="2012年各醫療層級各縣市護理師數",x = "縣市",y = "護理師人數")
```

coord_flip()



圖形輸出

- 利用RStudio UI介面存擋
- 命立列輸出
- 畫完圖之後，再存檔~~

```
ggsave('檔案名稱')  
ggsave("mtcars.pdf", width = 4, height = 4)  
ggsave("mtcars.png", width = 4, height = 4, dpi = 300)
```

當你學了螺旋丸，你一定想讓他變螺旋手裡劍！

傳說中的Plotly

- Plotly是一個資料視覺化的R套件，以簡單的方式，讓資料能夠產生互動的效果。
- 提供一個合作平台，使用者能夠將自己在R中繪製的圖存上屬於自己的Plotly平台上。
- [Plotly官方網站](#)
- 結合了各式各樣的API，包裝Python、R、Matlab、...等等
- 當然～ggplot2也能夠輕易地使用plotly轉換成互動式的圖表！！

套件安裝

- 直接從CRAN內下載就好囉！！

```
# Plotly is now on CRAN!
install.packages("plotly")
# install the latest development version (on GitHub) via devtools
```

- 或是從github上下載，但前提是先安裝devtools

```
# install.packages("devtools")
devtools::install_github("ropensci/plotly")
```

BarDemo(step1)

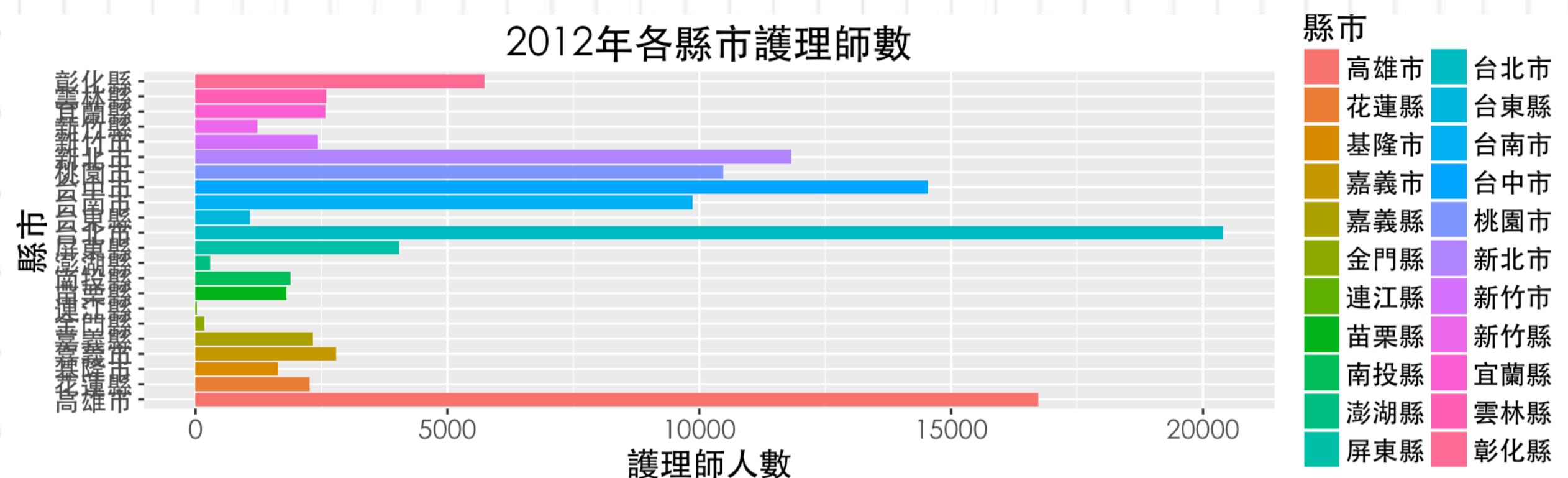
- 2012年各縣市的護理師人數呈現什麼趨勢呢？縣市之間差異很大嗎？
- 資料整理

```
temp_plot <- temp %>%
  filter(年度 == "2012") %>%
  group_by(縣市) %>%
  summarise(total = sum(人數))
```

BarDemo(step2)

- 先用ggplot2畫出

```
ggplot(temp_plot,aes(x = 縣市, y = total,fill = 縣市))+  
  geom_bar(stat = 'identity')+  
  thm() +  
  labs(title = "2012年各縣市護理師數", x = "縣市",y = "護理師人數") + coord_flip()
```



BarDemo(step3)

- 首先先將剛剛的圖存入一個物件

```
a <- ggplot(temp_plot,aes(x = 縣市, y = total ,fill = 縣市))+  
  geom_bar(stat = 'identity')+  
  theme() +  
  labs(title = "2012年各縣市護理師數",x = "縣市", y = "護理師人數") + coord_flip()
```

BarDemo(step4)

```
ggplotly(a)
```

lineDemo(step1)

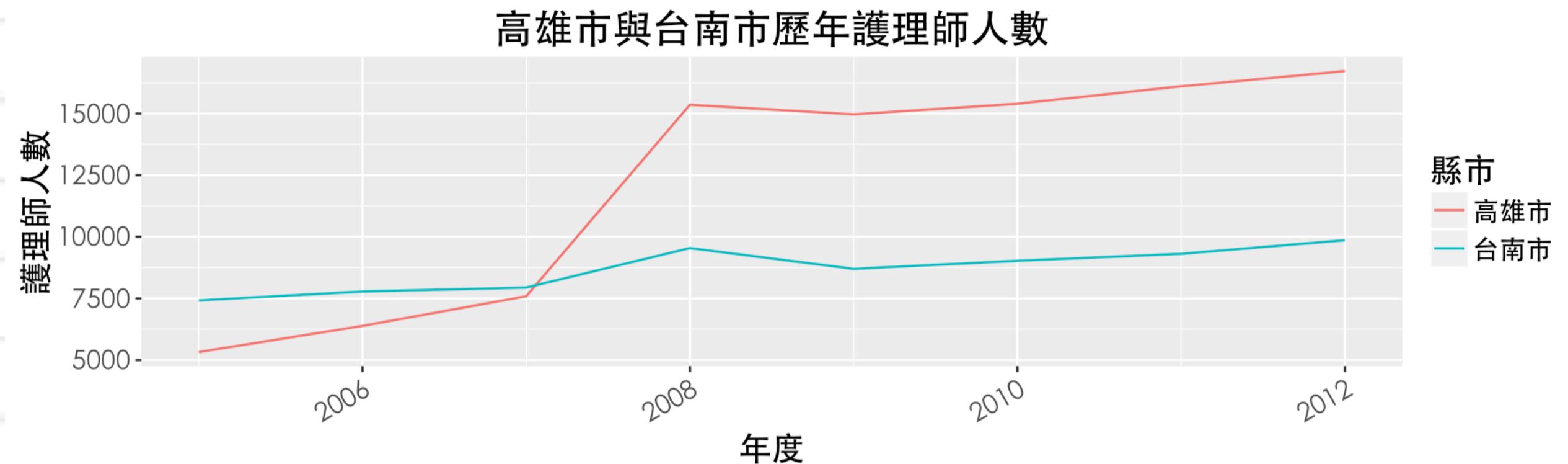
- 2005年~2012年高雄市與臺南市的護理人員人數是否有明顯的變化？

```
##  
temp_plot <- temp %>%  
  filter(縣市 == "臺南市" | 縣市 == "高雄市") %>%  
  group_by(年度, 縣市) %>%  
  summarise(護理人員數 = sum(人數))
```

lineDemo(step2)

```
a <- ggplot(temp_plot,aes(x = 年度,y = 護理人員數,group = 縣市,color = 縣市))+  
  geom_line() +  
  thm() +  
  labs(title = "高雄市與台南市歷年護理師人數",x = "年度",y = "護理師人數") +  
  theme(axis.text.x = element_text(angle = 30, hjust = 1))# 控制x軸字體傾斜程度
```

a



lineDemo(step3)

```
ggplotly(a)
```

PointDemo(step1)

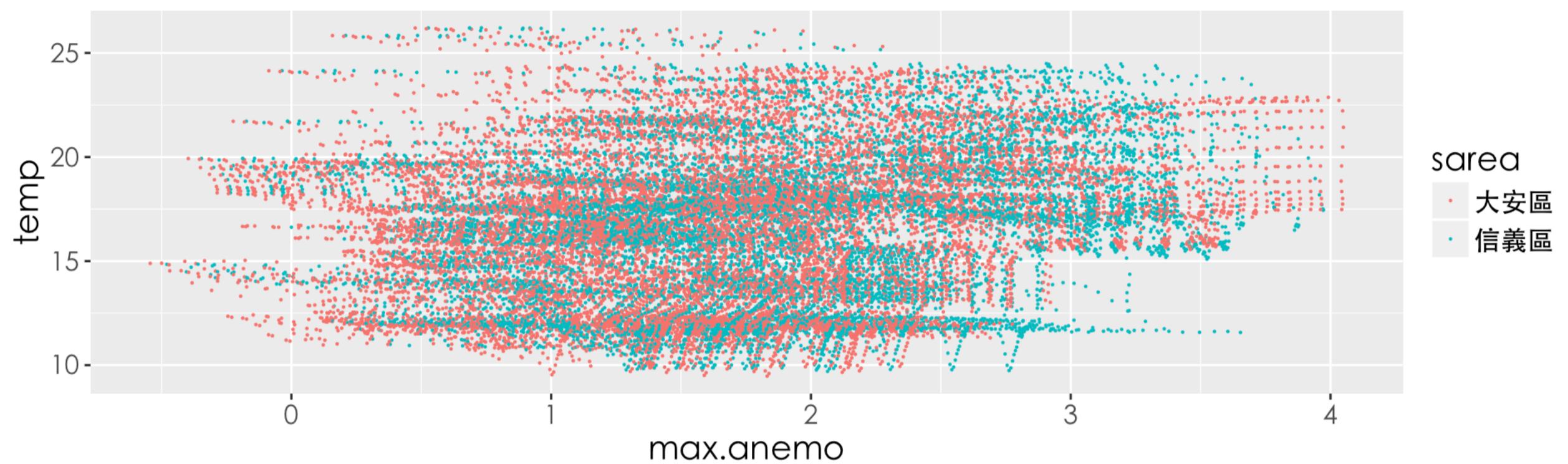
- 回到ubike資料，畫畫看信義區與大安區的max.anemo(最大風速)與temp(平均氣溫)是不是有關係呢？

```
ubike_temp <- ubike %>%
  filter(sarea == "信義區" | sarea == "大安區")
```

PointDemo(step2)

```
a <- ggplot(ubike_temp) +  
# aes裡的 colour 和 size 可依資料調整顏色和大小，使用 scale_size() 控制資料  
  geom_point(aes(x = max.anemo , y = temp, colour = sarea),size = 0.1)+  
  thm()
```

a



PointDemo(step3)

```
ggplotly(a)
```

學習資源

- [ggplot2 cheat sheet from RStudioInc.](#)
- [ggplot2 官方文件](#)
- [Rcpolorbrewer調色盤](#)
- [Rcpolorbrewer調色盤-官方教材](#)

掌握心法後，如何自行利用R解決問題

- 了解自己的需求
 - 多多交流
 - [Taiwan R User Group](#) , mailing list: Taiwan-useR-Group-list@meetup.com
 - ptt R_Language版
 - [R軟體使用者論壇](#)