R crawler note.

Using xml2 for crawling

<https://rpubs.com/skydome20/R-Note13-Web-Crawler-on-CIA-CREST-by-xml2>

text mining:

<http://rstudio-pubs-static.s3.amazonaws.com/12422_b2b48bb2da7942acaca5ace45bd8c60c.html>

參考資料:

進階的使用rvest爬Indeed職缺內容(英文)

http://stat4701.github.io/edav/2015/04/02/rvest\_tutorial/

<http://chihchengliang.github.io/DSC2015_Crawler/>

R字串相關函數教學

<https://www.ptt.cc/bbs/Statistics/M.1277714037.A.2CC.html>

<http://molecular-service-science.com/2015/01/18/text-processing-in-r-using-grep/>

文字探勘相關教學

<http://rstudio-pubs-static.s3.amazonaws.com/12422_b2b48bb2da7942acaca5ace45bd8c60c.html>

<http://www.slideshare.net/tw_dsconf/r-64232974>

<https://cran.r-project.org/web/packages/text2vec/vignettes/text-vectorization.html>

還沒看完的：

<https://rmarkdown.rstudio.com/lesson-3.html>

<https://medium.com/@kstseng/r-markdown-template-a4b0449a56d5>

<http://map-testing.blogspot.tw/2016/08/rstudio-markdown.html>

https://knowlab.wordpress.com/2016/12/05/以-r-markdown-輕鬆編輯資料分析報告（上）/

## test R pipe 'magrittr'

a\_list<-list(1:6,3:5,4:7)

a\_list%>%unlist%>%unique%>%sort%>%{

lapply(a\_list,setdiff,x=.)

}

a=1

f=function(a)(a+1)

f(a)

a%>%f

a%>%f()

a%>%f(.) ## . 代表前面的變數

## %<>% 傳入變數與值dat = data.frame(a = 1:3, b = 8:10)

dat = data.frame(a = 1:3, b = 8:10)

dat = dat %>% rbind(dat)

a=1

a%<>%add(1)

### crawler practice

<https://hk.saowen.com/a/8e2d738e015011cd6648742b975022e971c17f23117b9fca4ab59b7d68c2520d>

Sys.setlocale(category = "LC\_ALL",locale='cht')

install.packages('XML')

install.packages('RCurl')

install.packages('httr')

## teacher demo.1

rm(list=ls(all.names=TRUE))

library(XML)

library(RCurl)

library(httr)

Sys.setlocale(category = "LC\_ALL", locale = "cht")

urlPath <- "https://www.ptt.cc/bbs/movie/index.html" ##把網址抓下來

temp <- getURL(urlPath, encoding = "big5") # "big5"? utf8?

xmldoc <- htmlParse(temp) ## 轉成html的檔案

title <- xpathSApply(xmldoc, "//div[@class=\"title\"]", xmlValue)

title <- gsub("\n", "", title) #把\n弄掉

title <- gsub("\t", "", title) #把\t弄掉

author <- xpathSApply(xmldoc, "//div[@class='author']", xmlValue)

## get what u need ex: if u want article ; "div[@id]=\"main-content\"

(path <- xpathSApply(xmldoc, "//div[@class='title']/a//@href"))

date <- xpathSApply(xmldoc, "//div[@class='date']", xmlValue) #xmlvalue??

response<- xpathSApply(xmldoc, "//div[@class='nrec']", xmlValue)

alldata <- data.frame(title, author, path, date, response)

write.table(alldata, file = "movie.csv")

**RVEST CRAWLER NOTE**

如何爬蟲

Using xml for crawling

* Read XML and HTML with read\_xml() and read\_html().
* Navigate the tree with xml\_children(), xml\_siblings() and xml\_parent(). Alternatively, use xpath to jump directly to the nodes you’re interested in with xml\_find\_one() and xml\_find\_all(). Get the full path to a node with xml\_path().
* Extract various components of a node with xml\_text(), xml\_attrs(), xml\_attr(), and xml\_name().
* Convert to list with as\_list().
* Where appropriate, functions support namespaces with a global url -> prefix lookup table. See xml\_ns() for more details.
* Convert relative urls to absolute with url\_absolute(), and transform in the opposite direction with url\_relative(). Escape and unescape special characters with url\_escape() and url\_unescape().
* Support for modifying and creating xml documents in planned in a future version.

ALL : <https://yijutseng.github.io/DataScienceRBook/io.html#csv>

**note:**

* 日期轉換用 lubridate package  
  年/月/日格式的文字轉換為日期物件 : ymd()  
  月/日/年格式的文字轉換為日期物件 : mdy()  
  EX: ymd('2012/3/3')= "2012-03-03"
* strsplit 切割： 根據某種規則 strsplit(“Hello World”,””) 🡪 “Hello” “World”
* gsub取代：gsub(‘要取代的東西’, ‘取代成什麼’, ‘你的物件’)
* 回傳符合條件之向量位置(index) grep(搜尋條件,要搜尋的向量)

回傳每個向量是否符合條件(TRUE or FALSE) ***grepl***(搜尋條件,要搜尋的向量)

* nrow= 回傳列數 ncol= 回傳行數
* 處理遺漏值：只保留完整資料，那列只要有NA，就刪掉。  
  data[complete.cases(data),]

單變項的處理：data[!is.na(data)]

* names() 指定names給物件

**DPLYR**

Targeted for data analysis in *data.frames*

*remember to combine with ‘magrittr’*

|  |  |
| --- | --- |
| **select()** | ***select columns***   select(data, column1,column2..,starts\_with(‘’)) column1:column10 ; -column1   * starts\_with(‘’) * ends\_with(‘’) * matches(‘’) * one\_of() |
| **filter()** | ***filter rows***  filter(data,rowselection1, rowselection2..) \* row %in%\_\_\_\_\_\_ 選定column後，你想挑選的列名   * 裡面可以放條件式 |
| **arrange()** | ***re-order or arrange rows***  arrange(data,column1 , column2) arrange column1 by column2   * desc() order a column in descending order   EX: arrange(data,desc(column.name)) |
| **mutate()** | ***create new columns***  mutate(new variable=……,new variable=…..)  可以在同一行中狂增變項，一行搞定  如果只想留這兩個新的變項用transmute() |
| **summarise()** | summarise values summarize(name=function(),……….,***total=n()***) |
| **group\_by()** | * n() 個數 * n\_distinct(\_\_column\_\_\_\_) 不重複   allows for group operations in the “split-apply-combine” concept |

用哪一種方式group\_by

group\_by (\_\_variable\_\_) 然後可以用pipe operator 進行上面的function，後再併在一起。

**sample\_n**(data,你要幾筆) ; sample\_frac() take random samples

**rename**(data,new.column.name=old.column.name)

**data frame (line=…,text=…)**

**Using APPLY**

<http://blog.fens.me/r-apply/>

apply(X, MARGIN, FUN, ...)

X: matrix / dataframe / vector

Margin: 1=row ; 2=column

fun: function

後面放function 的內容

EX:

myFUN<- function(x, c1, c2) {

+ c(sum(x[c1],1), mean(x[c2]))

+ }

apply(x,1,myFUN,c1='x1',c2=c('x1','x2'))

lapply(X,Function) 針對data frame 跟 list

sapply(X, FUN, ..., simplify=TRUE, USE.NAMES = TRUE)

針對data frame 跟 list

* simplify: 是否数组化，当值array时，输出结果按数组进行分组

USE.NAMES: 如果X为字符串，TRUE设置字符串为数据名，FALSE不设置

sapply(a,function(x) matrix(x,2,2), simplify='array')

vapply(x,fun, FUN.VALUE =….)

x=vector matrix data.frame

FULL COVERAGE <https://yijutseng.github.io/DataScienceRBook/io.html#api>

**R 繪圖**

**## Basic plotting**

plot()

points(x=,y=,) 在原本圖上加點，先把資料選出來

**## Lattice**

install.packages(‘lattice’)

library(‘lattice’)

畫boxplot

bwplot(x=y~x,data=..,xlab=…)

畫3D圖

cloud(x=A~B+C | D, data=\_\_\_\_)

-A: Z 軸

-B/C: X Y軸

-D: 根據什麼分類

畫 scatter plot

xyplot(x=y~x,data=\_\_\_,group=\_\_\_\_,auto.key=list(…))

auto.key

* space ex:’top’ 在圖上方
* column 幾乘幾的標籤 group
* title 標題ㄋ
* cex.title= 1,2,3… 字體大小

使用panel function 來控制圖表內資訊

panel=function(){}

ex:

* panel,xyplot()
* panel.abline()

**https://plot.ly/ggplot2/**

<https://yijutseng.github.io/DataScienceRBook/vis.html#section-8.1>

**## ggplot**

**要先install(‘ggplot2’)**

**qplot**

qplot:用qplot可以畫出hist / scatter/ box ..etc

x=

y=

data=

geom= ‘histogram’ 直方圖 / ‘point’ 散佈圖 / ‘density’ 機率密度圖 / ‘boxplot’ 盒須圖

main

xlab

binwidth 每＿＿單位一區隔

fill 用什麼資料填滿顏色 (用於類別變項)

color 以顏色標註資料的某一變項

geom\_smooth(): using method = ‘loess’

facet = row直向分類~column橫向分類 ＃如果只設定單一方向，另一個用”.”表示

binwidth = 數字 代表分組間隔，x軸每多少分一組

**ggplot**

先建畫布，再把東西加上去

*Geometric Objects*

* geom\_histograms() / geom\_point() 點子圖 / geom\_line() 畫線 / geom\_polygon() / geom\_errorbar() / geom\_smooth() 劃趨勢線:method=’lm’ ; 預設是lowess

/ geom\_bar 直方圖

# ggplot(\_\_data\_\_) + …….

***前面要用Geometric object 包起來 (***

# aes()

* x=
* y=
* main=
* color= 某資料變相
* fill =用什麼資料填滿顏色

***)***

# facet\_grid(\_\_~\_\_) 將某個變項分類後分開畫 row直向分類~column橫向分類

# labs (x=,y=,title=…)跟main 一樣

# theme\_bw() 背景變黑白

# theme\_gray 預設樣式

# coord\_polar(‘’,start=0) 沿著Y，轉成圓餅圖ㄅ

# xlab; ylab ; labs(x=,y=), ggtitle()

**ggmap**

<https://blog.gtwang.org/r/r-ggmap-package-spatial-data-visualization/2/>

<https://yijutseng.github.io/DataScienceRBook/vis.html#choropleth-map>



install.packages(‘ggmap’,type=’source’)

library(ggmap)

library(mapproj)

get\_map

A<-get\_map(location=’\_\_\_\_\_’,zoom=\_\_\_)

location🡪c(lon=….,lat=…..) or 如果已經有內建國家，可以收錄

* maptype=指定地圖類型 ：   
   ‘terrain預設’; ‘roadmap’ ; ‘satellite’; ‘hybrid地圖加馬路’ ; ‘toner-life’黑白地圖 ; watercolor

zoom=? 放大倍率

language= ‘’ 地圖語言 ‘zh-TW’ 中文

ggmap(A) 負責畫出地圖本身

* darken:讓地圖變亮變暗 EX: darken=0.5

**結構：ggmap(\_\_) + geom\_point(\_\_\_\_\_\_\_)**

geom\_point() 加上資料點

- aes(x=\_\_\_,y=\_\_\_\_,size=\_\_\_\_\_) x=放longitude ; y=latitude ; size: 依據不同的量 讓圓圈大或小

- data=\_\_\_\_\_\_

ggplot()+

＃geom\_polygon（）

data=\_\_\_

aes(x=\_\_,y=\_\_,group=\_\_\_\_,fill=\_\_\_\_\_)

color=\_\_\_\_\_

size=\_\_\_\_\_

＃coord\_map（）

＃scale\_fill\_distiller（）

* name=’\_\_\_’ 地圖旁的條例的名字
* palette=\_’\_\_\_\_\_’ EX: YIGn (yellow to green); 可以用display.brewer.all看全部顏色
* breaks= pretty\_breaks(n=＿＿) 好的切割點

＃labs (title=\_\_\_\_\_)

**Chloropleth map面量圖**

install.packages(c("choroplethr","choroplethrMaps")) ##第一次使用前先安裝

library(chloroplethr) 只內建美國各州地圖與人口學資料

World development indicators 有很多資料，可以做面量圖，要上網查代碼：

code= “SP.POP.TOTL”…..

title=’’

year=

zoom參數設定想畫的國家，國家的名稱設定必須要和country.regions資料完全 吻合 EX: zoom=c('taiwan','japan','south korea','philippines')

choroplethr\_wdi(code="SP.POP.TOTL", year=2015,

title="2015 Life Expectancy",

zoom=c('taiwan','japan','south korea','philippines'))

more plotting info:<http://r4ds.had.co.nz/data-visualisation.html>

**## 輸出圖表**

png(‘.png’, width=..,height=..)

Using FB API

Facebook Crawler

<https://blog.gtwang.org/r/facebook-social-media-mining-with-r/>

<https://kangliping.wordpress.com/2017/04/27/r_text_mining_fanpage_rstudio/>

# idea 有點想畫花蓮縣所有的7-11

<https://geocoder.ccjeng.com>

<https://blog.ccjeng.com/2015/09/Batch-Geocoder.html>

[https://emap.pcsc.com.tw/#](https://emap.pcsc.com.tw/)

<https://data.gov.tw>