調整傳統季節分類曆法 (多變量分析期末報告)

李侑瑾、陳俊翔、張浩榜、許劭廷

國立東華大學應用數學所統計組

2018/06/14

Outline

前情提要

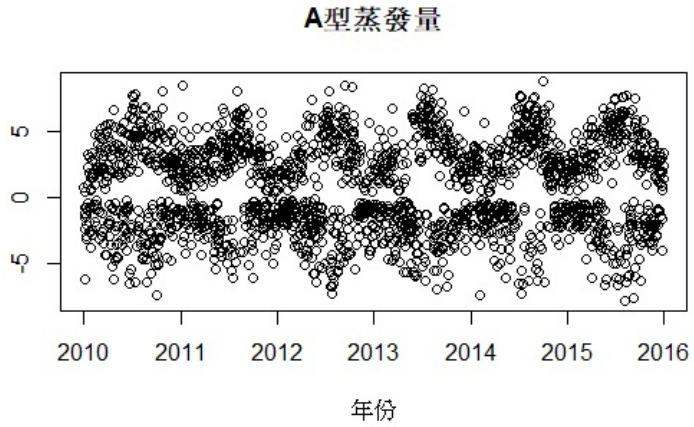
1. unsupervised learning

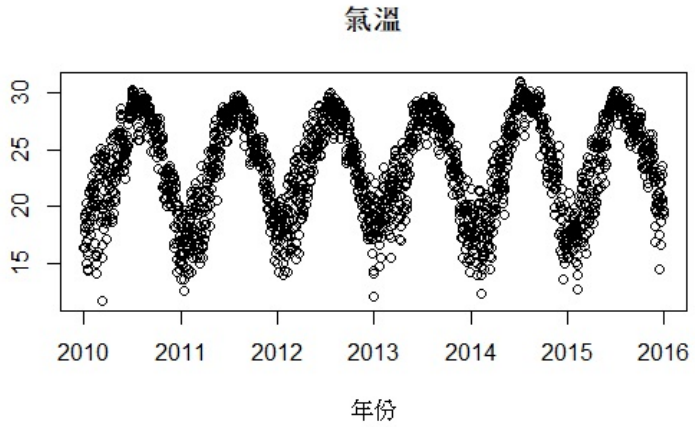
→ 做 cycle、找出分群法

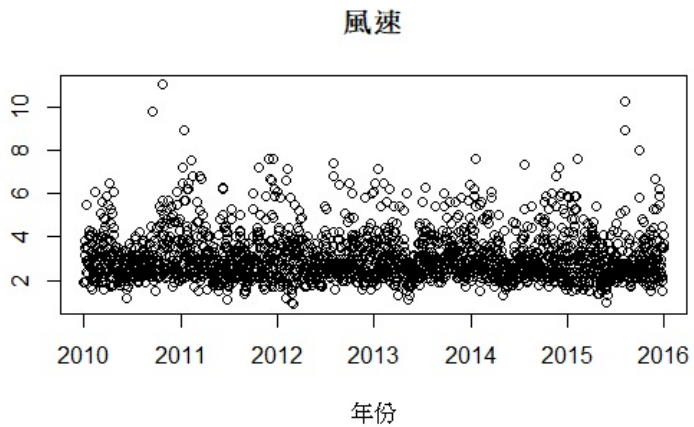
2. supervised learning

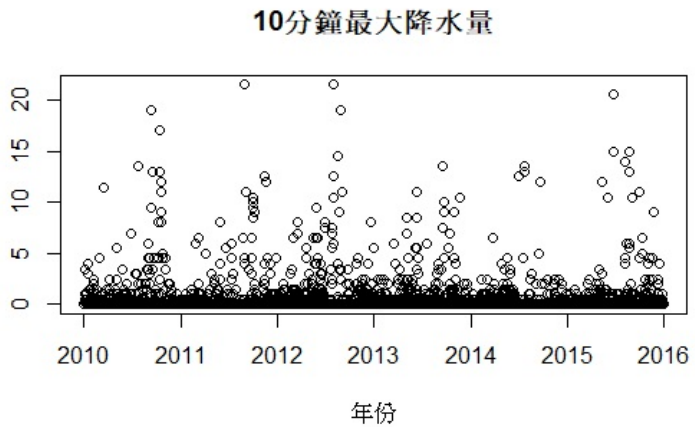
→ 根據現有春夏秋冬做分群

(續) 前情提要


A型蒸發量
-5
0
5
2010
2011
2012
2013
2014
2015
2016
年份


氣溫
15
20
25
30
2010
2011
2012
2013
2014
2015
2016
年份


風速
2
4
6
8
10
2010
2011
2012
2013
2014
2015
2016
年份


10分鐘最大降水量
0
5
10
15
20
2010
2011
2012
2013
2014
2015
2016
年份

註解頁

我們本來是有兩個計畫的，第一個是 unsupervised learning: 去做 cycle、找出分群法，另一個則是我們本次的報告，supervised learning: 根據現有春夏秋冬做分群當我們在做第一個計劃時出了一些問題，第一是做出的 cycle 跟現在的曆法幾乎一樣，而且也不是所有變數都能找出 cycle，就算有找出並做迴歸方法，因為是 unsupervised learning 所以會有太多分法都是合理的，每個職業關心的分法都不一樣，以我們現在的能力工程過於浩大；第二是還要配合時間去分，不然倘若 1.3.5 月一群，2.4.6 月一群，這樣根本沒有意義了。所以後來考量上訴因素就改成第二個計劃了。

為什麼要調整?

- 人為曆法
- 真實天氣

→ 調整曆法以貼合天氣

註解頁

現在的曆法可能跟真實天氣有所衝突，畢竟全球暖化等等天氣因素，造成幔股創下的曆法跟現在會出現些問題。我們想利用統計方法定義出符合現在的四季的曆法，是否應該調整現在的曆法。

資料與分類工具

- 中央氣象局花蓮測站 2010 年到 2018 年天氣資料

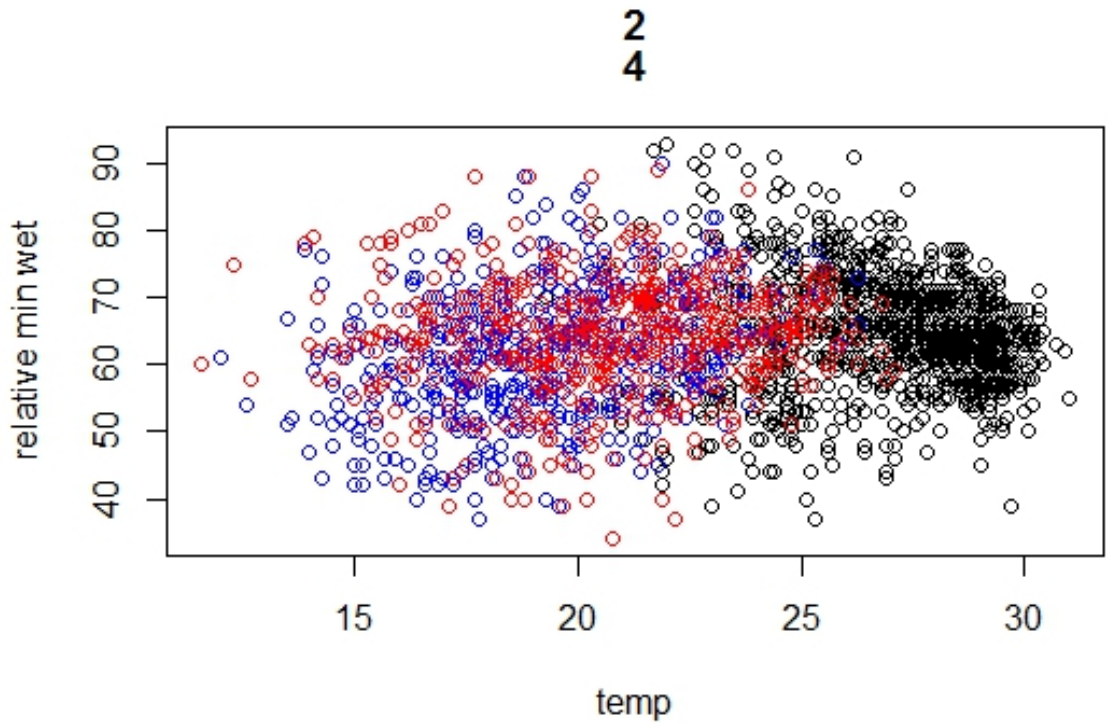
→ 氣壓、溫度、風、降水、日照

- SLR
- PCA
- CART
- AdaBoost

該怎麼做

以 2010 到 2015 年作為 Training data

1. 目標: 利用 training data 區分「冬春」以及「夏秋」

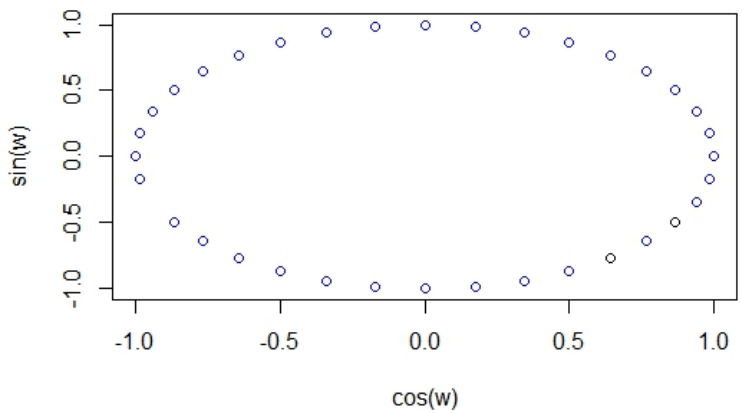


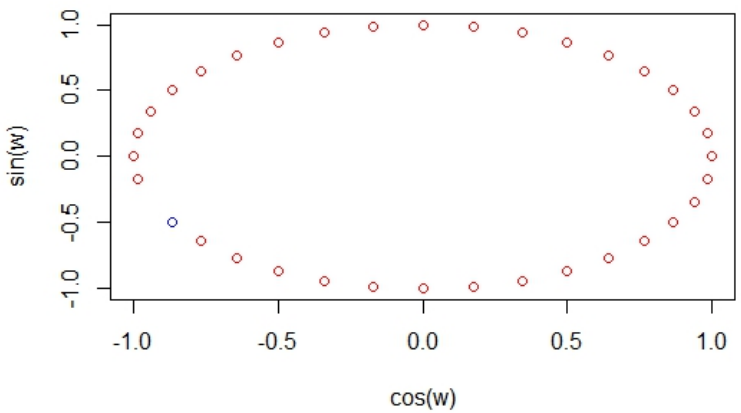
註解頁

利用 training data 區分「冬春」以及「夏秋」，但以現有技術難以再分四季了，像是此圖是對第二個變數以及第四個變數作圖，藍色跟紅色是現今曆法的冬天跟春天，很明顯是分不開的，而很多都是很難分開的，所以我們先以大方向區隔出「冬春」以及「夏秋」就好。

(續) 該怎麼做

2. 去除相關變數
3. 風向轉換 (角度 vs 向量)





註解頁

2. 去除相關變數

用 correlation 檢查其相關性，如果高度相關則擇一留下，像是溫度與當日最高溫有很高的相關性，就留下溫度即可。其實在做去除變數的時候，我們有其中四個變數有高度相關，但用 p-value 只有一個被變數需要去除，另外三個 p-value 都很大，後來發現是一個叫”Multicollinearity”，的現象，所以我們用”並陳”的方式只留其一。

3. 風向轉換 (角度 vs 向量)

在經過第二個步驟之後，我們將原有的 23 個變數減少成 15 個，那在這 15 個變數中，還有兩個是有關風向的變數，我們特別將它拿出來討論，進行第三個步驟。

因為在一般的認知中，風向是二維度的呈現，在資料中它是用角度呈現，也就是一維度的數值，所以這樣就會產生一個問題就是，當我們二維度的資料用一維度去呈現時，一定會產生某些資料的流失。以分別是 1 度跟 359 度做比較，所以他們差了兩度。實際上是非常接近的，但在數值上呈現卻差了 358 度。

(續) 該怎麼做

4. 做出 weak classify function
5. AdaBoost

→ 給予各個 weak classify function 權重，結合成一個決策函數。

6. 以 2010 到 2015 年來評估分群績效
7. 觀察 2016 到 2018 年的天氣是否有需要微調

註解頁

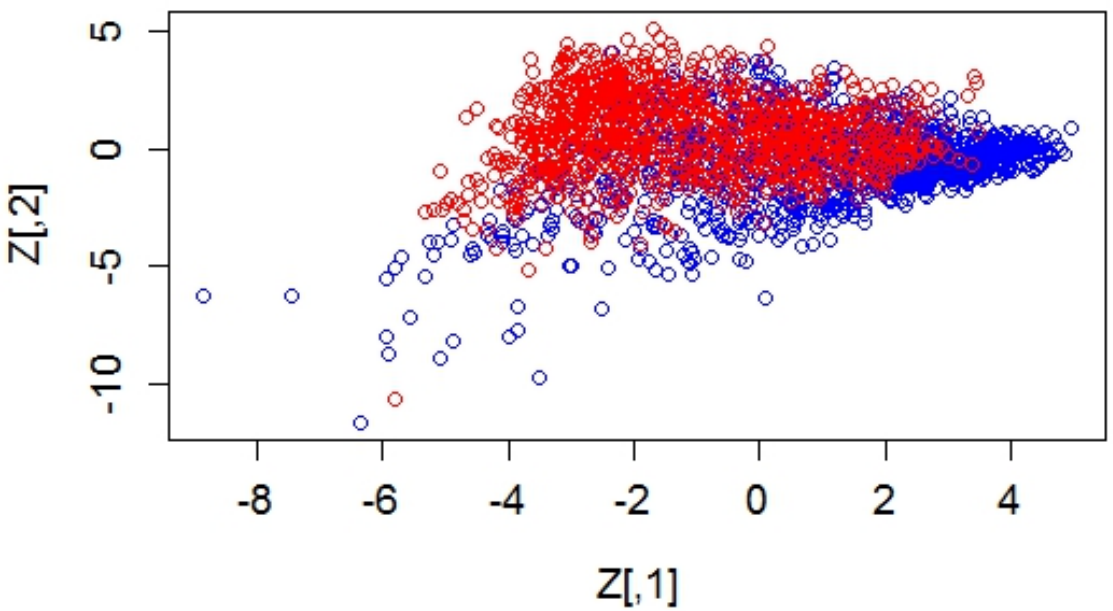
在決定使用哪些變數後，我們要開始處理這些變數，在處理變數的過程中我們會使用 rpart、SLR、PCA、weak classify function 這幾種方式將各組資料做分群。

在將各組資料作完分群後，我們要透過 AdaBoost 的方式給予各個分組適當的權重，並且將其結合成一個決策函數，在完成決策函數後，接著我們先以 2010 到 2015 年的資料來評估這個決策函數的分群績效，若該決策函數的績效是不錯的，那麼我們再以 2016 到 2018 年的天氣資料來做比較，決定是否現行的曆法需要因為氣候的變遷而有所異動。

weak classify function

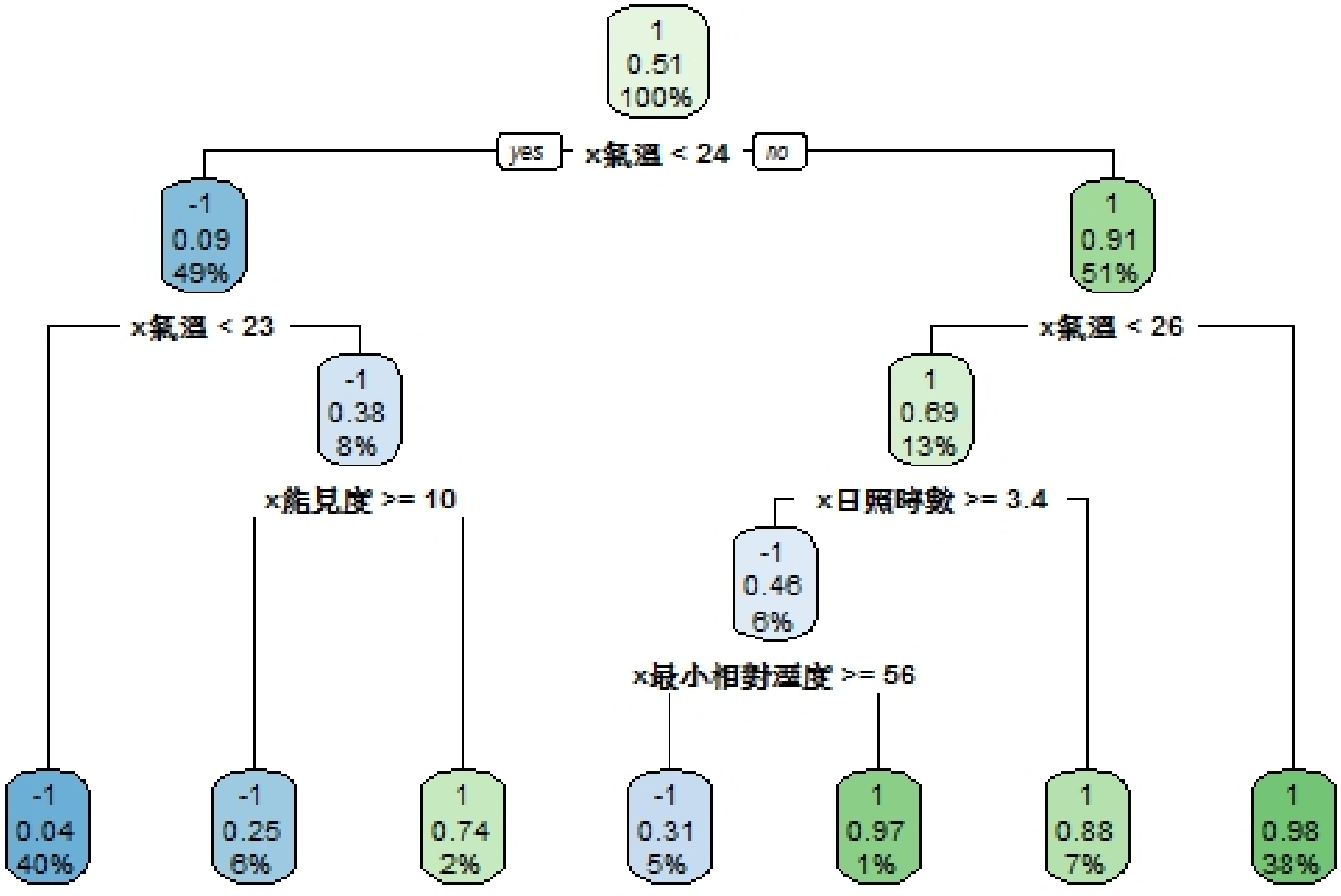
SLR、PCA

```
Coefficients:
                          (Intercept)  as.matrix(x[, c(1, 2, 10)])測站氣壓
                            -0.132638                          -0.004499
    as.matrix(x[, c(1, 2, 10)])氣溫  as.matrix(x[, c(1, 2, 10)])日照時數
                             0.206777                          -0.038344
```



(續)weak classify function

rpart 中的 CART 所做出來的決策樹



Adaboost

對應 classify function	adaboost 給予的權重
h1	0.977484768
h2	2.043227987
h3	0.518198453
h4	-0.017431753
h5	-0.020348786
h6	-0.171191291
h7	0.620454337
h8	-0.088147612
h9	-0.008179166
h10	-0.512029051

註解頁

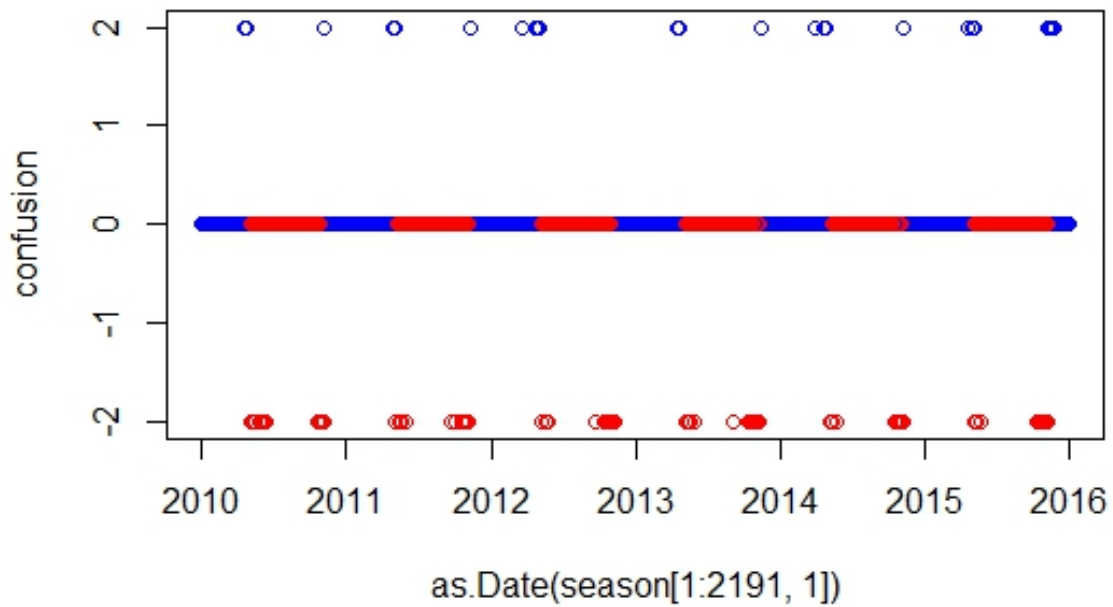
接下來我們利用 Adaboost 來找適合這 10 個弱分群函數的權重，所以每一個弱分群函數都會有屬於他們自己的權重，因此我們可以形成一個決策函數以便來區分冬春跟夏秋。

Testing

拿模型做回測

函數 \ 實際	冬春	夏秋
冬春	0.9628	0.1228
夏秋	0.0372	0.8772

函數 \ 實際	冬春	夏秋
冬春	1035	137
夏秋	40	979

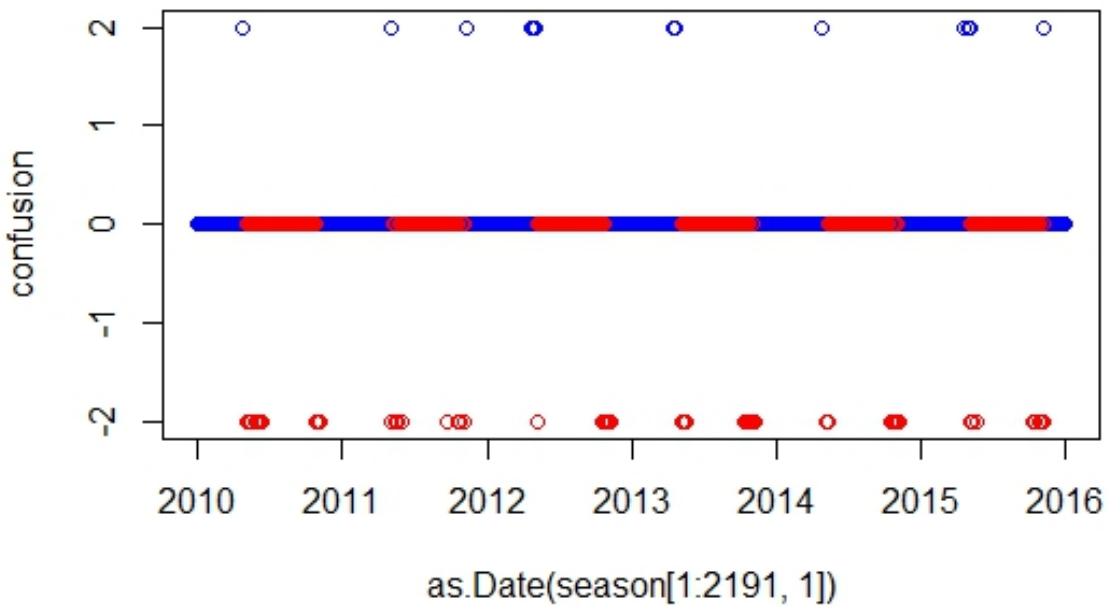


Testing: 經過平滑後

拿模型做回測

函數 \ 實際	冬春	夏秋
冬春	0.974	0.129
夏秋	0.026	0.871

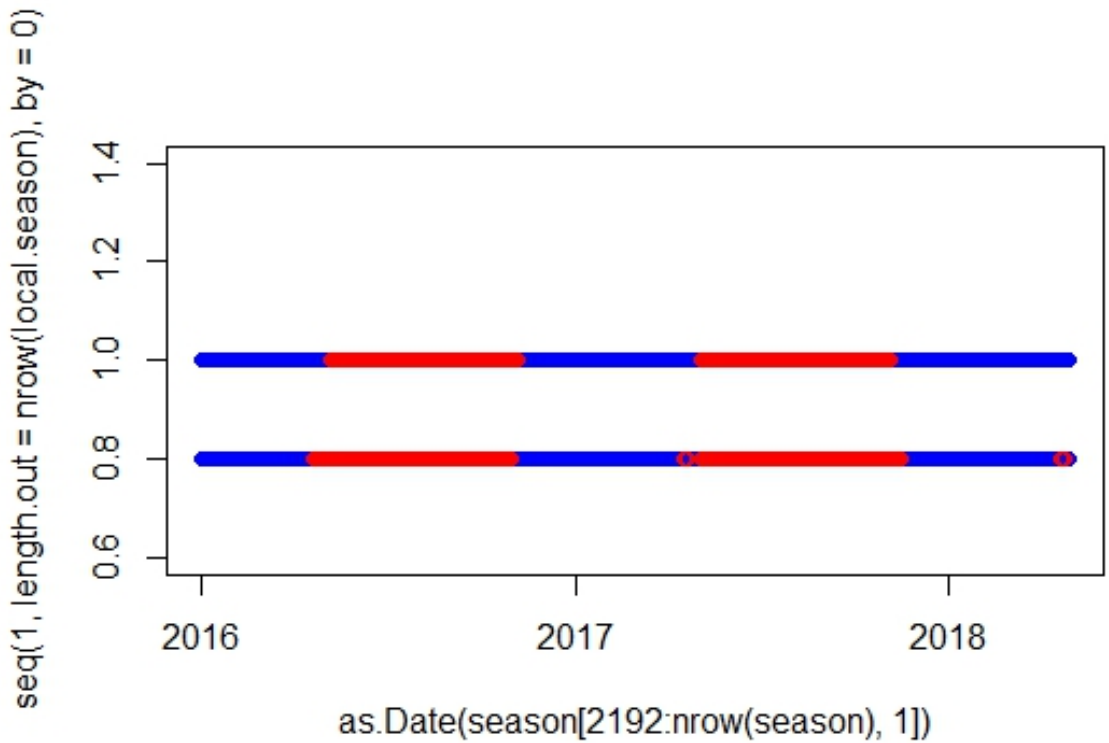
函數 \ 實際	冬春	夏秋
冬春	1047	144
夏秋	28	972



註解頁

拿著做好的決策函數回過頭去測試 2010 到 2015，發現到"冬春"以及"夏秋"交界處很容易分錯，且有些點不連續，因此我們使用一些小函數處理一些不連續的點。觀察 adaboost 的結果，發現它在交界處並沒有較明顯的特徵 (如值突然貼近 0 或非常遠離 0)，因此我針對每一個該被分成夏天的點，如果這個點的左邊以及右邊都是冬天，那麼就變成夏天吧；針對該被分成冬天的點也一樣。當然這樣沒辦法讓不連續完全消失，但可以讓整個圖變得更好讀。

檢測 2016 到 2018

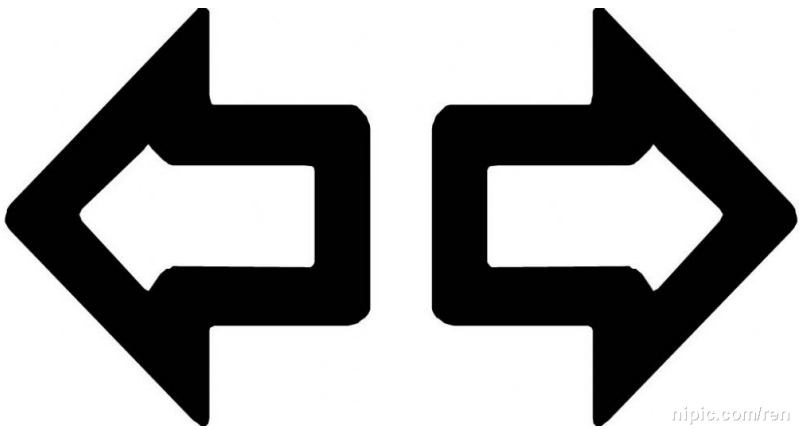


註解頁

我們拿 2010 到 2015 的資料為基準，觀察 2016 到 2018 的天氣型態，進而定論是否需要進行曆法調整，資料顯示曆法以及我們的決策函數所訂出來的"冬春"及"夏秋"相去不遠，決策函數的"夏秋"時間較長，但也長不過 7 天。

下一步

1. 取得過去資料
2. 等待未來資訊



註解頁

或許是因為訓練決策函數的資料點以及我們觀察的年份相去不遠，所以天氣型態沒有太大的改變。若我們能取得更久以前的資料，用那些資料當基準來觀察現在的天氣型態或許能發現很大的差別。又或是將我們的決策函數留下，觀察未來天氣型態是如何變化。

或者我們可以改變研究方式，因為我們想研究的是長期的氣候型態而非單純的天氣，可以使用 moving average、分時間區塊 (而非單純每日) 來進行分析。

QA

謝謝聆聽