

# R: Visualization on Maps

Ting-Shuo Yo

November 18, 2016

# 關於本課程

這段課程的主題是「如何在地圖上呈現資料」，包含以下概念：

- 相關的 R 套件
- 實際案例練習
  - 地點標示
  - 事件密度
  - 空間內差

本課程使用的套件：

- ggplot
- ggmap
- gstat
- sp

# 其他工具也做得到

- Excel
  - 進階功能Power Map，整合數據和圖像資料做決策
- Google Maps API
  - Google Maps JavaScript API: visualization 程式庫
- 其他服務，例如：
  - MapsData
  - Colore Maps

# R 的繪圖

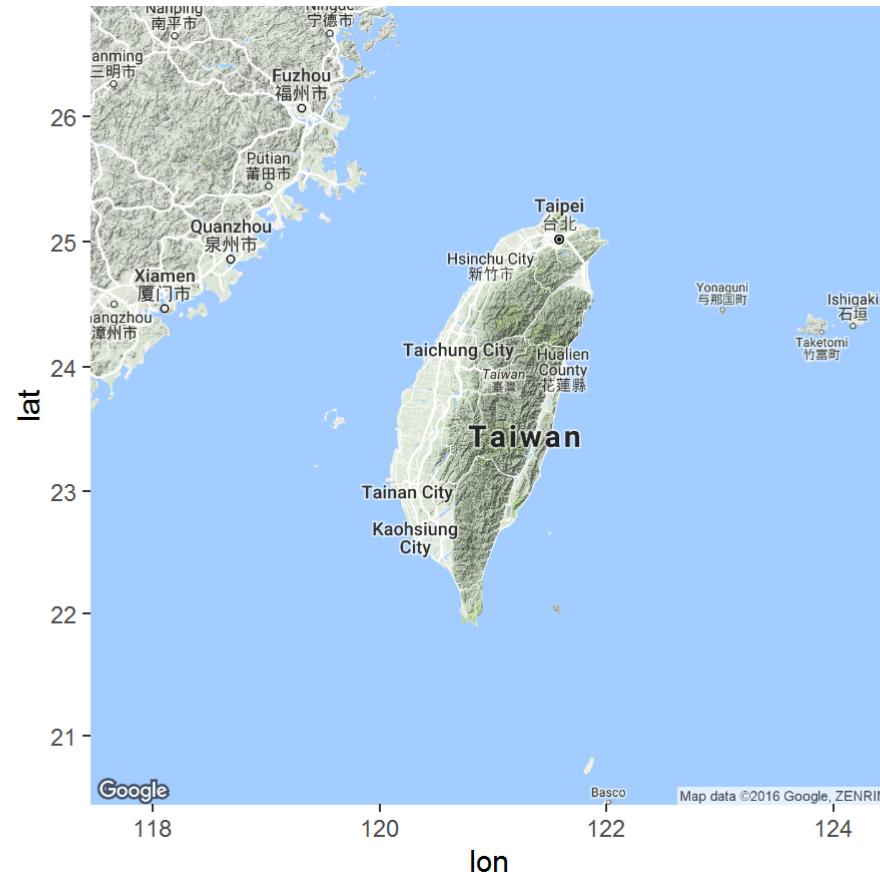
- 用範例學習：[R Graphics by](#)
  - 著名的參考書，網路版免費
  - 選你想要畫的圖，看程式碼，然後修改
- [The R Graph Gallery](#)
  - 更多的範例
- 動手做：
  - [Quick-R: Creating a Graph](#)

# ggmap 套件

- ggmap 套件是一個專門用來繪製地圖的 R 套件
- 使用 ggplot 的語法來結合地圖與資料並進行繪製
- 有多個地圖來源可以選擇：
  - Google Map
  - OpenStreetMap
  - Stamen Maps
  - CloudMade Maps
- 除了基本的資料點標示之外，還可以透過 ggmap 的所提供的函數來使用 Google 地圖 API 的各種功能

# ggmap 範例：台灣地圖

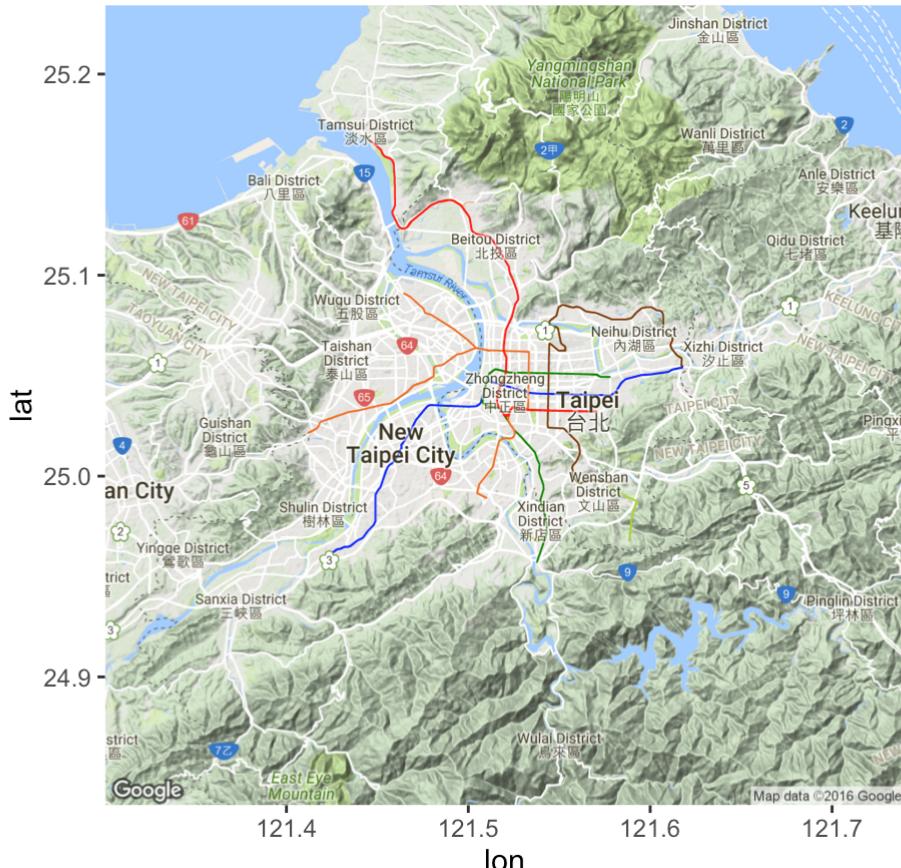
```
require(ggmap)  
map <- get_map(location = 'Taiwan', zoom = 7)  
ggmap(map)
```



# ggmap: get\_map()

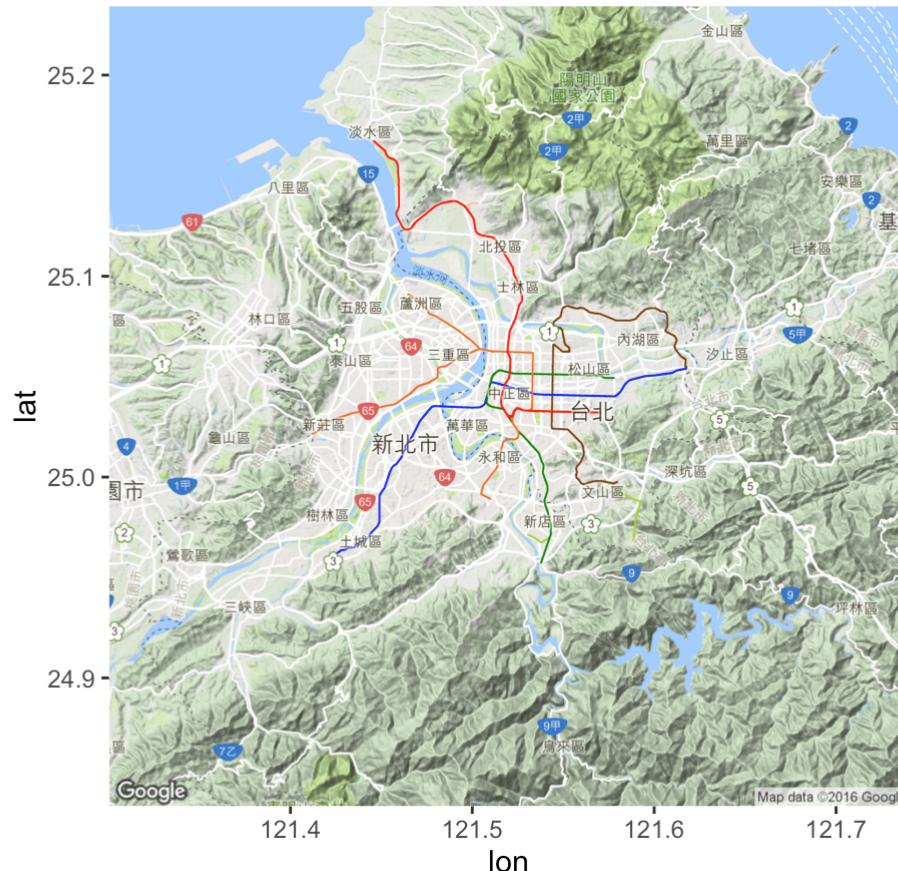
地圖的位置是透過 `location` 參數來指定，而 `zoom` 則是控制地圖的大小

```
tpe = c(lon=121.5197, lat=25.0356)
map <- get_map(location=tpe, zoom=11)
ggmap(map)
```



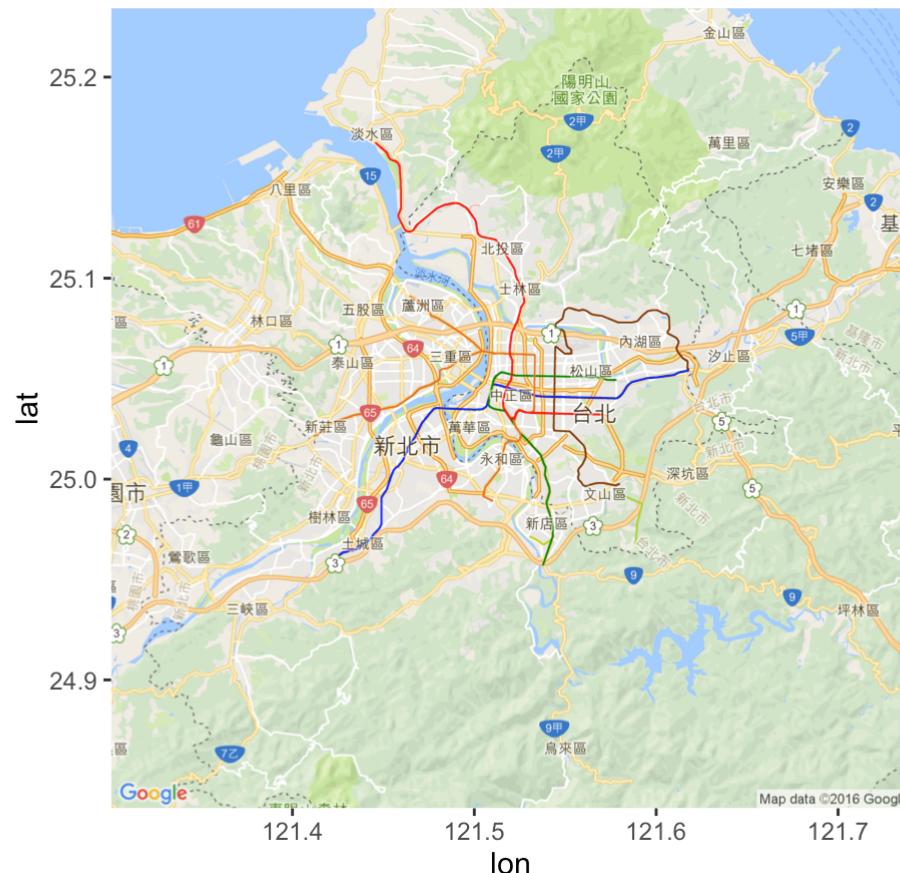
# language 文字標示的語言

```
map <- get_map(location=tpe, zoom=11, language="zh-TW")  
ggmap(map)
```



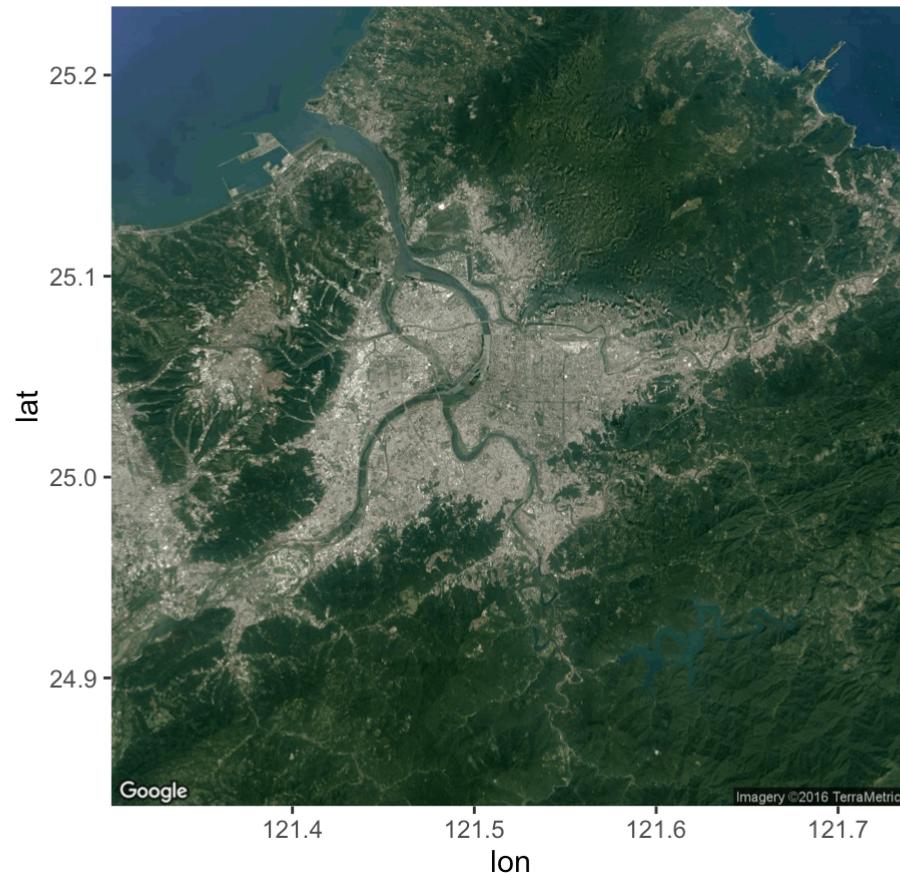
# maptype 指定地圖的類型

```
map <- get_map(location=tpe, zoom=11, language="zh-TW", maptype="roadmap")
ggmap(map)
```



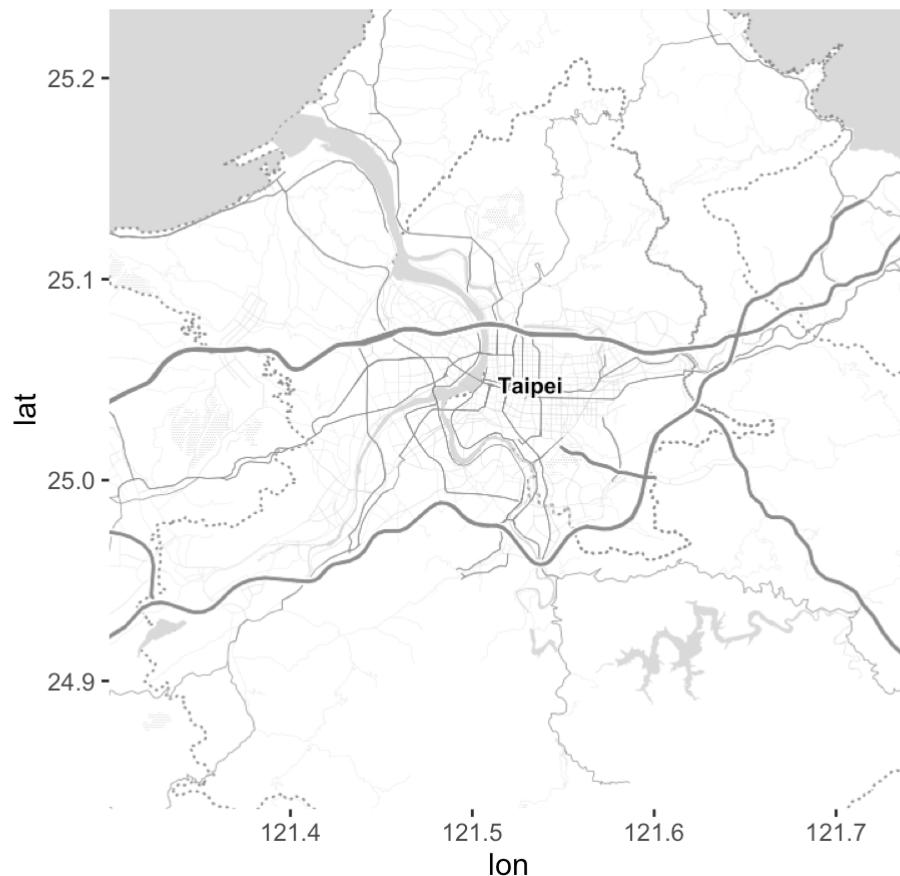
# maptype 指定地圖的類型

```
map <- get_map(location=tpe, zoom=11, language="zh-TW", maptype="satellite")
ggmap(map)
```



# maptype 指定地圖的類型

```
map <- get_map(location=tpe, zoom=11, language="zh-TW", maptype="toner-lite")
ggmap(map)
```



# 把資料放在地圖上

中央氣象局 2011-2016 年發生的地震資料

```
earthquake <- read.csv("cwb_earthquake.csv", stringsAsFactors = F, encoding="UTF-8")
str(earthquake)
```

```
## 'data.frame':    3879 obs. of  7 variables:
## $ No      : chr  "\u00a4p\u00b0\u03f0\xec" "105099"   "\u00a4p\u00b0\u03f0\xec" "\u00a4p\u00b0\u03f0\xec"
## $ Time    : chr  "2016/11/1 \u00a4U\u00a4\xc8 12:41:00" "2016/10/29 \u00a4U\u00a4\xc8 07:49:00" "201
## $ Lon     : num  122 122 122 122 121 ...
## $ Lat     : num  23.8 24.2 24.7 24.4 23.5 ...
## $ depth   : num  37 30.1 29.6 18.6 15.9 19.1 19.4 31.6 9.4 34.1 ...
## $ scale   : num  3.9 4.3 3.6 3.6 3.6 3.3 5.2 4.2 3.7 4 ...
## $ location: chr  "\u00aa\u1f6c\u00bf\u00a4\u00acF\u00a9\u00b2\u00abn\u00a4\xe8" 25.1  \u00a4\u00bd\u00a4
```

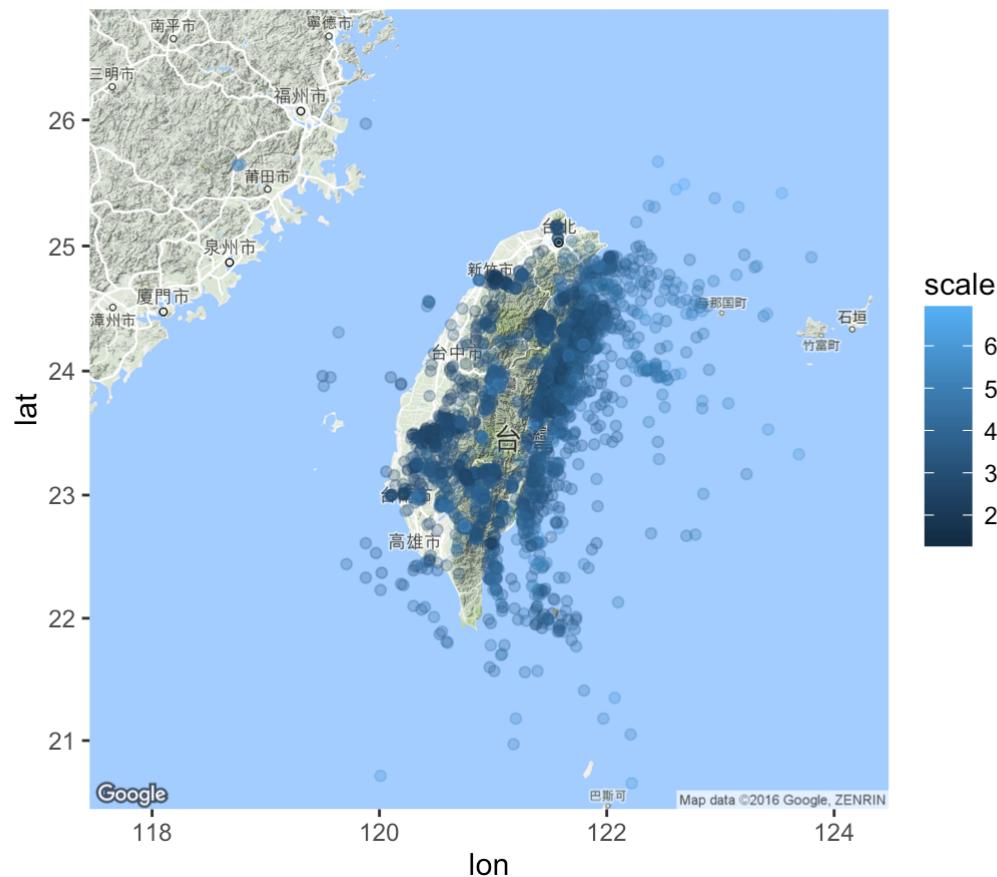
# 資料加工

```
require(dplyr)
# 新增「年份」欄位
earthquake$year <- substr(earthquake$Time, 1, 4)
# 挑出 2014~2016 的資料
eq1416 <- filter(earthquake, year %in% c("2014", "2015", "2016")) %>%
  select(Time, Lon, Lat, depth, scale, year)    # 省略不必要的欄位
str(eq1416)
```

```
## 'data.frame':    1754 obs. of  6 variables:
## $ Time : chr  "2016/11/1 \u00a4\u00a4\xc8 12:41:00" "2016/10/29 \u00a4\u00a4\xc8 07:49:00" "2016/1
## $ Lon  : num  122 122 122 122 121 ...
## $ Lat  : num  23.8 24.2 24.7 24.4 23.5 ...
## $ depth: num  37 30.1 29.6 18.6 15.9 19.1 19.4 31.6 9.4 34.1 ...
## $ scale: num  3.9 4.3 3.6 3.6 3.6 3.3 5.2 4.2 3.7 4 ...
## $ year : chr  "2016" "2016" "2016" "2016" ...
```

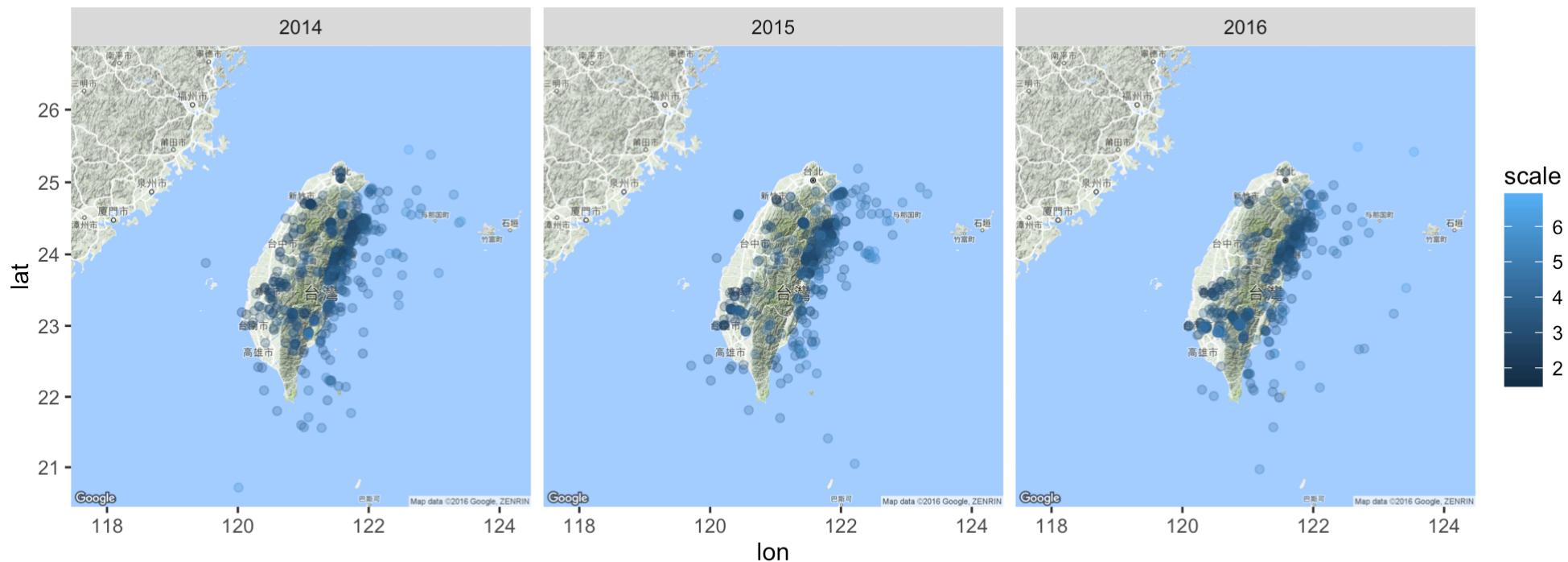
# 直接顯示資料

```
map <- get_map(location="Taiwan", zoom=7, language="zh-TW")
ggmap(map) + geom_point(aes(x=Lon,y=Lat,colour=scale), data=earthquake, alpha=0.3)
```



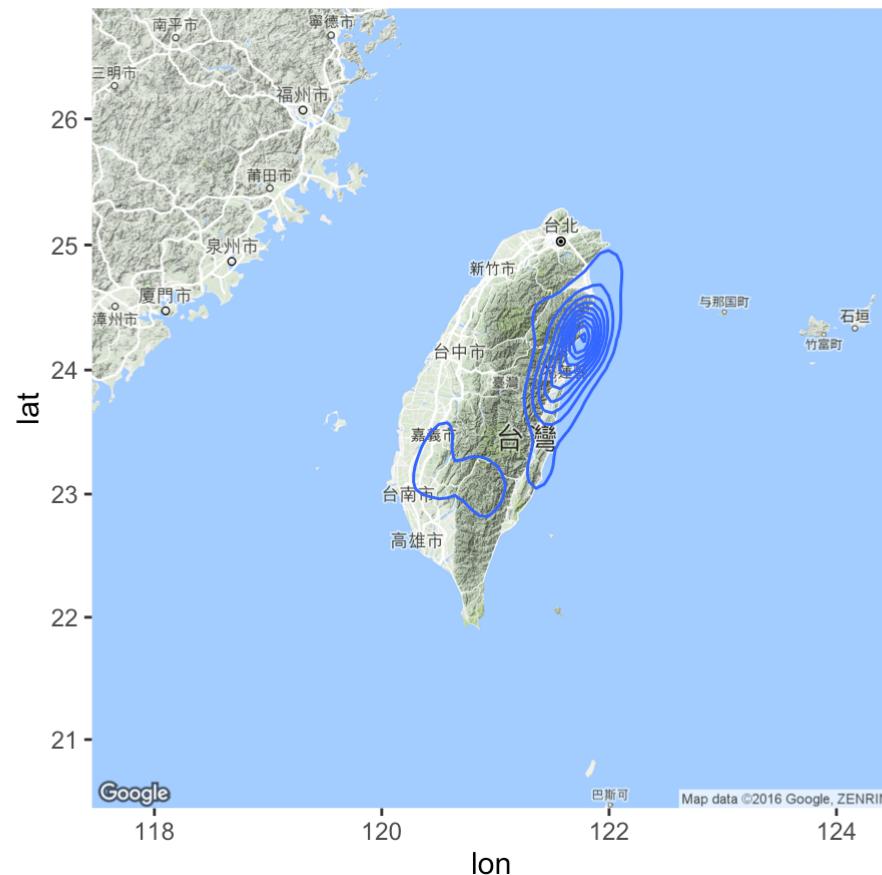
# 依年份顯示資料

```
ggmap(map) +  
  geom_point(aes(x=Lon,y=Lat,colour=scale), data=eq1416, alpha=0.3) +  
  facet_grid(~year)
```



# 用等高線顯示資料密度

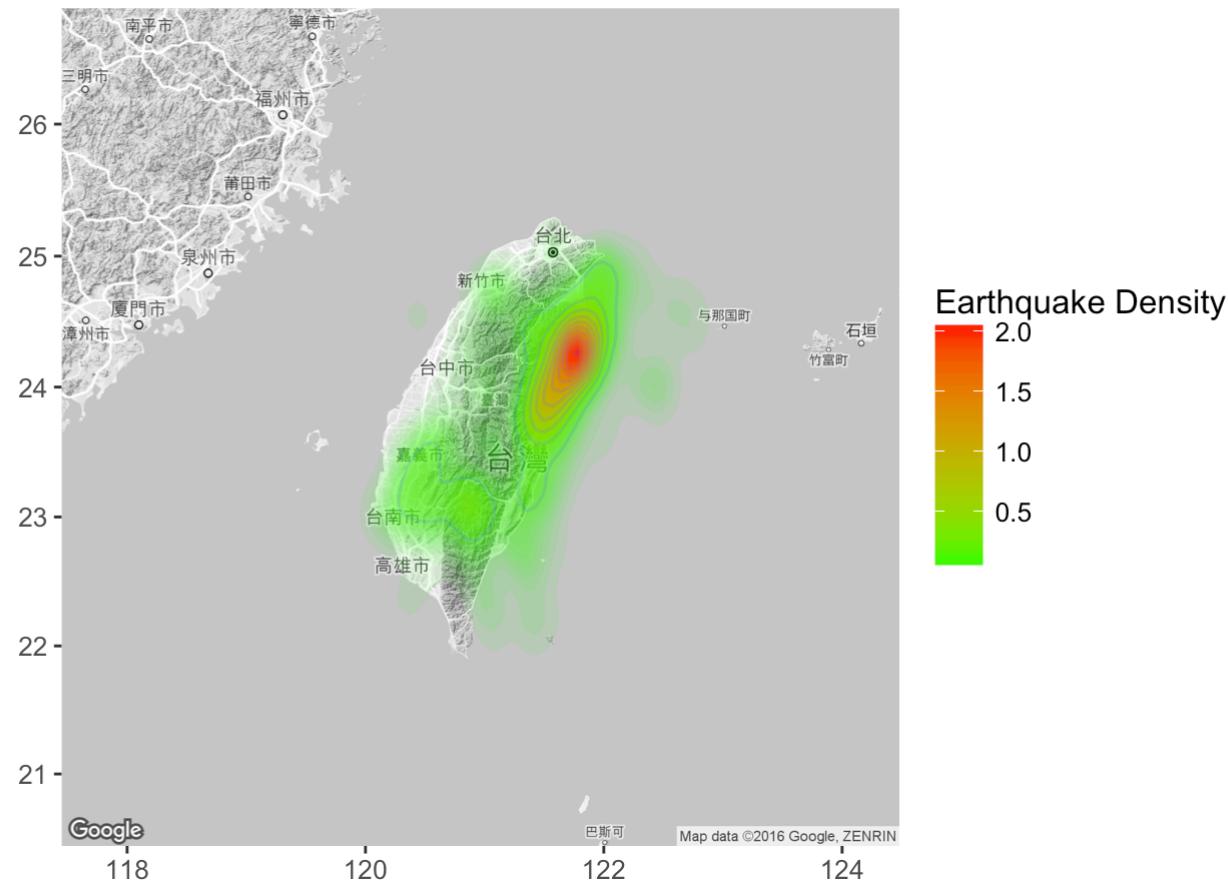
```
ggmap(map, extent="panel1", maprange=F) +  
  geom_density2d(data=eq1416, aes(x=Lon,y=Lat))
```



# 用顏色顯示資料密度

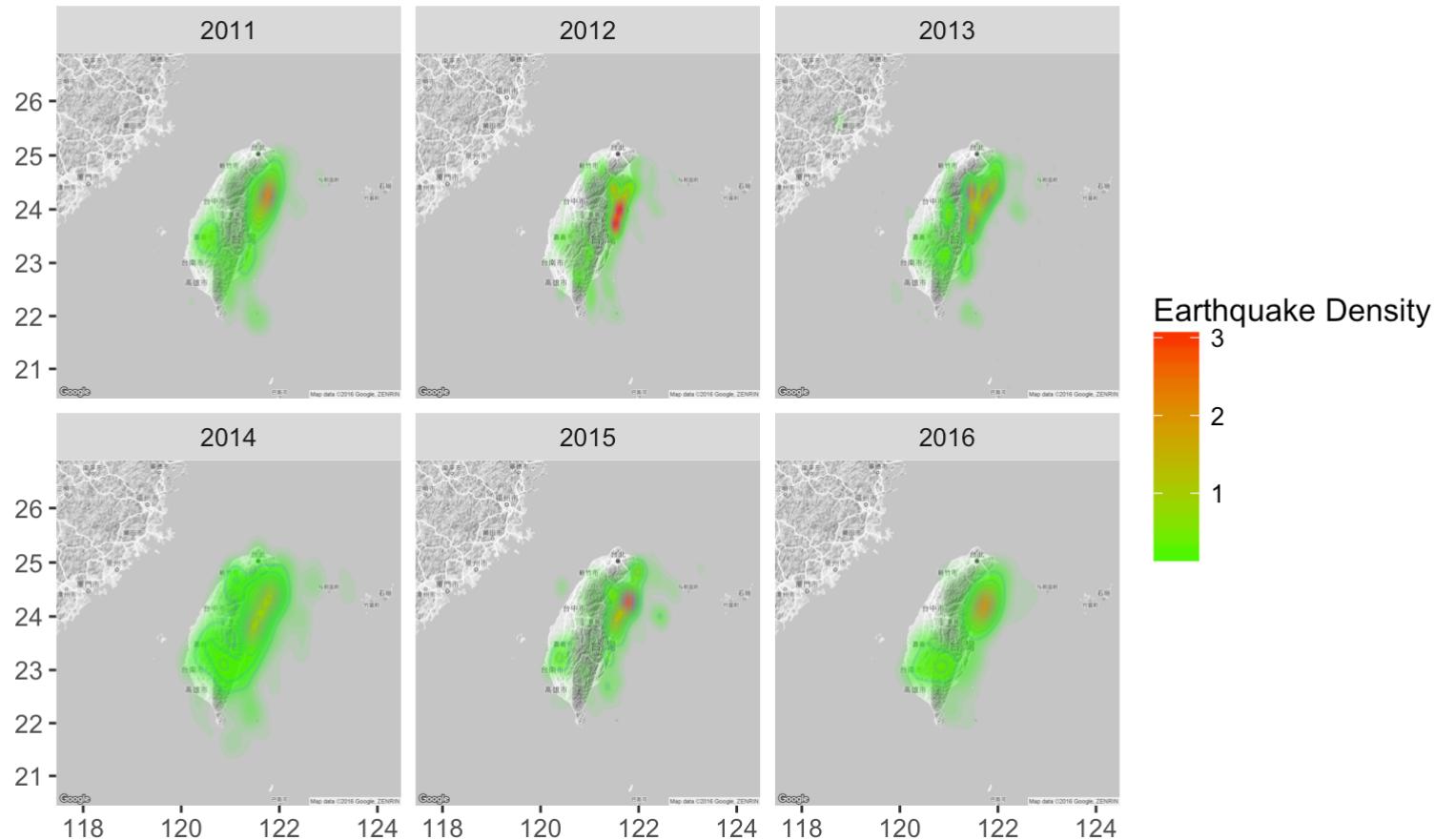
```
# 黑白地圖較適合疊上色塊
map <- get_map(location="Taiwan", zoom=7, language="zh-TW", color="bw")
# 如同前一張圖的地圖和等高線
ggmap(map, extent="panel", maprange=F) +
# 統計轉換：換成
  stat_density2d(data=eq1416, aes(x=Lon,y=Lat, fill=..level.., alpha=..level..),
                 size = 0.1, bins = 100, geom = 'polygon') +
# 色階及圖說設定
  scale_fill_gradient("Earthquake Density", low = "green", high = "red") +
  scale_alpha(range = c(0.10, 0.30), guide = FALSE) +
# 等高線移到較上面的圖層
  geom_density2d(data=eq1416, aes(x=Lon,y=Lat, alpha=0.6)) +
# 全圖主題設定
  theme(axis.title = element_blank(), text=element_text(size = 12))
```

# 用顏色顯示資料密度



# 逐年地震頻率

facet\_wrap(~year)



# 空間內插：將點狀觀測轉換到平面網格

有時候我們只有「點」的觀測，但是卻需要整個「面」的資料，例如：氣象觀測都是點狀的，但是數值模式需要的是「網格」資料，這時候就需要空間內插。

空間內插的意義：

- 資料統合
  - 整合在空間上無法相提並論的資料，例如家戶收入
- 預測實際上沒有觀測資料的點
  - 利用空間分布型態的資訊，來預測沒有觀察點的資料

# 空間內插方法的分類

- Global v.s. Local
  - **Global**: 假設一個全平面的分佈
  - **Local**: 只考慮鄰近的點的資料
- Exact v.s. Inexact
  - **Exact**: 內插後觀測點保留原始的觀測值
  - **Inexact**: 用內插值取代原始觀測值
- Deterministic v.s. Stochastic
  - **deterministic**: 忽略可能的誤差範圍，只提供單一值
  - **stochastic**: 提供區間估計

# 常見的空間內插法

- trend surface
  - 用資料來估計全平面的分佈
- IDW
  - Inverse Distance Weighting，局部的資料內插法，用觀測點資料的加權平均來預測未知資料點的估計值。通常權重與距離的某次方成反比。
- loess
  - Local Weighted Smoothing Regression
- spline
  - 曲面函數估計常用的數值方法
- kriging
  - 地理統計方法

# 在 R 中使用 IDW

需要的套件：

- **gstat**
- **sp**
- **maptools**

```
require(gstat)
require(sp)
require(maptools)
```

# 準備網格

```
# 設定資料座標
coordinates(earthquake) = ~Lon+Lat

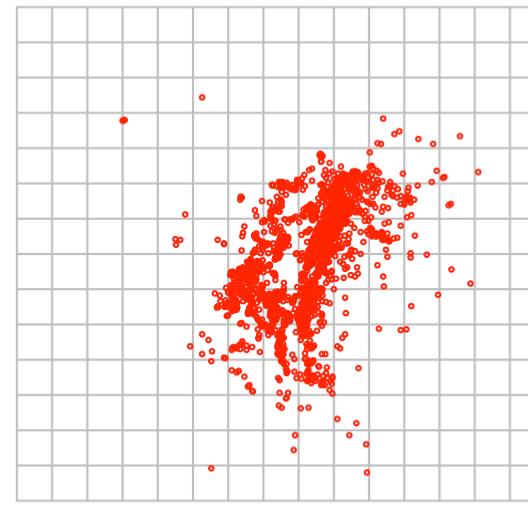
# 設定網格範圍
x.range <- as.numeric(c(117.5, 124.5))
y.range <- as.numeric(c(20.5, 27.2))

# 製作網格點
grd <- expand.grid(Lon = seq(from = x.range[1], to = x.range[2], by = 0.5),
                     Lat = seq(from = y.range[1], to = y.range[2], by = 0.5))

# 設定網格資料座標
coordinates(grd) <- ~Lon+Lat
gridded(grd) <- TRUE
```

# 看看網格的長相

```
plot(grd, cex = 1.5, col = "grey")
points(earthquake, pch = 1, col = "red", cex = 0.3)
```



# 呼叫 IDW 函數

網格準備完成，接著我們把地震規模（scale）內插到網格點上。

```
# 呼叫 IDW 函數進行空間內插
idw <- idw(formula=scale~1, locations=earthquake, newdata=grd, debug.level=0)
# 轉換資料格式以利繪圖
idw.output = as.data.frame(idw, stringsAsFactors=F)
names(idw.output)[1:3] <- c("lon", "lat", "value")
```

套件 stat 也包含 krige 函數，使用方法與 idw 類似，可以參考函數說明。

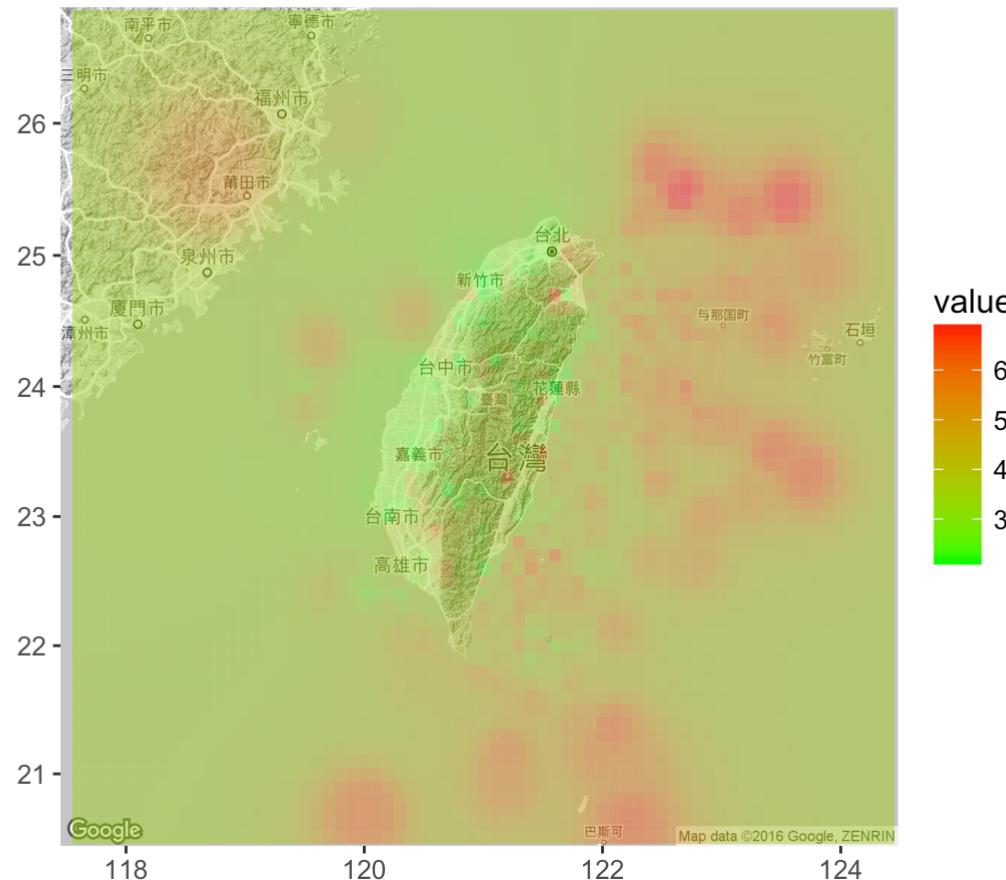
# 在地圖上繪製網格資料

現在，我們可以用 `geom_tile()` 把網格資料疊在地圖上了。

```
ggmap(map, extent="panel", maprange=F) +  
  geom_tile(data=idw.output, aes(x=lon, y=lat, fill=value), alpha=0.5) +  
  scale_fill_gradient(low = "green", high = "red") +  
  theme(axis.title = element_blank(), text = element_text(size = 12))
```

# 在地圖上繪製網格資料

網格資料的美觀程度，取決於網格的密度，但是網格越密，內插要算越久。以下是把網格間距從 0.5 度調降到 0.1 度的結果。



# 小結

我們示範了三種在地圖上顯示資訊的方式：

- 直接顯示資料值
- 顯示資料密度
- 顯示空間內插後的分佈

使用的套件：

- ggplot
- ggmap
- gstat
- sp

# 練習

上次我們練習合併了環保署的臭氧濃度和測站資料，請利用上面示範的三種方式顯示在地圖上。

可以參考解答：[modules/rppr03\\_visualization\\_on\\_map/exercise\\_map.r](#)

# 課程尾聲

- R 很容易上手，但跟其它程式語言一樣，需要常常使用才能熟練
- 線上課程
  - [R Programming](#) by John Hopkins University
  - [R軟體學會教學影片](#)
- R 的使用者和套件眾多，通常遇到的問題已經有人解決過了
  - [CRAN](#) (The Comprehensive R Archive Network)
  - [Stackoverflow](#)
  - [R-blogger](#)