

資料視覺化的奇幻之旅

彭其捷

資料視覺化的神兵利器 (上): 套裝軟體與程式篇

作者介紹

彭其捷鑽研網路服務多年，出版過 3 本 UX 使用者經驗專書，曾大量接觸設計、工程，使用者體驗等相關工作，近年因大數據&物聯網概念蓬勃發展，觀察到越來越多數據導向的服務興起。然而，艱澀的數據需要良好的設計輔助，才能創造良好的閱讀體驗，其中特別依賴資料視覺化的相關能力。因此，本專欄特別針對各類網路服務的資料呈現提出美學觀點，分享一些國內外資料視覺化的概念、工具與案例。

系列文章介紹

FINDIT 的目標是『發現趨勢，看見未來』，事實上，眾多的趨勢就隱藏在眾多的數據當中，等待著人們去發現、去解讀。透過『資料視覺化』的輔助，能夠把冷冰冰的數據圖像化，協助人們掌握趨勢，更能夠協助新創企業在創新創業過程中找出正確方向，或是幫助投資人找到潛在投資標的，是一項強大的武器。

本系列文章將從資料視覺化的概念開始著手，之後會陸續分享視覺化的經典案例，相關工具介紹等等，希望讀者們能夠在大數據時代，透過視覺化的輔助，說出好的故事。

Every Tool Forces You Down a Path

- Lisa Charlotte Rost （國際知名資料視覺化專家）



（圖來源：<https://vimeo.com/104301945>）

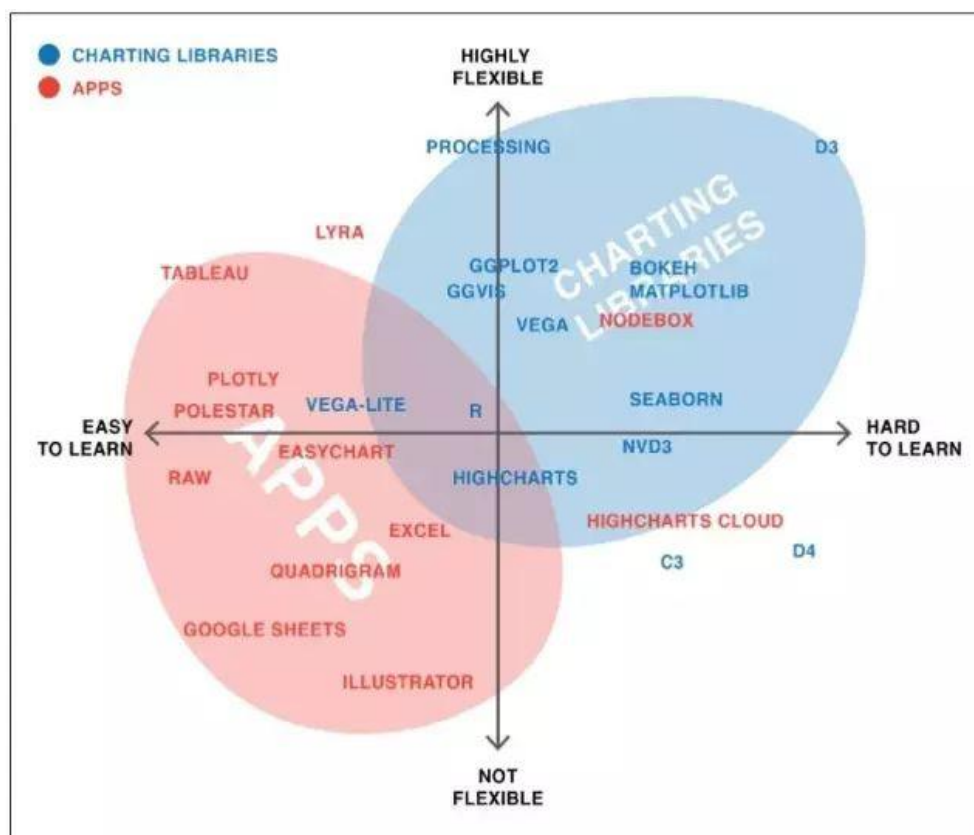
相信許多讀者都執行過視覺化作業，不過讀者們是不是偶爾也會想到：『為什麼別人可以做出這麼漂亮的圖呢？他們都是用了哪些工具完成的呢？』這就是本篇想要分享的內容，也就是各類視覺工具的介紹，期待能讓讀者了解更多工具應用的面向，或是挑選其中部分工具使用看看，說不定因此做出更棒的視覺化結果，得到老闆或是客戶的讚賞呢！

視覺化工具分類方式

視覺化工具非常多選擇，我們該怎樣找到適合自己的工具呢？本篇分享了三種分類方式，提供給讀者參考，分別是『容易上手程度』、『職務需求』、『工具本身類型』，介紹如下：

依容易上手程度區分

視覺化工具很常會用『好上手程度』區分，例如下圖就用『學習難度』與『彈性』製作了四軸象限示意圖：



資料來源：<http://www.36dsj.com/archives/79982>

圖 1、視覺化工具比較表，右上角多是視覺化元件庫，左下角多是可以直接應用的軟體工具

從上圖可看出，通常越容易學習的工具彈性也相對較低，而功能越強大的程式庫，學習曲線則相對陡峭，但我們現實當中也不一定需要做到非常複雜的功能，所以讀者可根據自身需求情境來選擇適合的工具。

依職務需求區分

另一種工具的分類方式則是回歸以人為本，也就是依照『職務需求』來進行分類，雖然這種分類並不是絕對的，但學習資源有限的狀況之下，團隊還是建議依照各職務需求重點來調度工具的使用，整理如下：

職務需求重點	適用職務	適合工具類型
精準呈現業務數據	企業主管、專案管理人員、研究人員	套裝軟體類
製作美麗的視覺圖表	視覺設計師、分析師、研究人員	套裝軟體類 線上視覺化工具類
搭配運算結果進行視覺呈現，與系統資料進行整合	開發人員、研究人員	線上視覺化工具類 程式語言類 前端程式外掛類

資料來源：本文整理。

依工具類型區分

延續上一段的分類，共有三種工具類型『線上視覺化工具類』、『程式語言類』、『前端程式外掛類』，本段落具體說明各使用類型的介紹與常見工具，如下：

使用情境	說明	常見工具
(1) 裝軟體類	指提供一個完整的視覺化環境，有時也附加資料處理的機制	Excel、Power BI、PowerPoint & Keynote、Tableau、Gephi、illustrator、Photoshop
(2) 式語言類	指的是撰寫客製化程式碼來產生視覺化的結果，彈性最高但也相對來說有學習的門檻	Processing、JavaScript、Python、R

(3) 端程式 外掛類	前端一般指的是 Web 端的顯示，目前視覺化的大宗還是以 PC 為主的視覺化結果呈現，許多人設計了好用的視覺函式庫，讓開發者能夠直接進行使用	Google Chart、HighChart、D3.js、Flot、Three.js、Mapbox、Bonsai、Leaflet、animate.css
(4) 上視覺 化工具 類	直接在線上進行視覺化作業，通常可上傳資料檔案，或是直接與外部資料進行介接等，非常方便的製作方式	Google Sheet、Plotly、Lyra、quadrigram、datahero、Highcharts Cloud、Easychart、NodeBox、plotDB、Vectr、Gapminder、Piktochart、Raw、Gliffy、Infogr.am、ai2html、Ease.ly、Creately、Wordle、giphy

資料來源：本文作者整理。

本篇內容優先談前三種作法的工(1) 各類可以執行視覺化作業的套裝軟體工具；(2) 透過程式來完成視覺化作業的套件；(3) 各種前端程式外掛套件。至於 (4) 線上第三方工具的部分，我們留到下一篇再繼續做介紹。

視覺化工具：套裝軟體類

(1) Excel

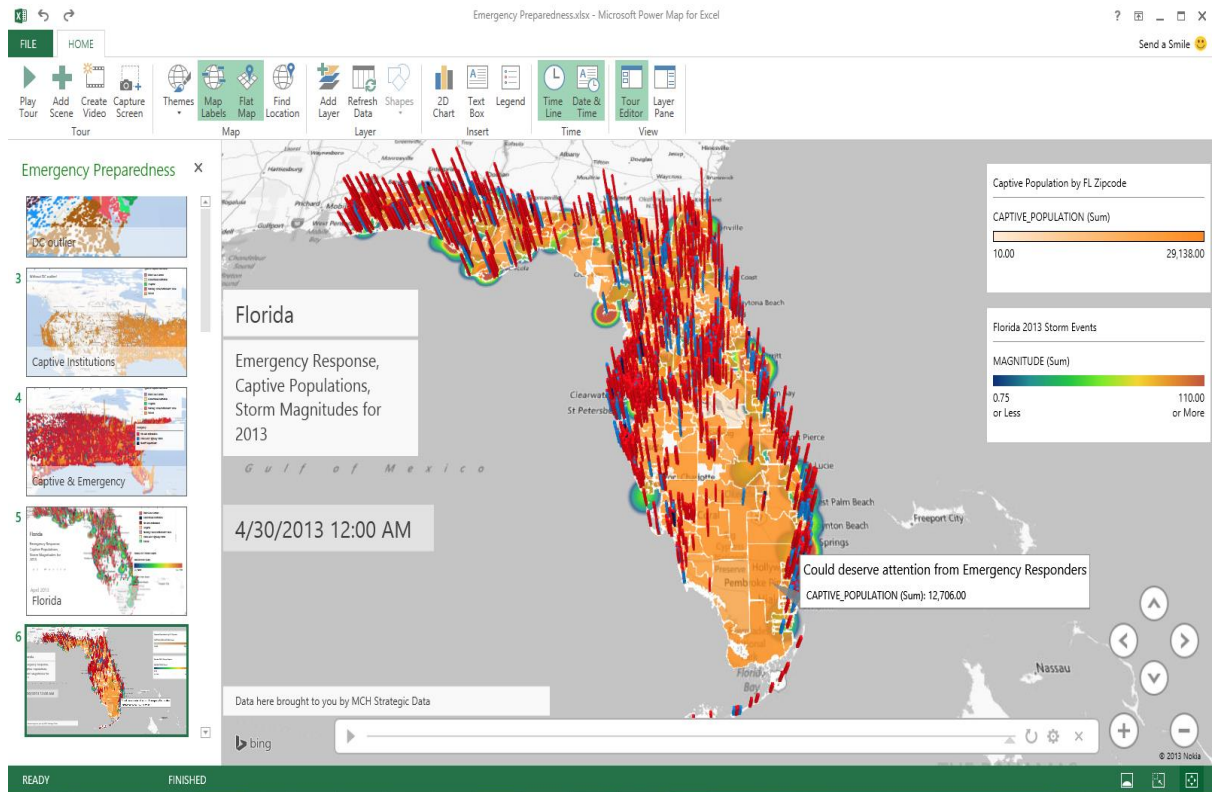
網址：<https://products.office.com/zh-tw/excel>

Excel 是一般人最常用的視覺化工具，已經在市場廣泛使用了數十年，許多人從學生時期就開始使用，不但對初學者來說很容易上手，對於進階使用者也提供很大的靈活性，搭配巨集或是函式庫後，也可做出進階統計圖表。另外，Excel 產出圖表也相容於 Powerpoint 跟 Word，因為同樣屬於微軟辦公室軟體系列。

Excel 近幾年也增加許多新功能，例如『Excel Online』的線上操作版本，或是提供像是『Power Map』之類的視覺外掛，或是跟『PowerBI』之类的外部工具整合，提升了整體的可用性。

但 Excel 也有若干缺點，像是不適合與外部程式介接，且對於初

學者來說不容易處理複雜的數據，且資料量過大時也有有若干問題產生。



資料來源：<https://powerbi.microsoft.com>

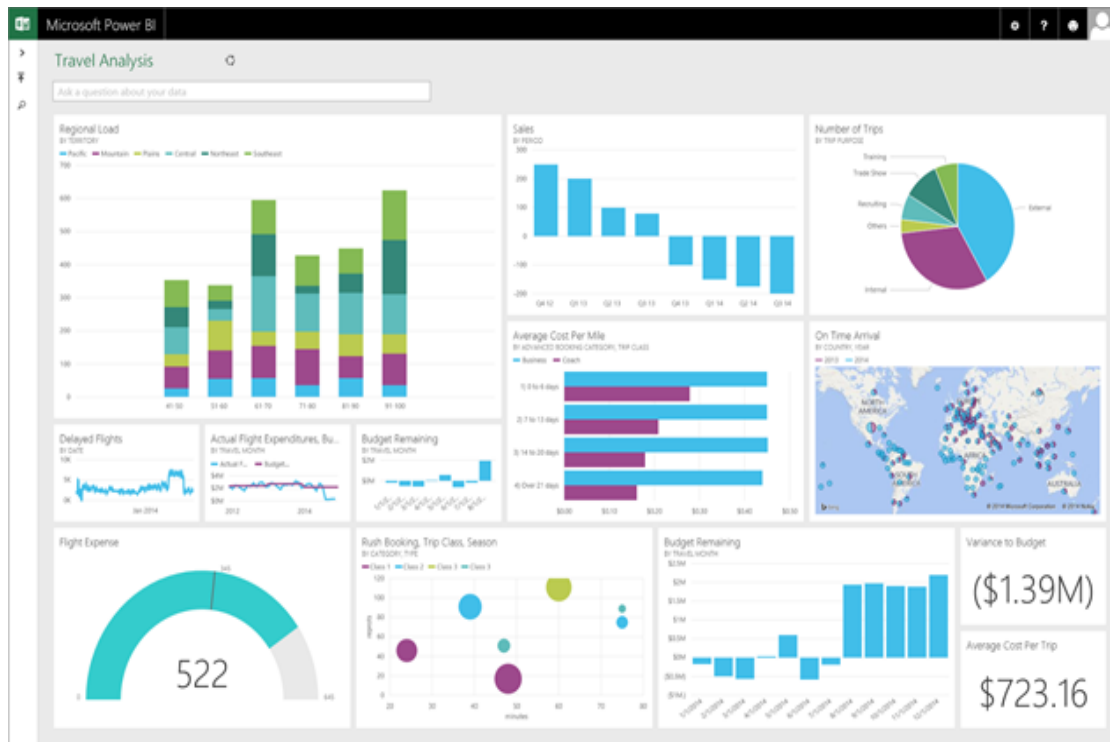
圖 2、Excel 是最常見的視覺化工具，搭配上視覺化工具也能做出華麗圖表

(2) Power BI

網址：<https://powerbi.microsoft.com>

Power BI 跟 Excel 都是微軟的產品，不過 Excel 比較偏向數據分析，但 Power BI 相對精簡且更強化資料視覺化的設計，兩者之間的整合性頗高。Power BI 的介面上也承襲了微軟的標準架構，所以相對容易上手。

Power BI 加入了許多資料視覺化的共通需求，像是多重資料匯入、拖拉操作、資料儀表板、協同作業機制、企業化使用等等，但如果要使用一些進階功能，就需要額外付費了，但也有許多人認為像是此類的商業軟體才是數據分析的優先選項，因為不用手動執行前置作業，省下許多時間。



資料來源：<https://powerbi.microsoft.com>

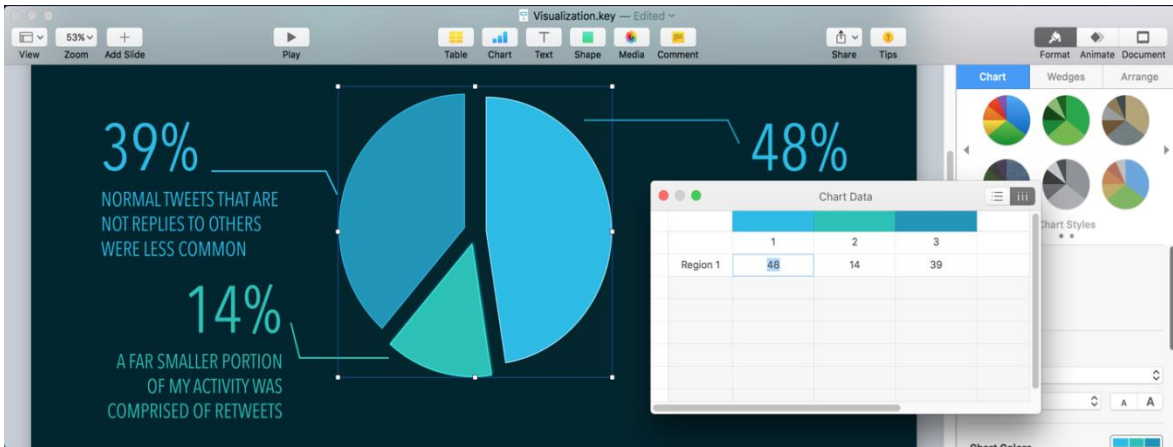
圖 3、Power BI 提供了大量的視覺化圖表，設計上也頗清爽

(3) Powerpoint & Keynote

網址 1：<https://products.office.com/zh-tw/powerpoint>

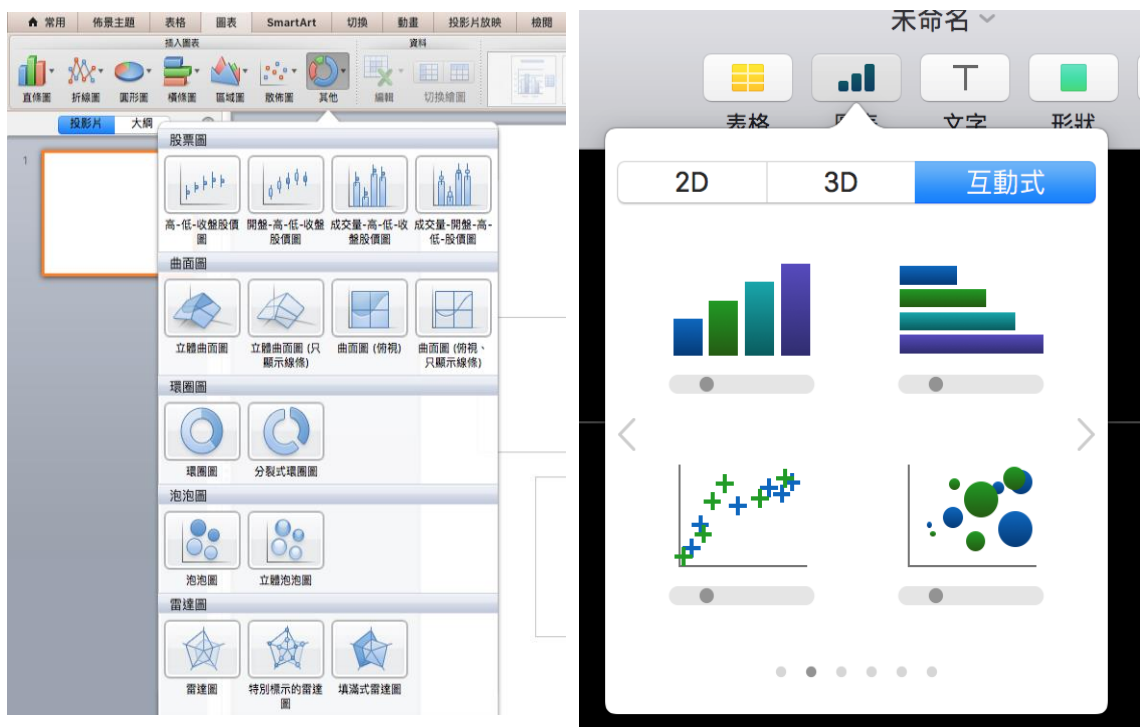
網址 2：<https://www.apple.com/tw/keynote/>

其實 PowerPoint 跟 Keynote 之類的簡報軟體，本身就是視覺化的好用工具，且許多人都已經使用的得心應手，兩項工具都可以輸入資料並轉換成各種統計圖表，只要搭配良好的色彩學與排版技巧，同樣能產出很棒的成果。不過，簡報軟體就比較無法搭配大量資料進行呈現，而是較擅長將整理過的統計數據執行轉換。



資料來源：<https://chrisardegna.com/blog/posts/keynote-data-vizualization/>

圖 4、善用簡報軟體，也能產出漂亮的視覺化圖表



資料來源：作者整理

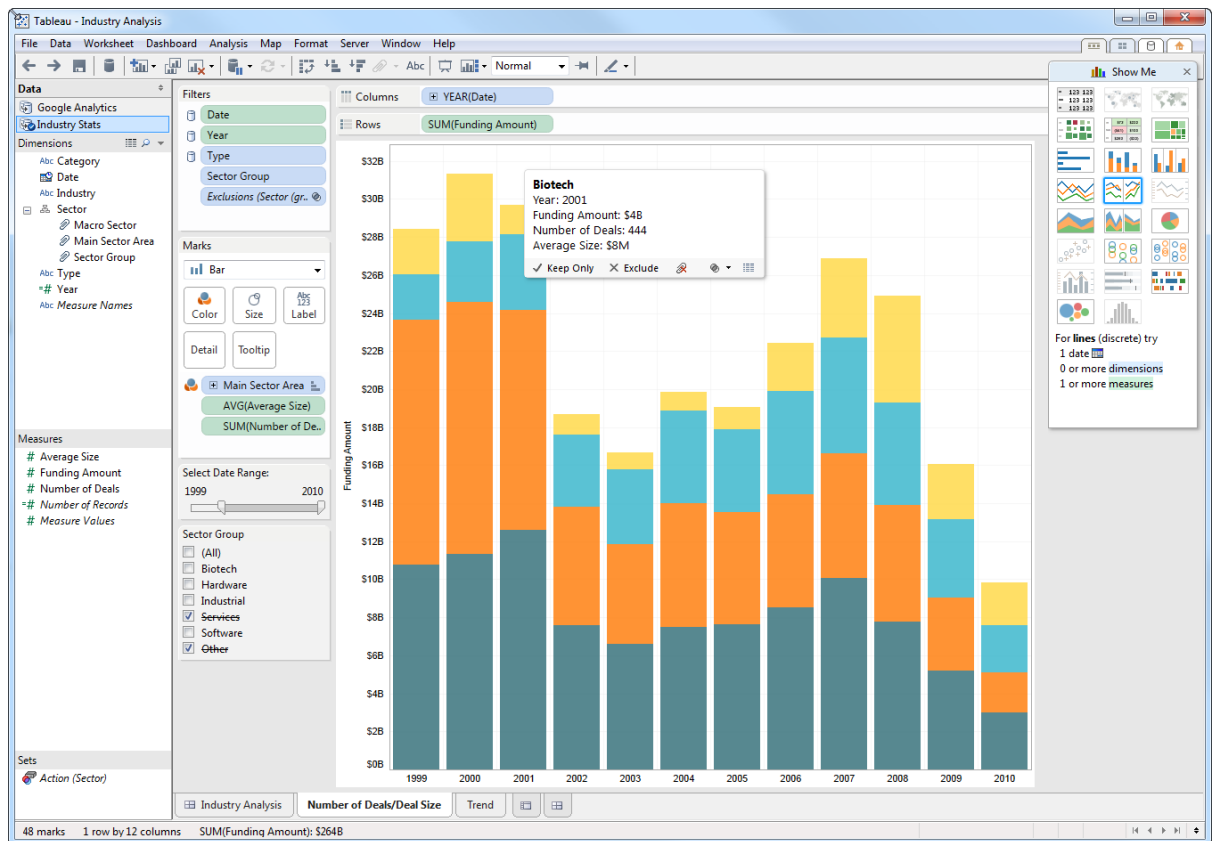
圖 5、簡報軟體提供許多視覺化的選項，也能切換 2D 跟 3D 圖

(4) Tableau

網址：<https://www.tableau.com/>

Tableau 不需要撰寫程式，只要將資料製作成指定的 key-value 格式（例如 .xls .csv 數據）後，丟進軟體就可以全自動生成各類視覺圖表，也能輸出為網頁格式，供使用者進行瀏覽。

Tableau 最近幾年異軍突起，成為許多企業指定使用的資料視覺化工具，Tableau 提供了優秀的操作經驗，像是各類拖拉手勢的操作，即使是新手也能快速學會基本操作。Tableau 區分了三種類別：Public、Desktop、Server 版本，建議讀者可先下載免費的 Public 玩玩看。



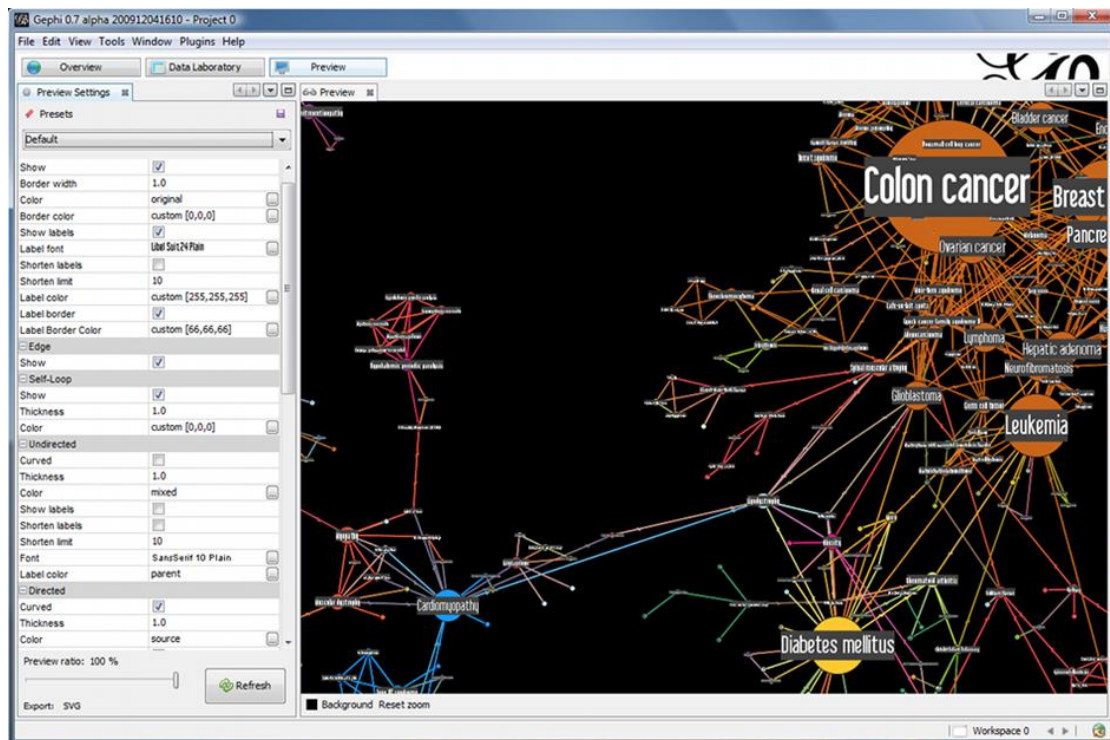
資料來源：<https://docs.treasuredata.com/articles/tableau-desktop-odbc>)

圖 6、Tableau 提供了優秀的操作體驗，也能快速切換不同圖表類型

(5) Gephi

網址：<https://gephi.org/>

Gephi 是繪製網路圖的好用工具，免費且開放原始碼，提供了很大的使用彈性，擅長繪製網絡關係，像是社群關聯性、節點關聯性、文件關聯性、生物關聯性等等，能夠承載大量資料，也方便執行資料整理等等，在科學研究領域頗為知名。



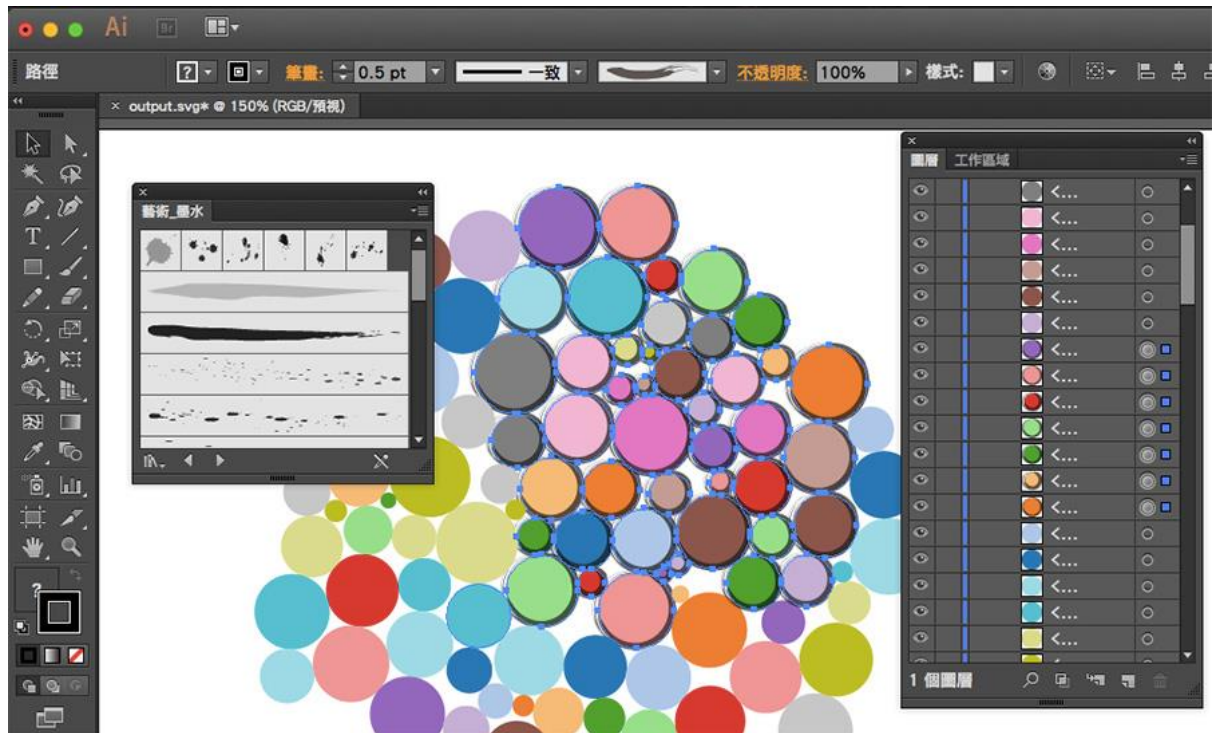
資料來源：<https://gephi.org/features/>

圖 7、Gephi 在網路圖的繪製上表現優秀

(6) illustrator

網址：<http://www.adobe.com/tw/products/illustrator.html>

illustrator 是向量繪圖工具，透過工具產生的視覺化結果，如果能使用視覺工具再加工，能夠更強化數據的視覺震撼力。此外，illustrator 可支援常見的『SVG』視覺化格式，所以甚至能直接編修工具產出的圖樣，提升視覺成效。



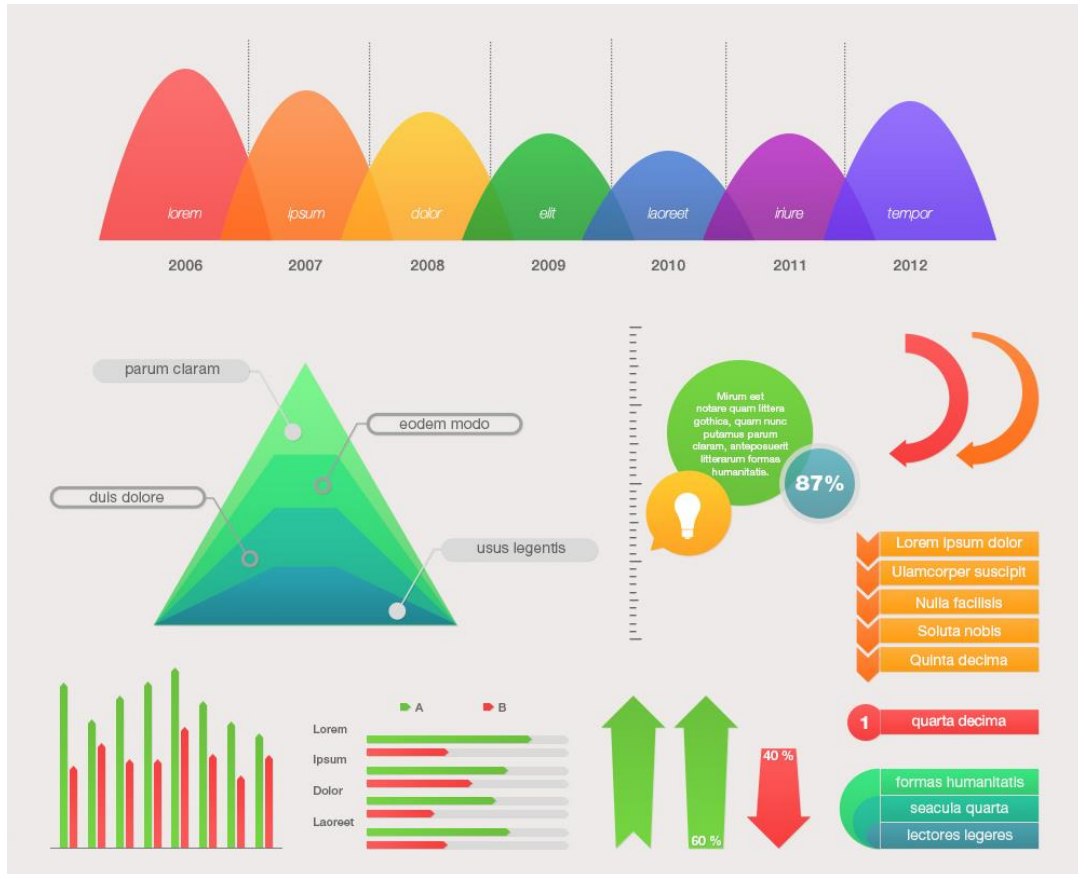
資料來源：<http://blog.infographics.tw/2015/04/d3js-adobe-illustrator-nodejs/>

圖 8、illustrator 可獨立使用來繪圖，也能編修 SVG 格式檔案

(7) Photoshop

網址：<http://www.adobe.com/tw/products/photoshop.html>

Photoshop 跟 illustrator 同樣都是視覺編修的經典工具，且設計人員都相當熟悉，跟 illustrator 的最大差異在於 Photoshop 產生的是『點陣式圖檔』格式，雖然視覺效果彈性更多元，但放大之後有失真的可能性。



資料來源：<https://www.pinterest.com/pin/127508233175687154/>

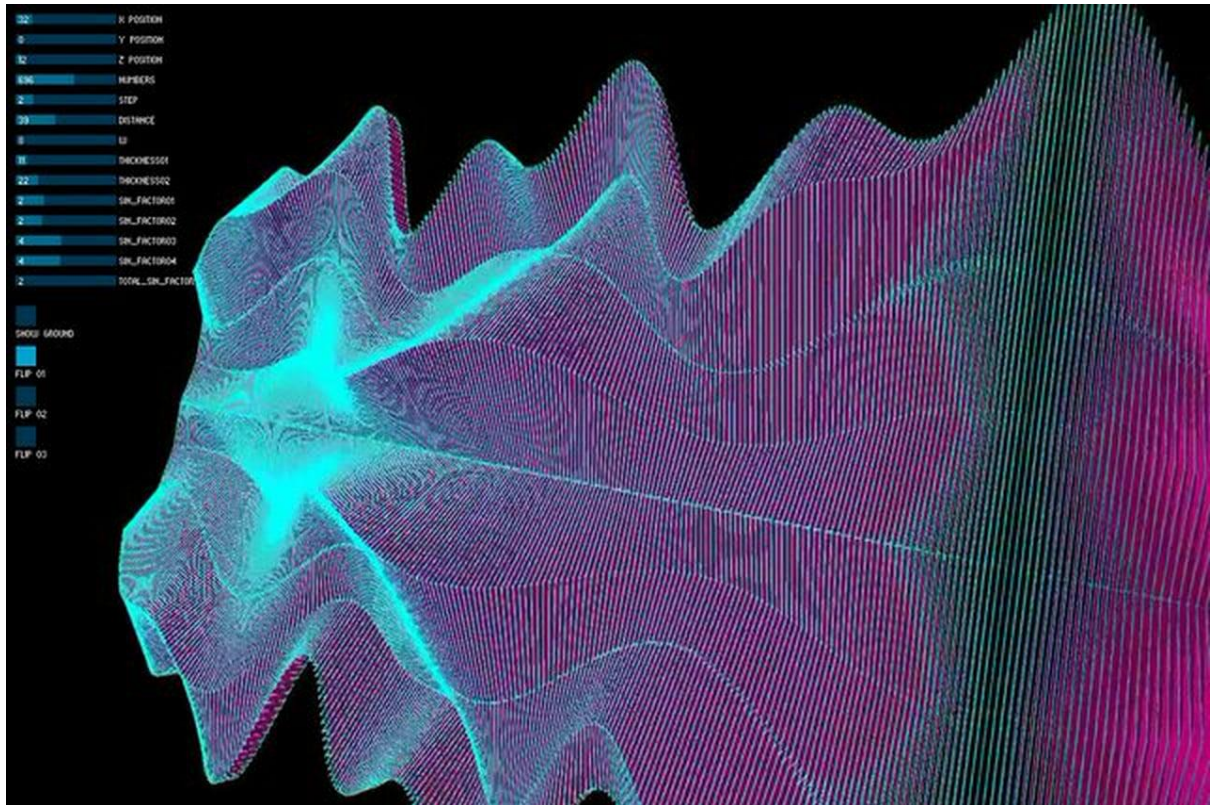
圖 9、直接用 Photoshop 著手製圖，是許多設計師的拿手技能

視覺化工具：程式語言類

(1) Processing

網址：<https://processing.org/>

在視覺化的程式語言當中，Processing 算是最適合新手入門的程式語言，已經封裝成更高階的語法，相當適合設計人士學習與操作，而正式執行時則會被轉譯成為 Java 語言。另外，Processing 也推出了 Javascript 套件 processing.js，讓網頁端的人方便導入使用，以 HTML5 的形式呈現，也能與系統結合。



資料來源：<https://vimeo.com/channels/processing>

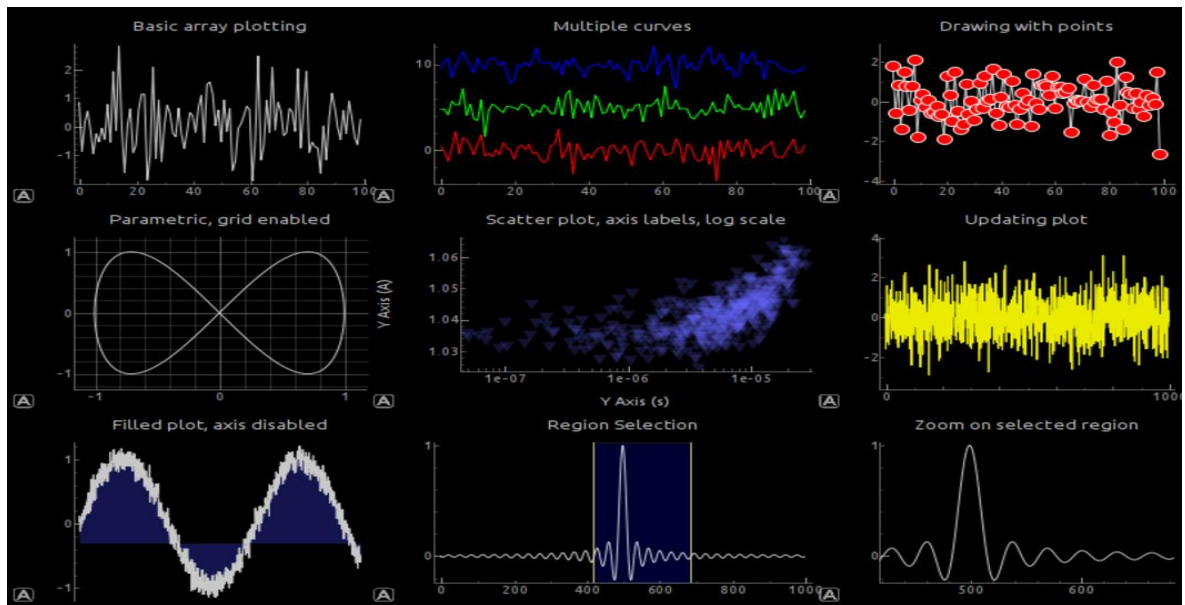
圖 10、Processing 擅長製作各種華麗的視覺作品

(2) Python

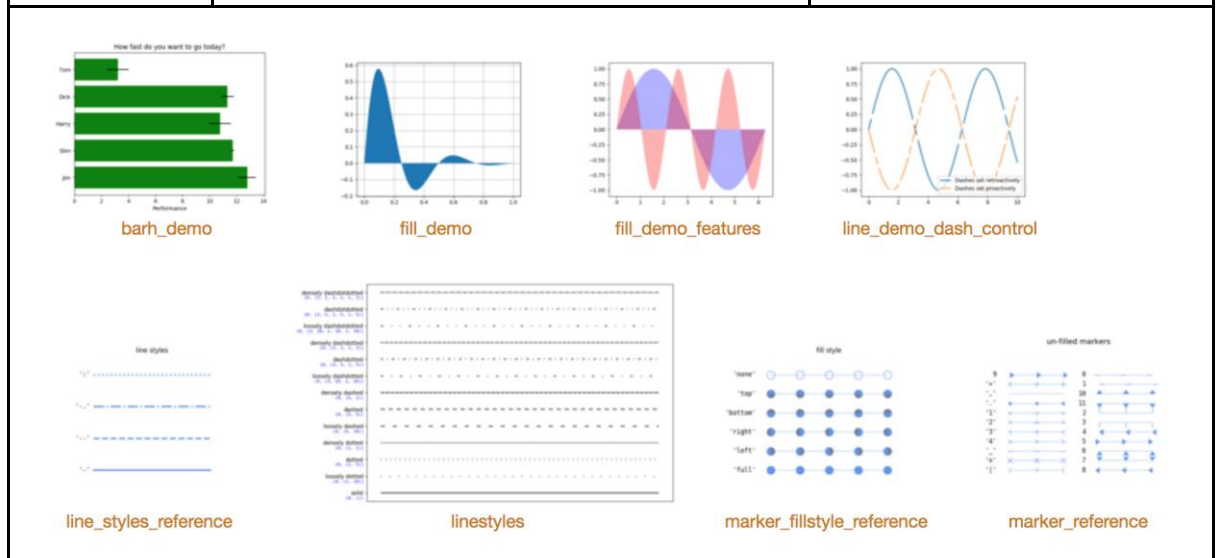
網址：<https://www.python.org/>

Python 是知名的程式語言，好用程度已經不用特別說明，近年更被資料科學領域大量採用，與 R 語言雙雙成為熱門程式語言，也發展出許多好用的視覺化套件，像是 PyQtGraph、Matplotlib、Boken 等等，介紹如下：

Pyhon 函式庫名稱	說明	網址
PyQtGraph	PyQt 是 Python 的應用函式庫，包含了許多函式可調用，而 PyQtGraph 則是從 PyQT 再延伸出來的視覺化函式庫，速度快且應用層面廣泛。	PyQtGraph http://www.pyqtgraph.org/



Pyhon 函式庫名稱	說明	網址
matplotlib	Python 繪製 2D 圖好幫手，且同時支援 IPython , jupyter 的開發環境，可以產生像是長條圖、熱度圖、散佈圖等等的常見圖表。	Matplotlib https://matplotlib.org/




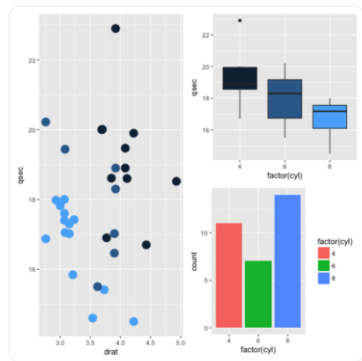
Python 函式庫名稱	說明	網址
	可將 Python 語法自動封裝成網頁呈現的 HTML、JavaScript 格式，在網頁前端進行顯示。	Boken http://bokeh.pydata.org/en/latest/
		

(3) R 語言

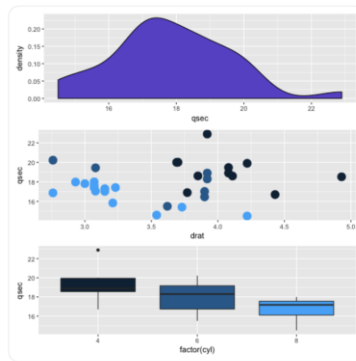
網址：<https://www.r-project.org/>

R 語言的設計，當初就是為了統計分析而推出，也因此有許多統計支援的語法、函式庫，例如變異數 Covariance (cov) 的計算、相關統計分布的運算等等。也因此發展出多樣的視覺呈現套件，像是 ggplot2、ggvis、shiny 等等，介紹如下：

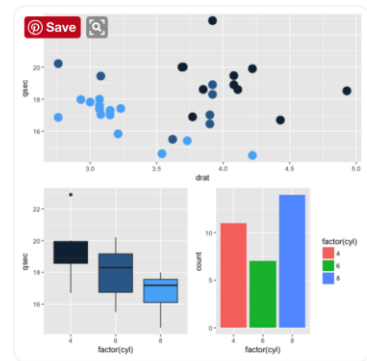
R 函式庫名稱	說明	網址
 ggplot2	R 常見的視覺化函式庫，透過 R 語言直接產出若干統計圖表，擁有非常多種圖表的呈現支援，產出的結果多為靜態。	ggplot2 http://ggplot2.org/



#261 Multiple graphs on same page

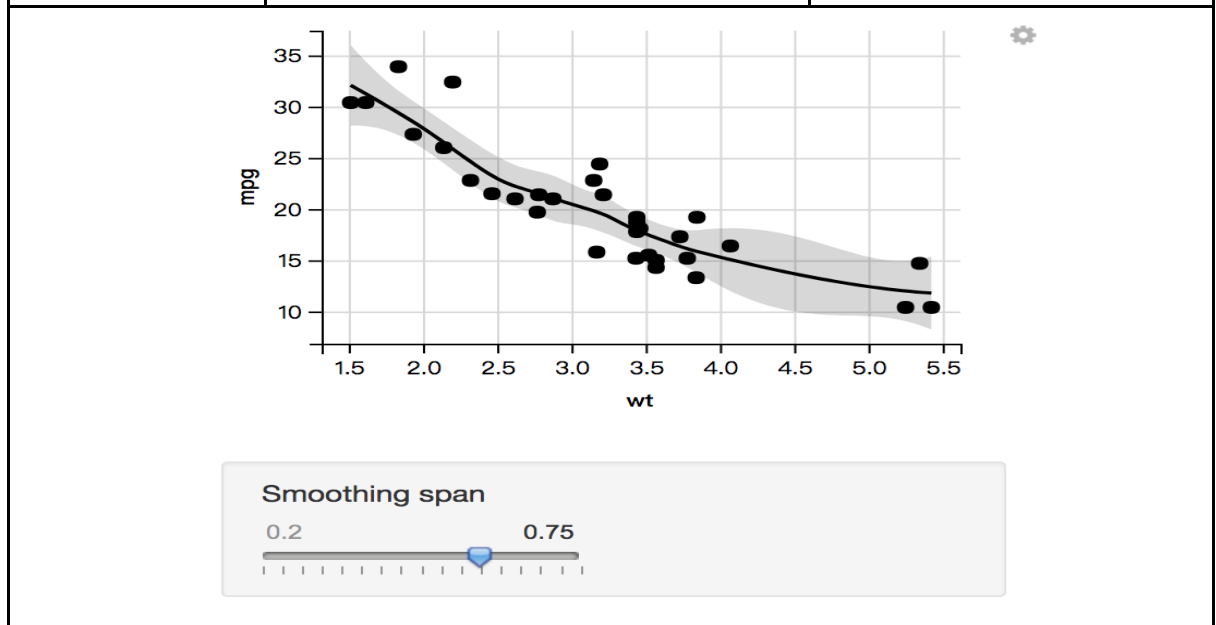



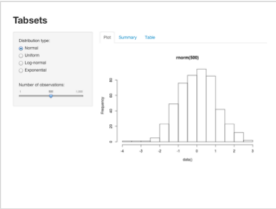

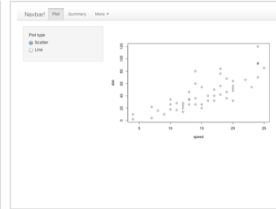
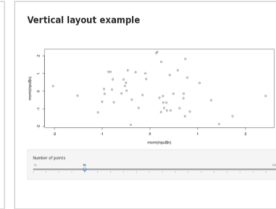
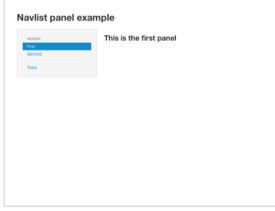
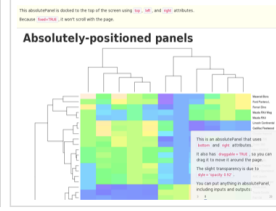

#261 Multiple graphs on same page



#261 Multiple graphs on same page

R 函式庫名稱	說明	網址
ggvis	語法跟 ggplot2 類似，但相對於 ggplot2 能夠做出更有互動性的圖表，也能輸出到網頁當中呈現。	ggvis http://ggvis.rstudio.com/



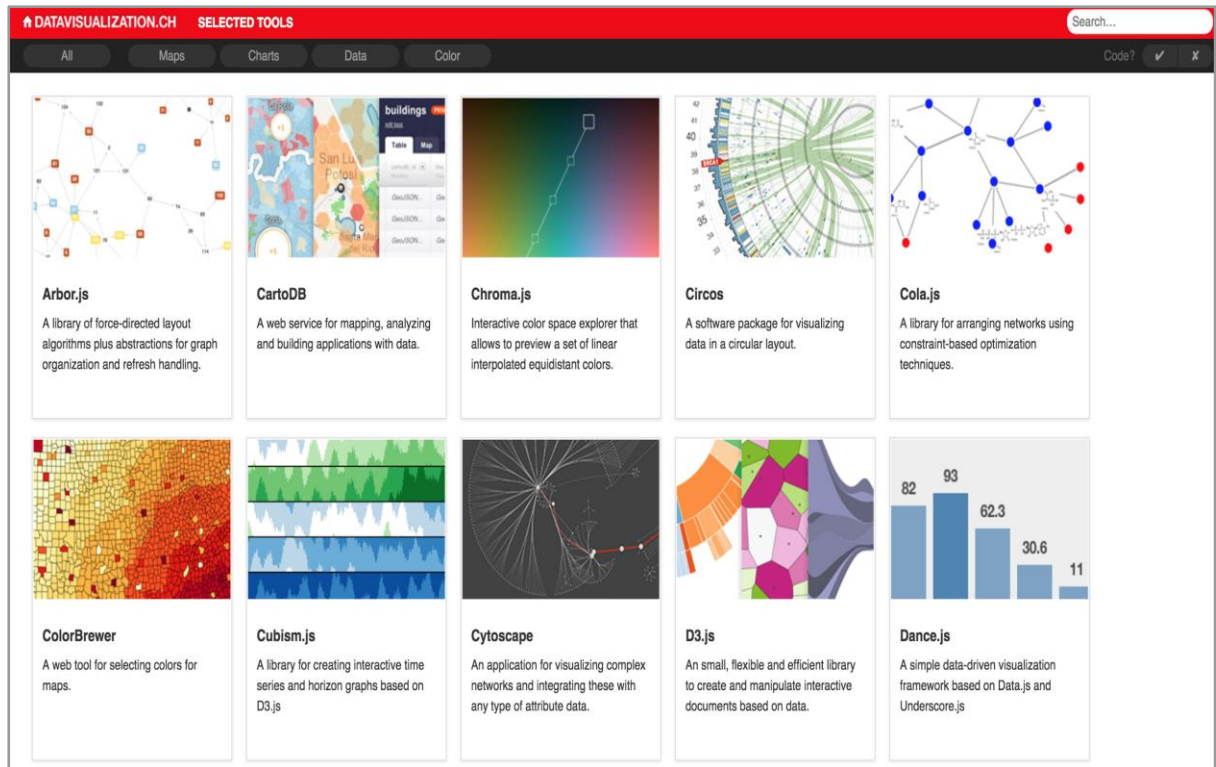
R 函式庫名稱	說明	網址	
	透過 R 語言的撰寫，可自動轉譯為網頁前後端程式碼，對於不熟悉網頁程式，但熟悉 R 語言的人來說非常方便。	shiny https://shiny.rstudio.com/	
			
Tabsets	Plot plus three columns	Navbar Example	Vertical Layout
			
Retirement simulation	navlistPanel example	Absolutely-positioned panels	Including HTML, text, and Markdown files

(4) JavaScript

網址：<https://www.javascript.com/>

隨著世界逐漸雲端化，各類服務架構都搬到網站上面，而 JavaScript 是賦予網頁靈魂的程式語言，負責各種複雜的視覺操作與互動邏輯，也能整合各類網頁元件，成為全世界最主流的網頁視覺互動程式語言，間接讓『前端工程 (F2E)』的領域在近幾年逐漸興起。

網頁視覺化的大量需求，催生許多高手設計出各類 JavaScript 視覺函式庫，最常見的就是 D3.js，提供了超級大的邏輯彈性，讓使用的人創造出許多美麗且互動性十足的資料視覺化作品。同時間還不斷有新的 JavaScript 函式庫一直出現，本篇文章就挑選了其中 7 款，於下一個段落進行介紹。



資料來源：<http://selection.datavisualization.ch/>

圖 11、由 **datavisualization.ch** 網站整理的 **JavaScript** 視覺化函式庫介紹網站

視覺化工具：前端程式外掛類

網頁是常見的視覺化介面呈現地點，而網頁的視覺則依賴前端程式（Html + Css + JavaScript）的協力撰寫來完成，現今各式各樣的網路媒體、系統資訊、新聞資訊等都依賴前端程式來完成，本段落將會介紹一些知名的前端視覺函式庫，只要整合業務與資料邏輯，就能做出各類美麗圖表。

(1) D3

網址：<https://d3js.org/>

D3 已經成為全球最最知名的資料視覺化函式庫之一，提供了視覺創作與互動程式的最大彈性，透過 SVG 格式與 HTML 完美整合，檔案小且完全免費。

D3 最大的缺點在於學習門檻高，因為彈性需求的關係，許多視覺圖表的基本功能都要手動設定，不像許多函式庫都會偷偷幫使用者處理好，D3 屬於較進階的視覺化工具。

Data-Driven Documents



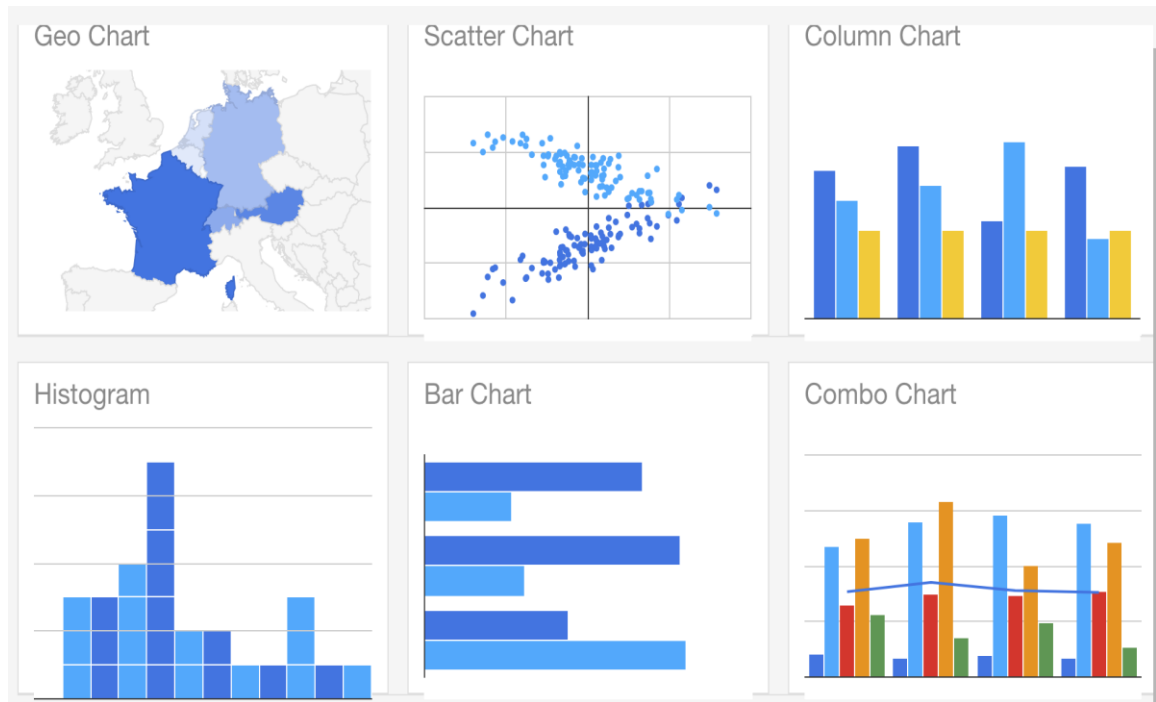
資料來源：<https://d3js.org/>

圖 12、D3 視覺化圖表多元，可做出各種變化

(2) Google Chart

網址：<https://developers.google.com/chart/>

由 Google 推出的視覺化工具，也是免費使用，提供了非常棒的製圖環境，只要將 Google Chart 函式庫引入，即可馬上進行使用，也能搭配動畫與互動功能，最大的優點應該是由知名公司所推出，所以更能確保未來會持續維護。



資料來源：<https://developers.google.com/chart/interactive/docs/gallery>

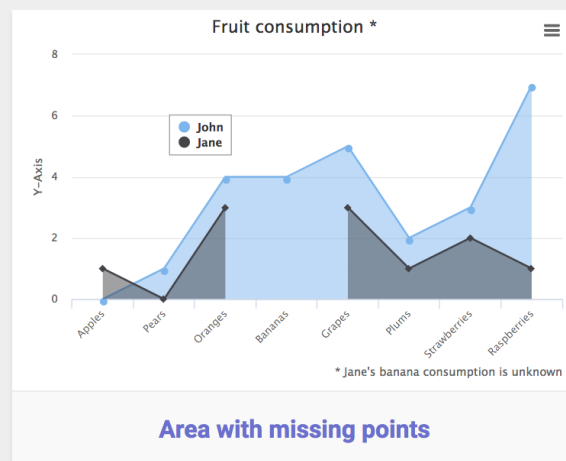
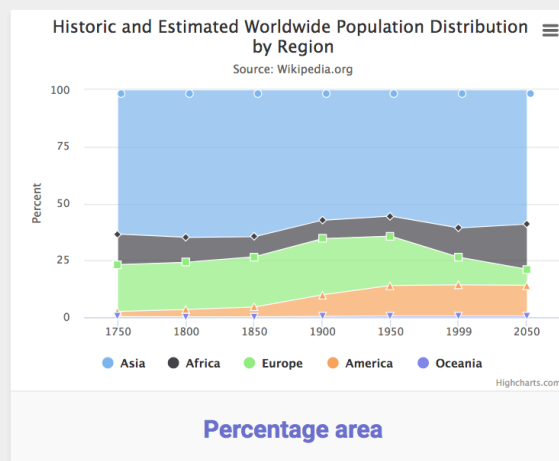
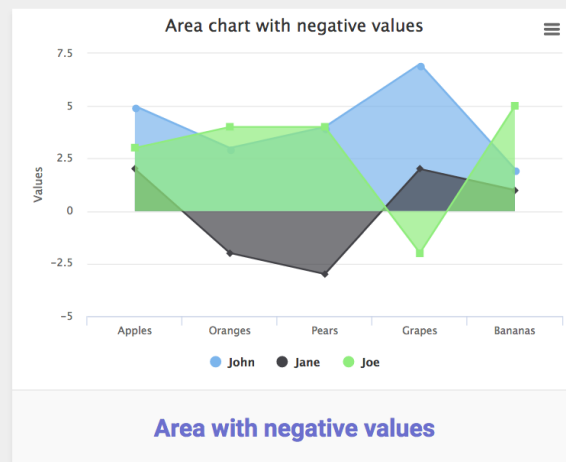
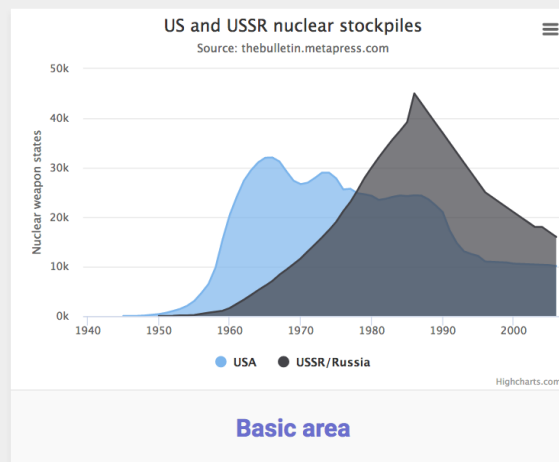
圖 13、Google Chart 視覺化圖表

(3) HighChart

網址：<https://www.highcharts.com/>

Highcharts 由 JavaScript 所撰寫而成，非商用免費而進階使用需進行付費，特色是相關瀏覽器相容性跟擴充性都做的很好，也支援非常多種類的視覺圖表，實際套用專案也很容易，是資料視覺化的殺手級應用。

Area charts



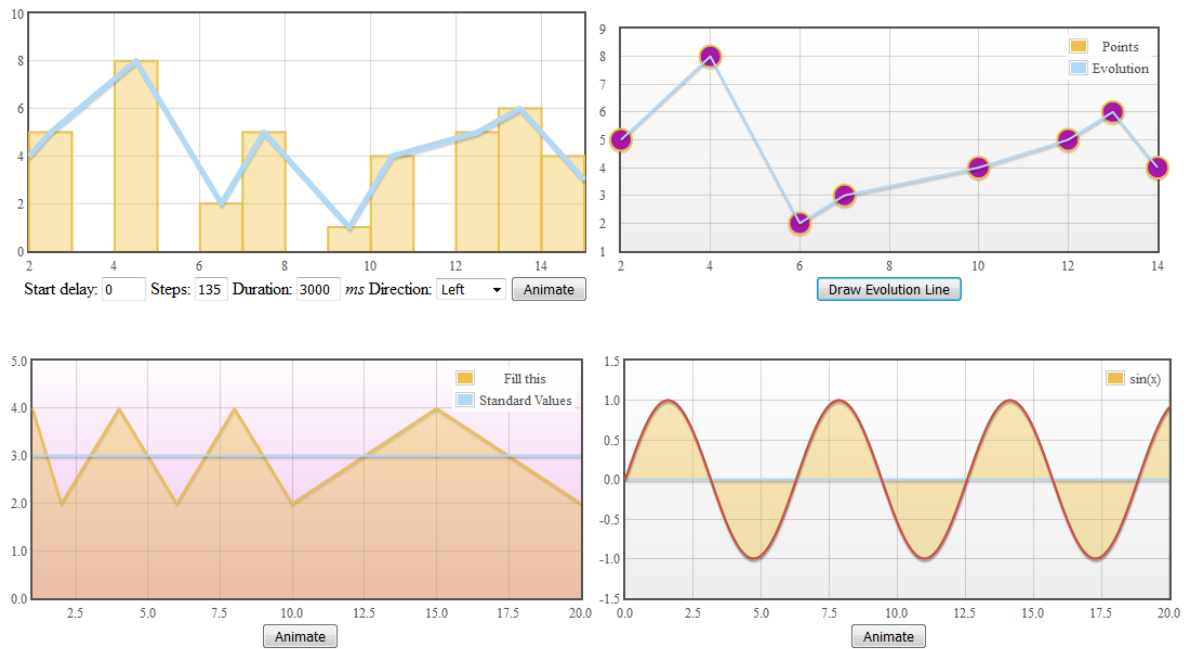
資料來源：<https://www.highcharts.com/demo>

圖 14、HighChart 視覺化圖表，線上有非常多的 demo 圖表可觀看

(4) Flot

網址：<http://www.flotcharts.org/>

Flot 函式庫發展多年，特性是與 jQuery 函式庫的完美整合且免費，對應的瀏覽器支援度也處理的很好，甚至向下支援到 Internet Explorer 6+ 環境。



資料來源：<http://javascript-html5-tutorial.net/2016/01/16/animated-charts-in-jquery-flot-animator.html>

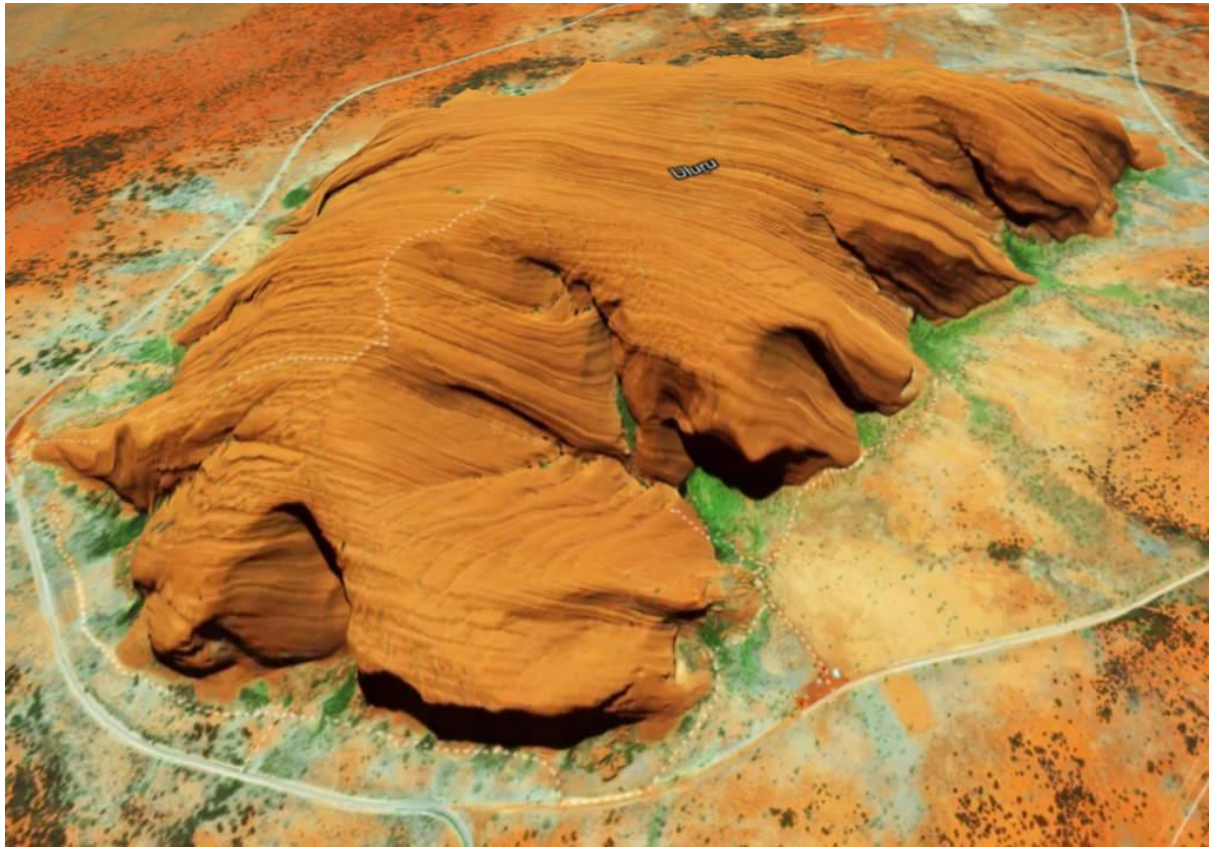
圖 15、Flot 視覺化圖表

(5) Three.js

網址：<https://threejs.org/>

Three.js 如同它的名字，就是 Three + JavaScript 的意思，也就是透過 JavaScript 實現 3D 空間視覺的呈現，其背後的技術是 WebGL（瀏覽器呈現 3D 視覺的一套規範），原生的 WebGL 程式碼不易撰寫，但 Three.js 將 WebGL 包裝成為高階 3D 程式語言，讓我們在網頁創造豐富的 3D 資料視覺化。

這邊要補充說明，並非所有的瀏覽器都支援 WebGL 技術，但現今主流的瀏覽器都已經有支援，像是最新的 Chrome、Firefox、Safari、IE11+ 等等，所以現今才會越來越多網頁敢採用 WebGL 製作。



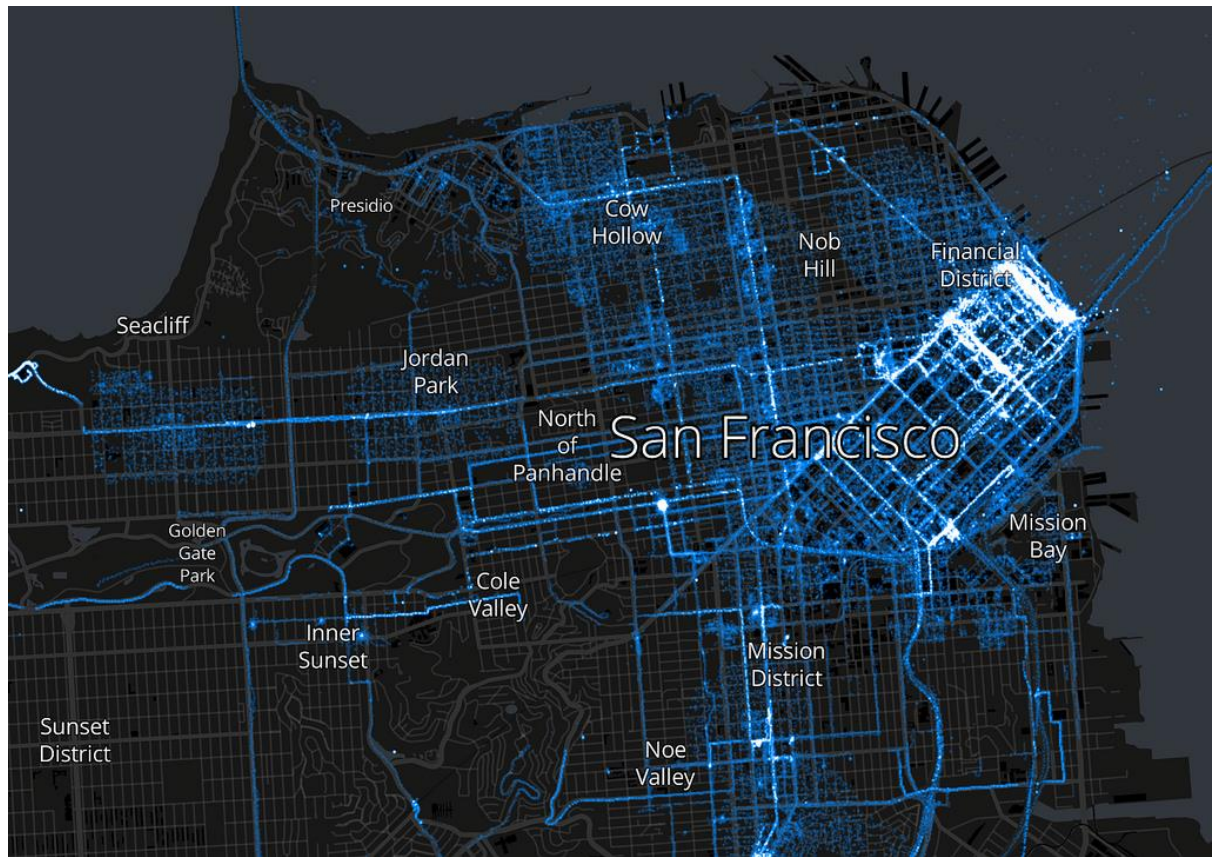
資料來源：<https://www.mapbox.com/blog/3d-terrain-threejs/>

圖 16、透過 Three.js 引擎製作的 3D 地質視覺網頁

(6) Mapbox

網址：<https://www.mapbox.com/>

經緯度數據是常見的視覺化資料型態，而整合 3D 圖層、GIS 來呈現地理位置等等也是視覺化常見需求。Mapbox 基於『開放街圖』的概念執行地理位置視覺架構服務，利用 OpenGL 技術，產生輕量化的向量圖檔，可在不同裝置上渲染視覺地理資訊。



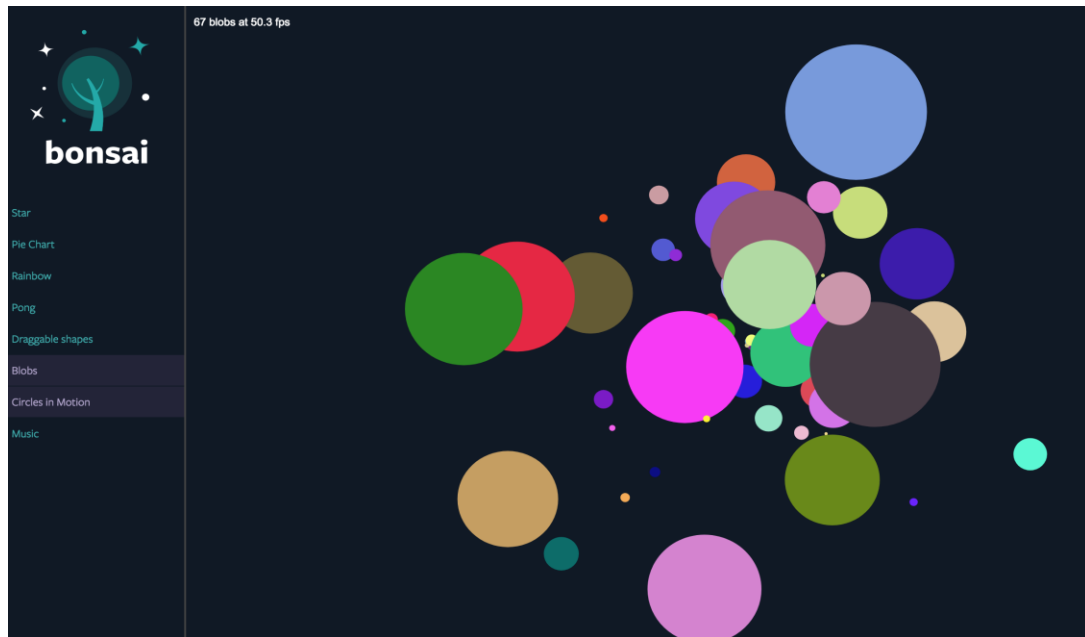
資料來源：http://mapbox6.rssing.com/chan-4301981/all_p18.html

圖 17、Mapbox 是地理資訊視覺化的重要函式庫

(7) bonsai

網址：<https://bonsaijs.org/>

bonsai 的強項在於動畫效果，透過 SVG 格式與各類動態 API 來實現，像是顏色設定、梯度變化設定、灰階設定、透明度設定等等，等於是將動畫效果進行高階語法包裝，讓開發者能夠更輕鬆實現各類互動視覺。



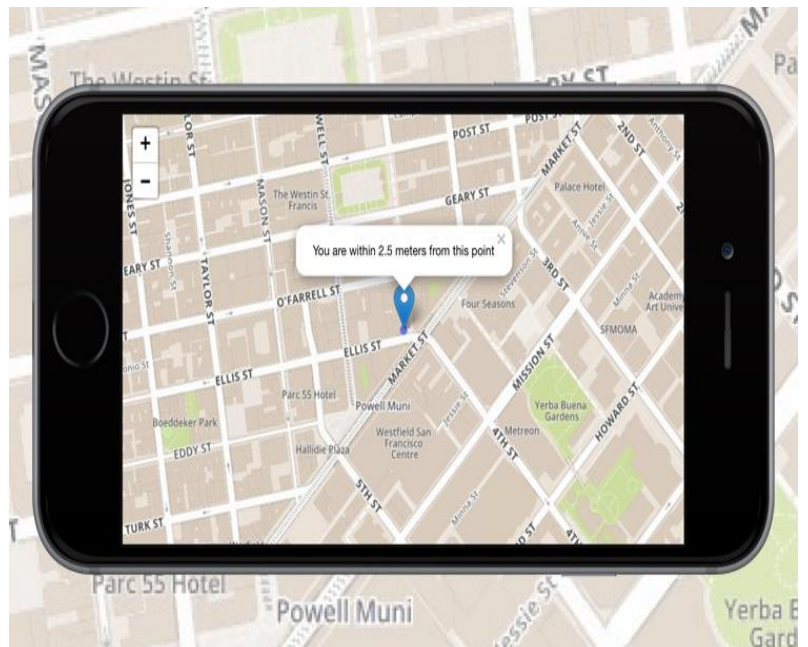
資料來源：<http://demos.bonsaijs.org/demos/blobs/index.html>

圖 18、bonsai 製作的 3D 動態視覺網頁

(8) Leaflet

網址：<http://leafletjs.com/>

Leaflet 設計的目標，就是在手機上呈現良好的互動地圖，最大的特色在於架構簡單、效能優秀、實用性佳，也達成跨瀏覽器的標準需求，且在手機上呈現地圖常常會有效能的議題，此部分 Leaflet 設計的很良好，且也能讓較低階的瀏覽器也能執行順暢。



資料來源：<http://leafletjs.com/examples.html>

圖 19、Leaflet 套件可供手機介面視覺化呈現地理資訊

(9) animate.css

網址：<https://daneden.github.io/animate.css/>

上面介紹了許多套件，大多是屬於 JavaScript 的套件，但 animate.css 則是屬於 CSS 技術的工具。對於設計師來說，良好的資料視覺動畫需要準確控制速度與節奏，但是這類顯示的細節如果必須自行寫程式則頗為麻煩，這時就可透過 animate.css 套件來達成像是『彈力』、『減速』、『加速』、『煞車』等等動態效果，加強整體的互動體驗。



來源：<http://blog.infographics.tw/2016/11/animate-css/>

圖 20、animate.css 套件提供的各種物理視覺引擎

說在最後

本篇零零總總介紹了許多視覺化套裝軟體與程式外掛，其實各自工具都有擅長的強項，讀者不用覺得是全部要學起來，建議讀者思考自己對哪些工具特別有興趣，可以挑選幾項深入鑽研，了解工具背後設計的邏輯，先選擇幾個熟悉後，之後就能評估各種工具的優劣，並挑選出真正適合自己的需求的工具。

下個月的大主題跟本月相同，都是『資料視覺化的神兵利器』系列，但會更專注分享各類線上視覺化服務，敬請讀者期待。