THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Class 1 – Lesson 1

介紹

Ian H. Witten

Department of Computer Science
University of Waikato
New Zealand

weka.waikato.ac.nz

使用Weka進行更深入的資料探勘

...這是一門實用課程，講解如何使用Weka的高級功能完成資料探勘 (不需要學習編程，只學習使用Weka的交互介面)

... 延伸自*Data Mining with Weka的前期課程*

...學習一些資料探勘的基本理論

Ian H. Witten

University of Waikato, New Zealand

使用Weka進行更深入的資料探勘

- **此課程假定你已知：**
  - *何謂資料探勘以及它為何有用？*
  - *「簡單第一」的規範*
  - *安裝Weka而且使用 Explorer介面*
  - *一些受歡迎的分類器演算法以及過濾法*
  - *使用Weka 中的分類器和過濾器且知道如何能夠得到更多關於他們的資訊*
  - *結果評估，避免訓練/測試中的陷阱*
  - *解釋Weka的輸出和視覺化你的資料集*
  - *完整的資料挖掘過程*
- **以上可藉由觀看*Data Mining with Weka*課程複習**
- **(複習: 在YouTube WekaMOOC頻道觀看)**

使用Weka進行更深入的資料探勘

- 如你所知, Weka是
  - 新西蘭特有的鳥？
  - 資料探勘工作平台:
    Waikato Environment for Knowledge Analysis

在資料探勘上使用的機器學習算法

- 超過一百種用來分類的演算法
- 75種用來資料預處理的演算法
- 25種用來協助特徵選擇的演算法
- 20種用來聚類、找到關聯規則...等等的演算法

使用Weka進行更深入的資料探勘

你將學習到：

- *Experimenter, Knowledge Flow interface, Command Line interfaces*
- *處理大數據(big data)*
- *文本探勘(Text mining)*
- *監督式和非監督式過濾器(Supervised and unsupervised filters)*
- *離散(discretization)和取樣(sampling)*
- *屬性選擇方法*
- *屬性選擇和過濾專用的元分類器(Meta-classifiers)*
- *規則和樹的區別：rules vs. trees(產生規則)*
- *關聯規則(Association rules)以及聚類(clustering)*
- *成本敏感評估和分類*

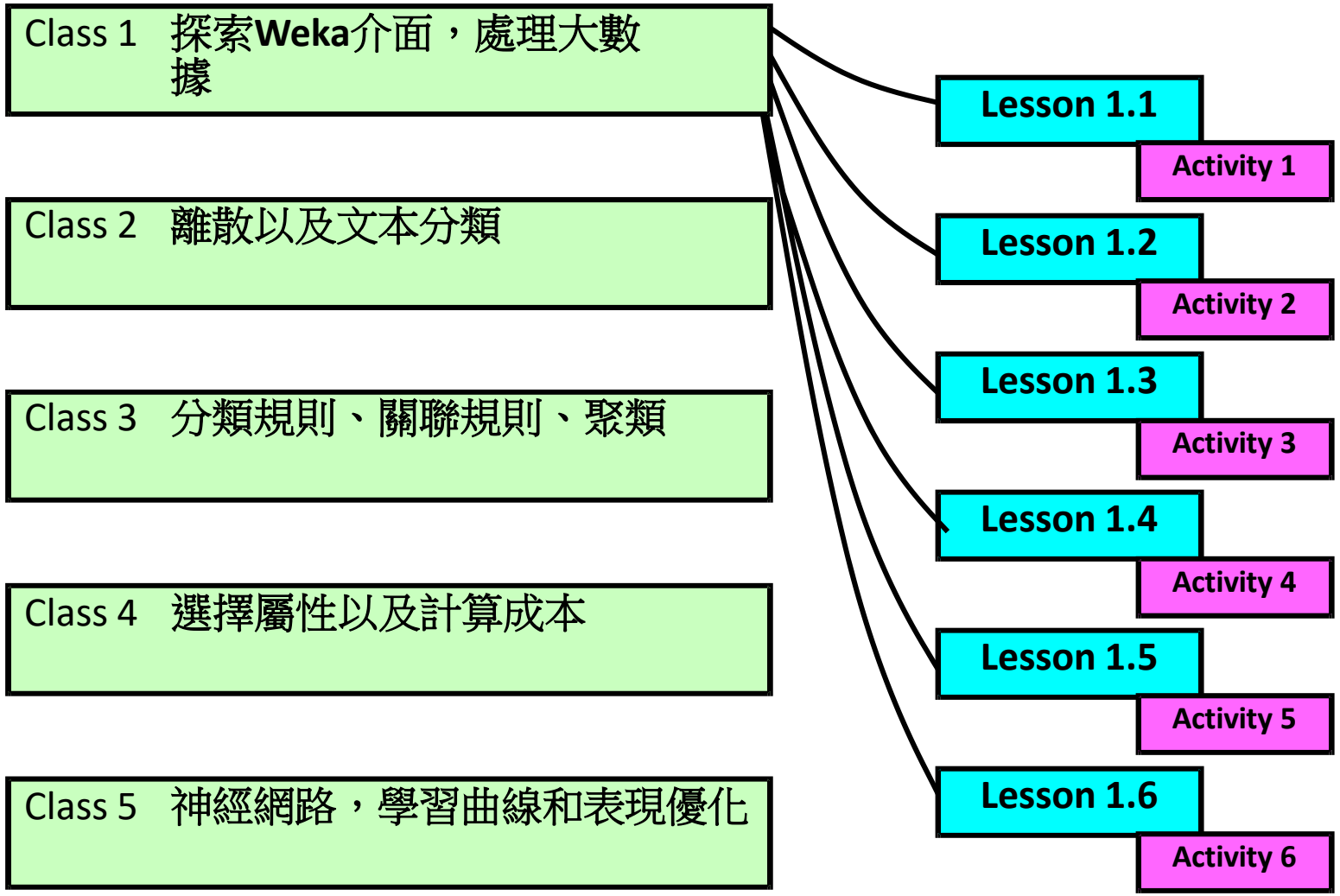
在你自己的資料上使用… 而且了解你在做什麼！

Class 1: 探索Weka介面，處理大數據

- Experimenter 介面
- 使用Experimenter來比較分類器
- Knowledge Flow介面
- Simple Command Line介面
- 處理大數據
  - *Explorer: 1百萬個實例, 25種屬性*
  - *Command line 介面：effectively unlimited*
  - *在活動中你將處理多筆百萬筆實例的資料集*

Course organization

這門課和上門課的結構完全相同，共5部分。每部分包含6節課。
Class1有六節課，每節課都有一個短小的視頻，且有課後練習。



Download Weka now!

下載自：

http://www.cs.waikato.ac.nz/ml/weka

針對Windows, Mac, Linux作業系統

Weka 3.6.11

Weka最新的穩定版本

包含本課程的資料集

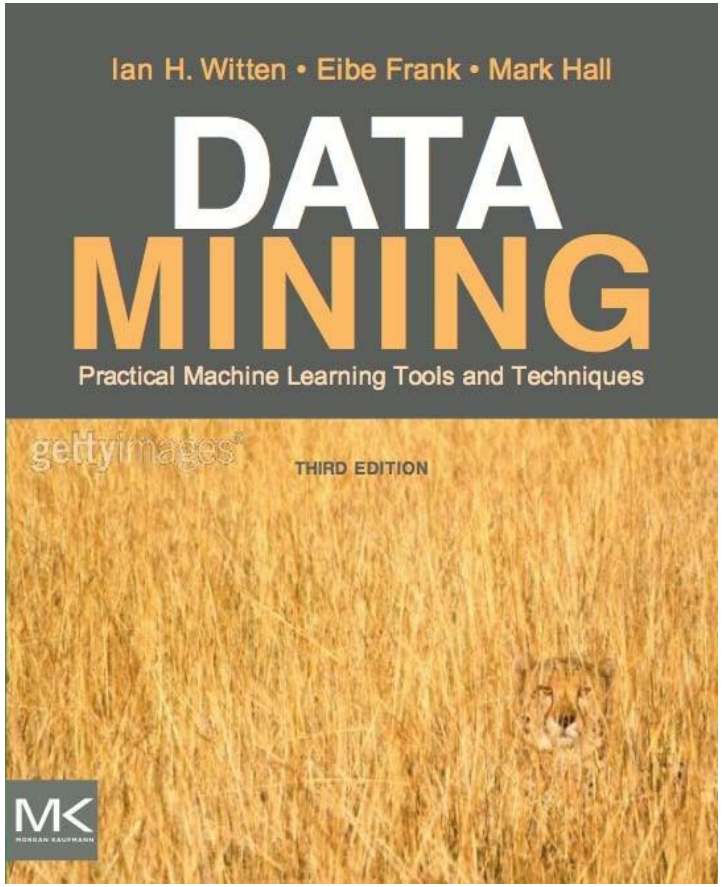
取得正確的版本很重要！

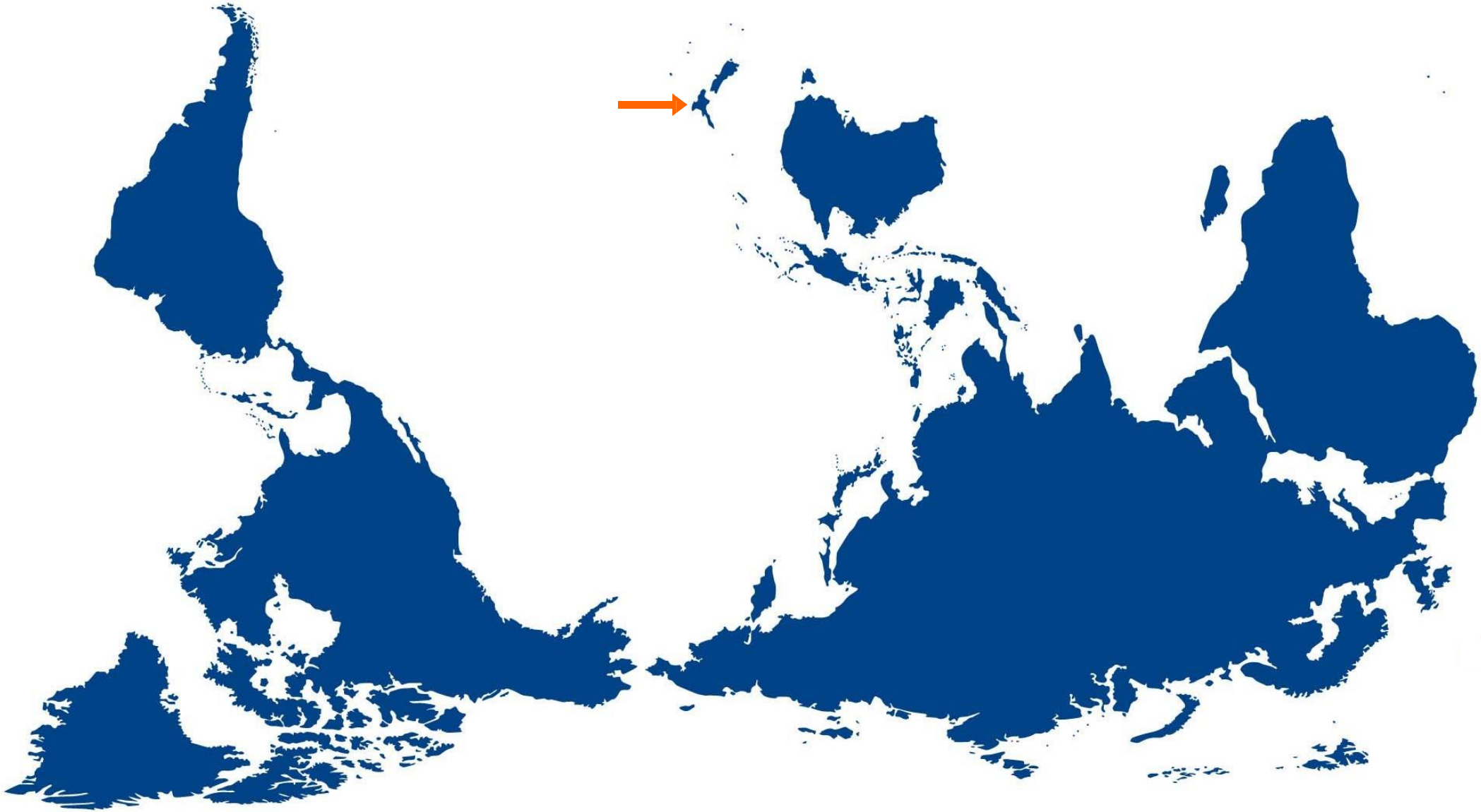
Textbook

這本參考書深入地探討資料探勘和Weka：

Data Mining: Practical machine learning tools and techniques,
by Ian H. Witten, Eibe Frank and Mark A. Hall. Morgan Kaufmann, 2011

出版商製作了關於這門課的電子書，可在線上取得。





THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Class 1 – Lesson 2

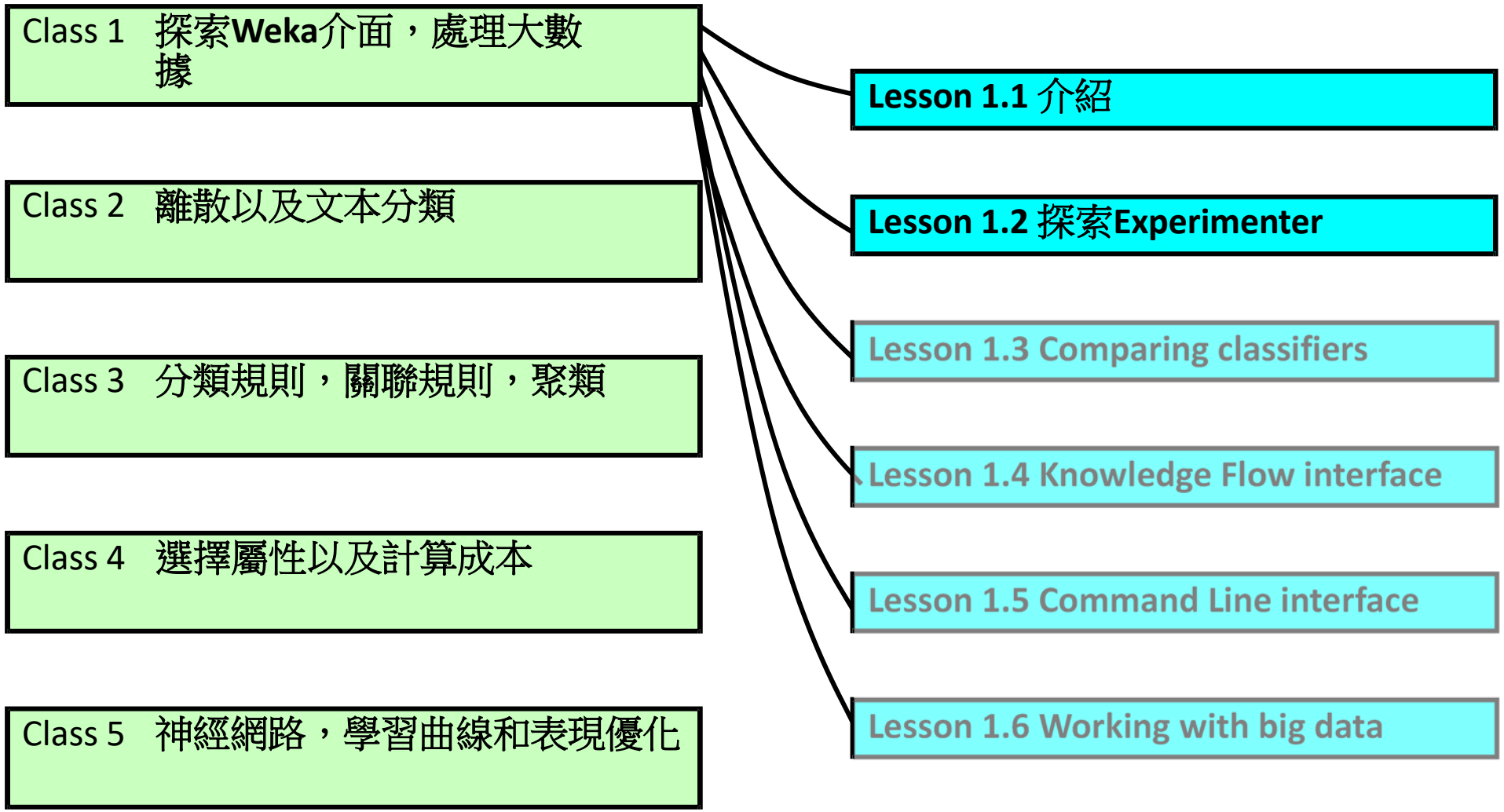
探索Experimenter

Ian H. Witten

Department of Computer Science
University of Waikato
New Zealand

weka.waikato.ac.nz

Lesson 1.2: 探索Experimenter



Lesson 1.2: 探索Experimenter

試用分類器/過濾器

性能的比較

工作臺

圖形化介面

指令行界面

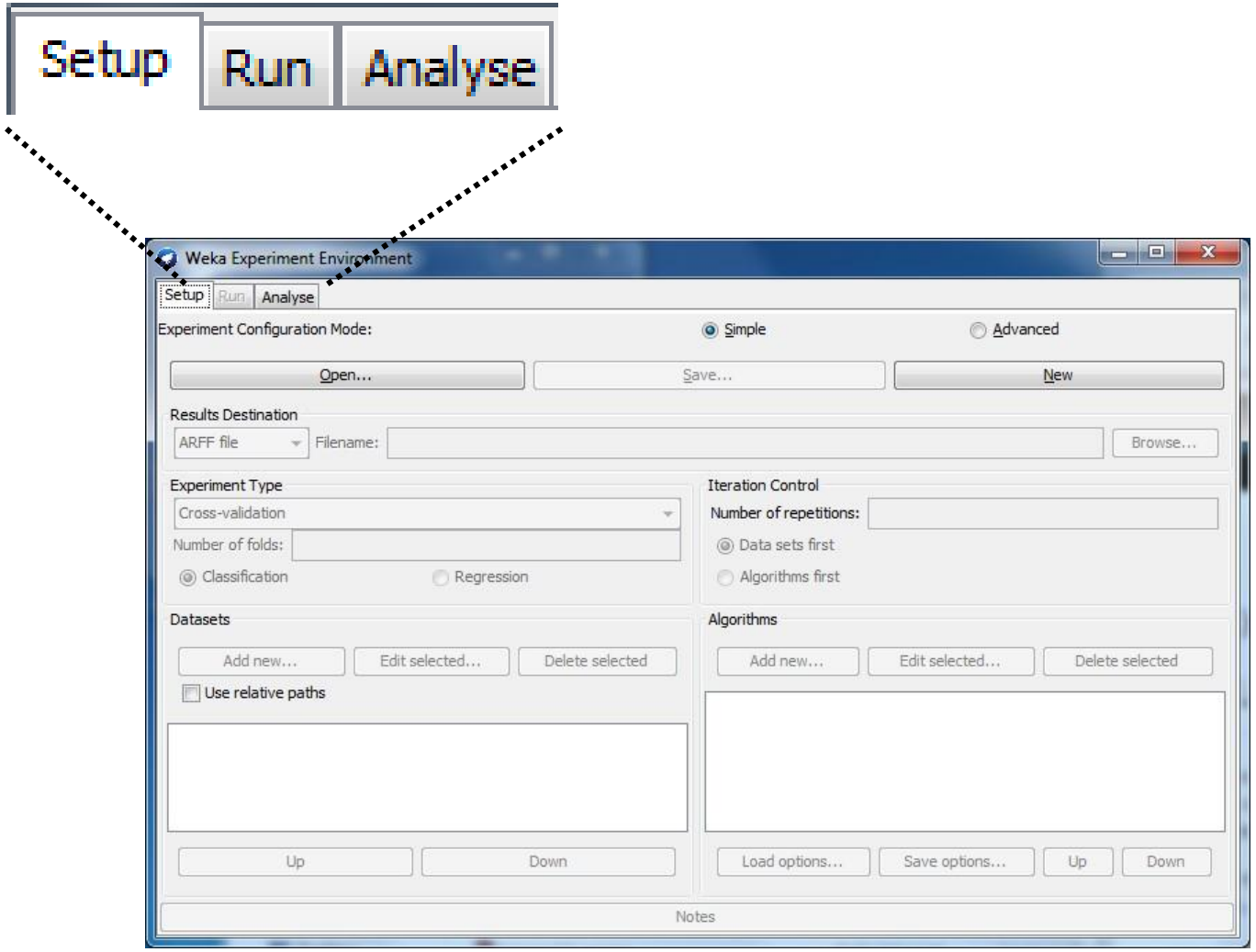
Lesson 1.2: 探索Experimenter

使用Experimenter來...

- 計算分類算法對於某一資料集的平均數和標準差
  ...或是多個資料集設置多個算法
- 找出對於特定的資料集，某個分類器是否就優於另一種分類器
  ...以及它們的差別是否在統計意義上顯著不同
- 檢驗同一算法的不同參數的效果
- 通過ARFF文檔顯示測試結果
- 有時，Experimenter可能運算幾天甚至幾周的時間
  ...它能使用多台機器同時運算

Lesson 1.2: 探索Experimenter

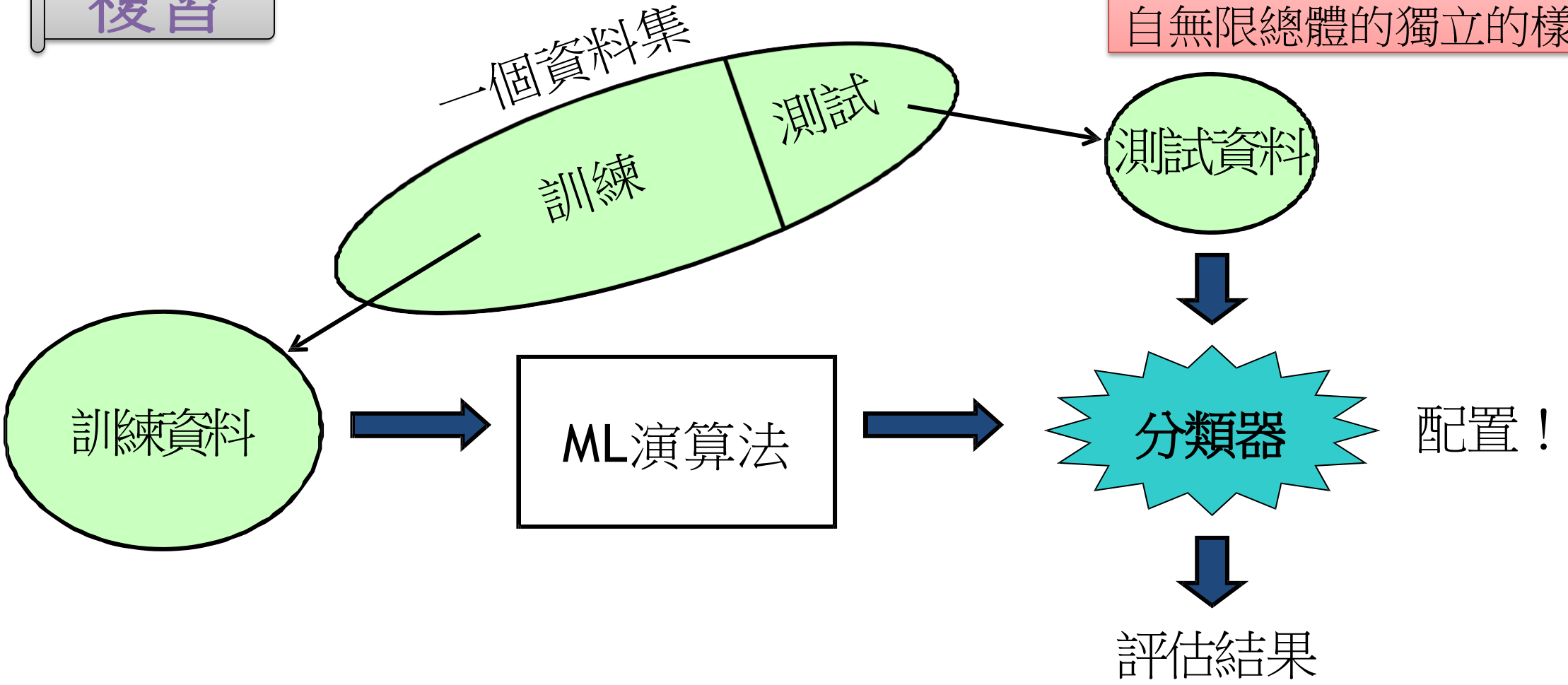
Experimenter包含三個面板：Setup面板、Run面板和Analyse面板。



Lesson 1.2: 探索Experimenter

這是課程Data Mining with Weka第2.3節中的投影片。機器學習一個基本的假設是這些數據是取自無限總體的獨立的樣本數集。

複習



基本準則：訓練和測試資料都是分別取自無限總體的獨立樣本。

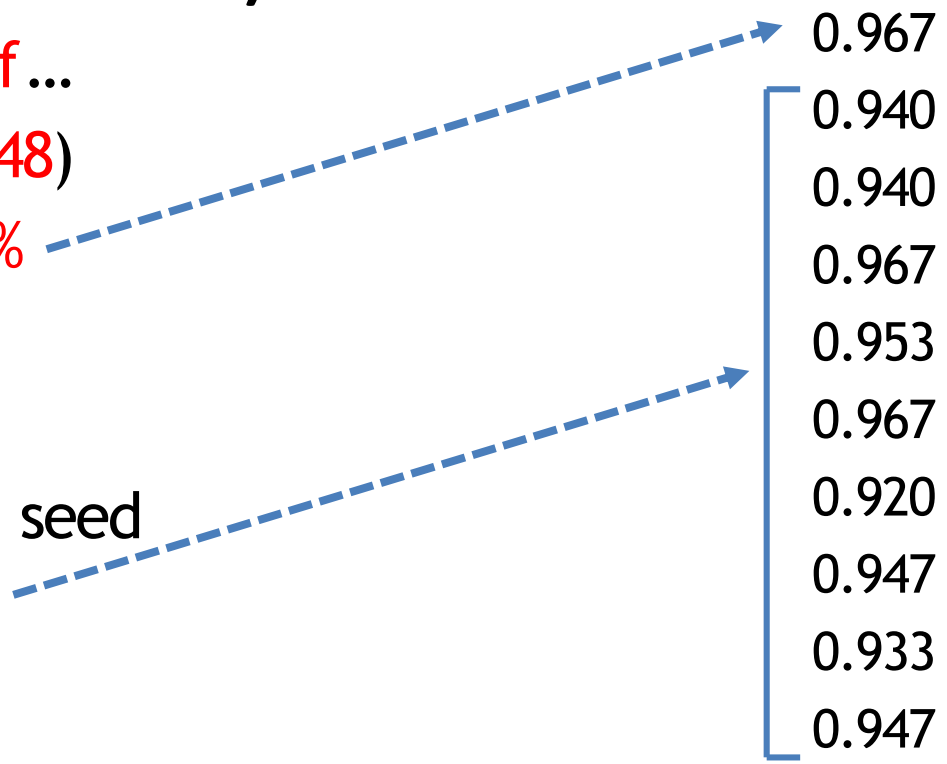
Lesson 1.2: 探索Experimenter

複習

在上門課2.3節中提到我們選擇一個名為segment-challenge的資料集、選擇算法J48以及使用比例分割方法來評估。通過評估，得到一個準確率。然後，我們使用不同的隨機種子來重覆這一步驟，最終得到10個不同的準確率。

評估J48在segment-challenge上的表現 (*Data Mining with Weka,* Lesson 2.3)

- 使用 segment-challenge.arff ...
- 使用J48 (trees資料夾下的 J48)
- 設定 percentage split 為 90%
- 執行: 96.7% 準確率
- 重複執行Repeat
- [More options] Repeat with seed 2, 3, 4, 5, 6, 7, 8, 9 ,10



Lesson 1.2: 探索Experimenter

最後，我們計算樣本平均數，方差和標準差。

複習

評估J48在segment-challenge上的表現 (*Data Mining with Weka,* Lesson 2.3)

樣本平均值 $\bar{x} = \frac{\sum x_i}{n}$

變異數 $\sigma^2 = \frac{\sum (x_i - \bar{x})^2}{n - 1}$

標準差 σ

0.967
0.940
0.940
0.967
0.953
0.967
0.920
0.947
0.933
0.947

$\bar{x} = 0.949$, $\sigma = 0.018$

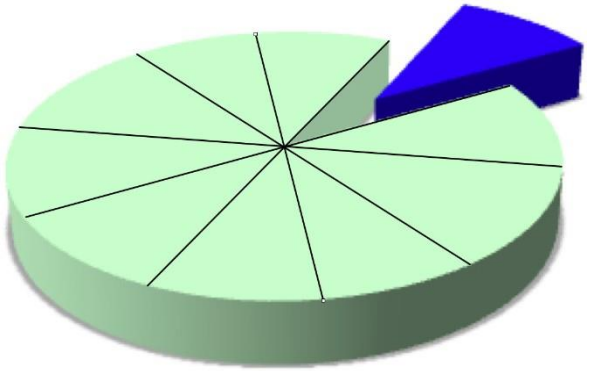
Lesson 1.2: 探索Experimenter

在Data Mining with Weka第2.5節中，我們學習了10折交叉驗證，也就是把資料集分成10組，每次保留一組測試數據，然後平均十次的結果。

複習

10層(folds)的交叉驗證(***Data Mining with Weka,*** **Lesson 2.5)**

- 只分割一次，但是將資料集分成10份(層)
- 輪流拿出每一份作為測試資料，其他的份作為訓練資料
  (每一份資料會有1次被拿來當測試資料，9次當訓練資料)
- 對結果取平均



分層的交叉驗證

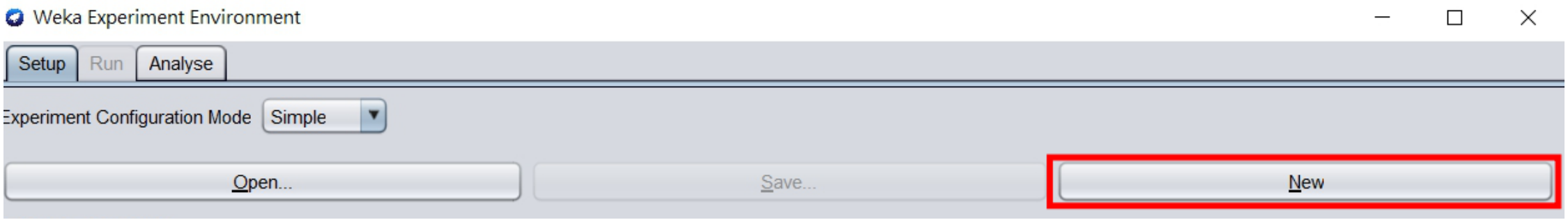
- 確保每一份的類值(class value)的比例大致相同

Lesson 1.2: 探索Experimenter

1. 開啟Weka程式，於Weka GUI Chooser介面左鍵單擊Experimenter按鈕

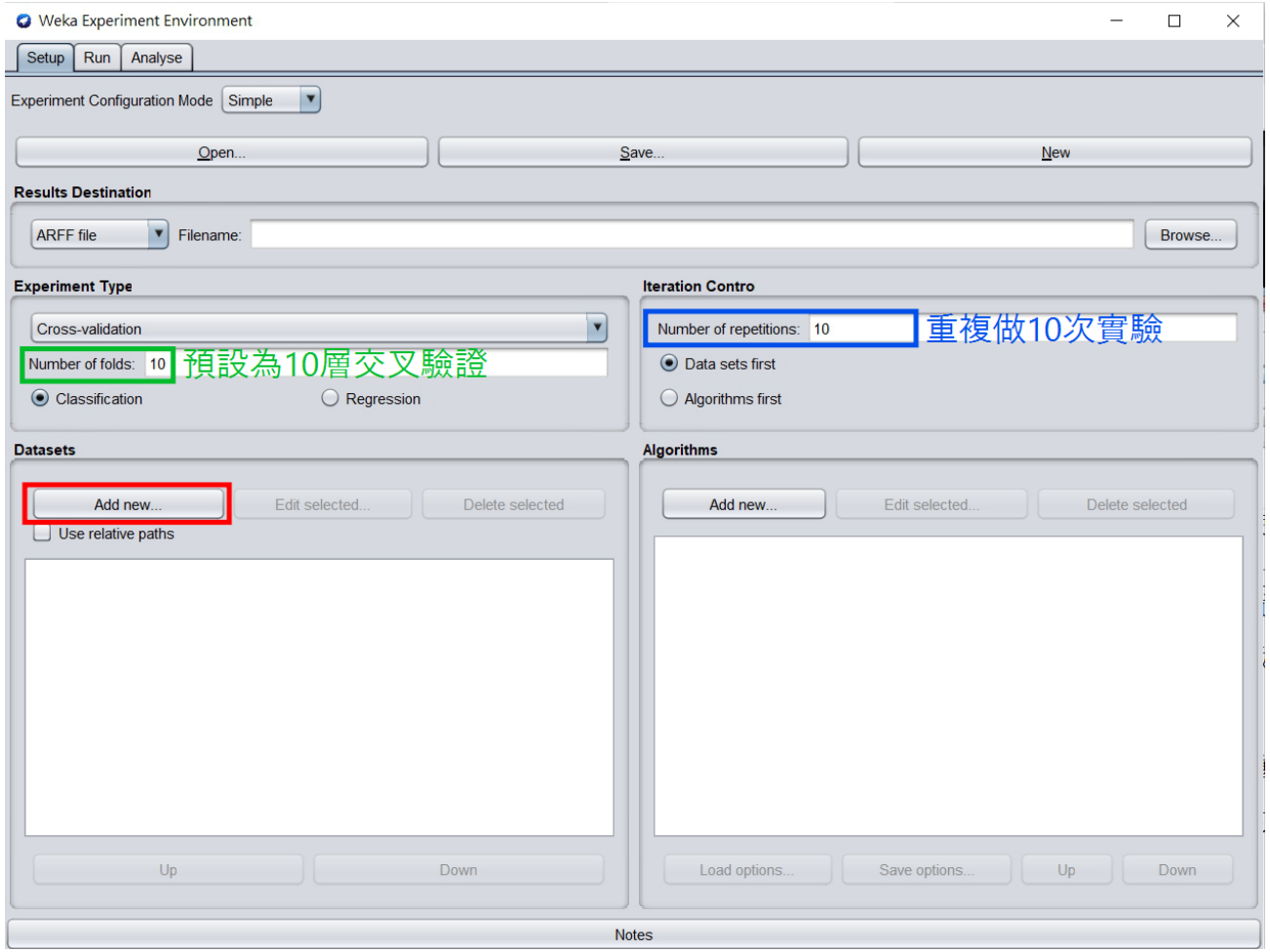
Lesson 1.2: 探索Experimenter

2. 在Setup面板，以左鍵單擊New鈕，新增一個實驗



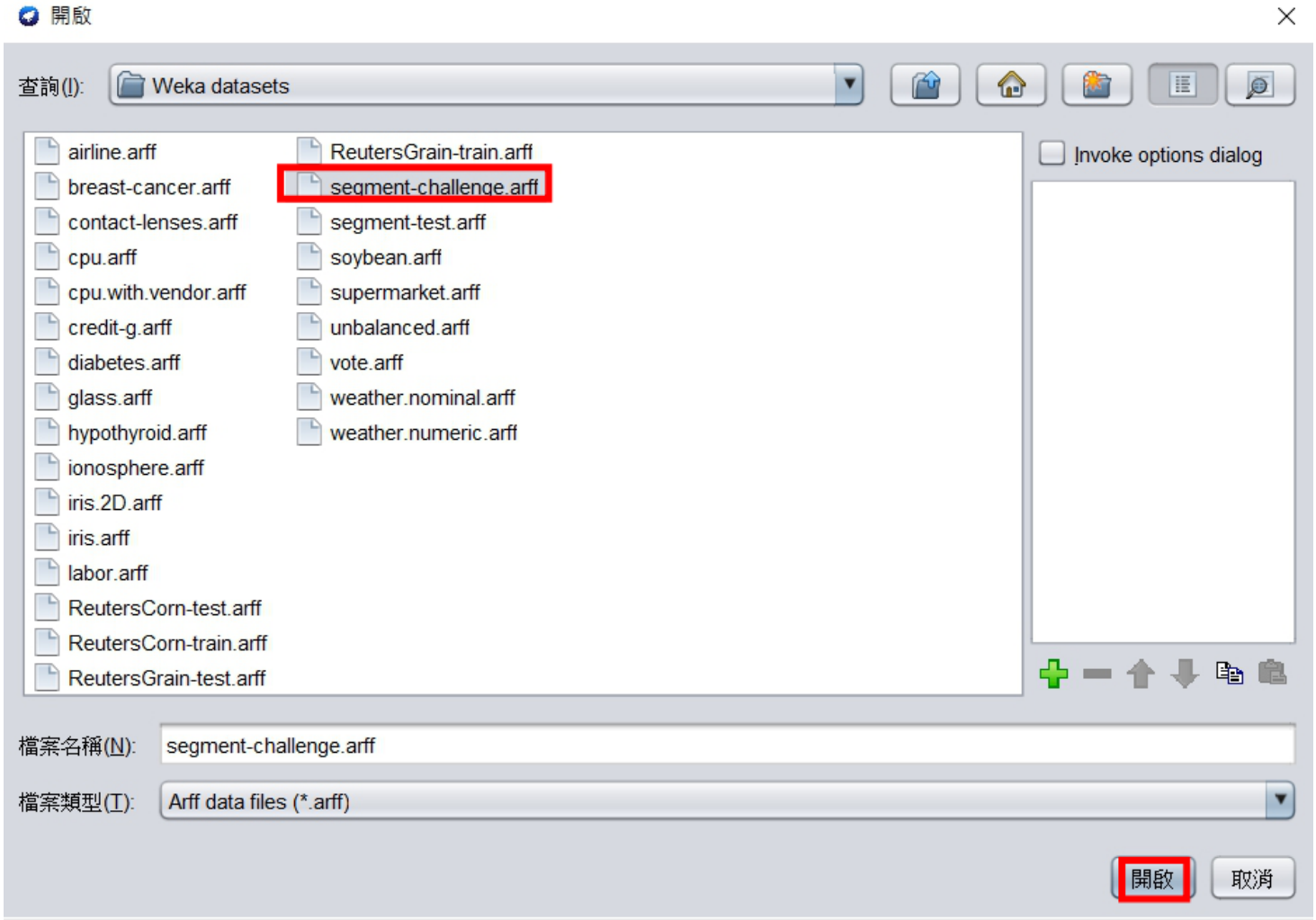
Lesson 1.2: 探索Experimenter

3. 左鍵單擊圖中紅框中的Add new...鈕，新增資料集



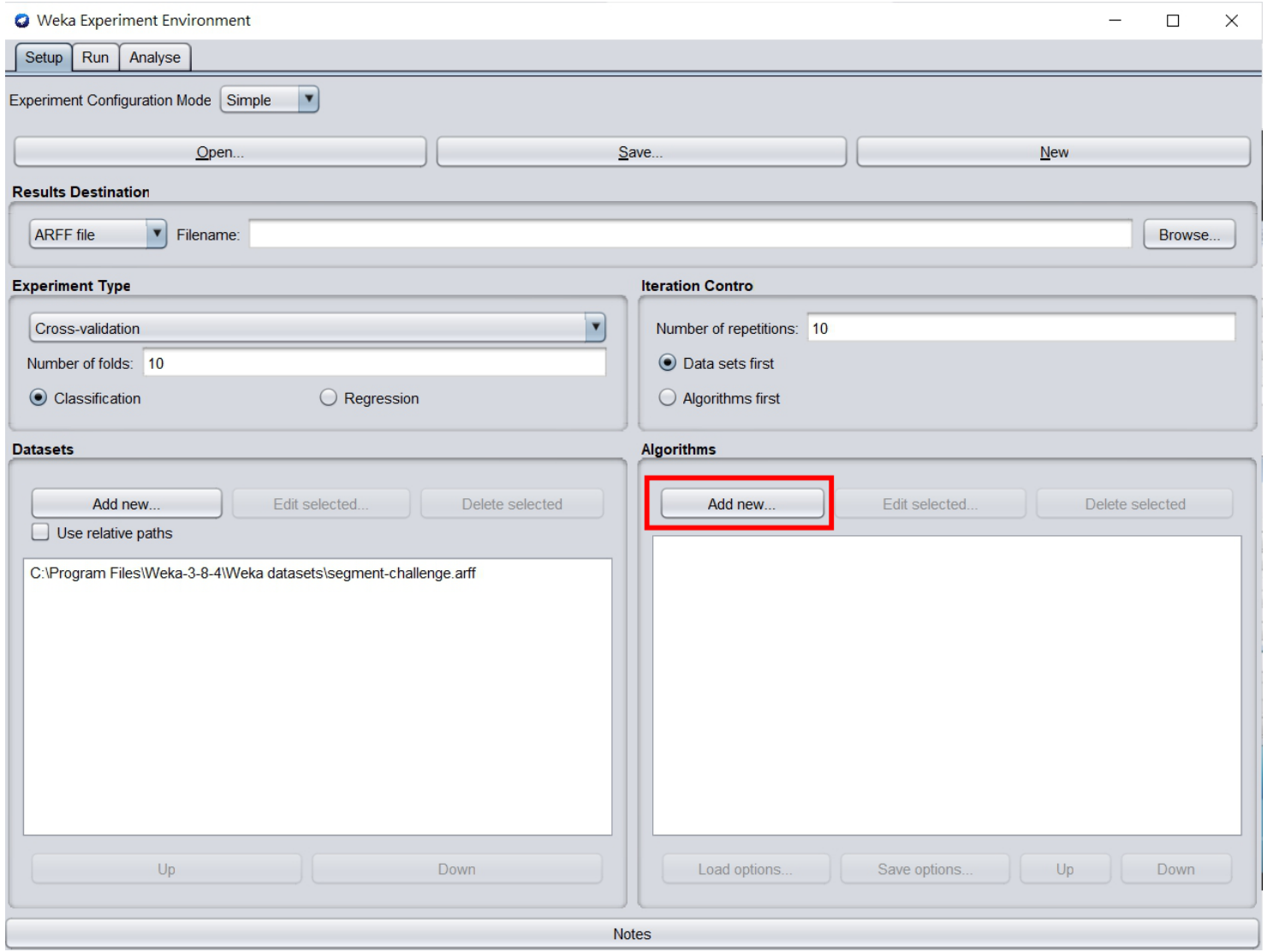
Lesson 1.2: 探索Experimenter

4.進入自行複製的Weka datasets資料夾，左鍵單擊segment-challenge.arff 的檔案後，再以左鍵單擊下方”開啟”以載入此資料集



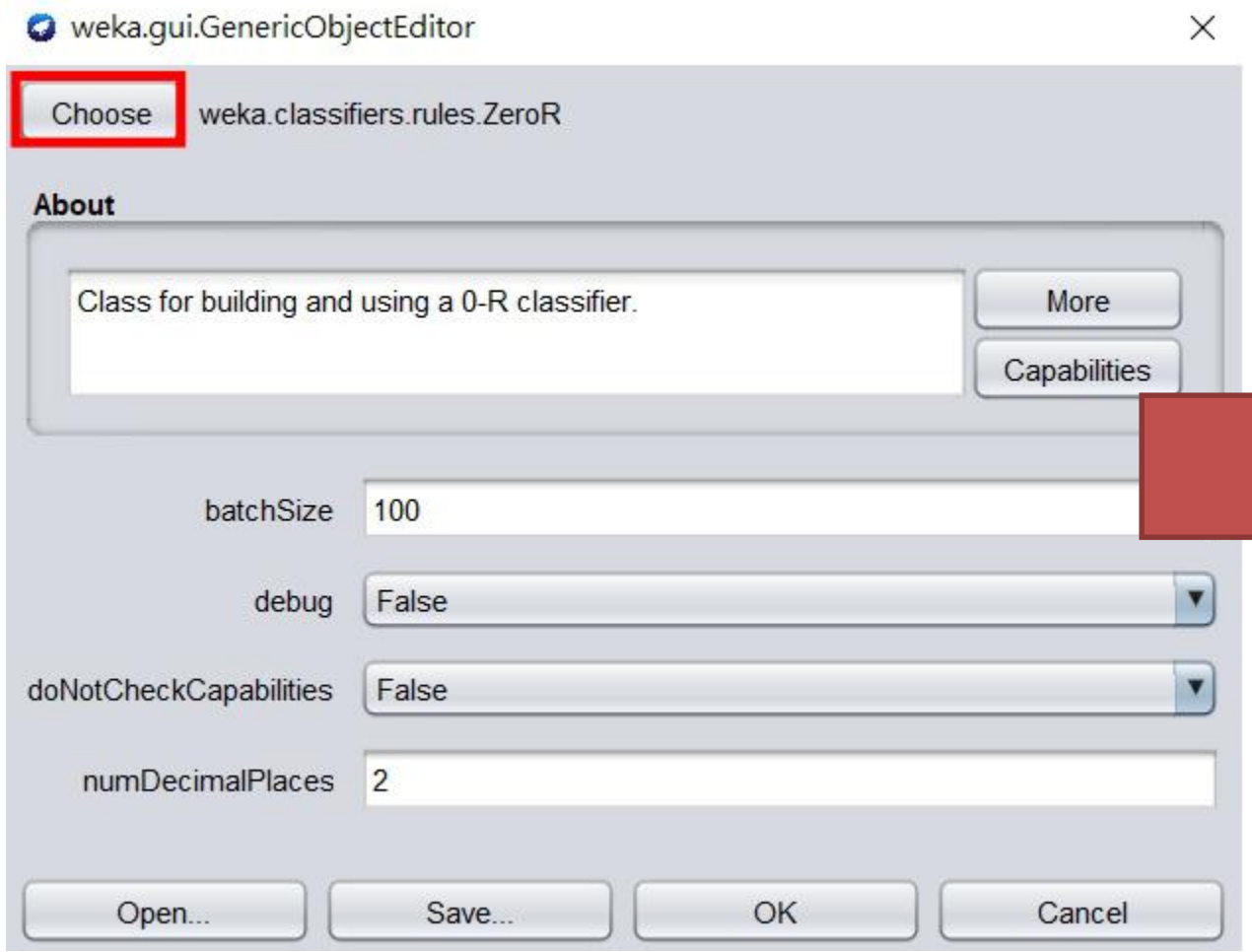
Lesson 1.2: 探索Experimenter

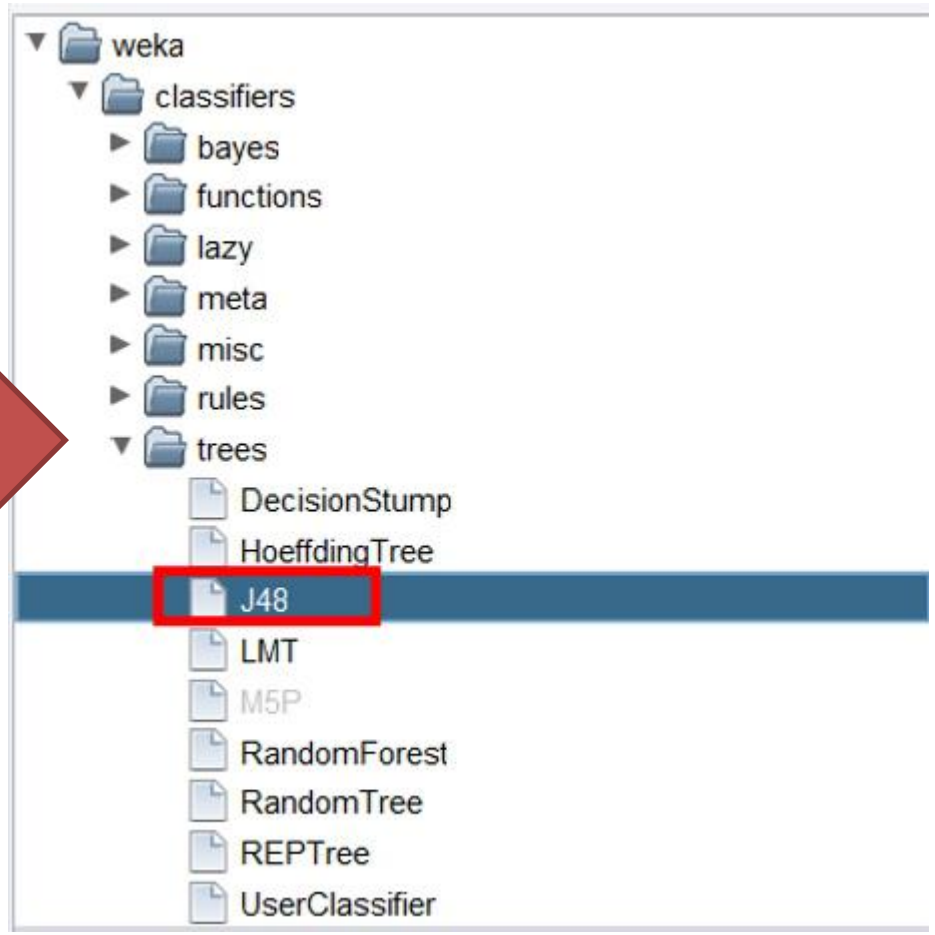
5. 左鍵單擊圖中紅框中的Add new...鈕，新增演算法



Lesson 1.2: 探索Experimenter

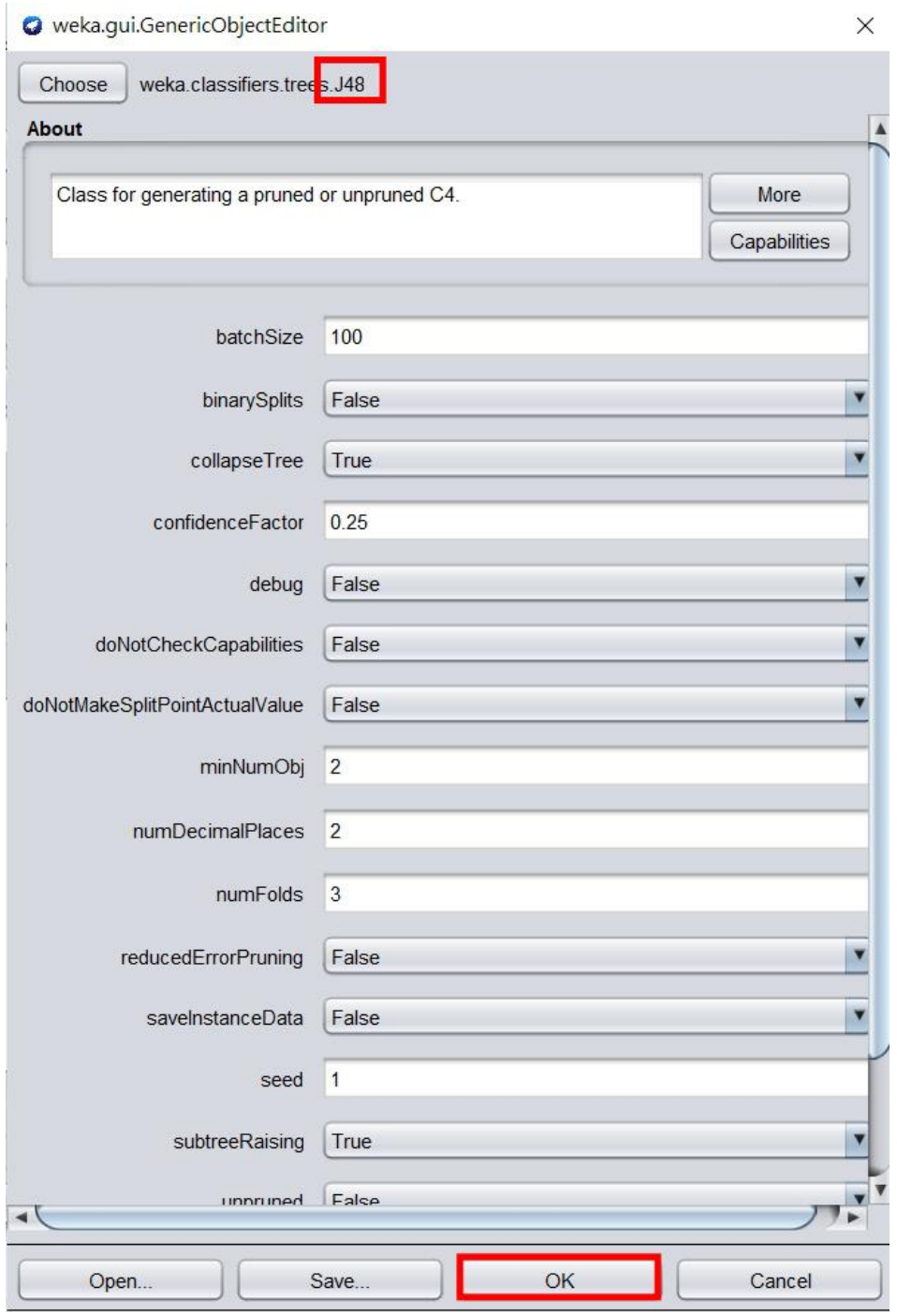
6. 在出現的視窗中左鍵單擊Choose鈕，選擇trees資料夾下的J48分類器





Lesson 1.2: 探索Experimenter

7. 確認已選擇J48分類器後，左鍵單擊下方的OK鈕

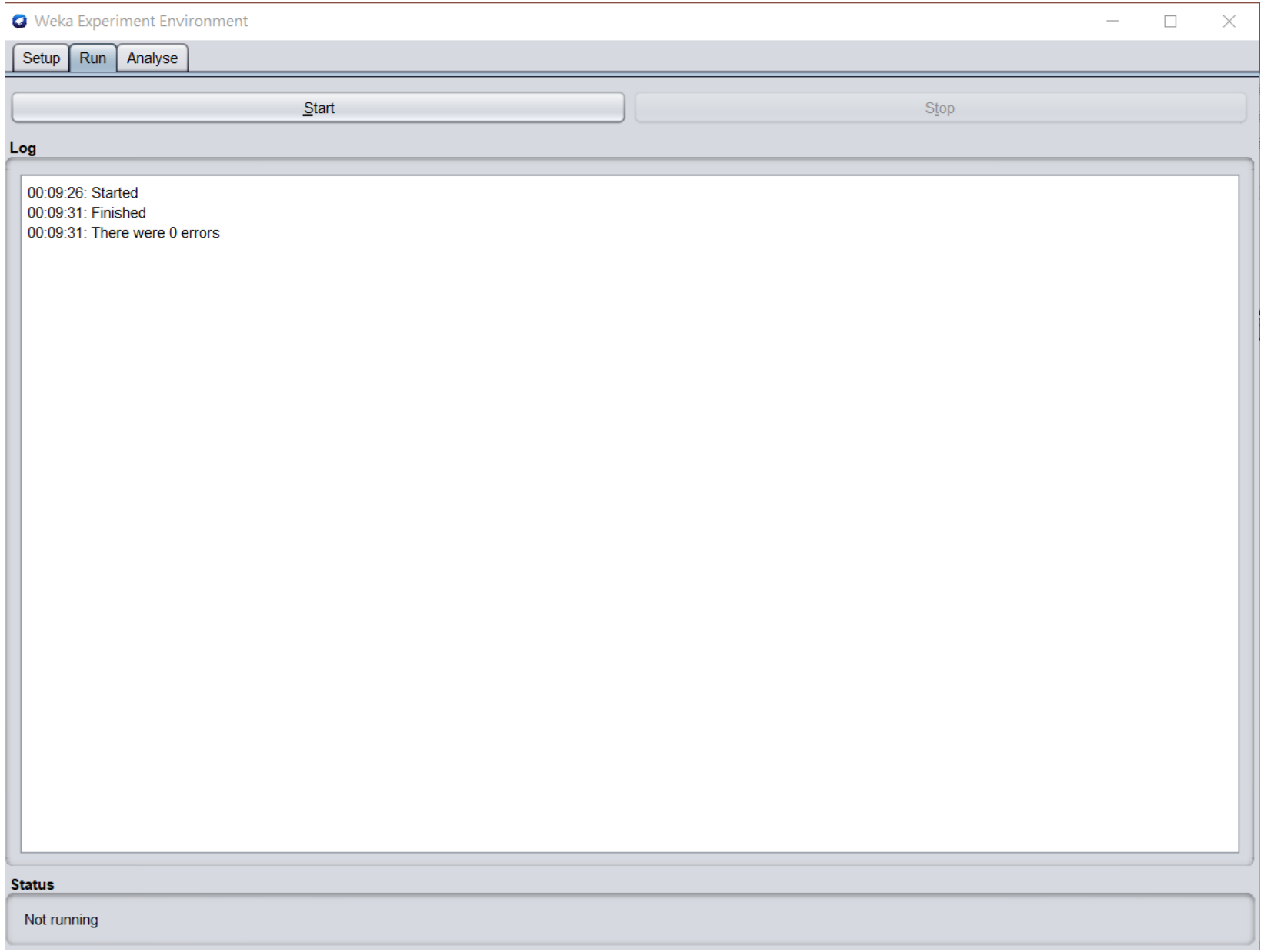


Lesson 1.2: 探索Experimenter

8.切換到Run的面板，左鍵單擊Start鈕

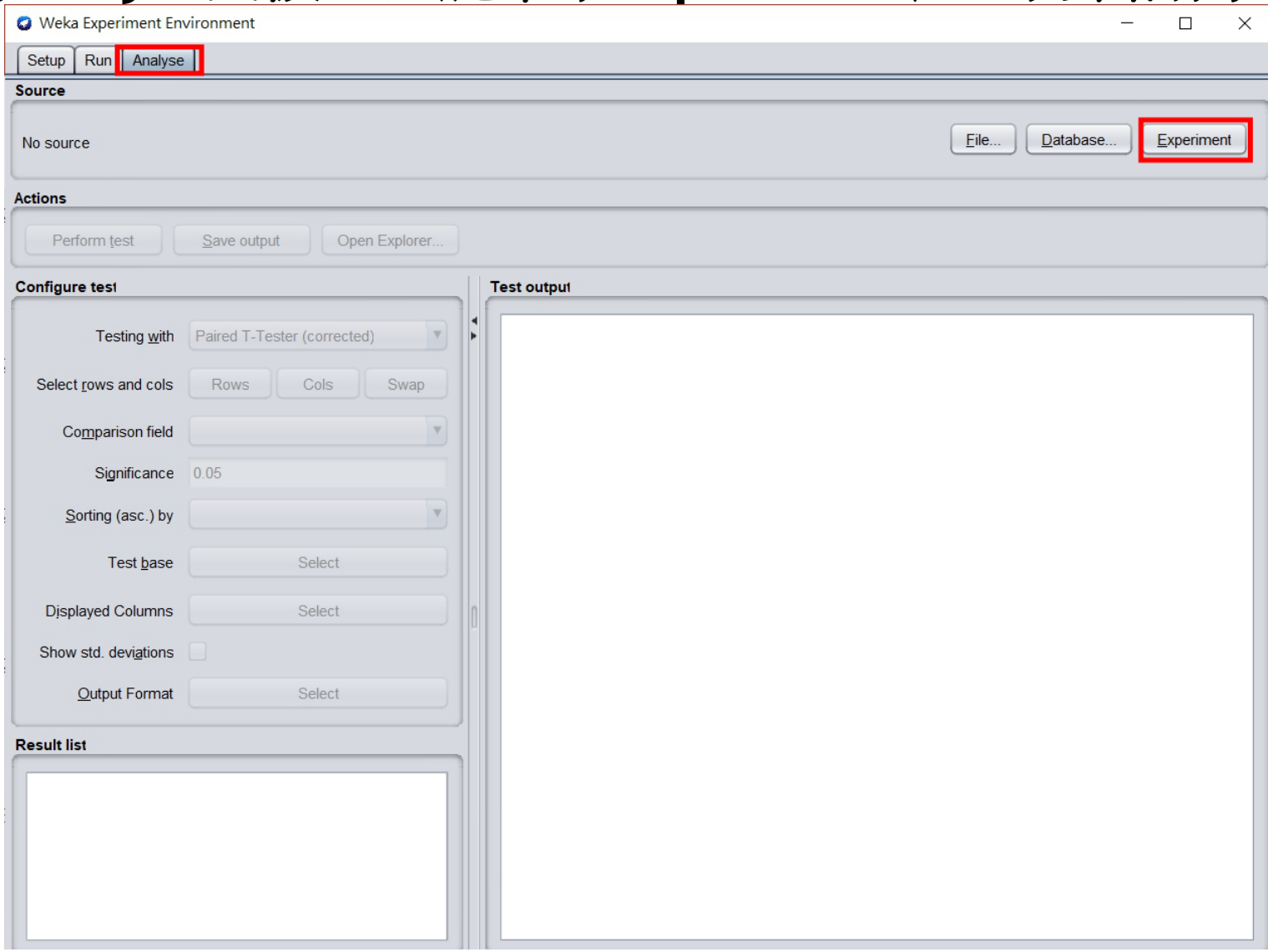
Lesson 1.2: 探索Experimenter

▼結果展示



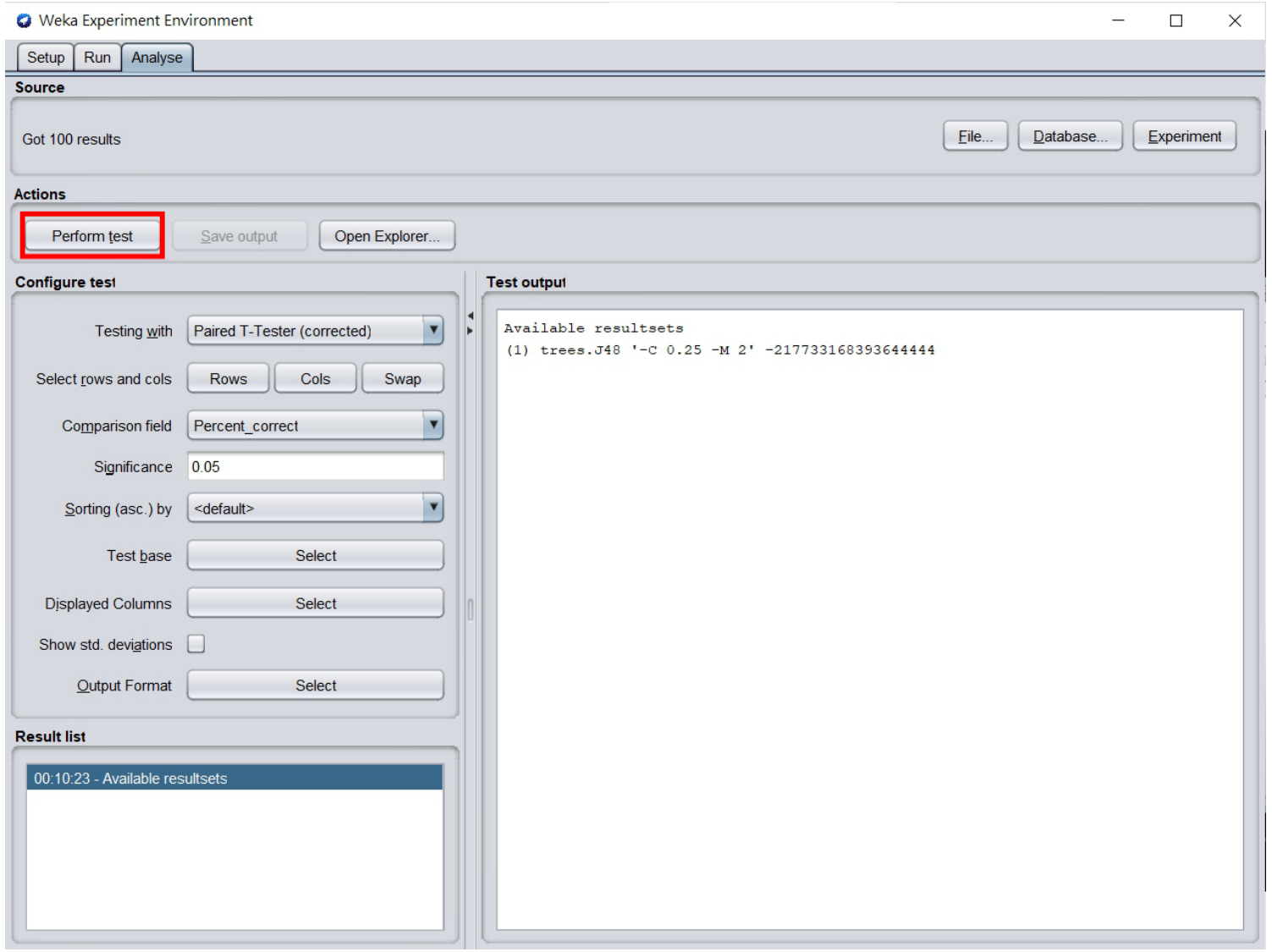
Lesson 1.2: 探索Experimenter

9.切換到Analyse面板，左鍵單擊Experiment鈕，分析剛才的實驗結果



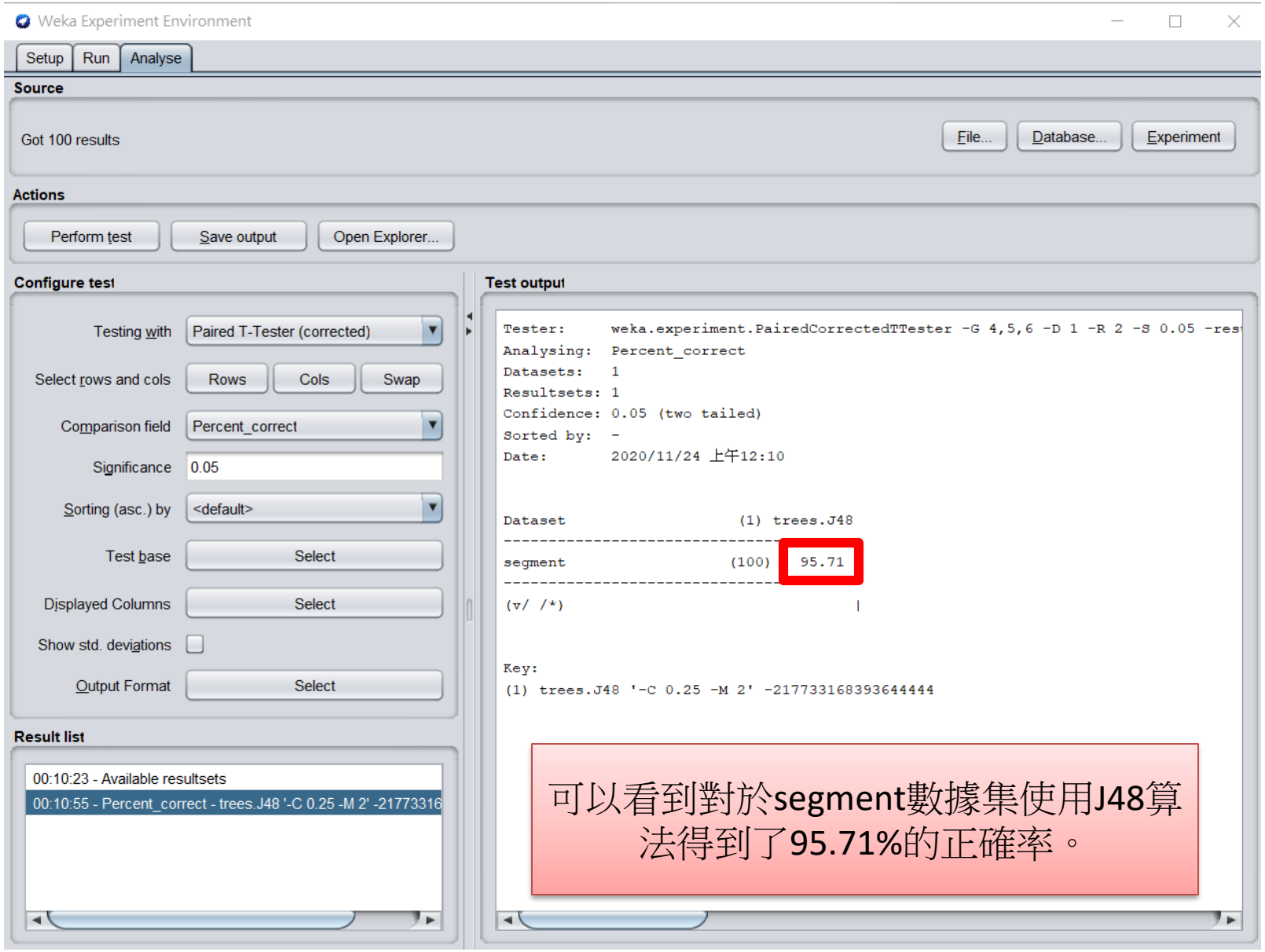
Lesson 1.2: 探索Experimenter

10.左鍵單擊Perform test



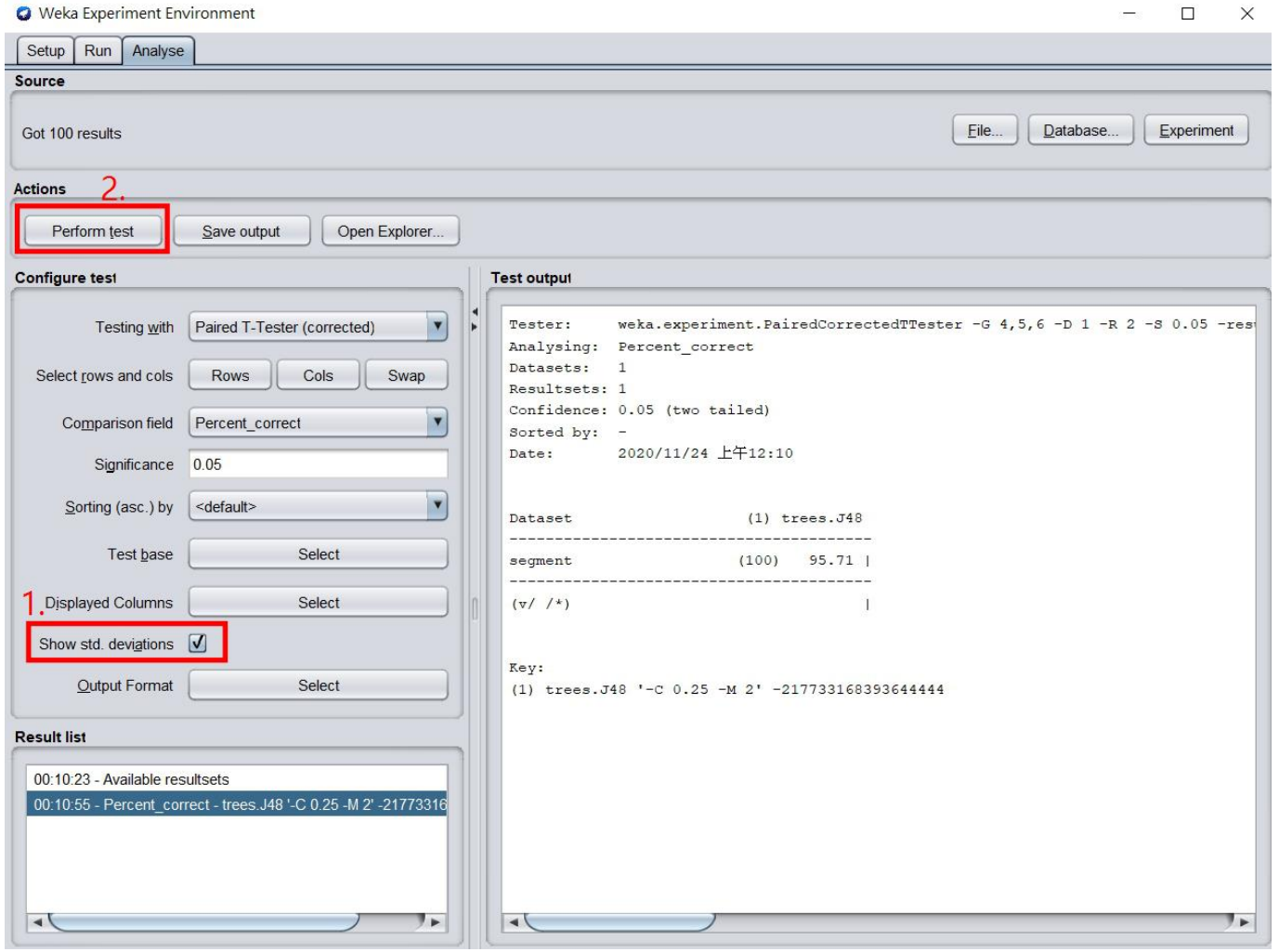
Lesson 1.2: 探索Experimenter

▼結果展示



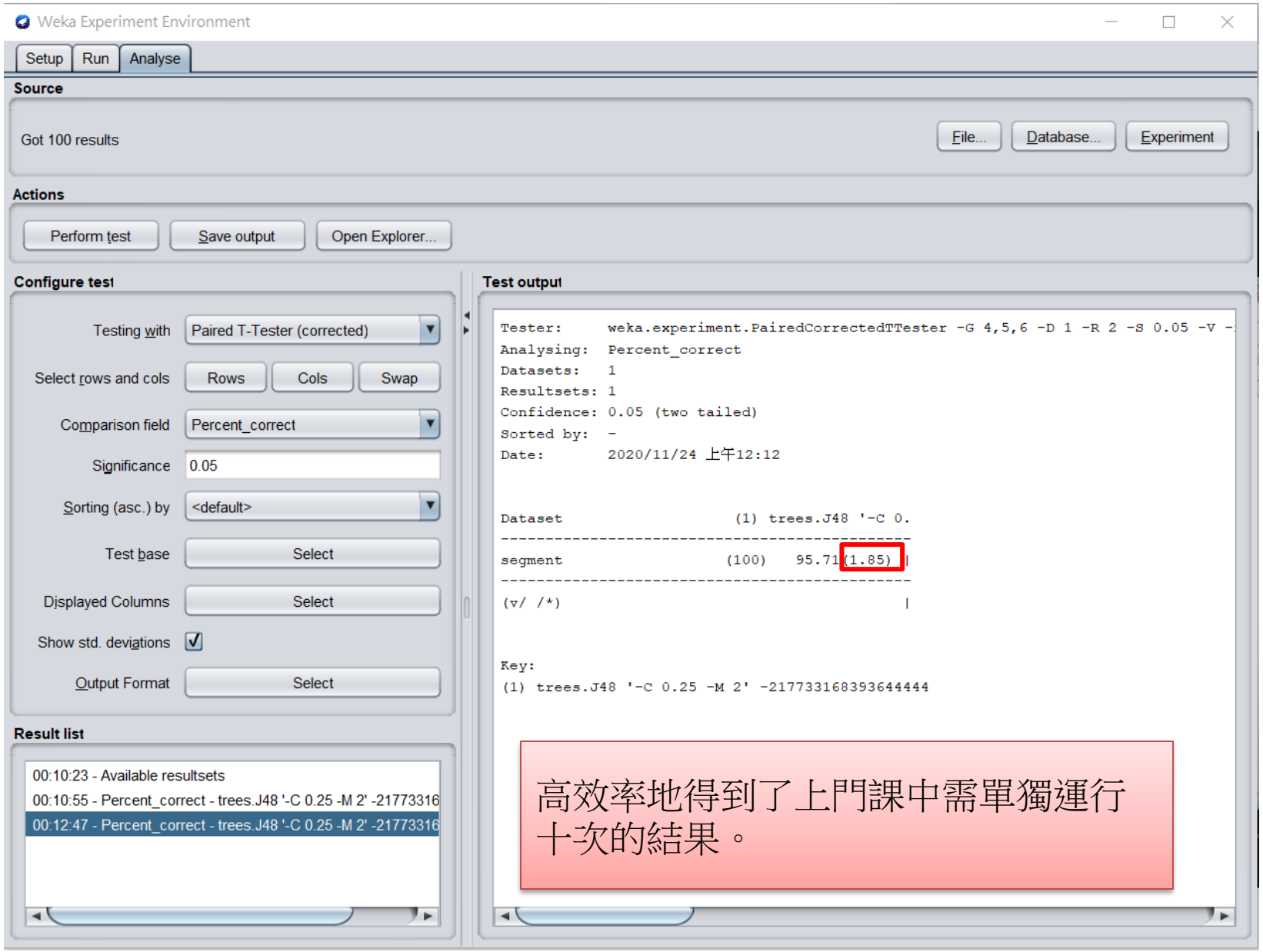
Lesson 1.2: 探索Experimenter

11.再來查看標準差。勾選Show std. deviations，並以左鍵單擊Perform test鈕再次運行測試。

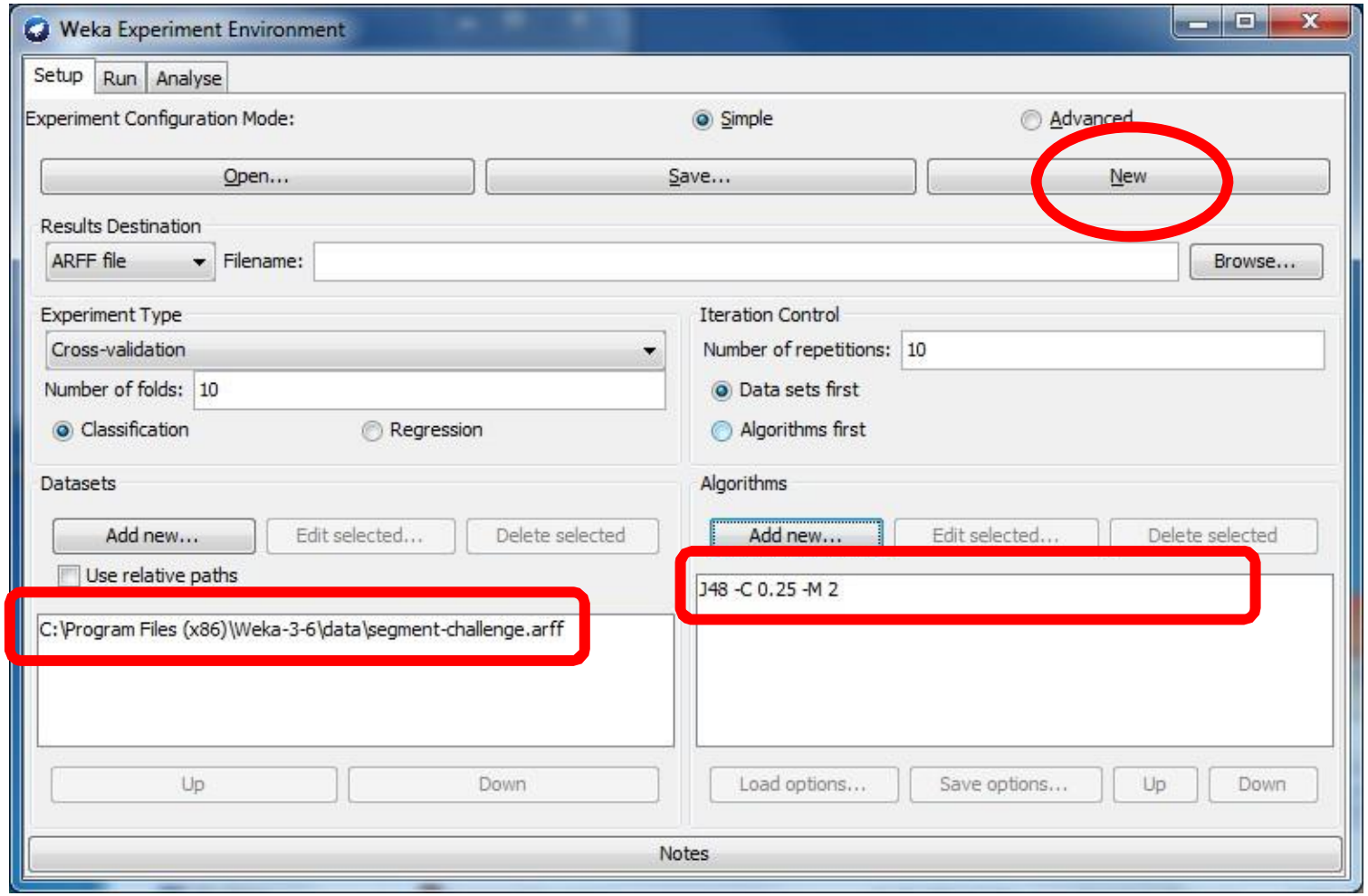


Lesson 1.2: 探索Experimenter

▼結果展示



Lesson 1.2: 探索Experimenter



總結

Setup 面板

- 點選 *New*
- 注意預設值s
  - 10層交叉驗證，重複實驗10次
- 在*Datasets區域中*, 點選 *Add new*開啟檔案segment-challenge.arff
- 在*Algorithms區域中*, 點選 *Add new*開啟分類器trees資料夾下的J48

Run 面板

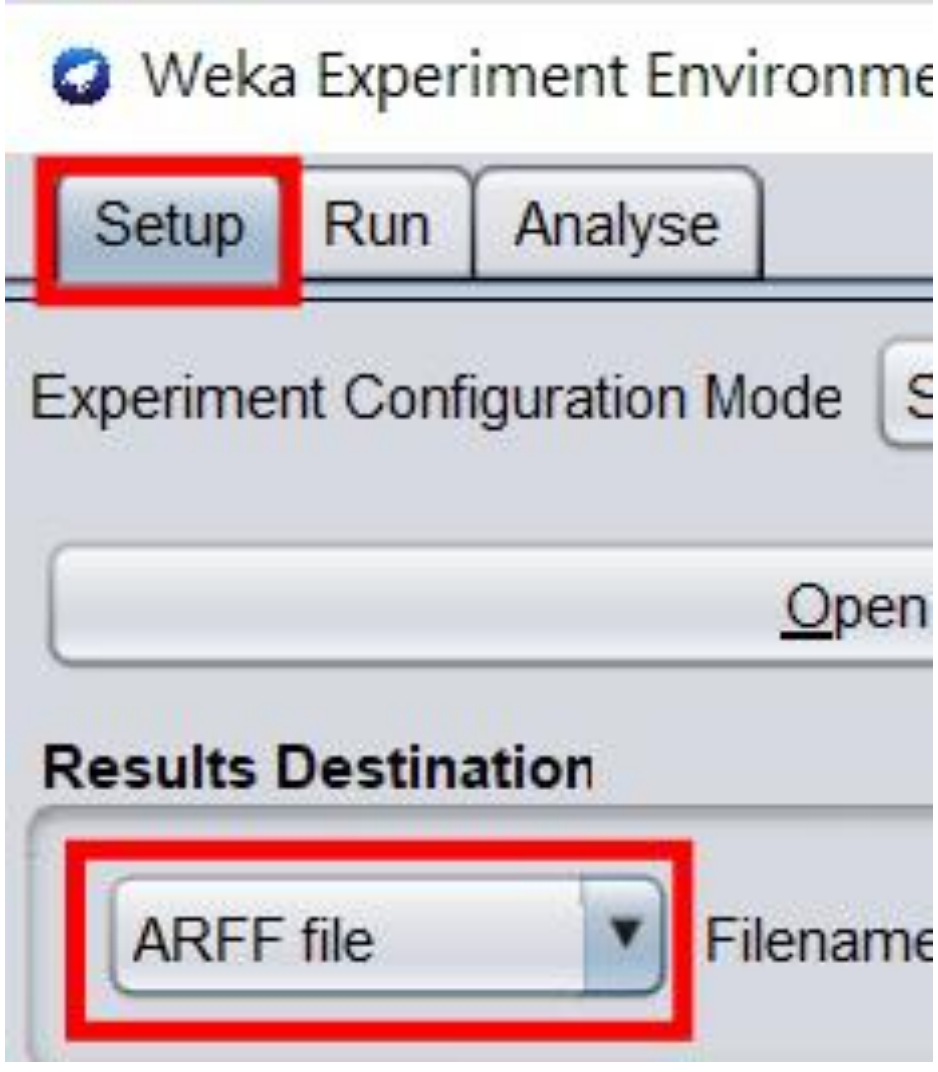
- 點選 *Start*

Analyse 面板

- 點選 *Experiment*
- 勾選 *Show std. deviations*
- 點選 *Perform test*

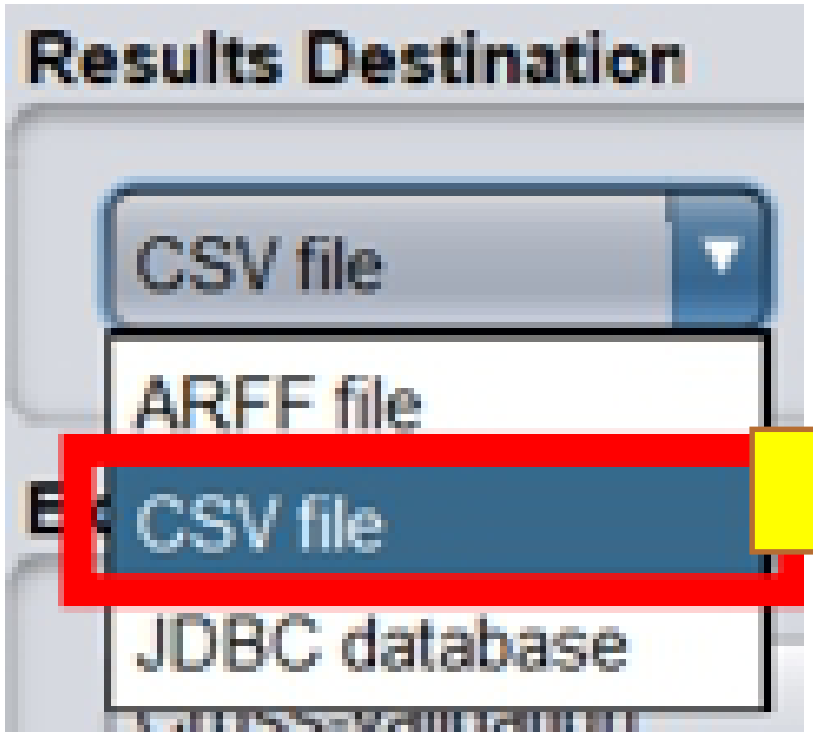
Lesson 1.2: 探索Experimenter

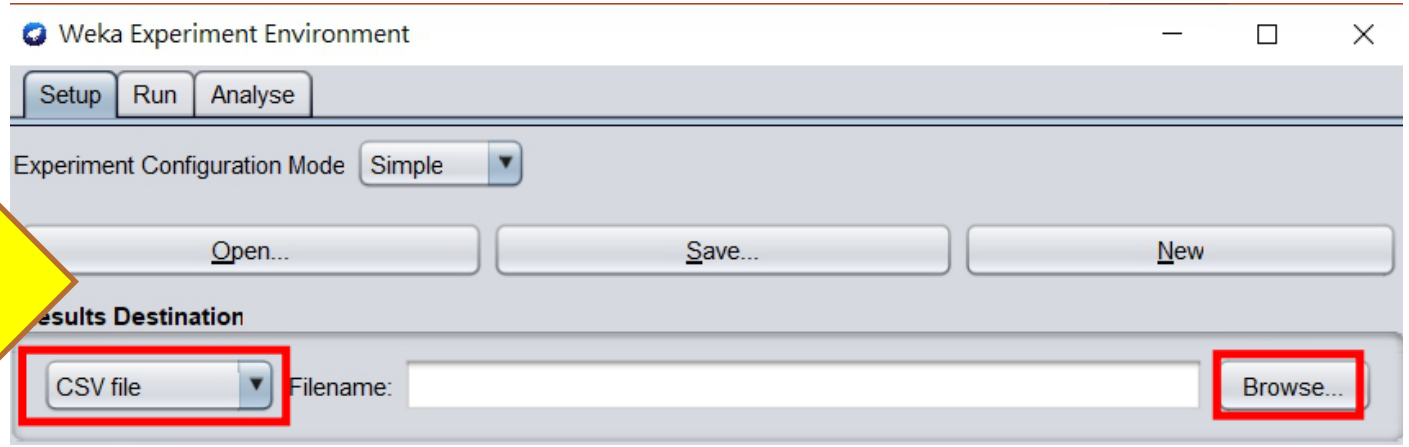
12.切換到Setup面板，左鍵單擊Result Destination區域中的下拉式選單(圖中紅框處)



Lesson 1.2: 探索Experimenter

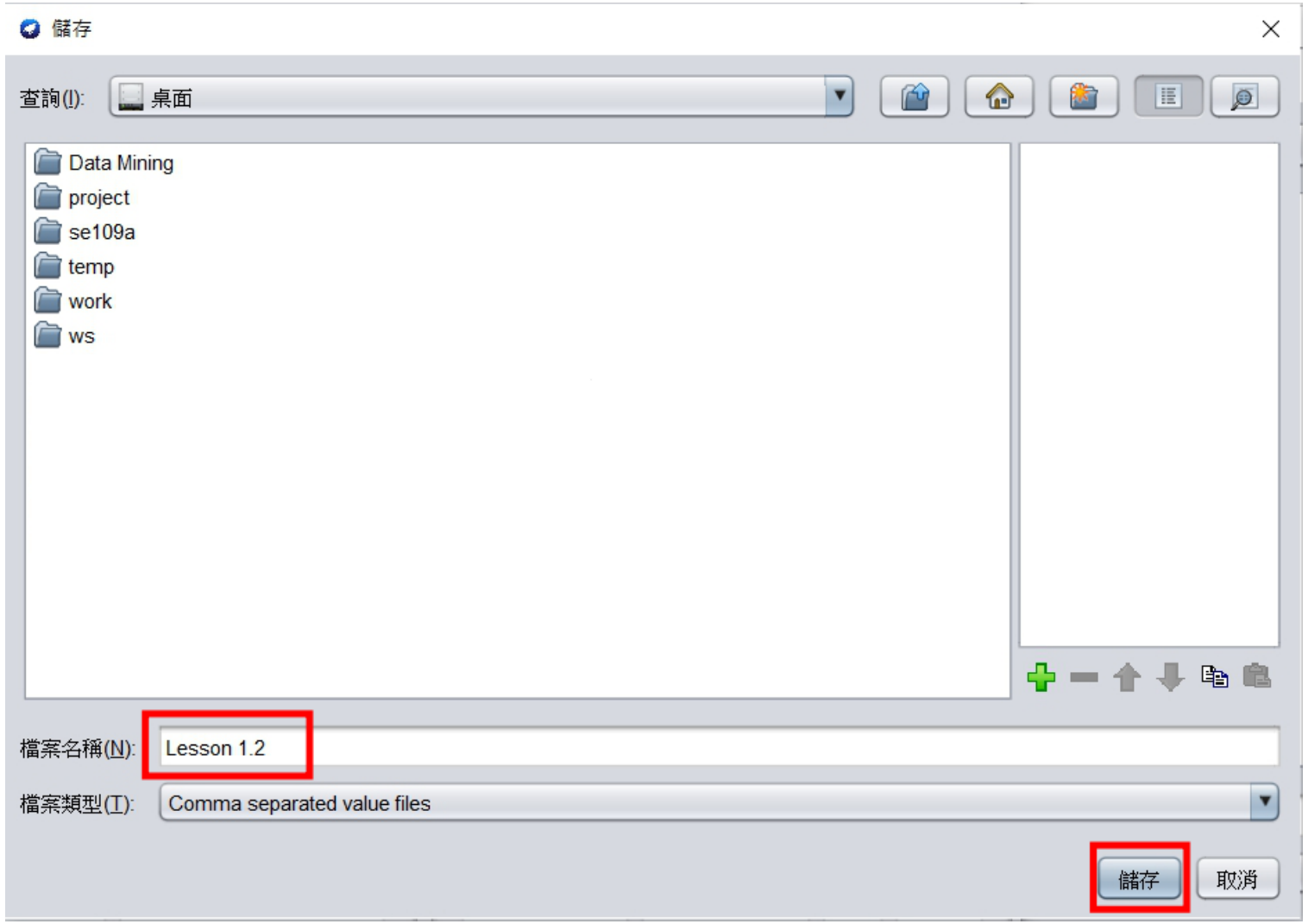
13.在出現的選單中，左鍵單擊CSV file。接著，按下右方的Browse鈕設定檔案儲存位置。





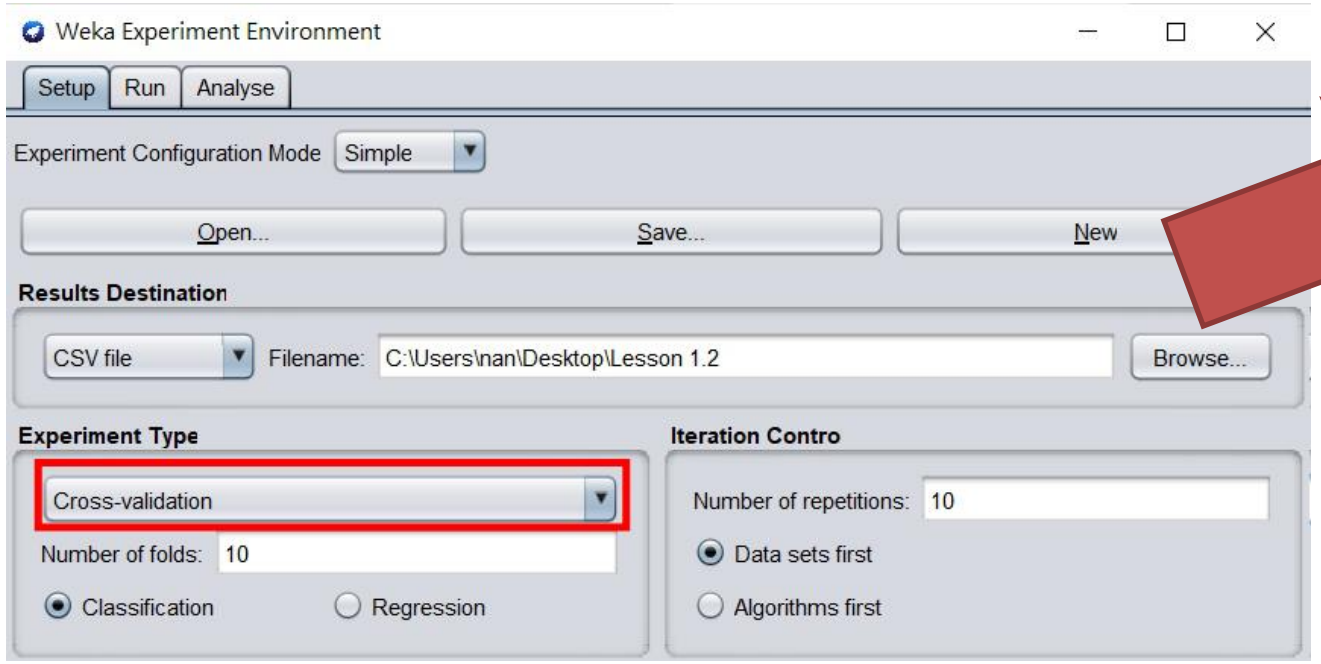
Lesson 1.2: 探索Experimenter

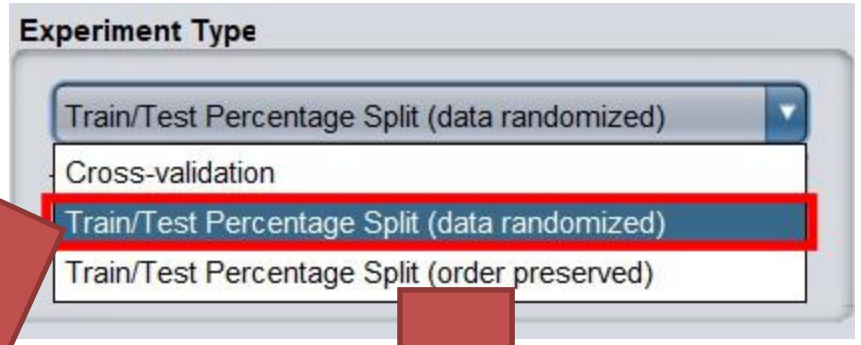
14.我們將檔案存在桌面，並在檔案名稱後的輸入框中填上 Lesson 1.2，然後按下下方儲存按鈕。

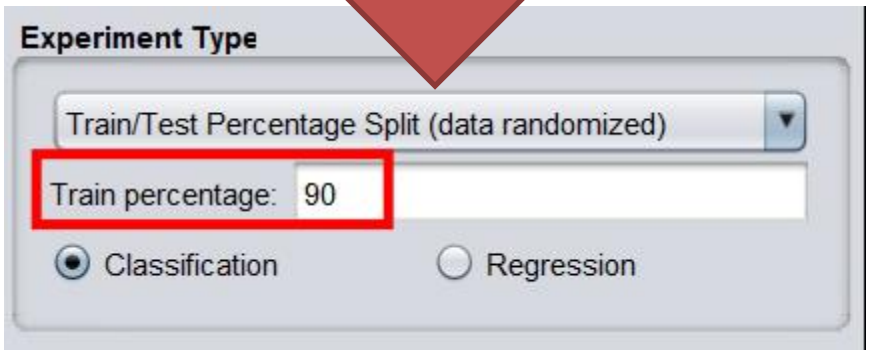


Lesson 1.2: 探索Experimenter

15.接著做比例分割。
左鍵單擊Experiment Type區域下的下拉式選單，選擇Train/Test Percentage Split選項。接著在下方的Train percentag後的輸入框填上90。





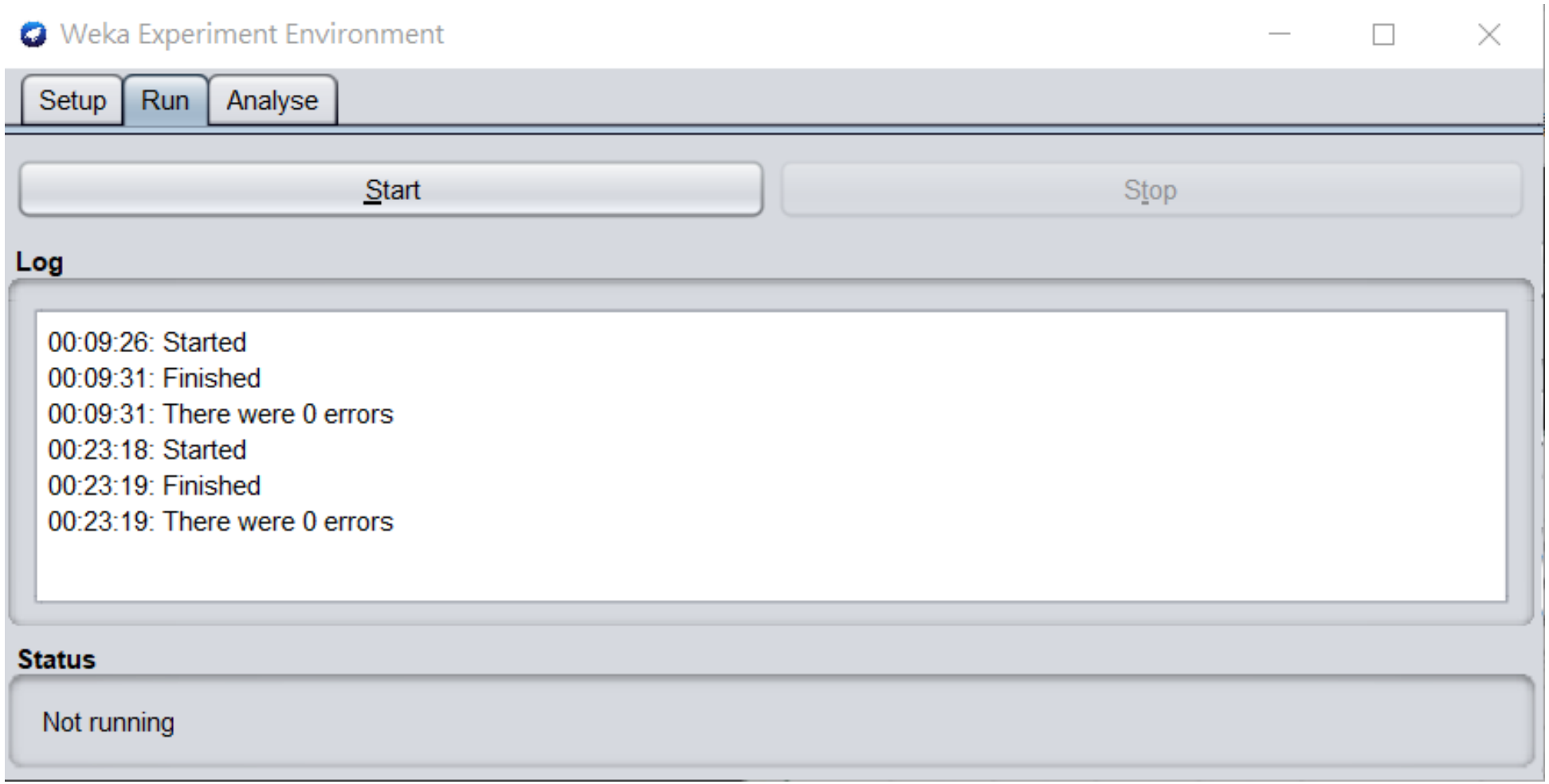


Lesson 1.2: 探索Experimenter

16.切換到Run面板，並按下Start。

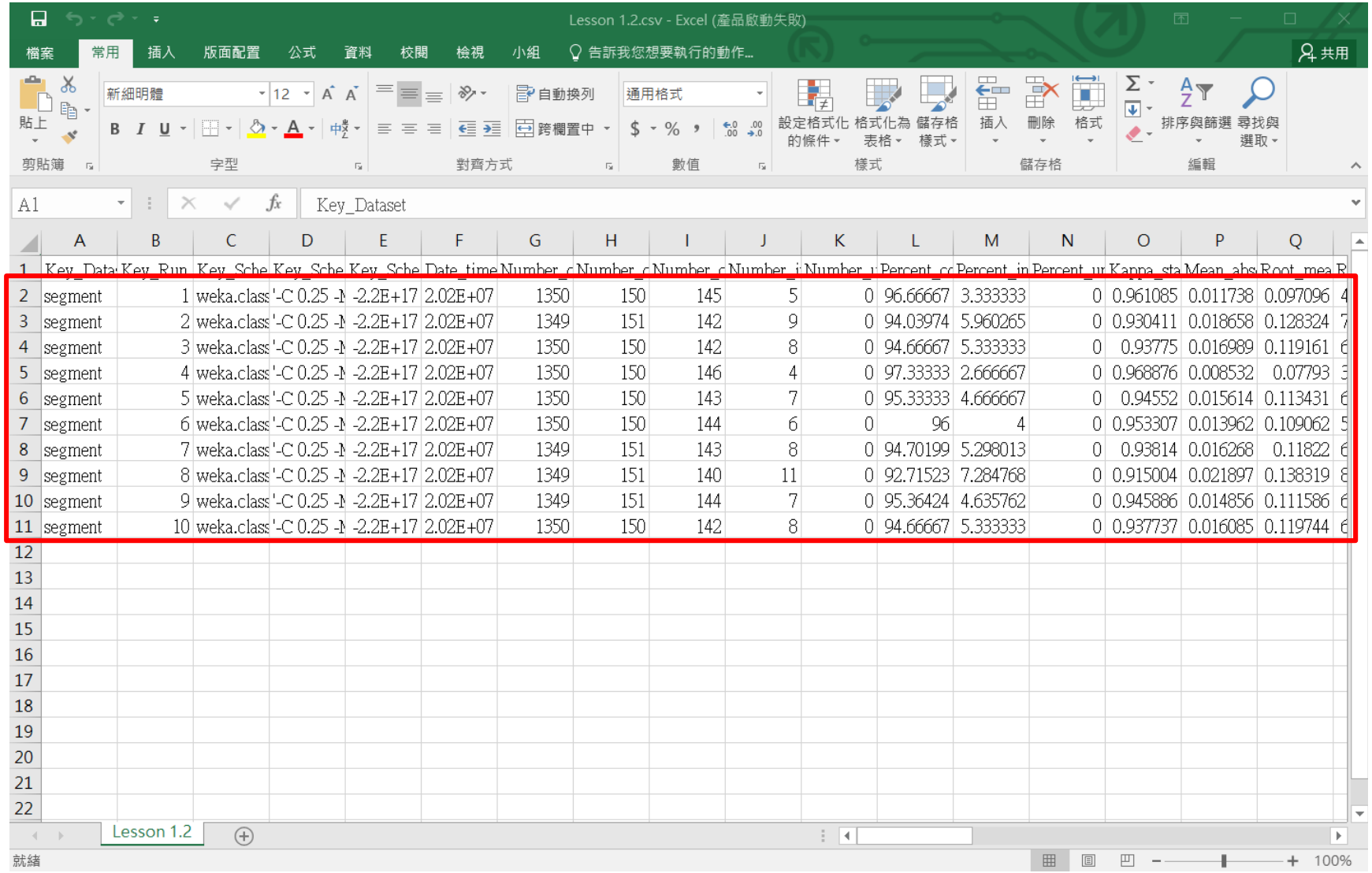
Lesson 1.2: 探索Experimenter

▼結果展示



Lesson 1.2: 探索Experimenter

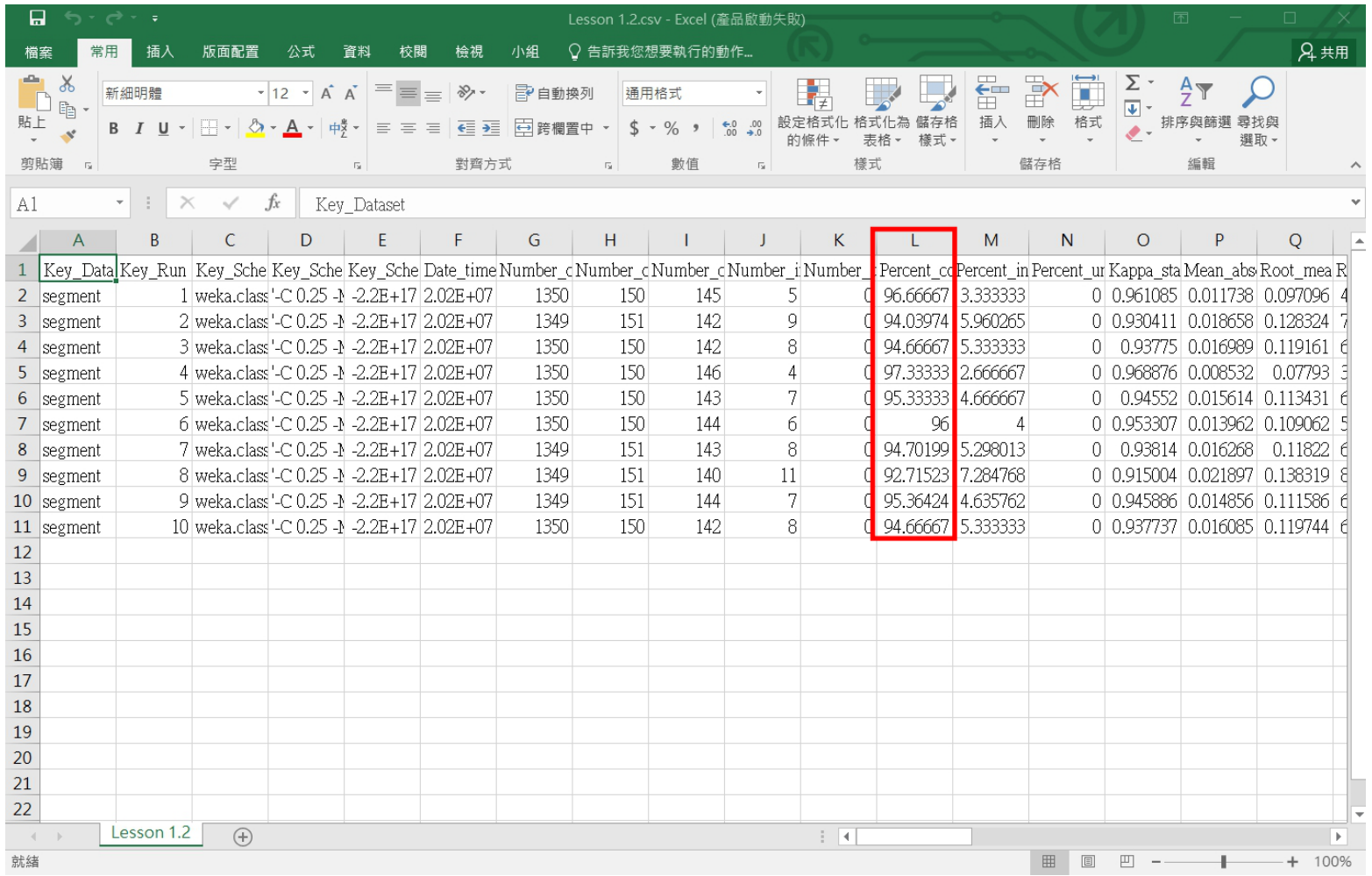
17.查看輸出的CSV文檔。我們重複實驗十次，這些是這十次的運行記錄。



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Key_Data	Key_Run	Key_Sche	Key_Sche	Key_Sche	Date_time	Number_c	Number_c	Number_c	Number_i	Number_u	Percent_co	Percent_in	Percent_un	Kappa_sta	Mean_abs	Root_mea
2	segment	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961085	0.011738	0.097096
3	segment	2	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1349	151	142	9	0	94.03974	5.960265	0	0.930411	0.018658	0.128324
4	segment	3	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.93775	0.016989	0.119161
5	segment	4	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	146	4	0	97.33333	2.666667	0	0.968876	0.008532	0.07793
6	segment	5	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.94552	0.015614	0.113431
7	segment	6	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953307	0.013962	0.109062
8	segment	7	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1349	151	143	8	0	94.70199	5.298013	0	0.93814	0.016268	0.11822
9	segment	8	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1349	151	140	11	0	92.71523	7.284768	0	0.915004	0.021897	0.138319
10	segment	9	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1349	151	144	7	0	95.36424	4.635762	0	0.945886	0.014856	0.111586
11	segment	10	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.937737	0.016085	0.119744
12																	
13																	
14																	
15																	
16																	
17																	
18																	
19																	
20																	
21																	
22																	

Lesson 1.2: 探索Experimenter

18.其中Percent_correct這是十次中每次運行的正確率。表中還有很多信息，包括開始時間、運行時間、和其他很多信息。



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Key_Data	Key_Run	Key_Sche	Key_Sche	Key_Sche	Date_time	Number_c	Number_c	Number_c	Number_i	Number_	Percent_co	Percent_in	Percent_ur	Kappa_sta	Mean_abs	Root_mea
2	segment	1	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961085	0.011738	0.097096
3	segment	2	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1349	151	142	9	0	94.03974	5.960265	0	0.930411	0.018658	0.128324
4	segment	3	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.93775	0.016989	0.119161
5	segment	4	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1350	150	146	4	0	97.33333	2.666667	0	0.968876	0.008532	0.07793
6	segment	5	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.94552	0.015614	0.113431
7	segment	6	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953307	0.013962	0.109062
8	segment	7	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1349	151	143	8	0	94.70199	5.298013	0	0.93814	0.016268	0.11822
9	segment	8	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1349	151	140	11	0	92.71523	7.284768	0	0.915004	0.021897	0.138319
10	segment	9	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1349	151	144	7	0	95.36424	4.635762	0	0.945886	0.014856	0.111586
11	segment	10	weka.class	'-C 0.25 -N	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.937737	0.016085	0.119744
12																	
13																	
14																	
15																	
16																	
17																	
18																	
19																	
20																	
21																	
22																	

Lesson 1.2: 探索Experimenter

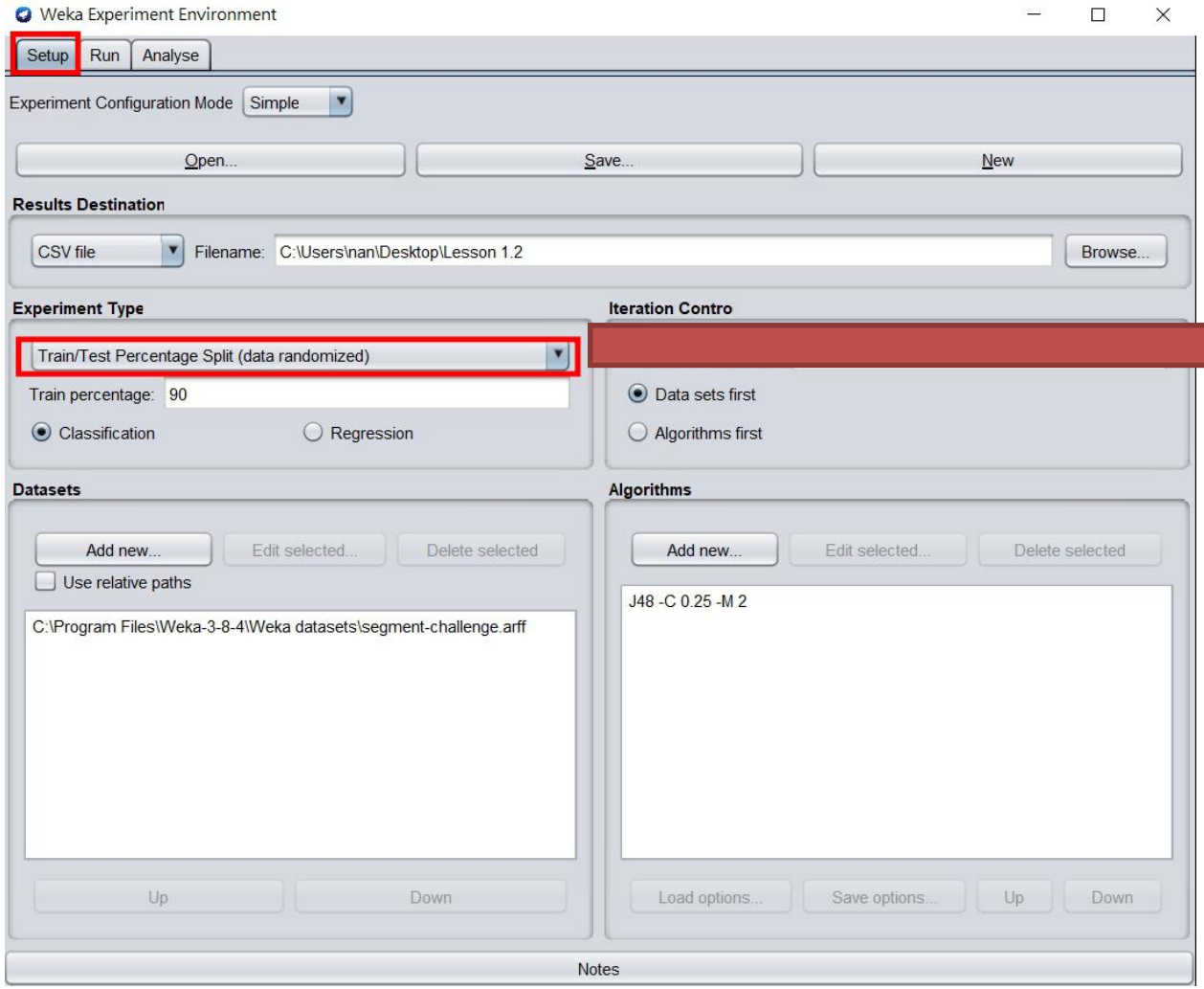
接著，再次運行交叉驗證的實驗

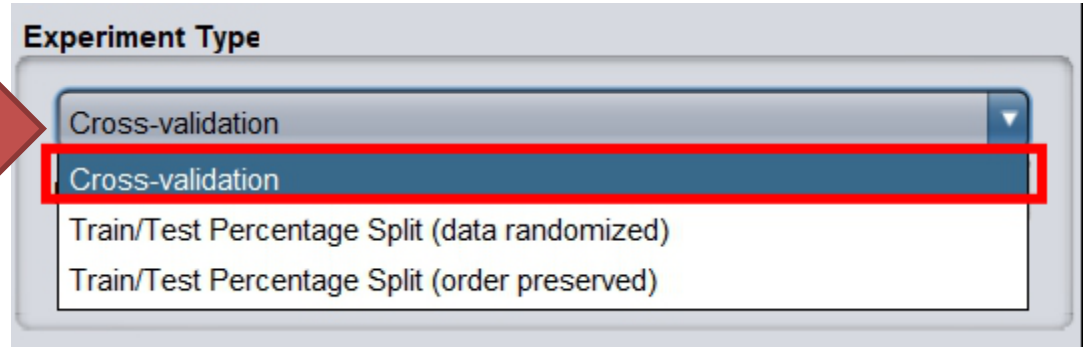
- 打開結果的電子表單

Lesson 1.2: 探索Experimenter

現在要做十層交叉驗證，並將結果寫入文檔。

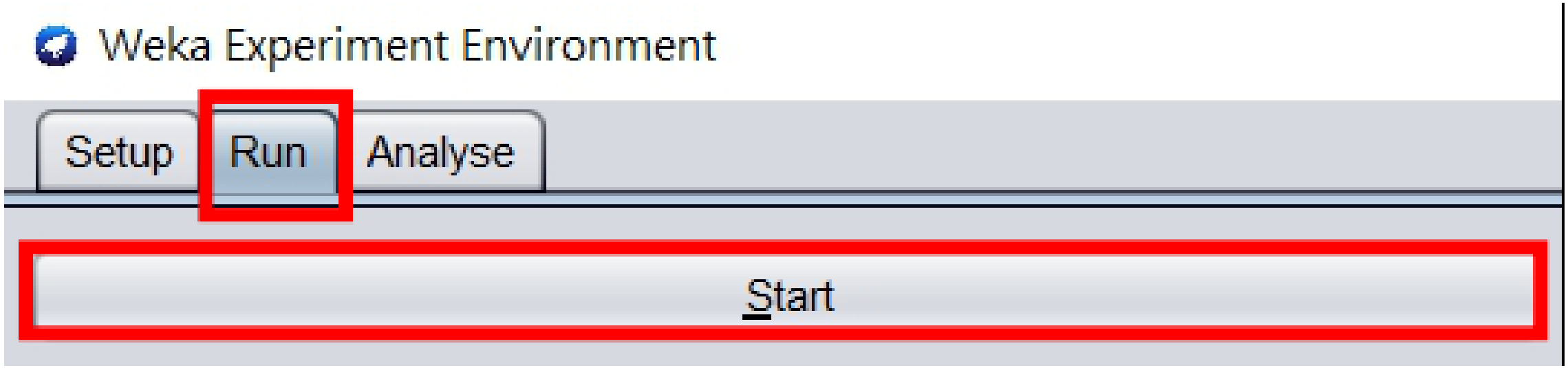
19.切換到Setup面板，左鍵單擊Experiment Type區域中的下拉式選單，選擇Cross-validation。





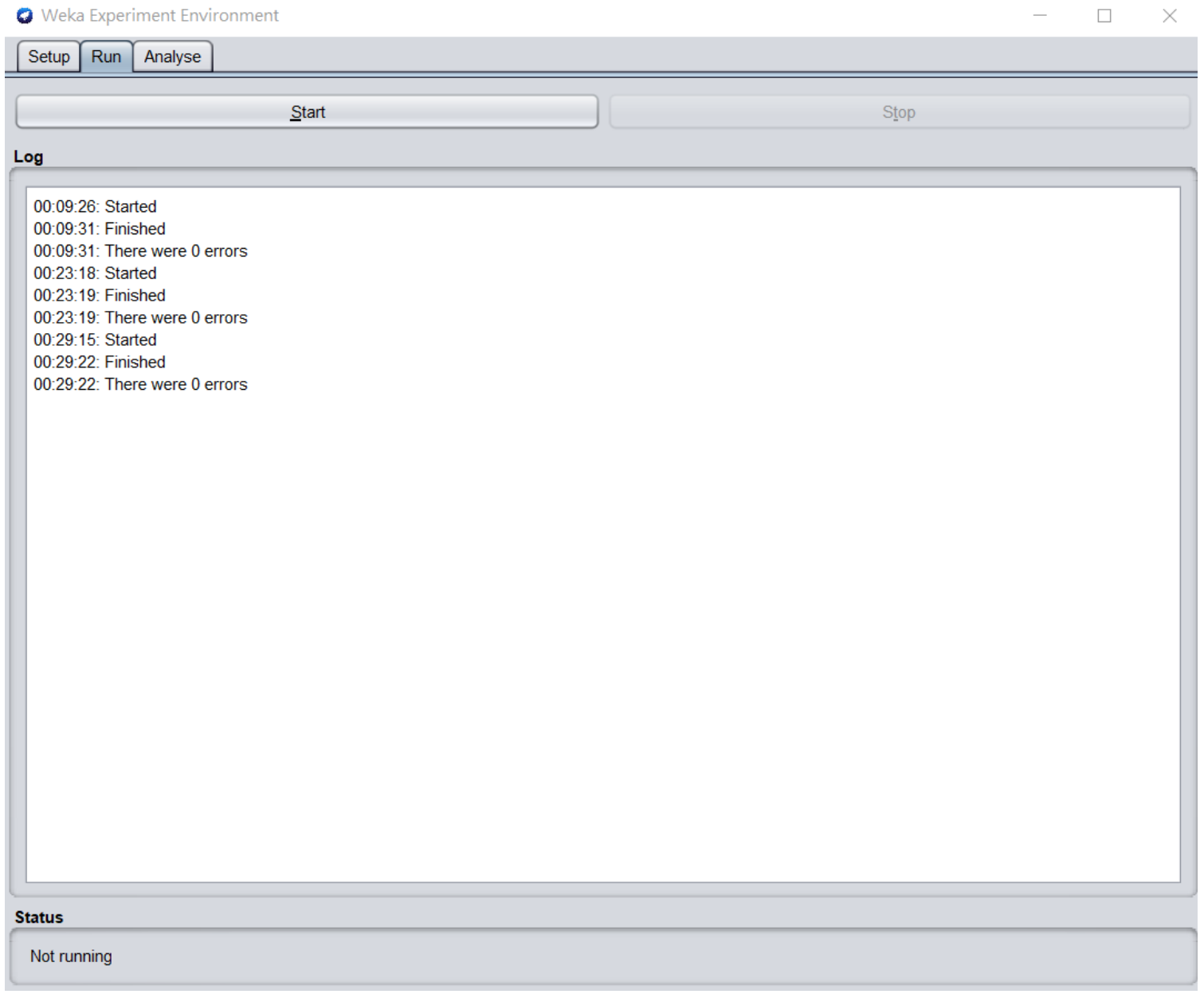
Lesson 1.2: 探索Experimenter

20.切換到Run面板，左鍵單擊Start



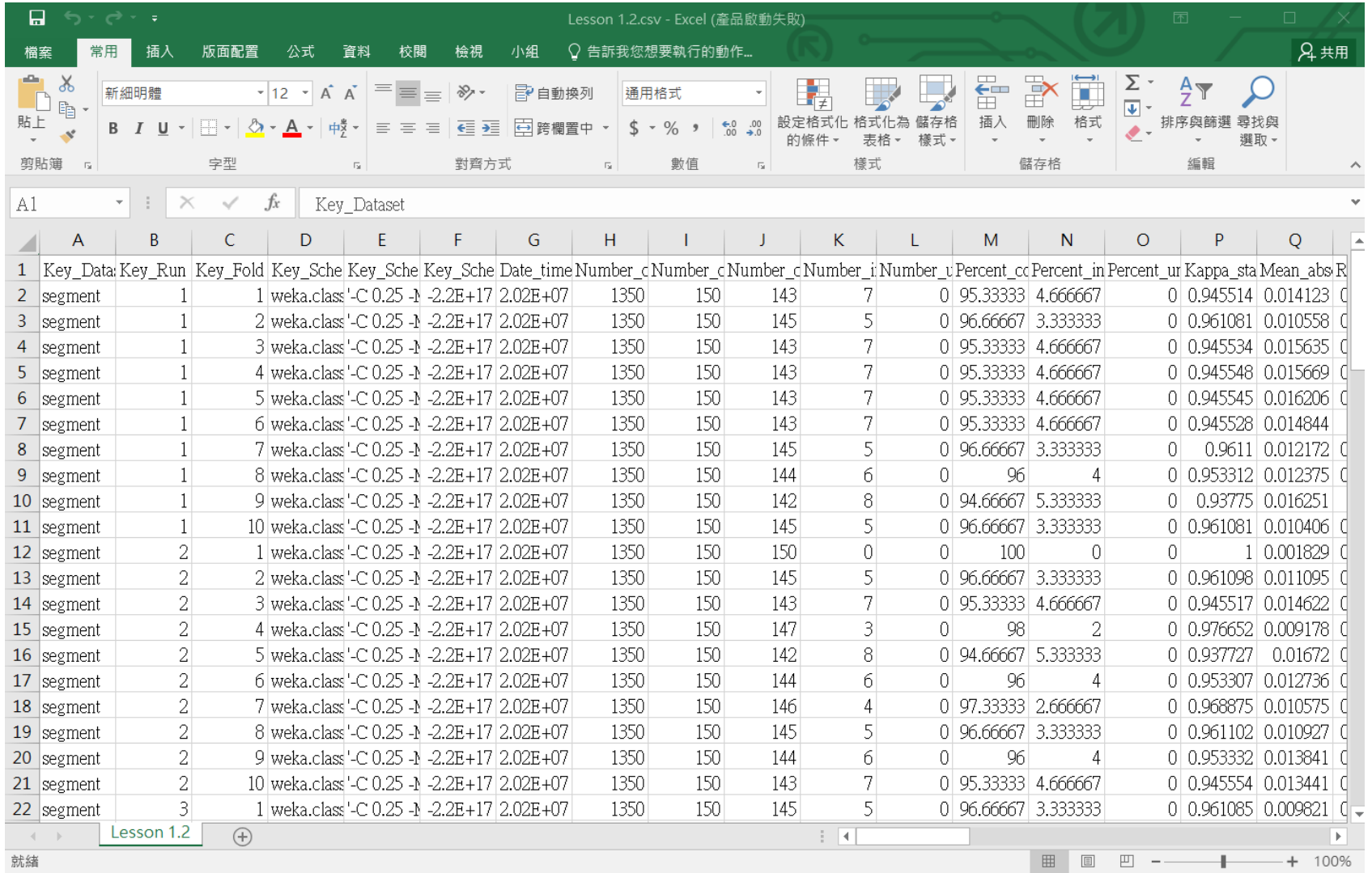
Lesson 1.2: 探索Experimenter

▼結果展示



Lesson 1.2: 探索Experimenter

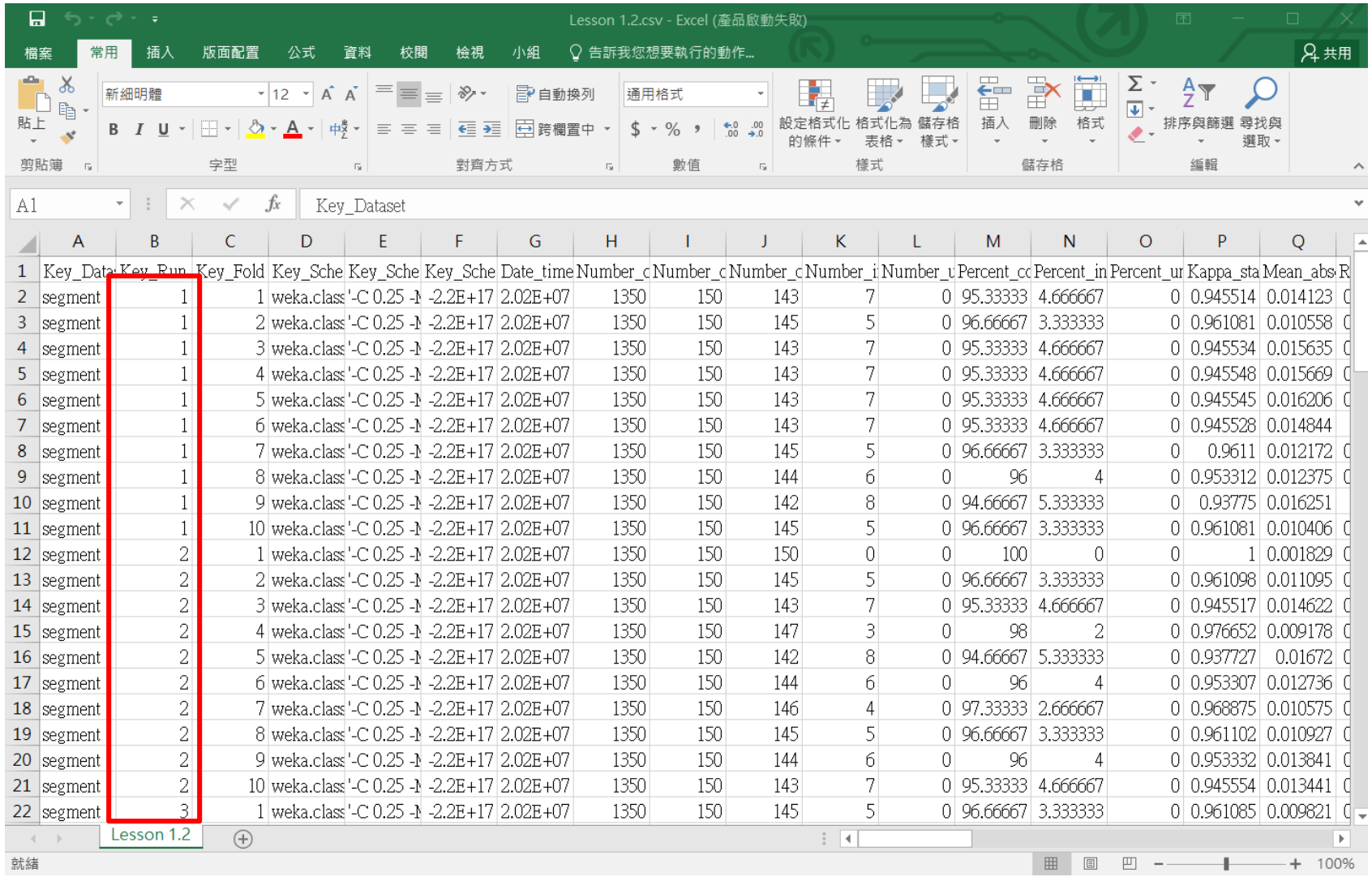
21.文檔已經完成輸出，再次打開檔案Lesson 1.2查看。



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Key_Data	Key_Run	Key_Fold	Key_Sche	Key_Sche	Key_Sche	Date_time	Number_c	Number_c	Number_c	Number_i	Number_u	Percent_cc	Percent_in	Percent_ur	Kappa_sta	Mean_abs
2	segment	1	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945514	0.014123
3	segment	1	2	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961081	0.010558
4	segment	1	3	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945534	0.015635
5	segment	1	4	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945548	0.015669
6	segment	1	5	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945545	0.016206
7	segment	1	6	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945528	0.014844
8	segment	1	7	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.9611	0.012172
9	segment	1	8	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953312	0.012375
10	segment	1	9	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.93775	0.016251
11	segment	1	10	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961081	0.010406
12	segment	2	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	150	0	0	100	0	0	1	0.001829
13	segment	2	2	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961098	0.011095
14	segment	2	3	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945517	0.014622
15	segment	2	4	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	147	3	0	98	2	0	0.976652	0.009178
16	segment	2	5	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.937727	0.01672
17	segment	2	6	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953307	0.012736
18	segment	2	7	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	146	4	0	97.33333	2.666667	0	0.968875	0.010575
19	segment	2	8	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961102	0.010927
20	segment	2	9	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953332	0.013841
21	segment	2	10	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945554	0.013441
22	segment	3	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961085	0.009821

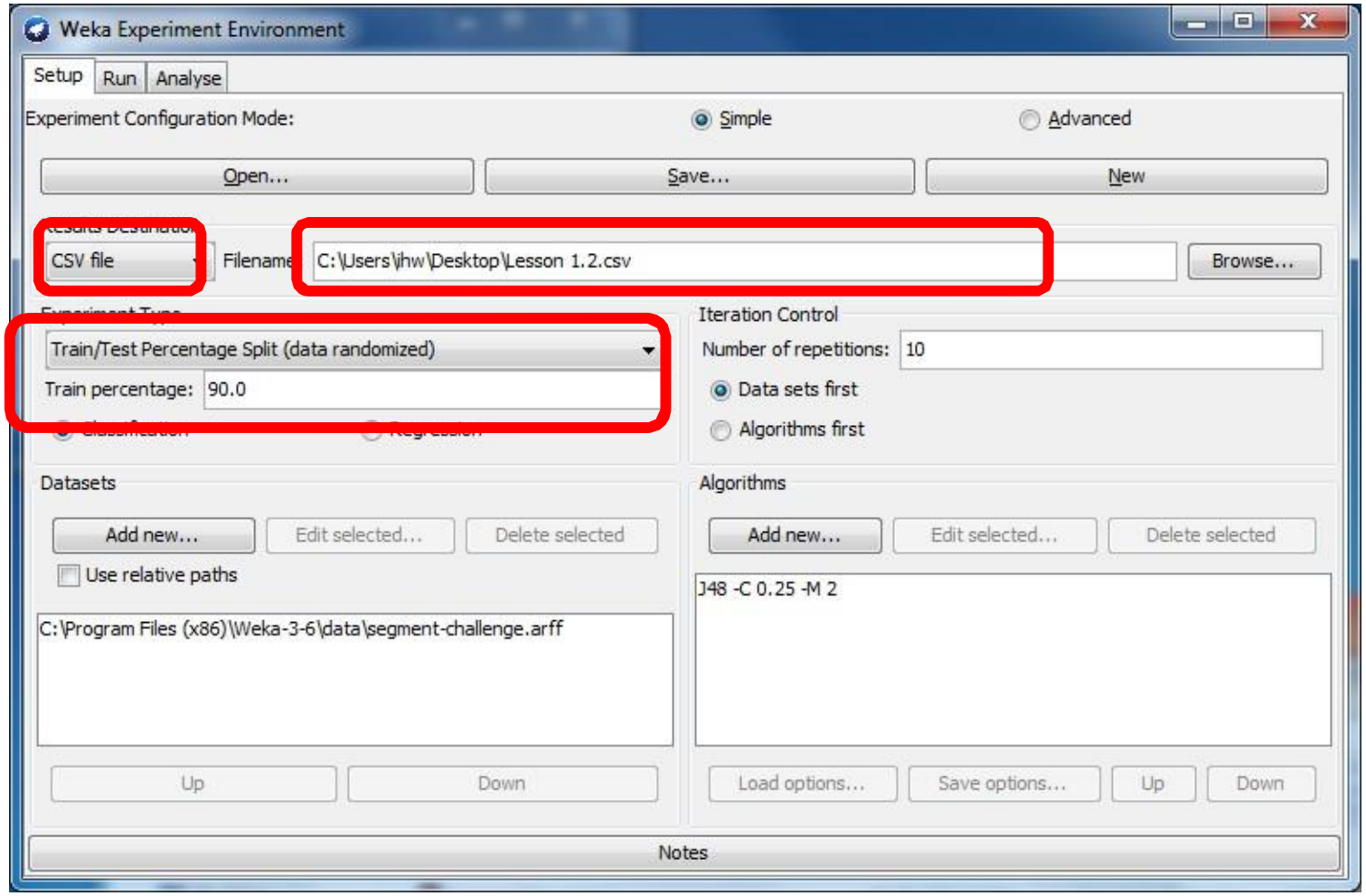
Lesson 1.2: 探索Experimenter

▼我們運行了十次，十次的十層交叉驗證。



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Key_Data	Key_Run	Key_Fold	Key_Sche	Key_Sche	Key_Sche	Date_time	Number_c	Number_c	Number_c	Number_i	Number_u	Percent_cc	Percent_in	Percent_ur	Kappa_sta	Mean_abs
2	segment	1	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945514	0.014123
3	segment	1	2	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961081	0.010558
4	segment	1	3	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945534	0.015635
5	segment	1	4	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945548	0.015669
6	segment	1	5	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945545	0.016206
7	segment	1	6	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945528	0.014844
8	segment	1	7	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.9611	0.012172
9	segment	1	8	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953312	0.012375
10	segment	1	9	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.93775	0.016251
11	segment	1	10	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961081	0.010406
12	segment	2	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	150	0	0	100	0	0	1	0.001829
13	segment	2	2	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961098	0.011095
14	segment	2	3	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945517	0.014622
15	segment	2	4	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	147	3	0	98	2	0	0.976652	0.009178
16	segment	2	5	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	142	8	0	94.66667	5.333333	0	0.937727	0.01672
17	segment	2	6	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953307	0.012736
18	segment	2	7	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	146	4	0	97.33333	2.666667	0	0.968875	0.010575
19	segment	2	8	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961102	0.010927
20	segment	2	9	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	144	6	0	96	4	0	0.953332	0.013841
21	segment	2	10	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	143	7	0	95.33333	4.666667	0	0.945554	0.013441
22	segment	3	1	weka.class	'-C 0.25 -M	-2.2E+17	2.02E+07	1350	150	145	5	0	96.66667	3.333333	0	0.961085	0.009821

Lesson 1.2: 探索Experimenter



總結

為了得到詳細的數據

回到***Setup*** 面板

- 選擇.csv檔
- 為結果命名
- *Train/Test Split*；*90%*

Lesson 1.2: 探索Experimenter

總結

切換至*Run面板*

- 點擊*Start*
- 打開結果的電子表格

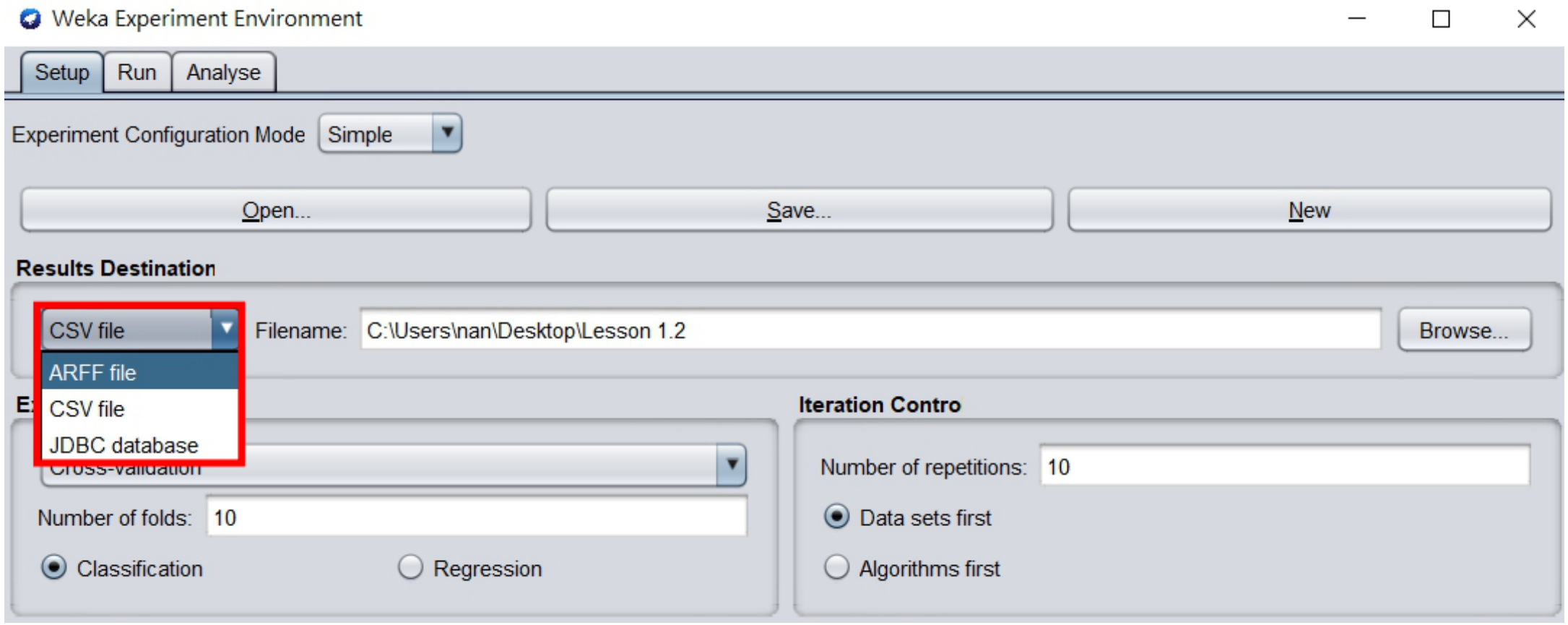
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Key_Datas	ey_Ru	ey_Scheme	Key_Scheme	Key_Sche	Date_time	Number_of_	Number_	Number_	Number_	Number_	Percent_correct	cent_incorre	Percer	Kappa_s	Mean_a	Root_me
2	segment	1	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1350	150	145	5	0	96.7	3.3	0	0.9611	0.012	0.09709
3	segment	2	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1349	151	142	9	0	94.0	6.0	0	0.9304	0.019	0.12832
4	segment	3	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1350	150	142	8	0	94.7	5.3	0	0.9377	0.017	0.11916
5	segment	4	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1350	150	146	4	0	97.3	2.7	0	0.9689	0.009	0.07793
6	segment	5	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1350	150	143	7	0	95.3	4.7	0	0.9455	0.016	0.11343
7	segment	6	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1350	150	144	6	0	96.0	4.0	0	0.9533	0.014	0.10906
8	segment	7	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1349	151	143	8	0	94.7	5.3	0	0.9381	0.016	0.11822
9	segment	8	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1349	151	141	10	0	93.4	6.6	0	0.9227	0.02	0.13130
10	segment	9	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1349	151	144	7	0	95.4	4.6	0	0.9459	0.015	0.11158
11	segment	10	ers.trees.J48	'-C 0.25 -M 2	-2.2E+17	2.01E+07	1350	150	142	8	0	94.7	5.3	0	0.9377	0.016	0.11974
12																	
13																	

Lesson 1.2: 探索Experimenter

複習Experimenter的功能。
下圖是Setup面板中的Open...、Save...和New...按鈕的功能

Lesson 1.2: 探索Experimenter

Setup面板中的Results Destination中，我們可以選擇ARFF文檔 / CSV文檔 / 數據庫文檔，並為輸出文檔命名，。

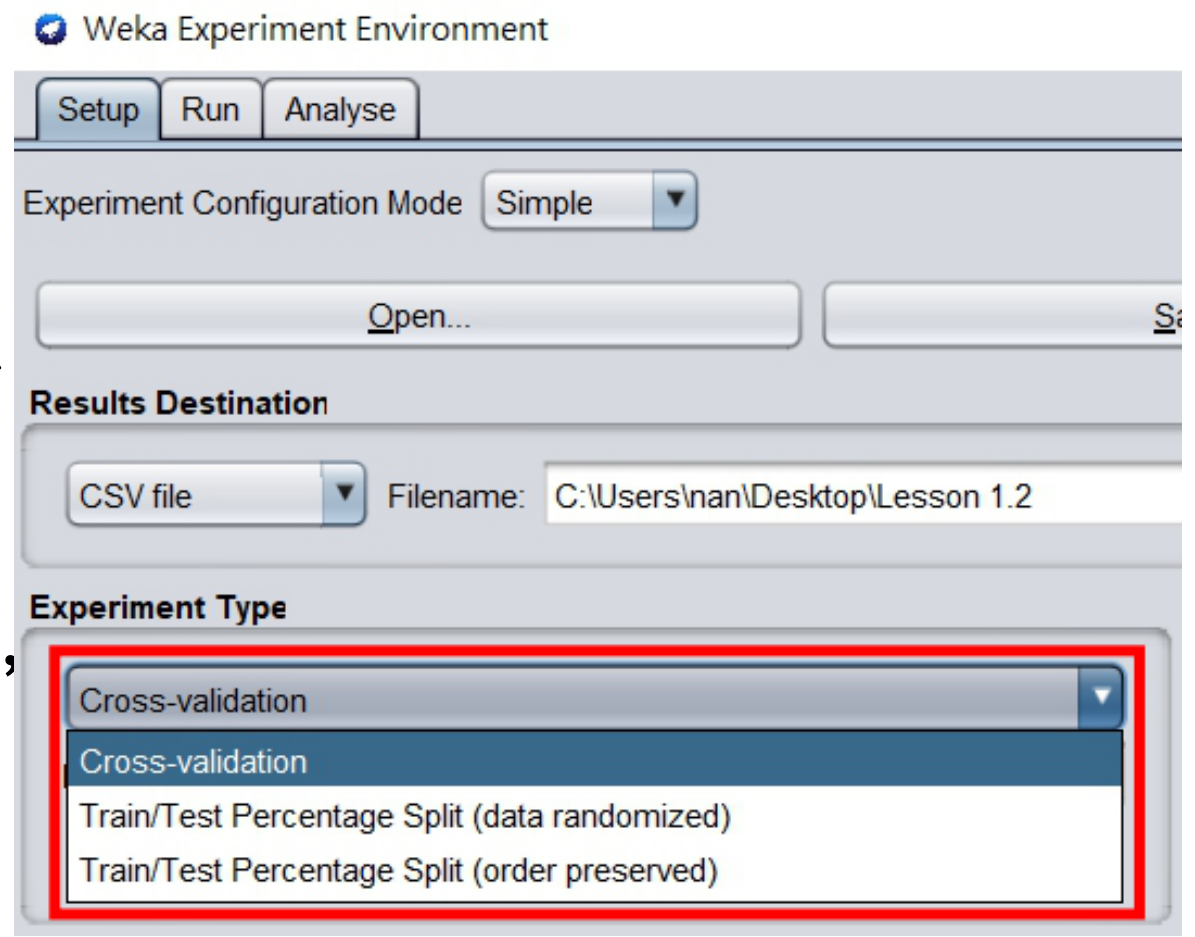


Lesson 1.2: 探索Experimenter

Setup面板中的**Experiment Type**的下拉式選單，可以做交叉驗證或者百分比分割。

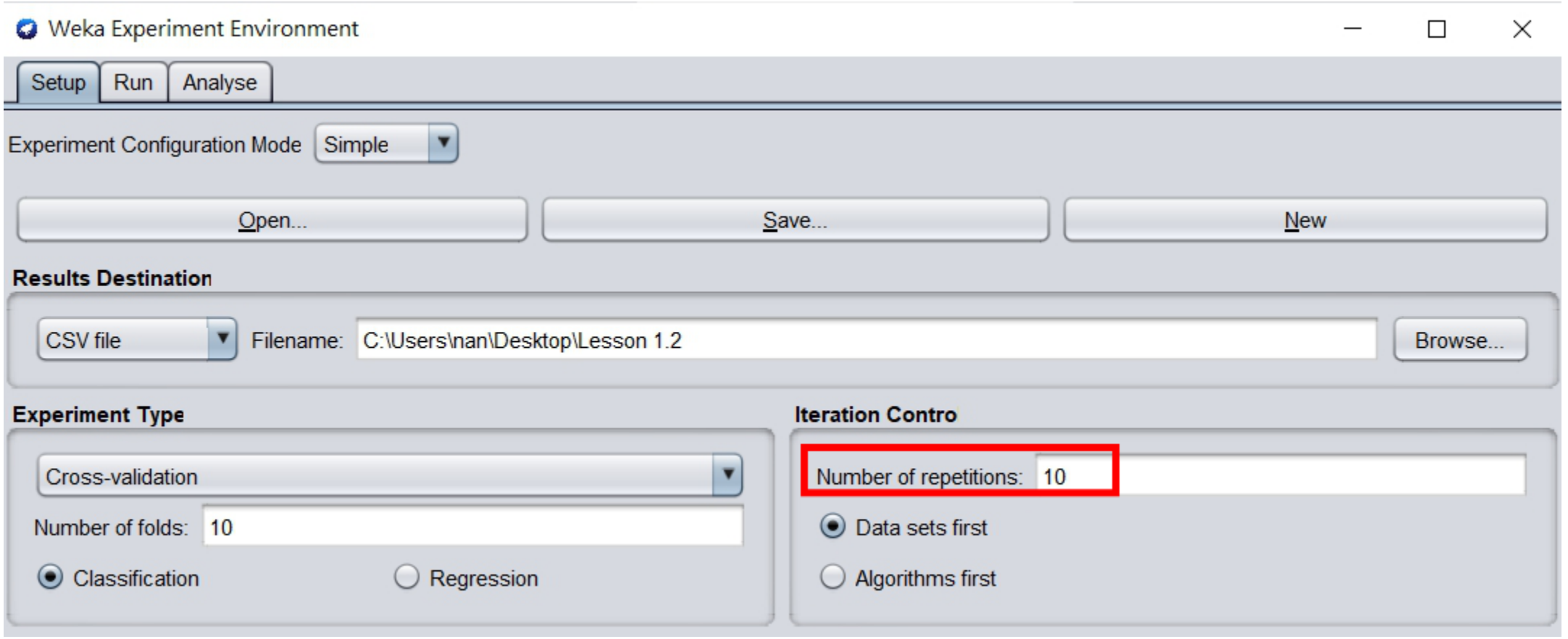
實際上，因為**Experimenter**不支持使用獨立的測試文檔，我們可以在做百分比分割時設置運行的順序，將訓練資料集和測試資料集綁定在一起。設定好順序，指定合適的百分比後，後面的資料就會成為測試資料。

但一般來說，我們不會這樣做，我們會隨機百分比。



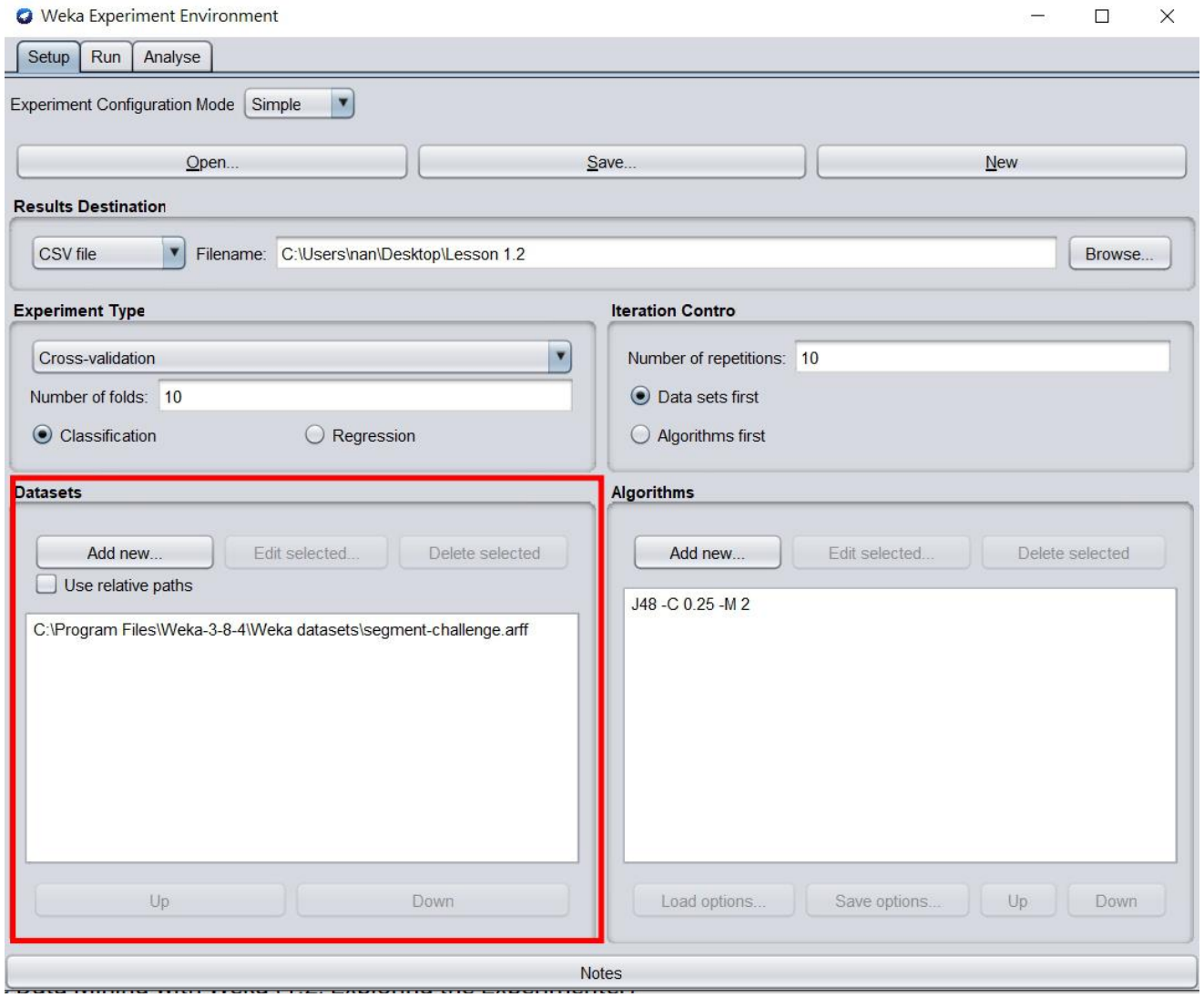
Lesson 1.2: 探索Experimenter

從Setup面板中的Iteration Contro區域內的Number of repetitions可以得知重複的次數。這裡我們重複了十次，也可以選擇重複一百次。



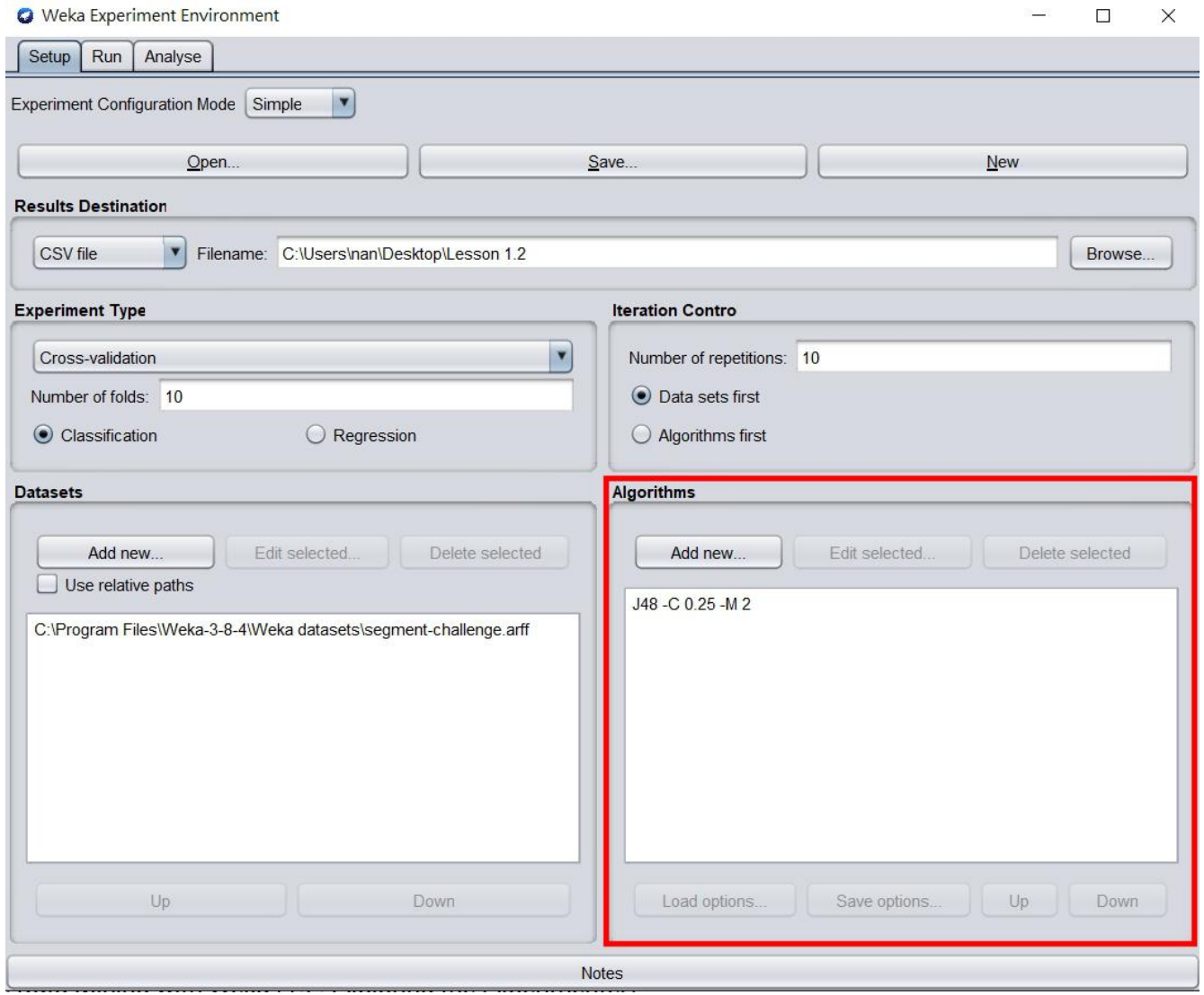
Lesson 1.2: 探索Experimenter

Setup面板中的Datasets區域內，我們可以增加新的資料集，也可以刪除資料集。



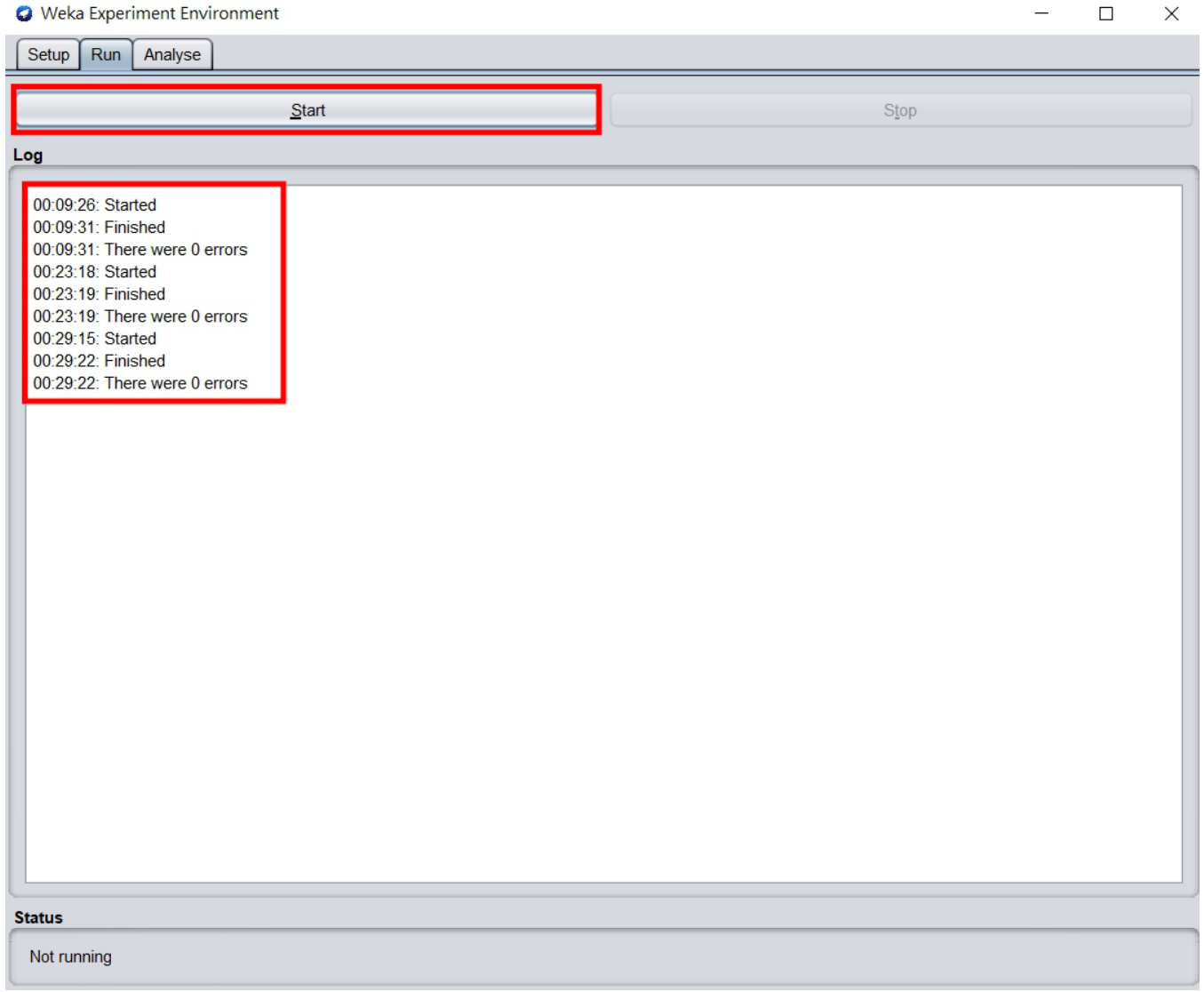
Lesson 1.2: 探索Experimenter

在Setup面板中的Algorithms區域內，可以輸入演算法。



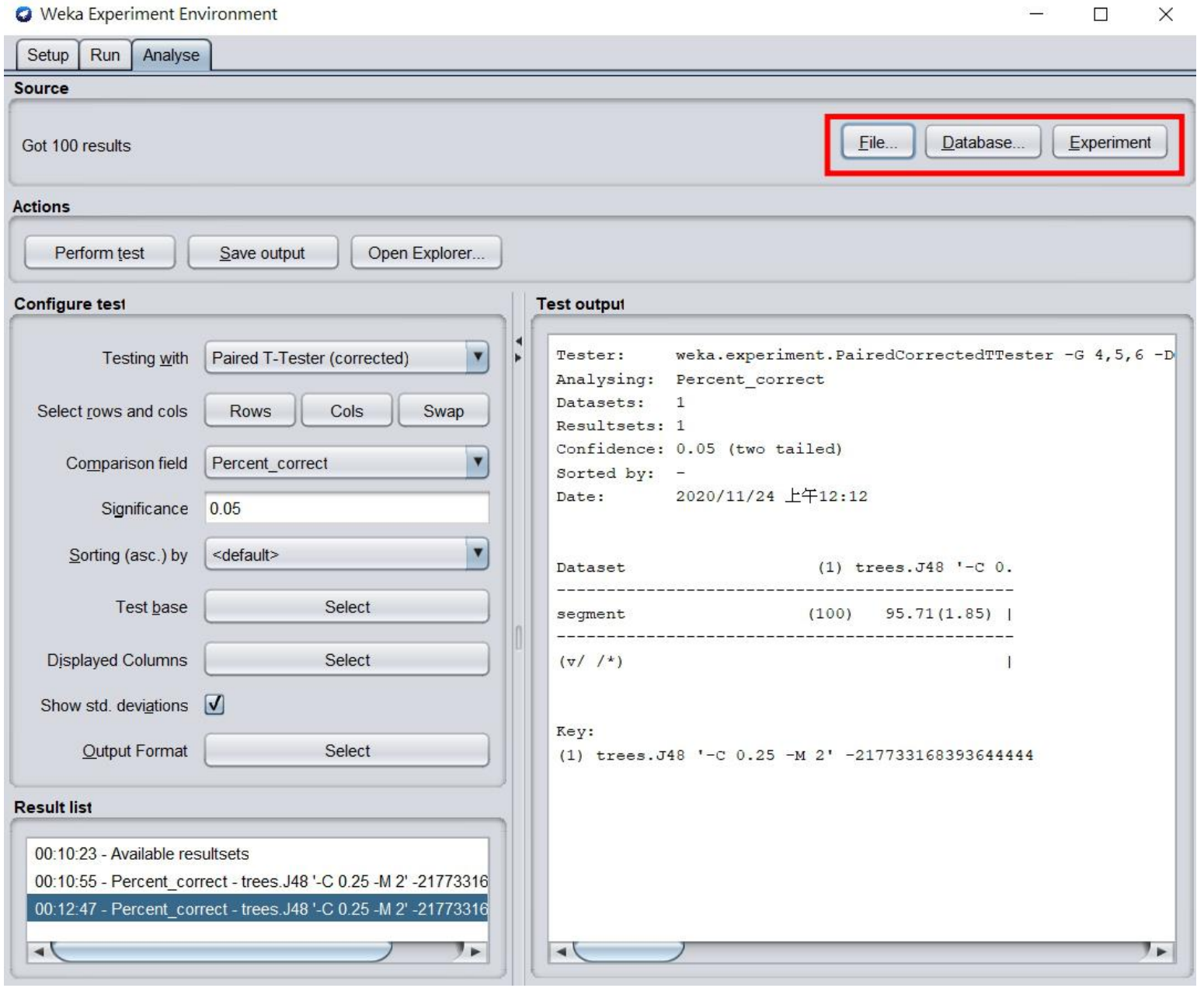
Lesson 1.2: 探索Experimenter

Run面板中的操作，我們只點擊Start紐，並在下方Log區域內監控錯誤。



Lesson 1.2: 探索Experimenter

在Analyse面板中，我們可以從File...或Database鈕導入結果，但我們通常做的是點擊Experiment獲取實驗數據。



Lesson 1.2: 探索Experimenter

Setup面板

- 儲存/ 載入實驗
- 將結果存在Arff檔...或是資料庫
- 在Train/Test split中保留順序 (不能做重複實驗)
- 使用多個資料集和多個分類器
- Advanced mode(進階模式)

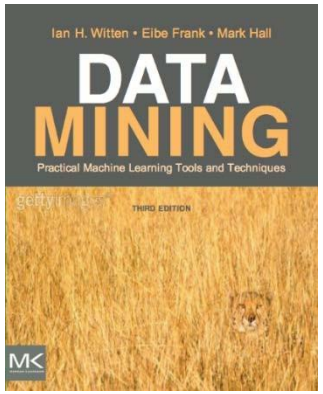
Run 面板

Analyse 面板

- 從.csv or Arff檔案載入結果 ...或是從資料庫
- 有許多選項

Lesson 1.2: 探索Experimenter

- 開啟Experimenter
- Setup, Run, Analyse面板
- 評估在一個資料集上使用一個分類器
  - … 使用交叉驗證, 重複 10 次
  - … 使用比例分割, 重複 10 次
- 檢測電子表單的輸出
- 在Analyse面板中可以得到平均值和標準差
- 在Setup和Run面板中的其他選項



課程文本

- Chapter 13 *The Experimenter*

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Class 1 – Lesson 3

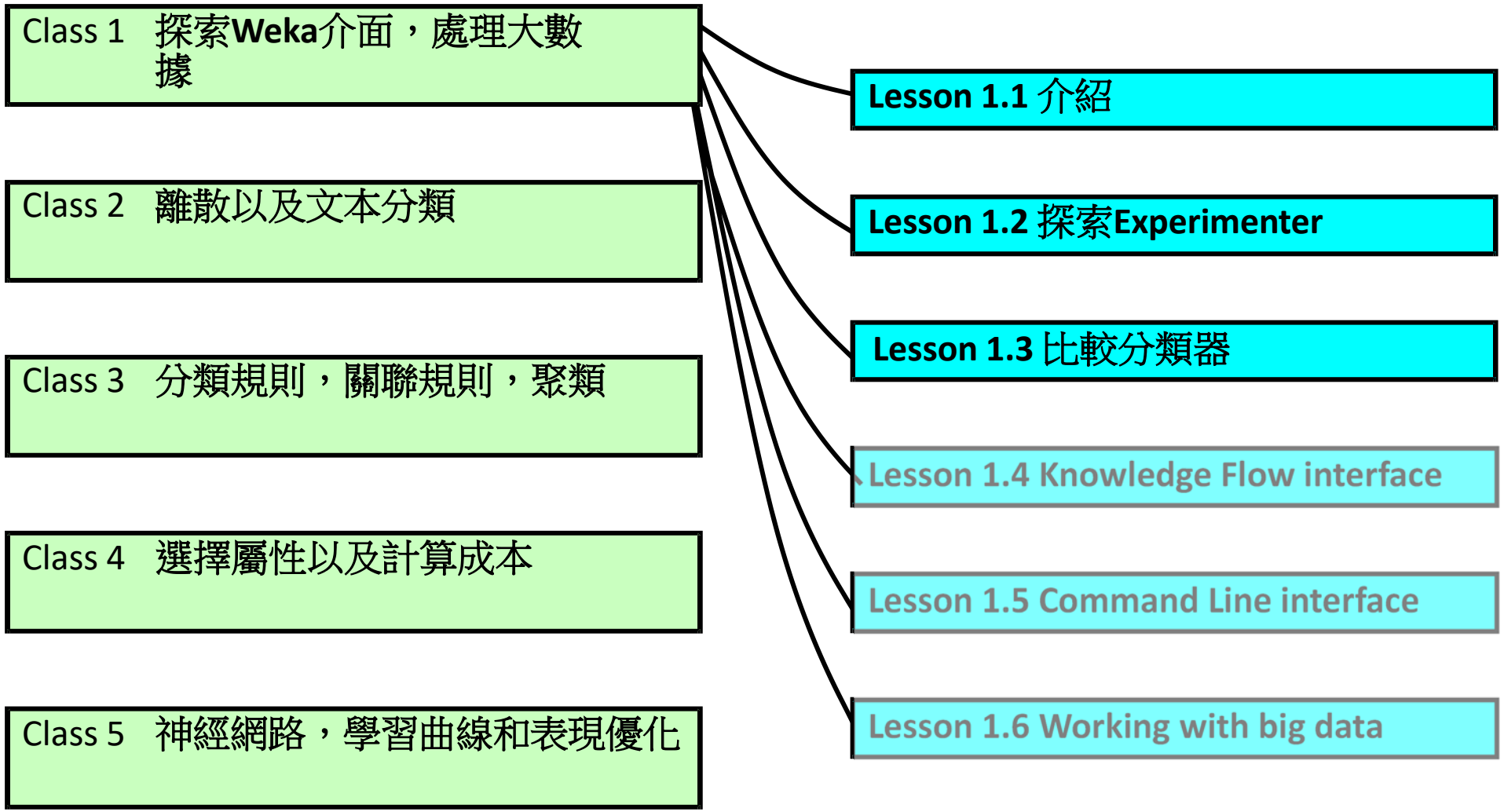
比較分類器

Ian H. Witten

Department of Computer Science
University of Waikato
New Zealand

weka.waikato.ac.nz

Lesson 1.3: 比較分類器



Lesson 1.3: 比較分類器

針對Iris資料集，J48是否優於ZeroR或OneR？

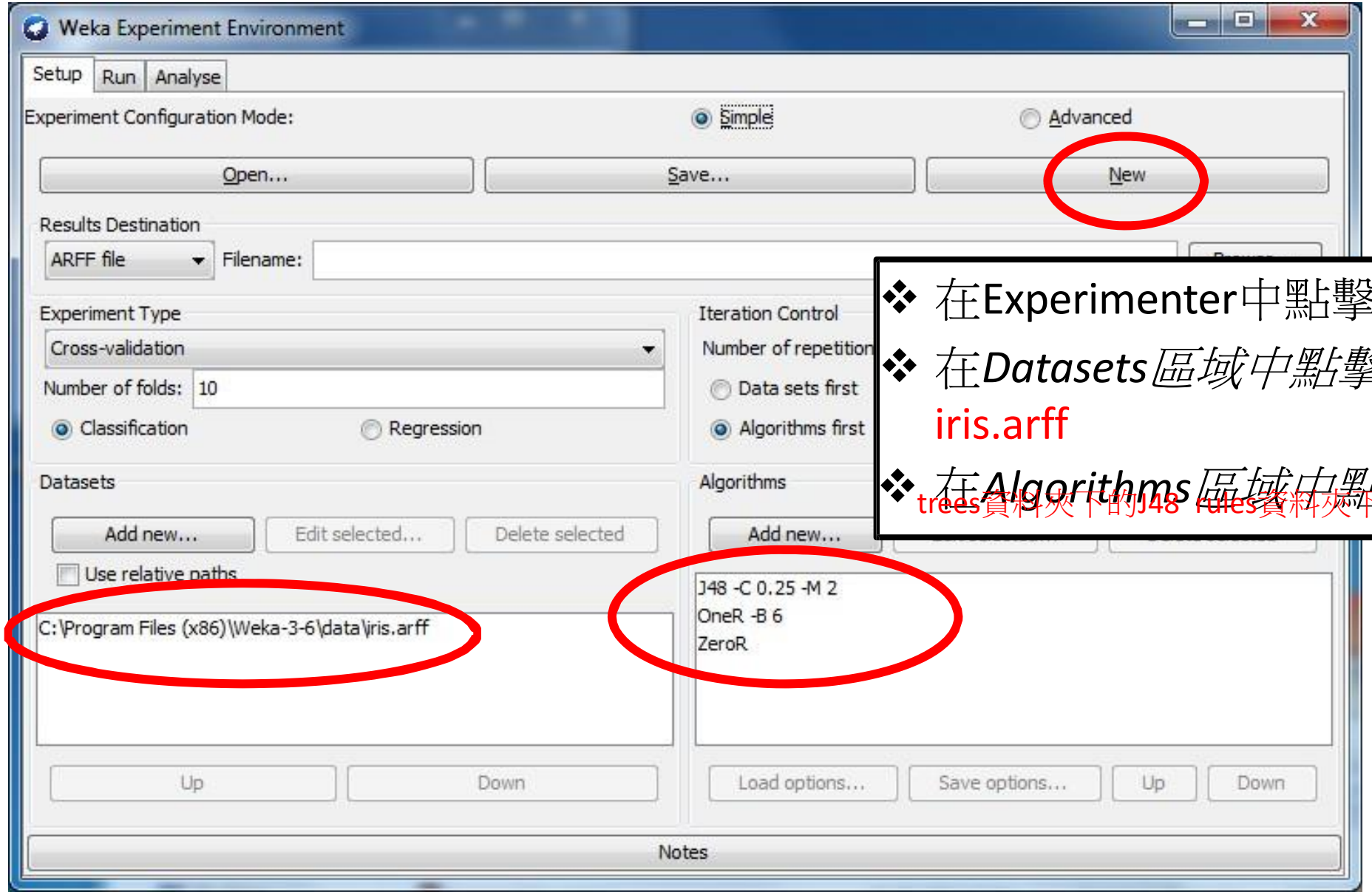
- 在Explorer中開啟iris.arff檔案
- 使用交叉驗證，並用下列分類器評估分類器準確率

ZeroR (rules>ZeroR)	33%
OneR (rules>OneR)	92%
J48 (trees>J48)	96%

但對比結果的可信度是多少？

如果我們選擇不同的隨機種子結果會不同...

Lesson 1.3: 比較分類器



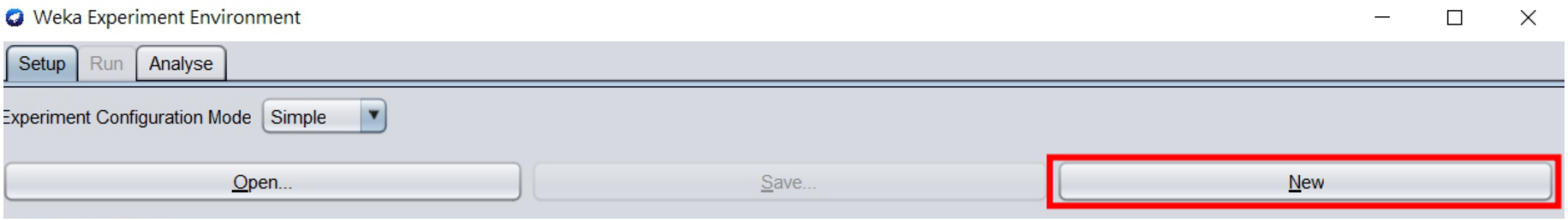
- 在Experimenter中點擊*New*鈕
- 在*Datasets*區域中點擊*Add new*以開啟iris.arff
- 在*Algorithms*區域中點擊*Add new*開啟trees資料夾下的J48、rules資料夾中的OneR、rules資料夾下的ZeroR

Lesson 1.3: 比較分類器

1. 開啟Weka程式，於Weka GUI Chooser介面左鍵單擊Experimenter按鈕

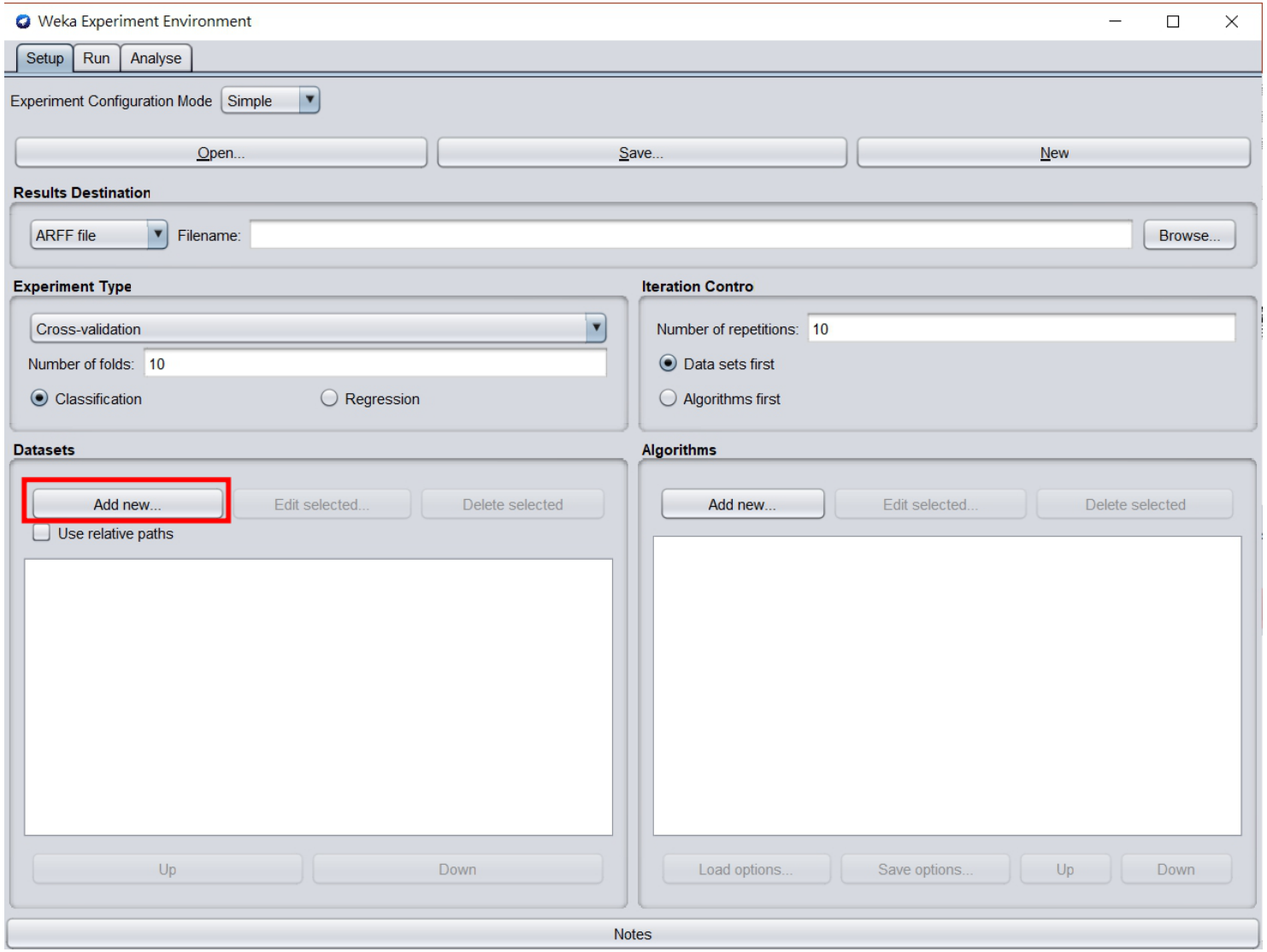
Lesson 1.3: 比較分類器

2. 在Setup面板，以左鍵單擊New鈕，新增一個實驗



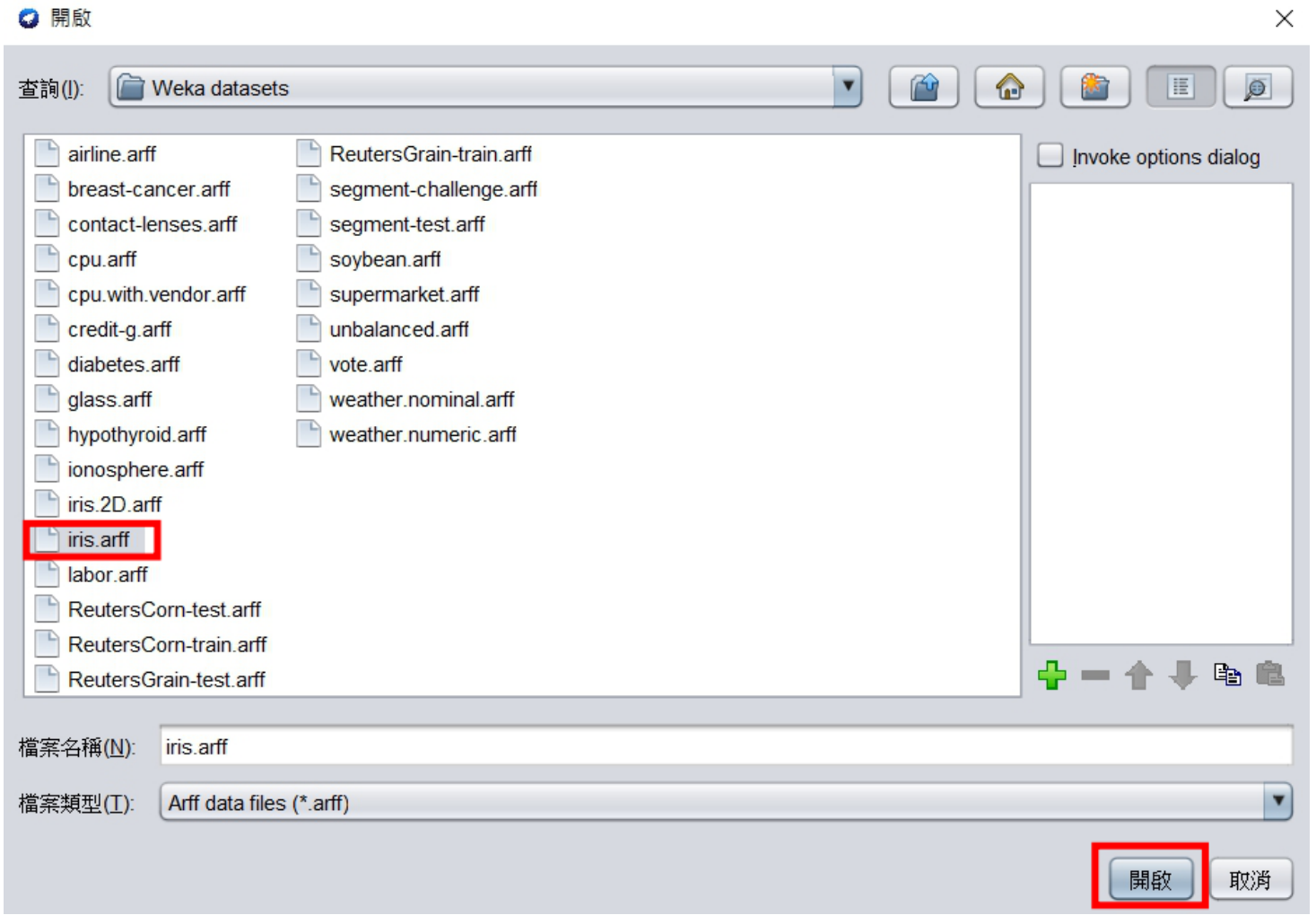
Lesson 1.3: 比較分類器

3. 左鍵單擊圖中紅框中的Add new...鈕，新增資料集



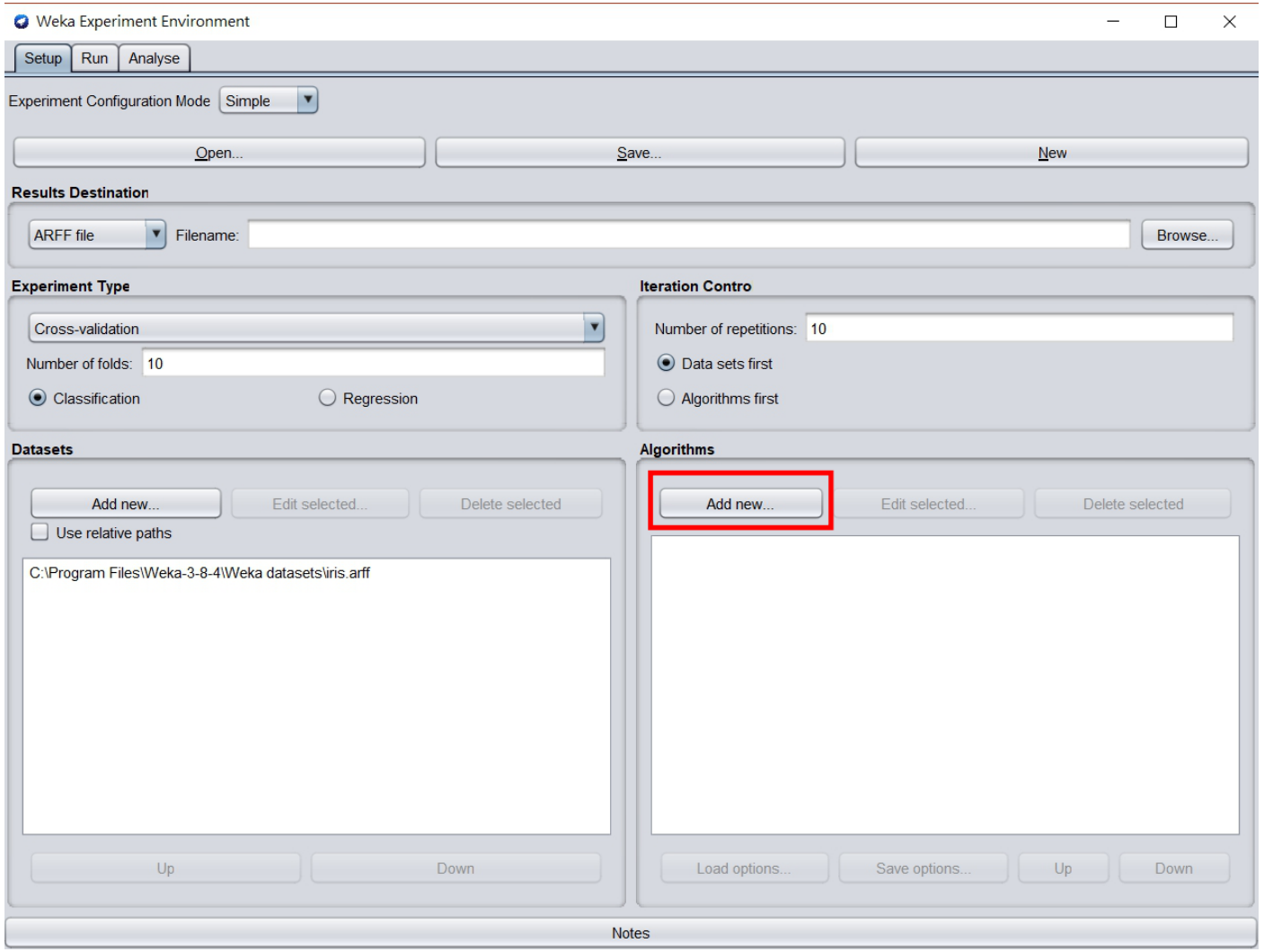
Lesson 1.3: 比較分類器

4.進入自行複製的Weka datasets資料夾，左鍵單擊iris.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集



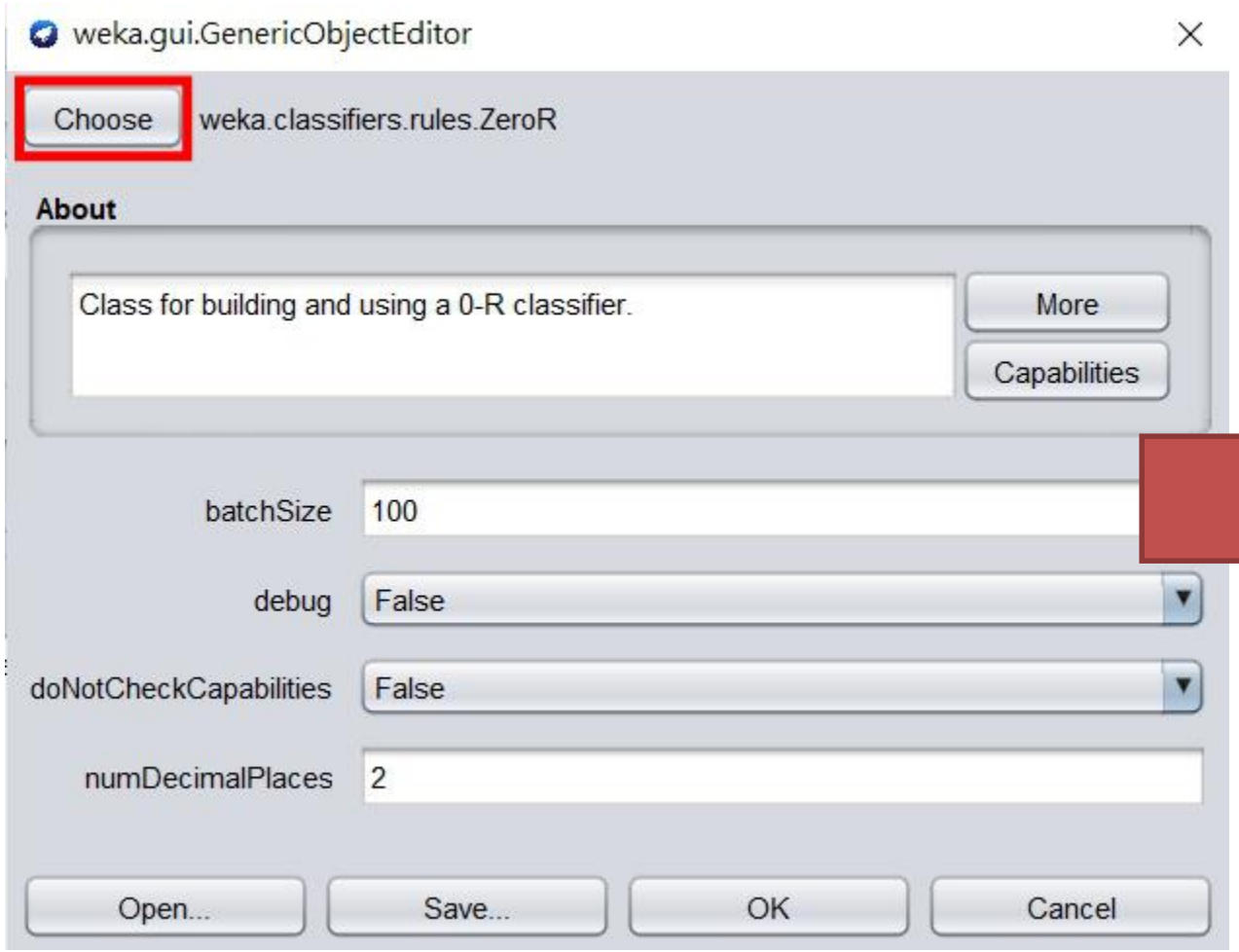
Lesson 1.3: 比較分類器

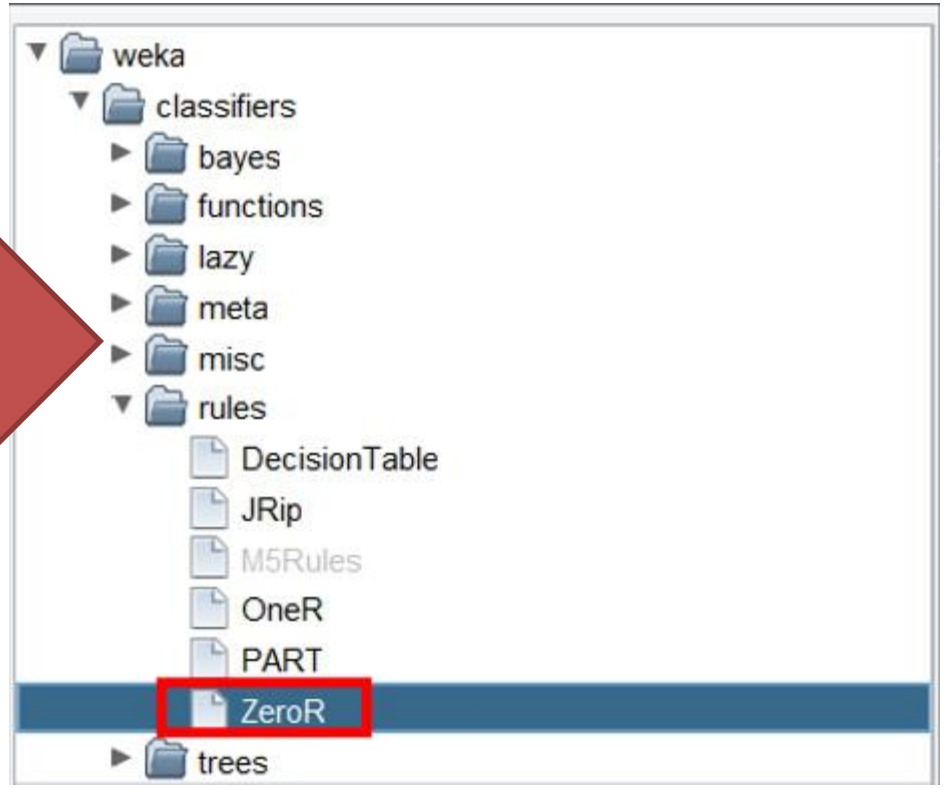
5.左鍵單擊Algorithms區域內的Add new...按鈕



Lesson 1.3: 比較分類器

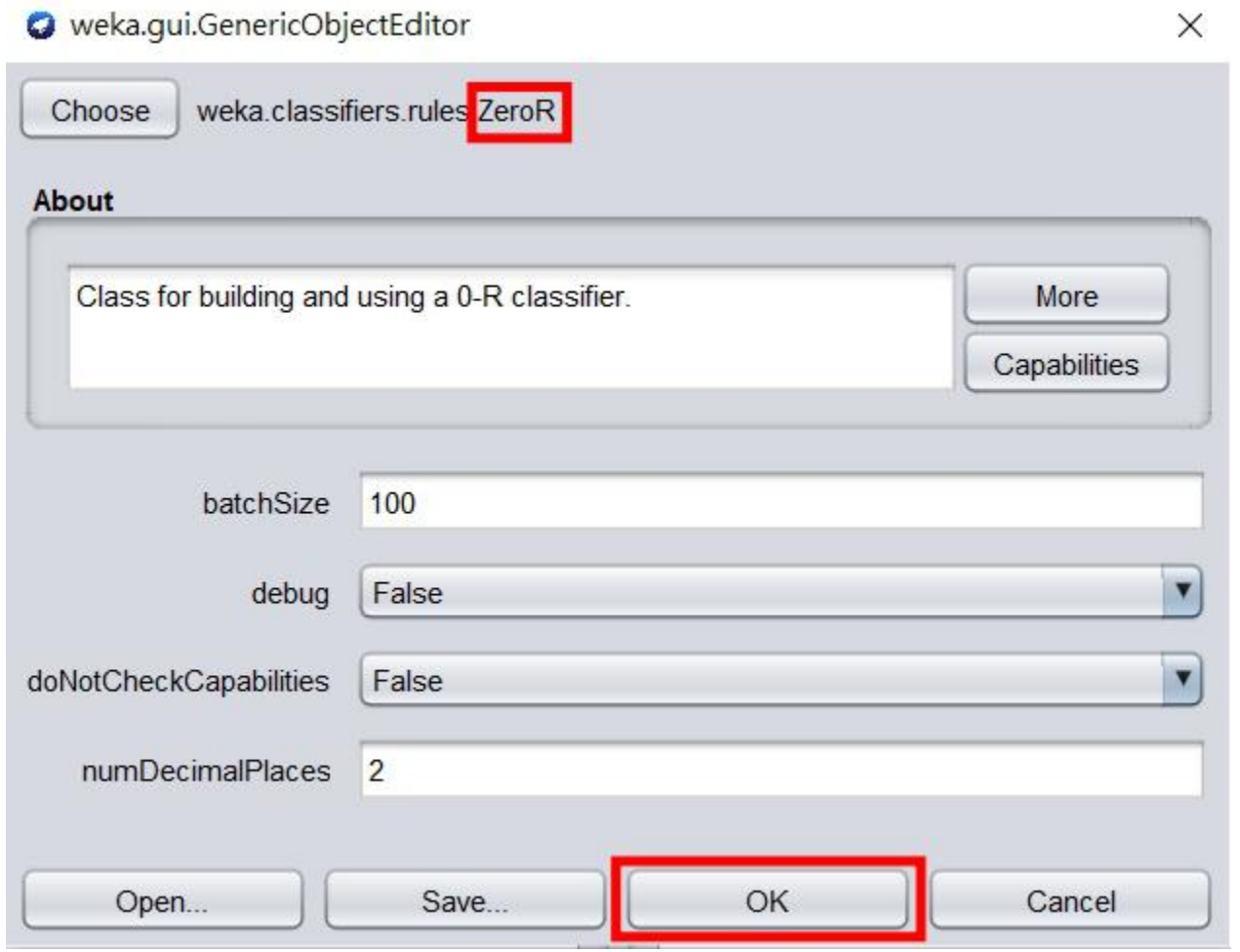
6.在出現的視窗中左鍵單擊Choose按鈕，並在出現的選單中選擇rules資料夾下的ZeroR分類器





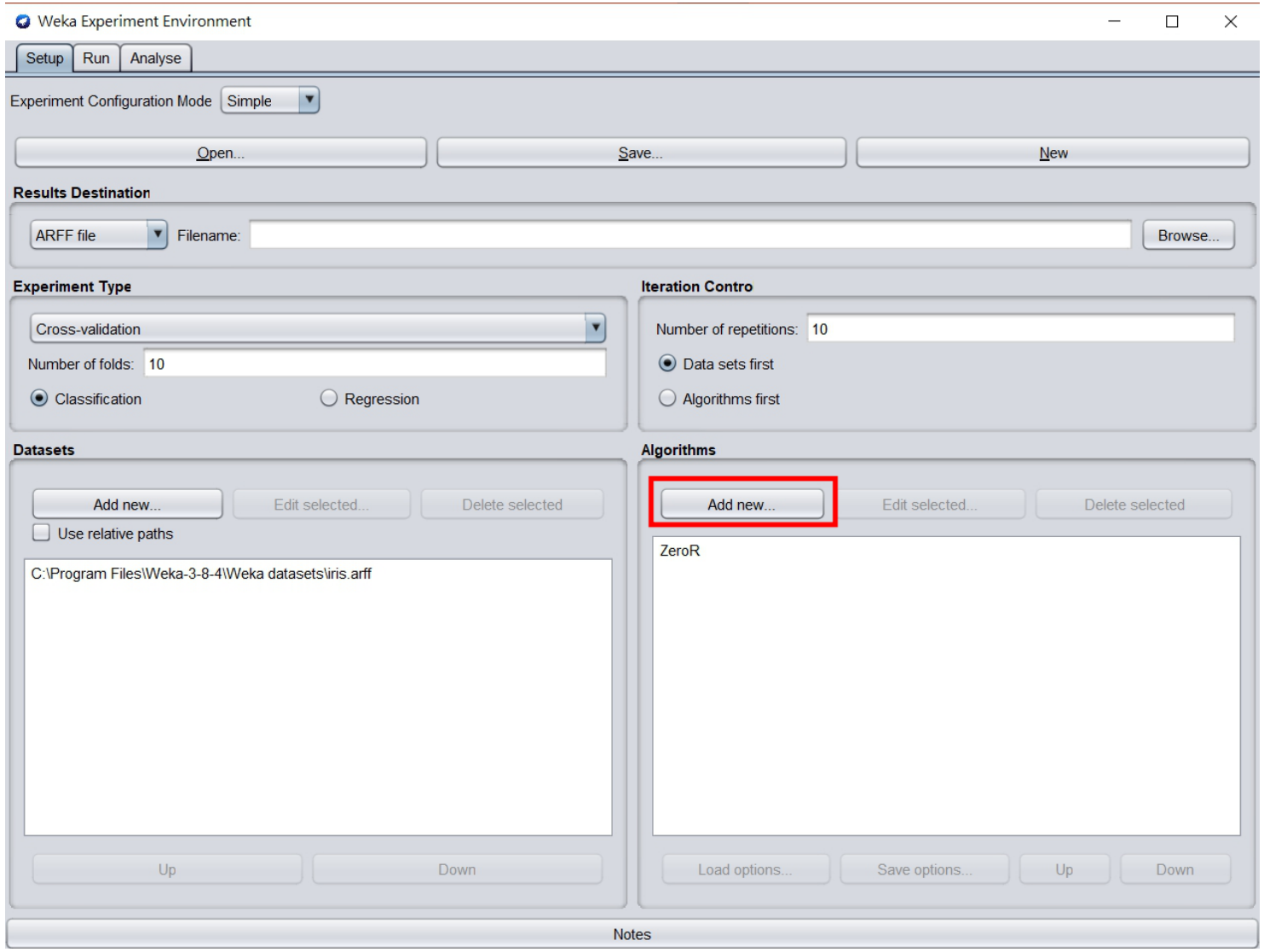
Lesson 1.3: 比較分類器

7.確認選擇ZeroR分類器後按下視窗下方OK按鈕



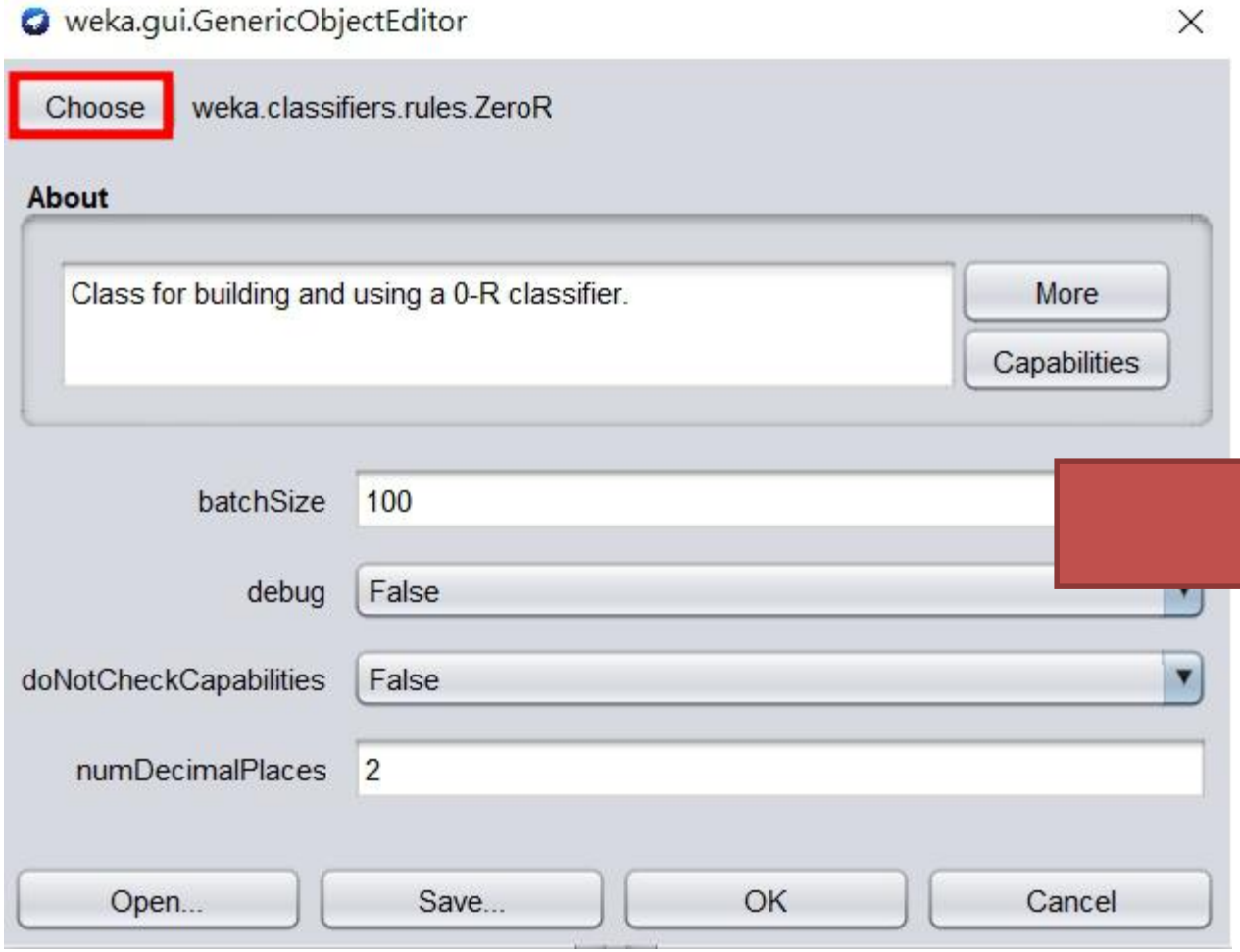
Lesson 1.3: 比較分類器

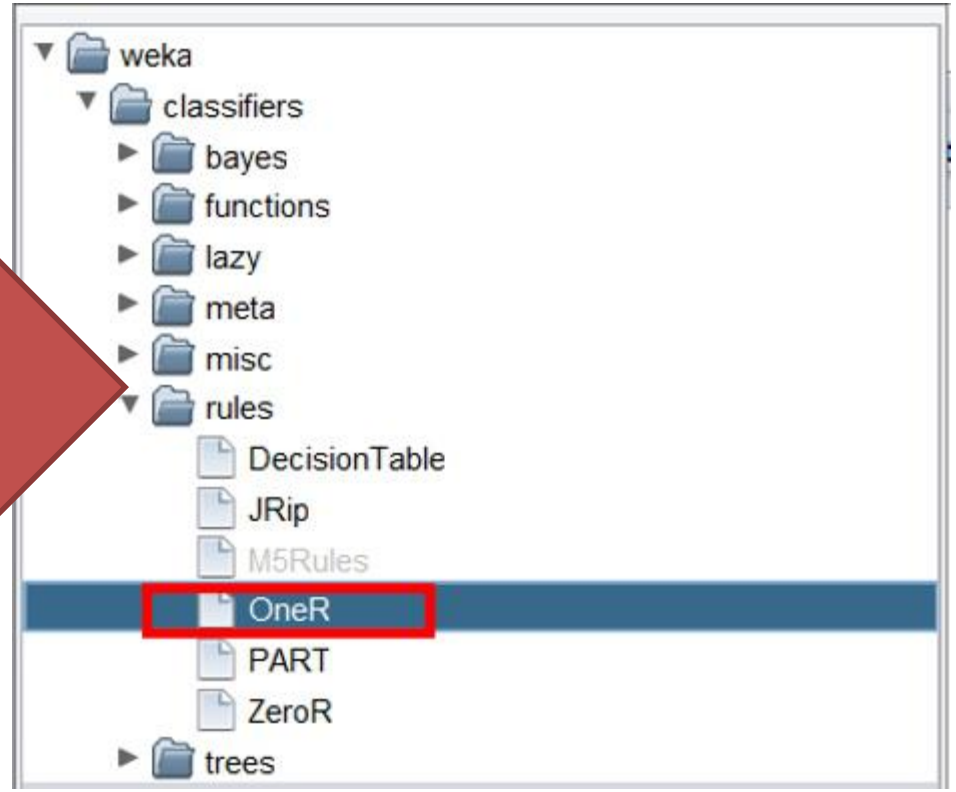
8. 再次以左鍵單擊Algorithms區域內的Add new...按鈕



Lesson 1.3: 比較分類器

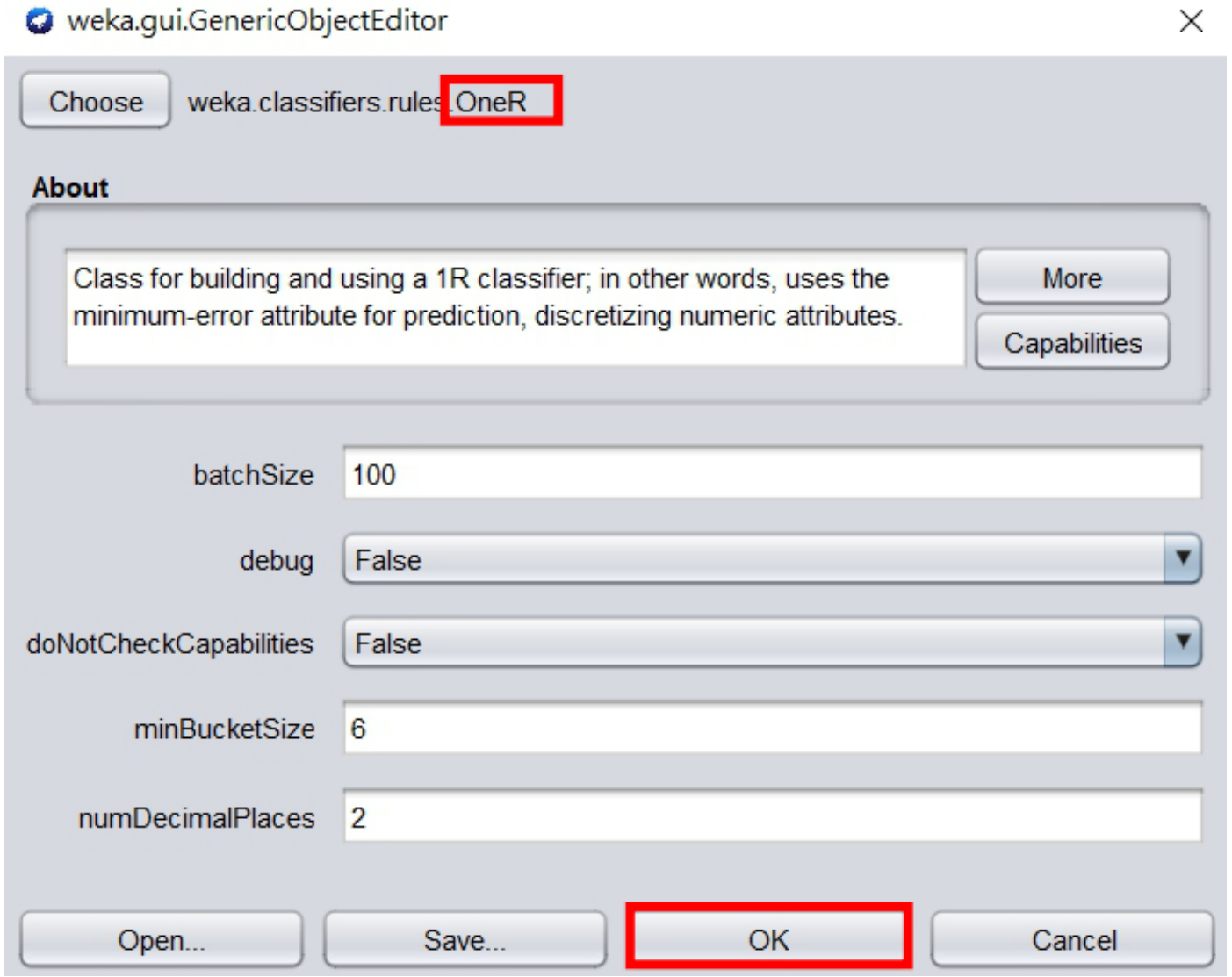
9.在出現的視窗中左鍵單擊Choose按鈕，並在出現的選單中選擇rules資料夾下的OneR分類器





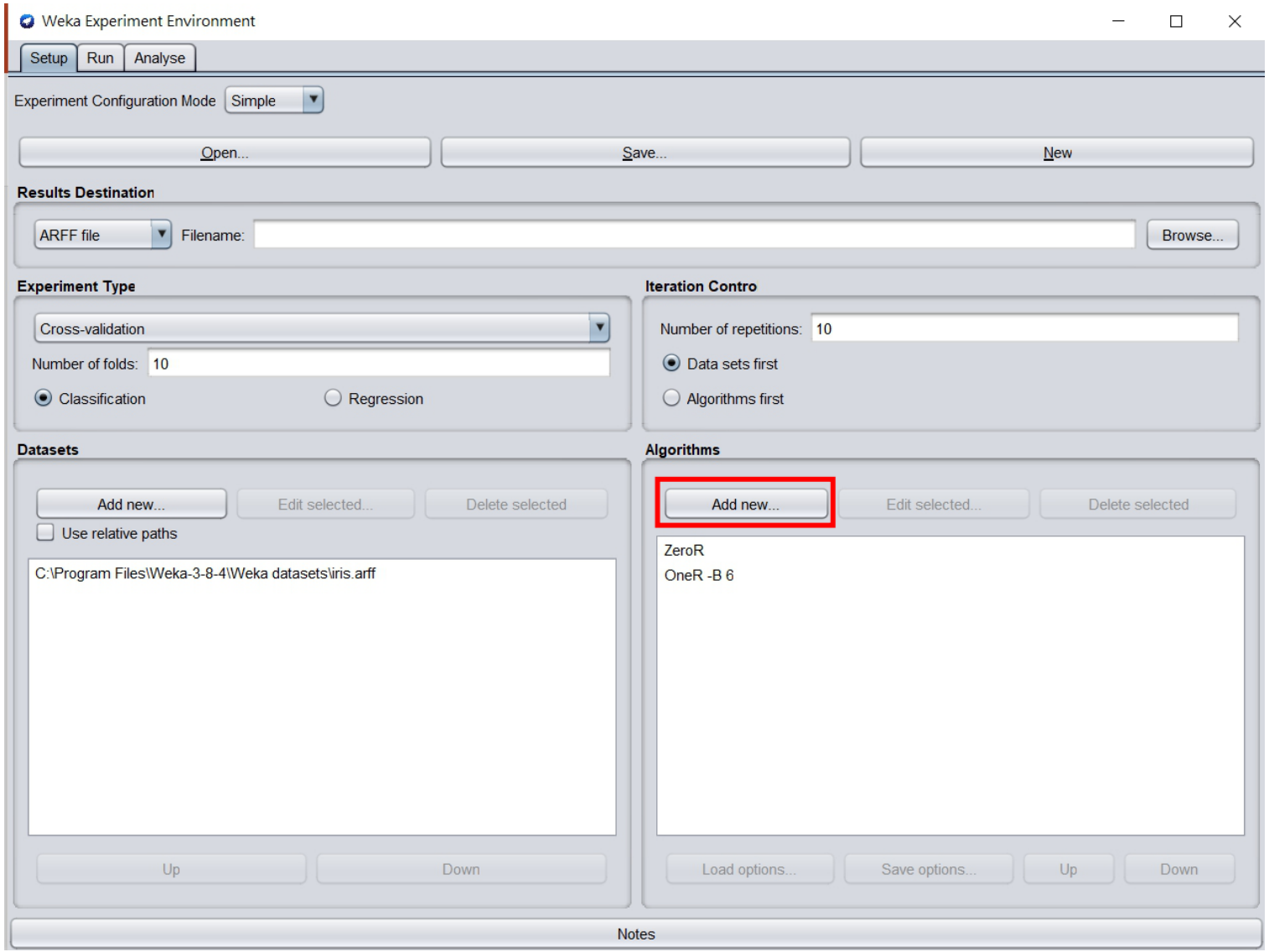
Lesson 1.3: 比較分類器

10.確認選擇OneR分類器後按下視窗下方OK按鈕



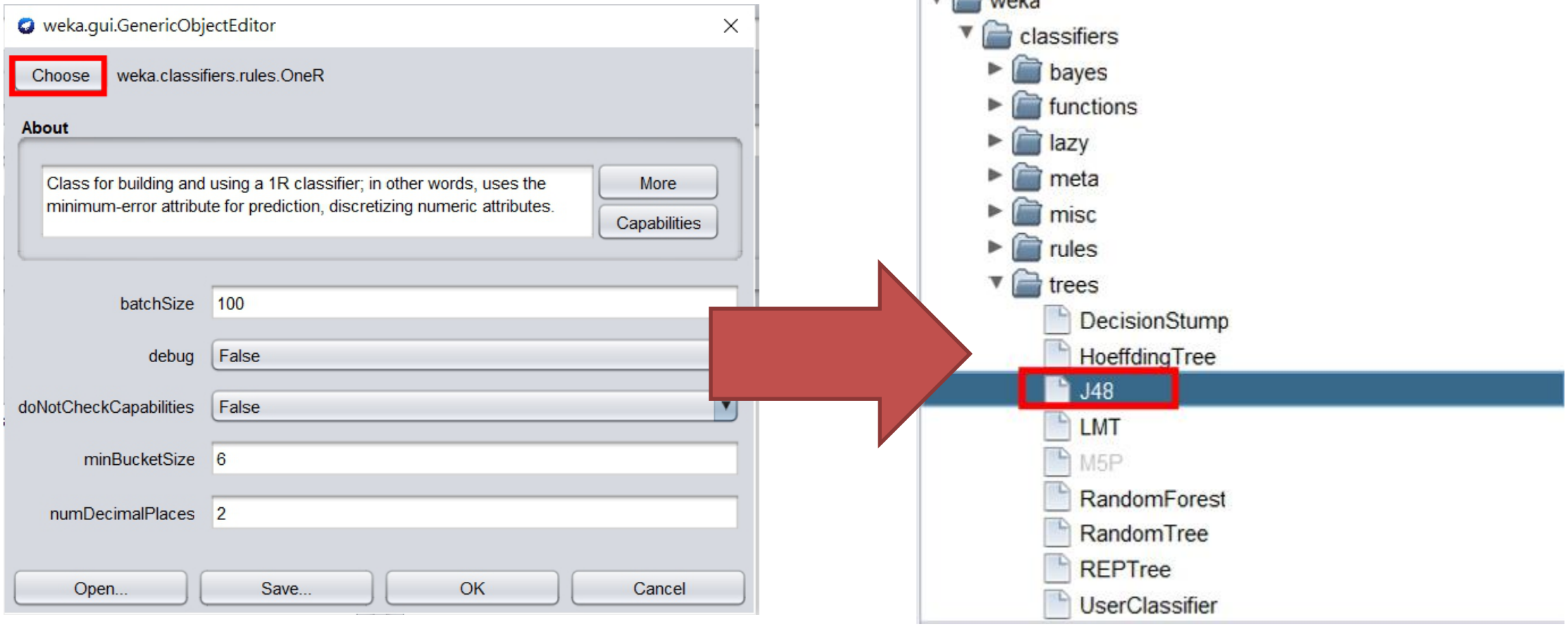
Lesson 1.3: 比較分類器

11. 再次以左鍵單擊Algorithms區域內的Add new...按鈕



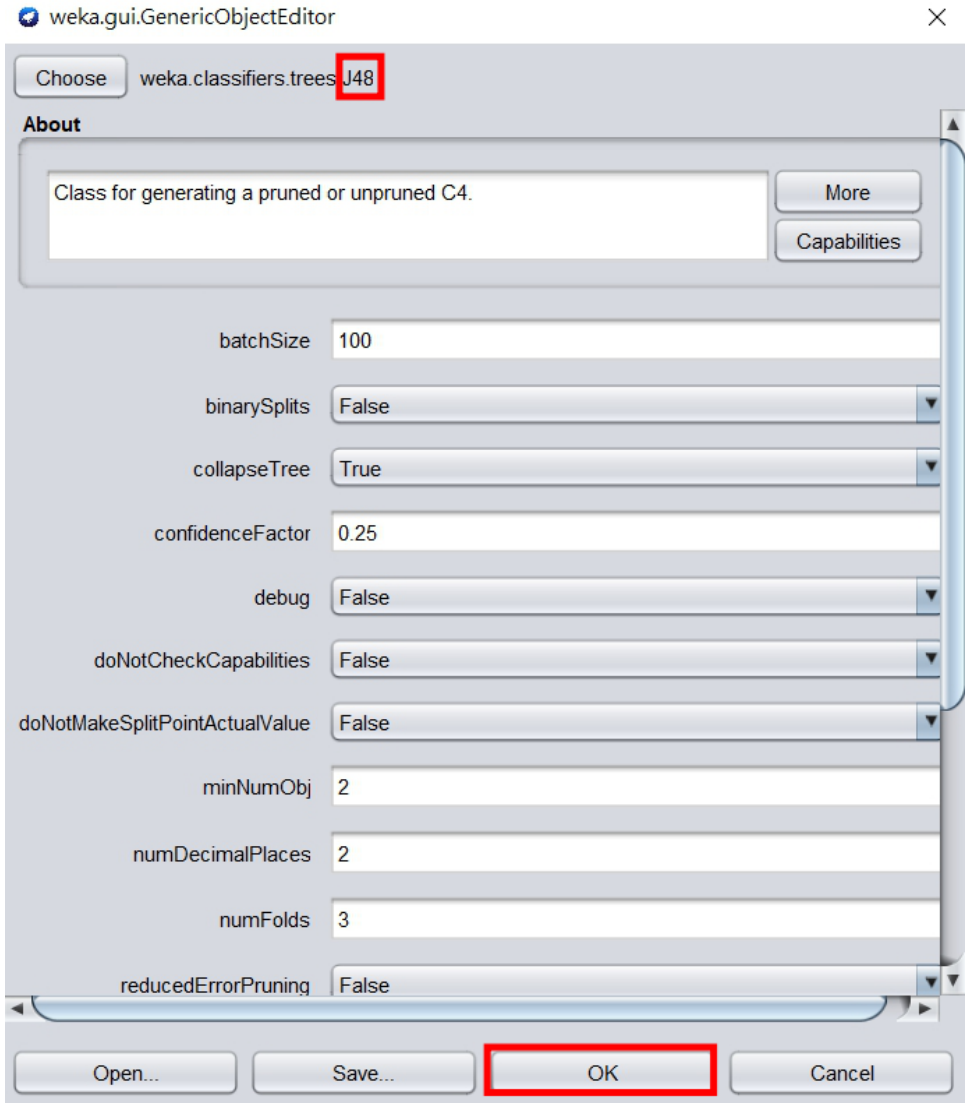
Lesson 1.3: 比較分類器

12.在出現的視窗中左鍵單擊Choose按鈕，並在出現的選單中選擇trees資料夾下的J48分類器



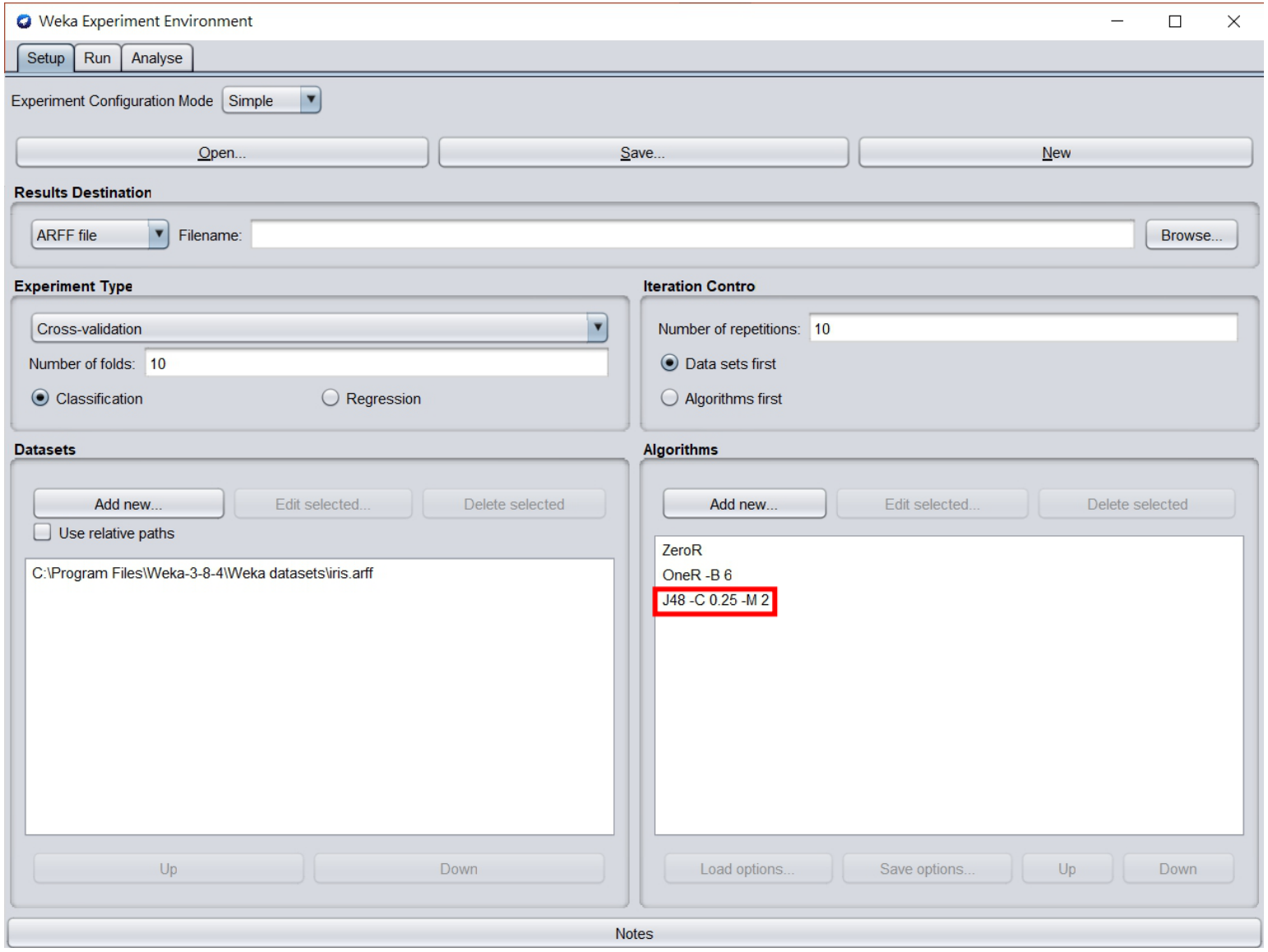
Lesson 1.3: 比較分類器

13.確認選擇J48分類器後按下視窗下方OK按鈕



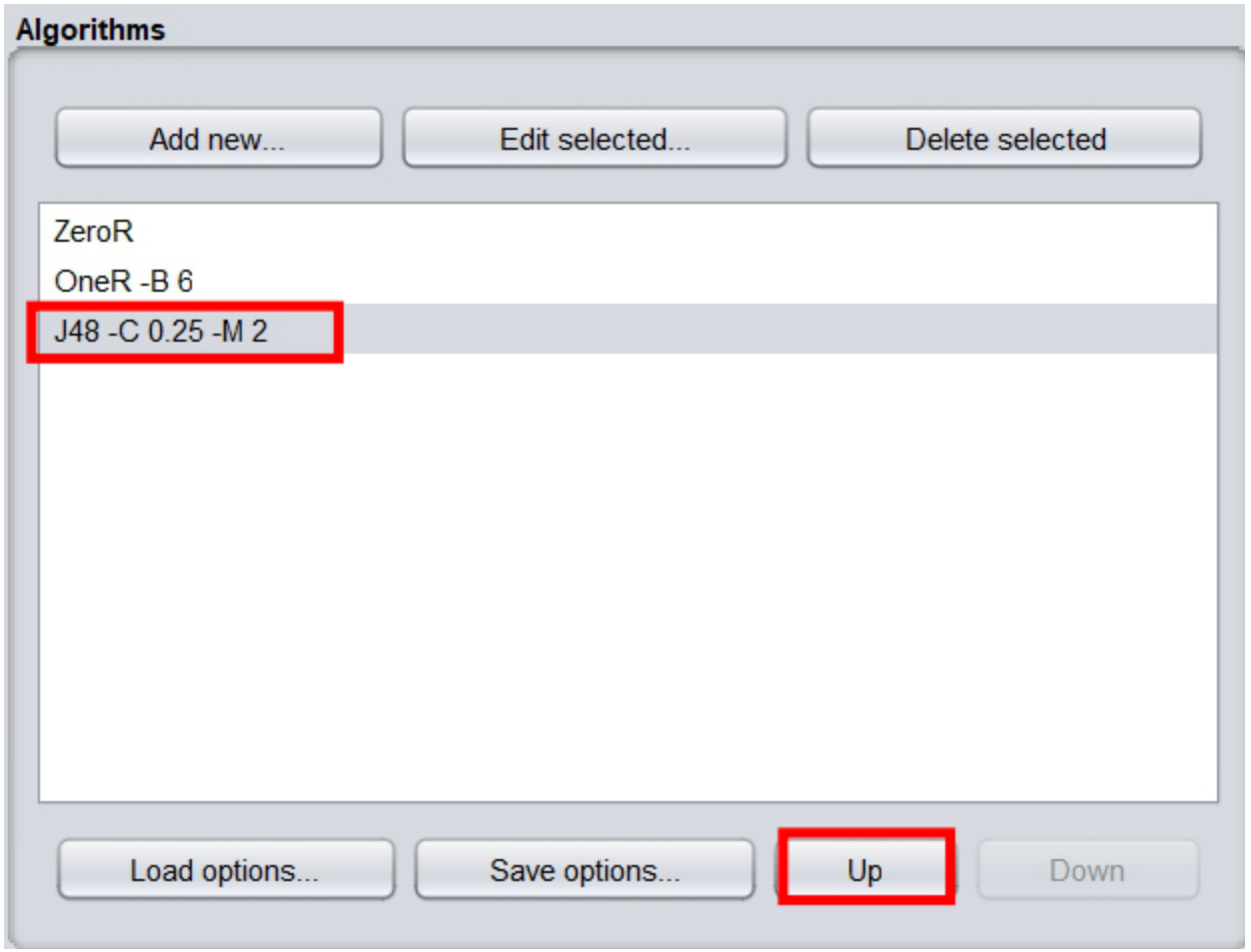
Lesson 1.3: 比較分類器

14. 在Algorithms區域內，左鍵單擊分類器列表中的J48



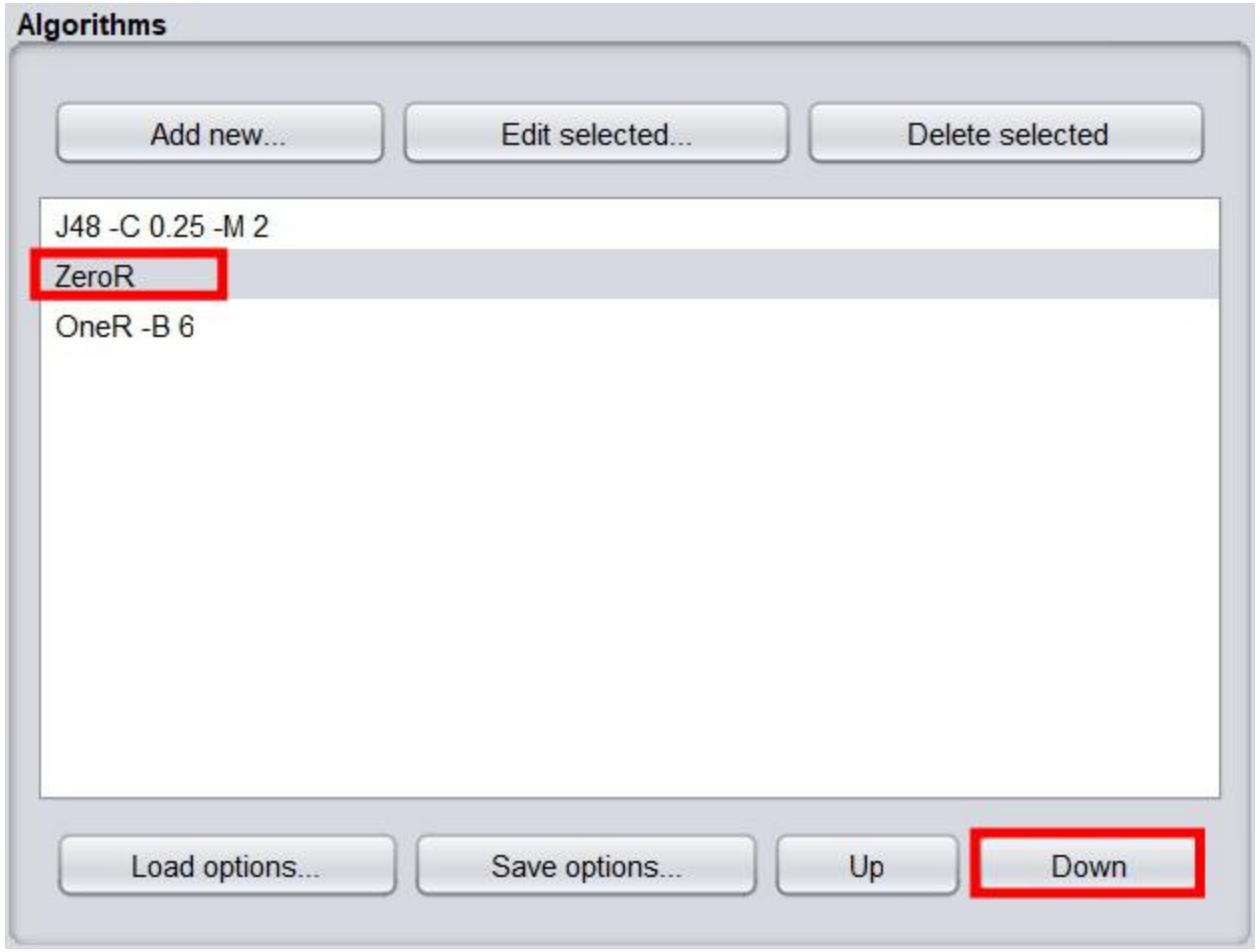
Lesson 1.3: 比較分類器

15.確認按下J48分類器後，左鍵雙擊右下方的UP按鈕



Lesson 1.3: 比較分類器

16.左鍵單擊分類器列表中的ZeroR分類器，在以左鍵單擊右下方的Down按鈕



Lesson 1.3: 比較分類器

最後分類器列表的順序應如下：

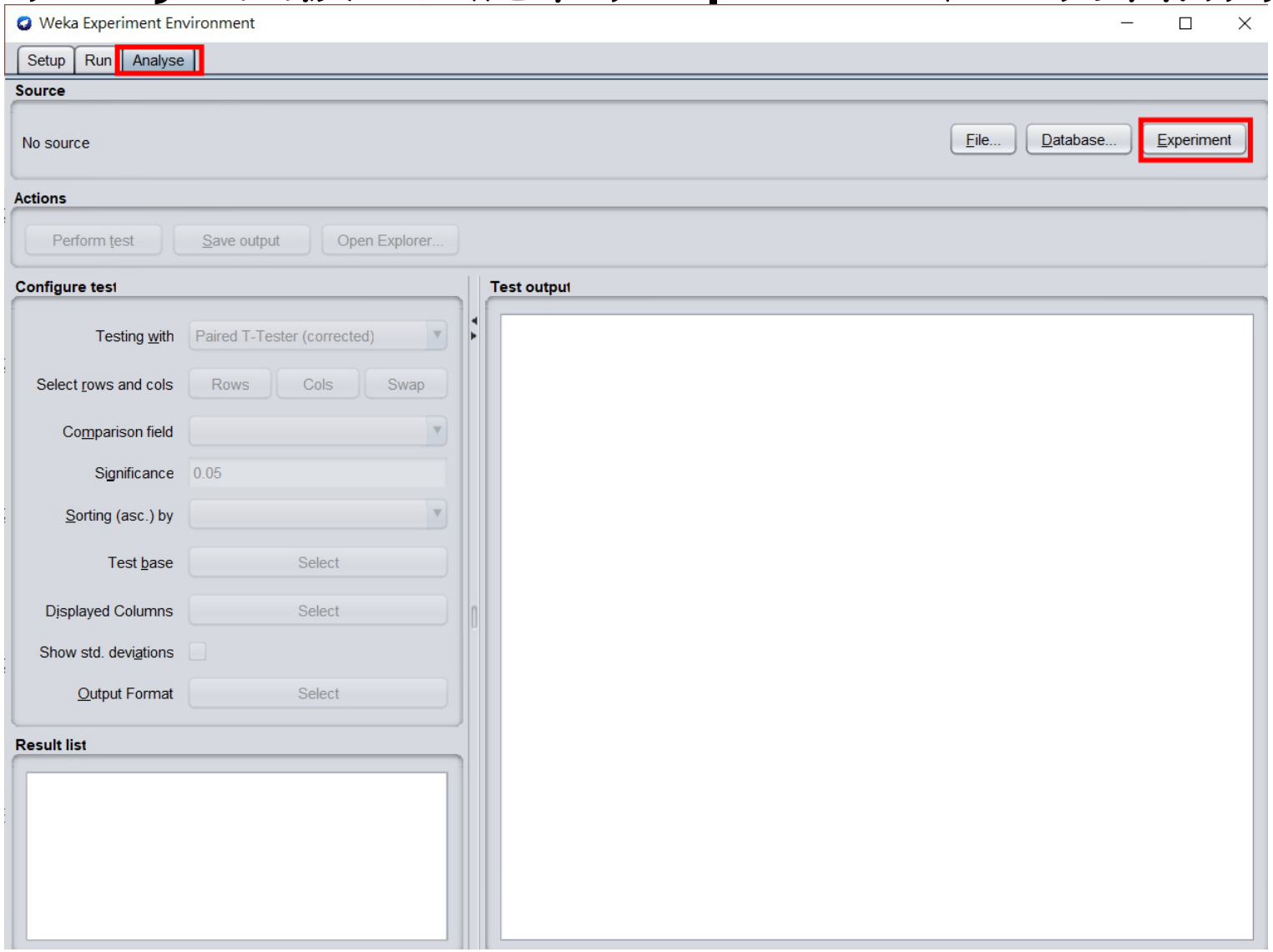
```
J48 -C 0.25 -M 2
OneR -B 6
ZeroR
```

Lesson 1.3: 比較分類器

17. 切換到Run面板，以左鍵單擊Start按鈕，並等待結果執行完畢

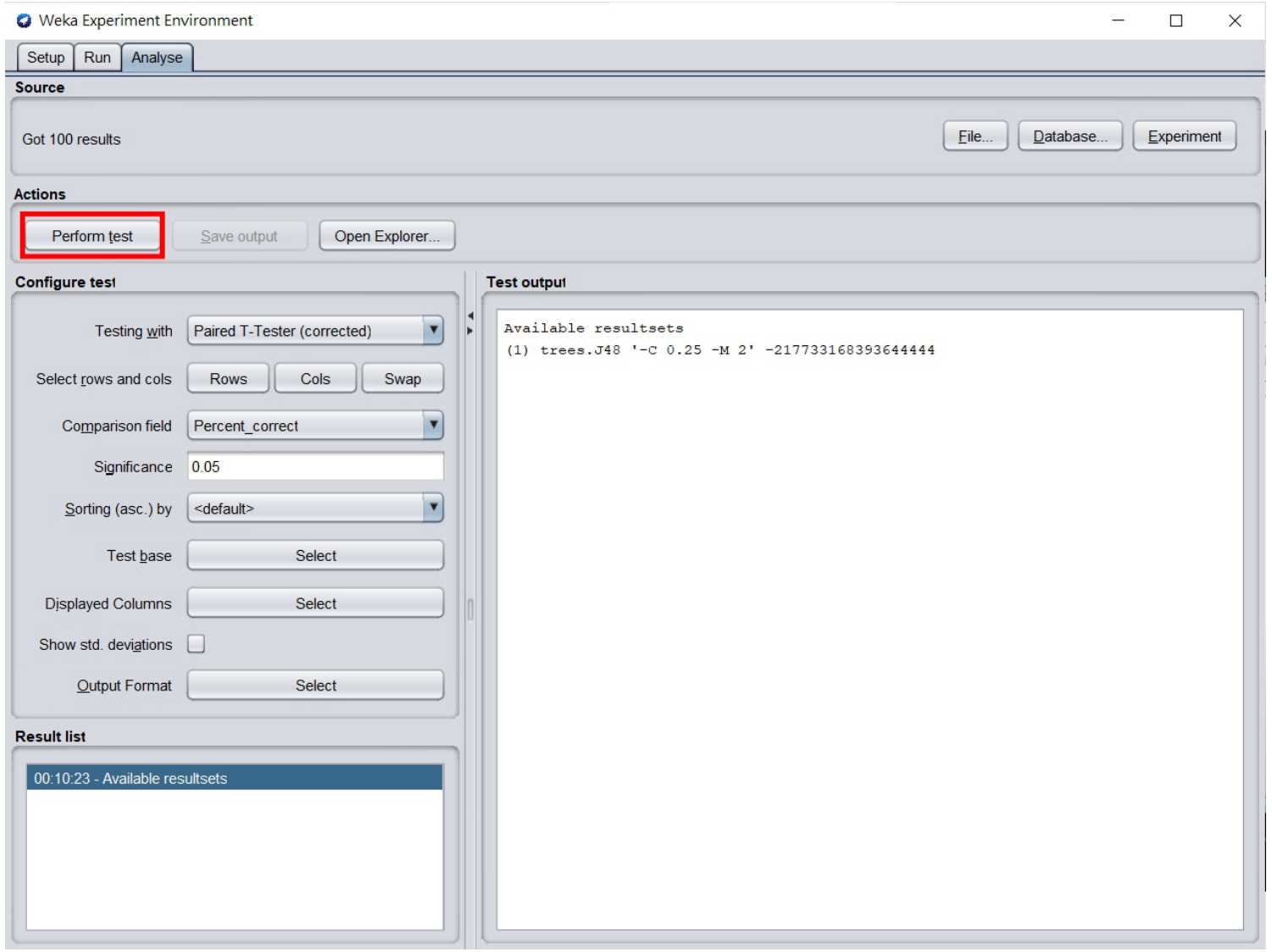
Lesson 1.3: 比較分類器

18.切換到Analyse面板，左鍵單擊Experiment鈕，分析剛才的實驗結果



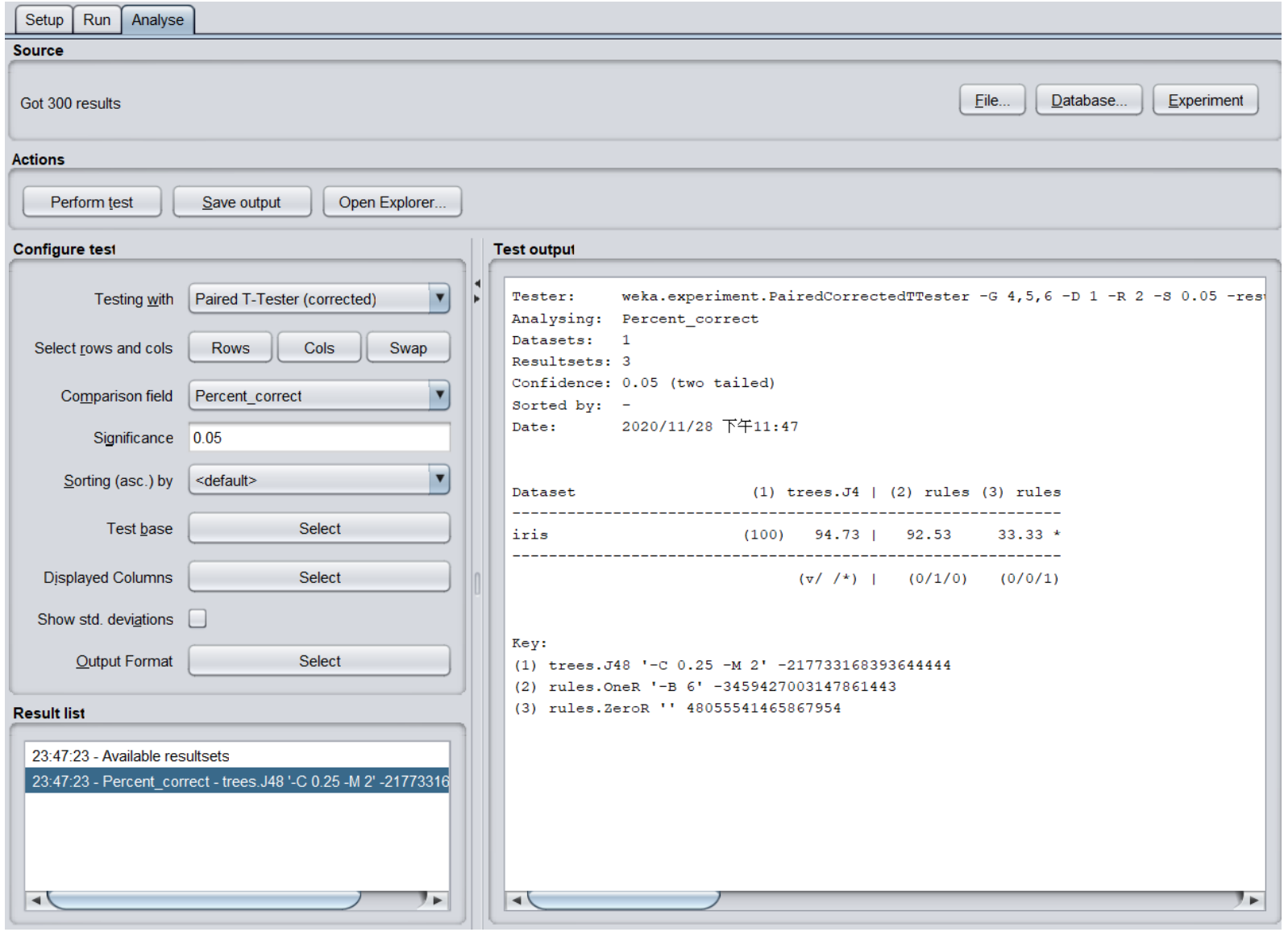
Lesson 1.3: 比較分類器

19.左鍵單擊Perform test



Lesson 1.3: 比較分類器

▼執行結果



Lesson 1.3: 比較分類器

```
Dataset                    (1) trees.J4 | (2) rules (3) rules
-------------------------------------------------------------
iris                  (100)   94.73 |   92.53     33.33 *
-------------------------------------------------------------
                            (v/ /*) |   (0/1/0)   (0/0/1)


Key:
(1) trees.J48
(2) rules.OneR
(3) rules.ZeroR
```

v 遠優於 (significantly better)

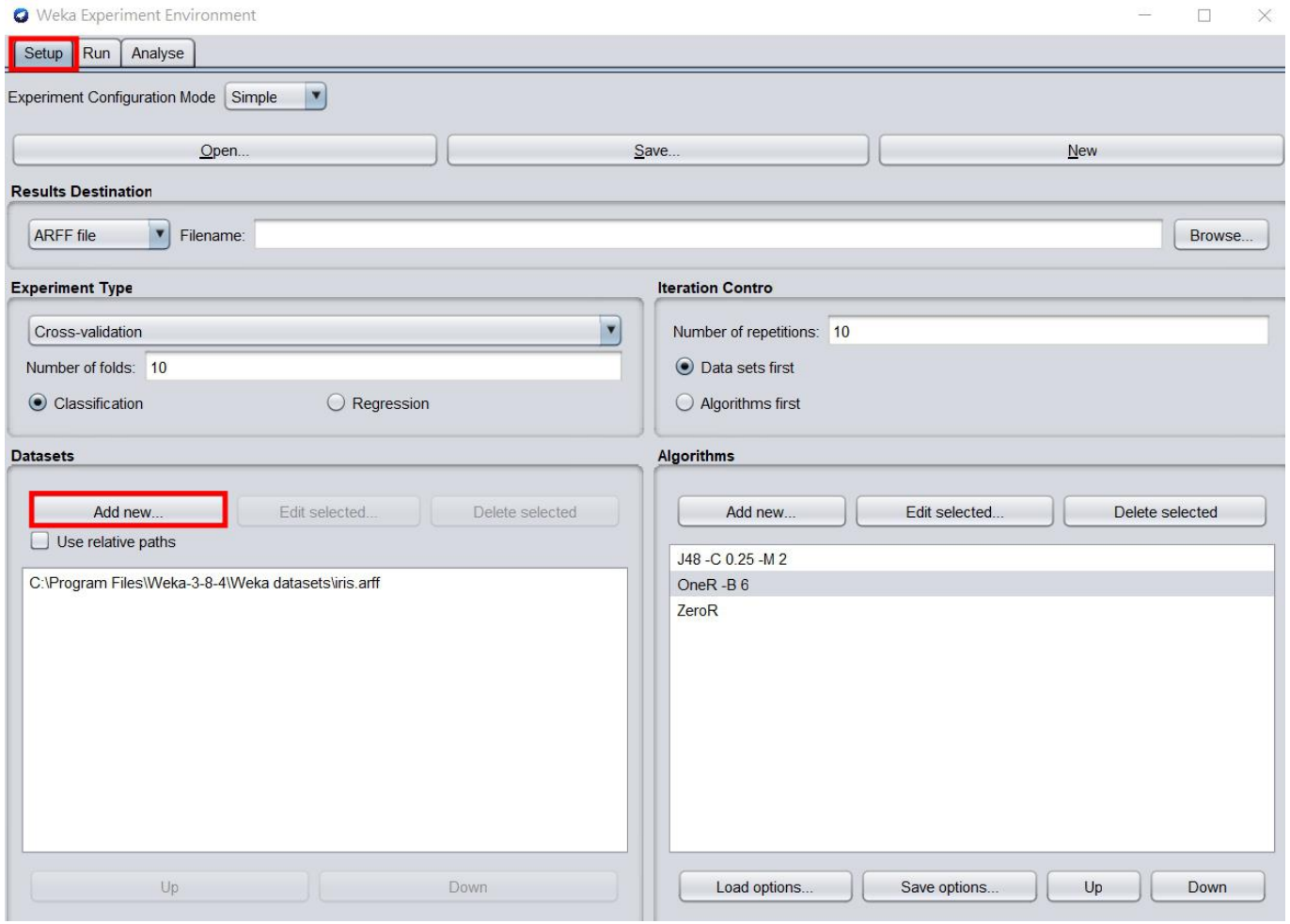
* 遠不如 (significantly worse)

- ZeroR (33.3%) 的表現結果遠不如J48 (94.7%)
- 不能斷定OneR (92.5%)遠不如 J48
- … 根據統計意義上5% level
- J48 似乎優於ZeroR: 極確定並非出自偶然
- …且優於OneR; 但可能出於偶然；不適用統計意義上的5% level

Lesson 1.3: 比較分類器

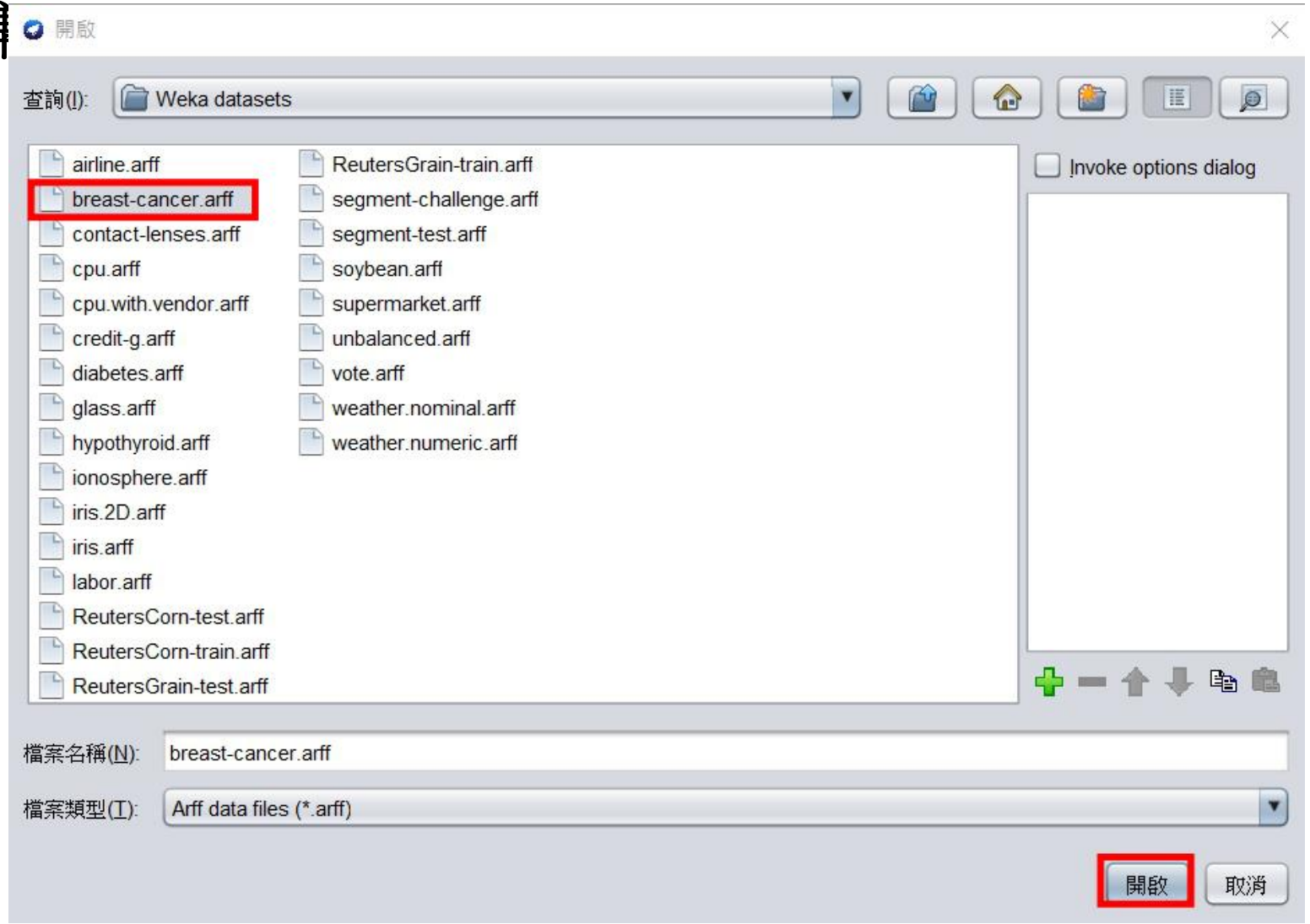
接著，我們試著導入更多的資料庫。

1. 切換到Setup面板，左鍵單擊Datasets區域內的Add new…按鈕



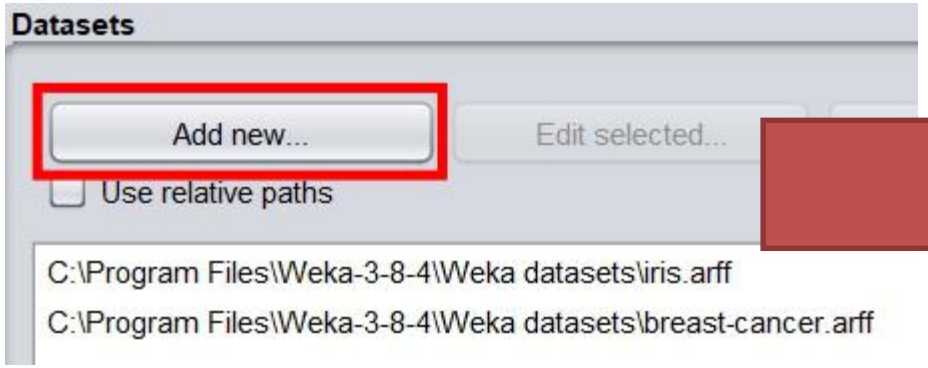
Lesson 1.3: 比較分類器

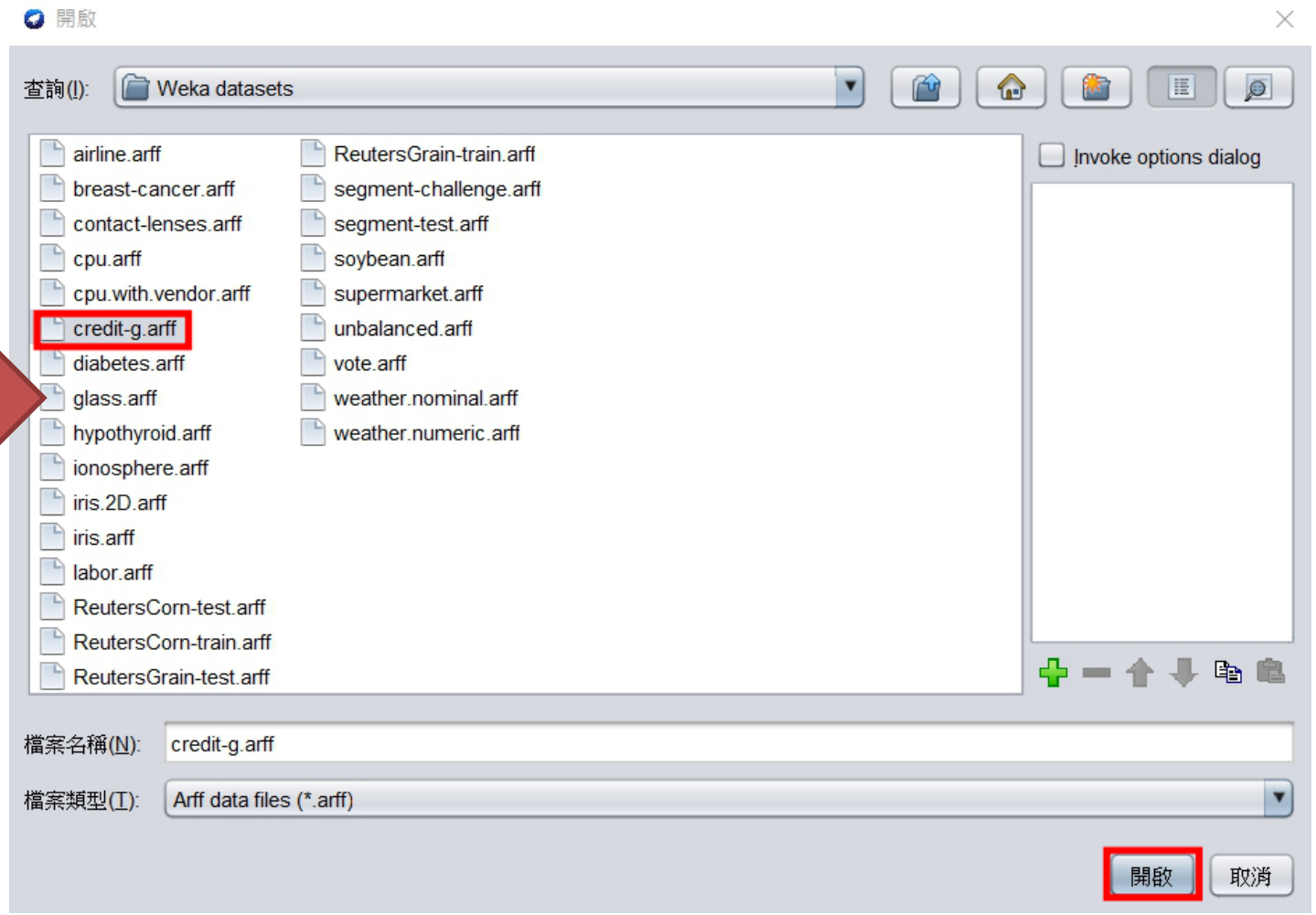
2.左鍵單擊breaset-cancer.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集



Lesson 1.3: 比較分類器

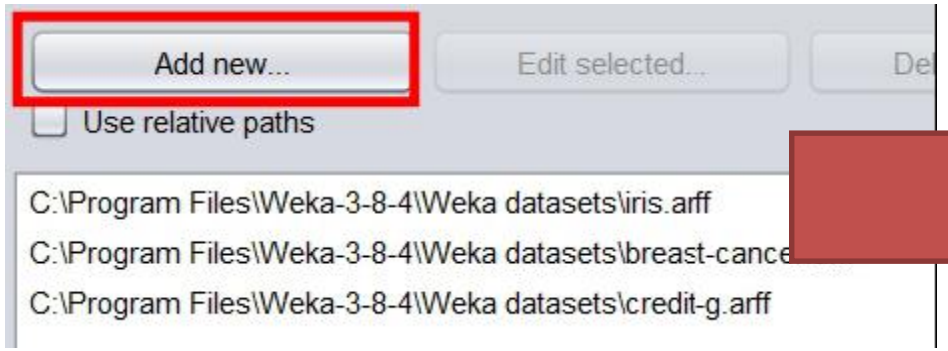
3.再次以左鍵單擊Datasets區域內的Add new...按鈕，左鍵單擊credit-g.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集

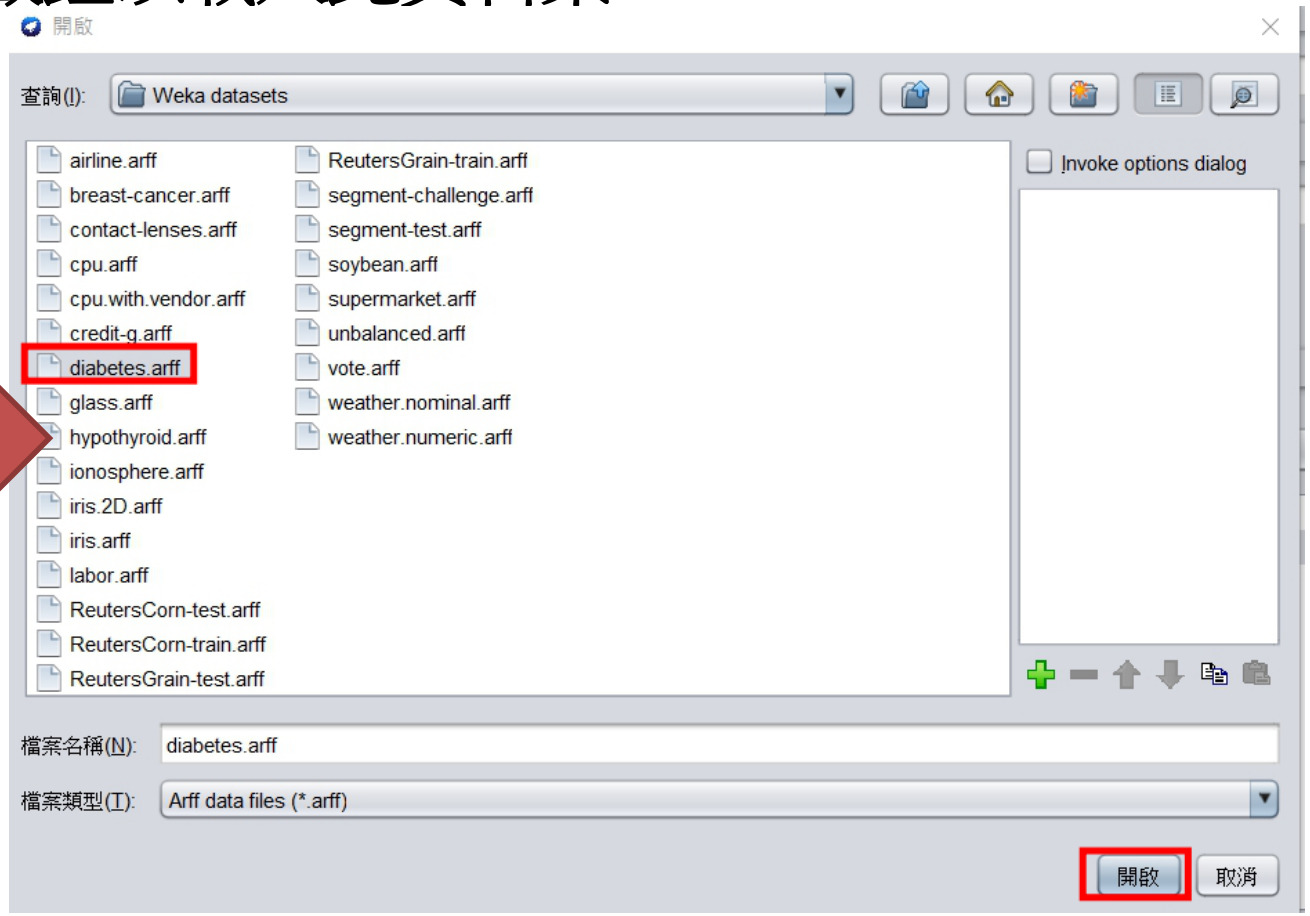




Lesson 1.3: 比較分類器

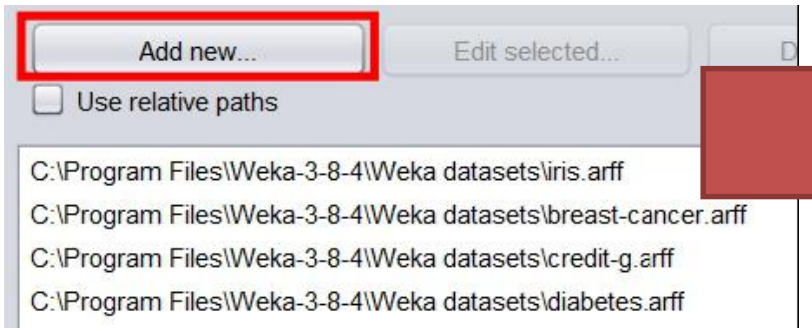
4.再次以左鍵單擊Datasets區域內的Add new...按鈕，左鍵單擊diabetes.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集

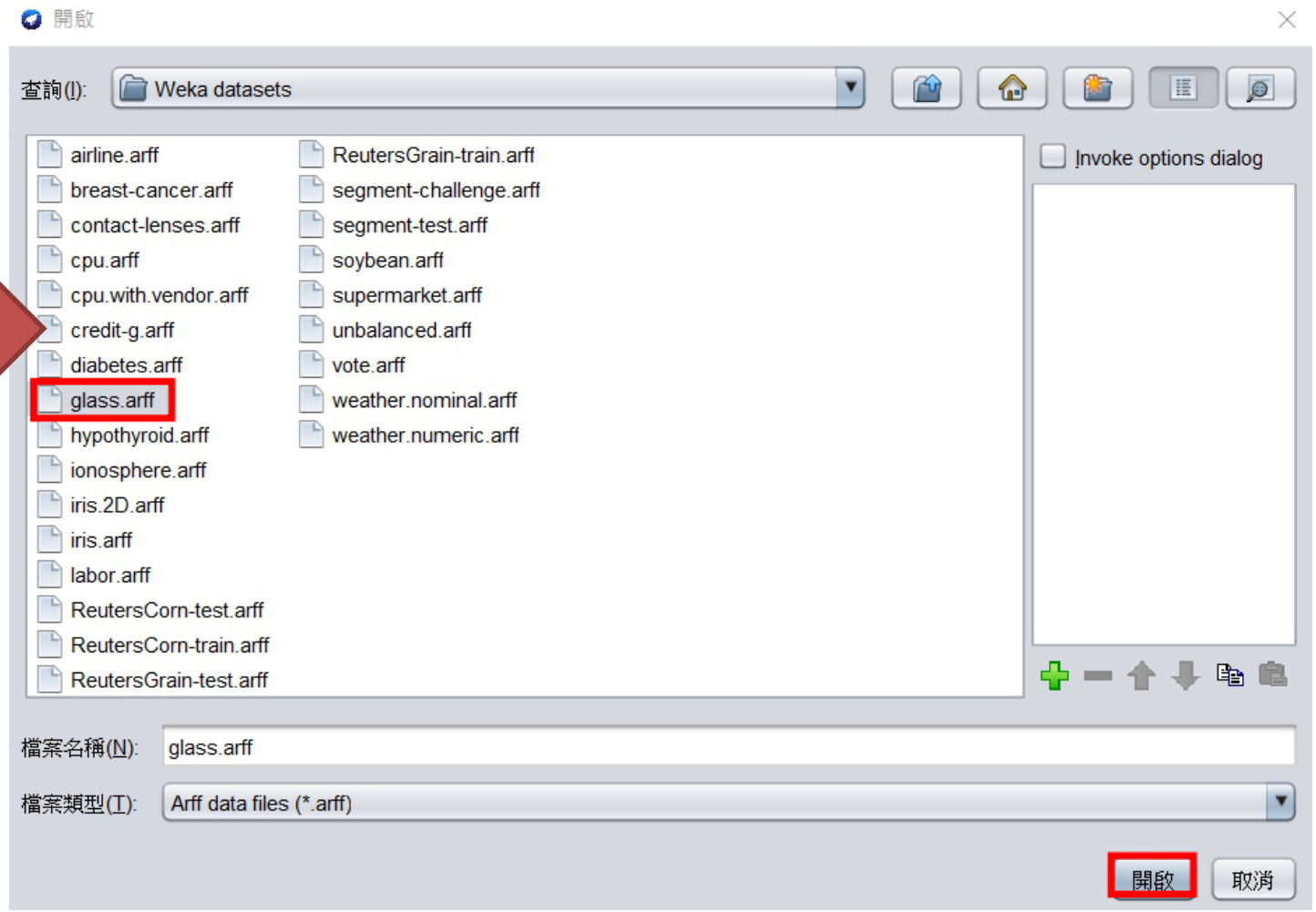




Lesson 1.3: 比較分類器

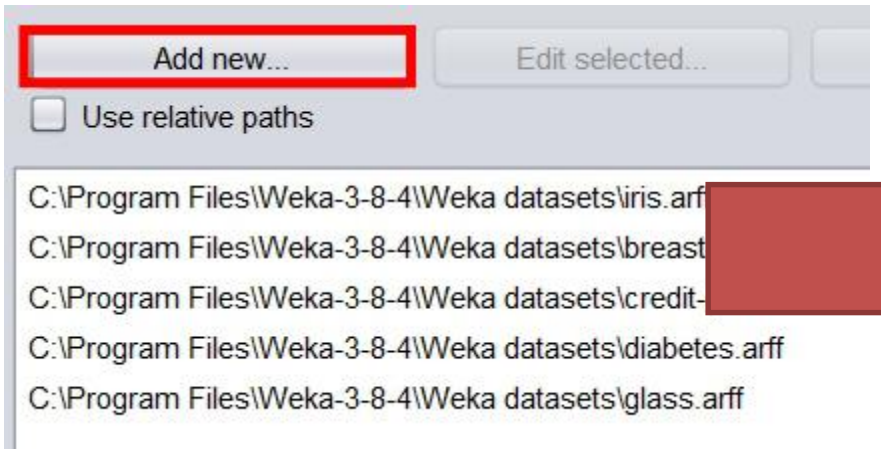
5.再次以左鍵單擊Datasets區域內的Add new...按鈕，左鍵單擊glass.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集

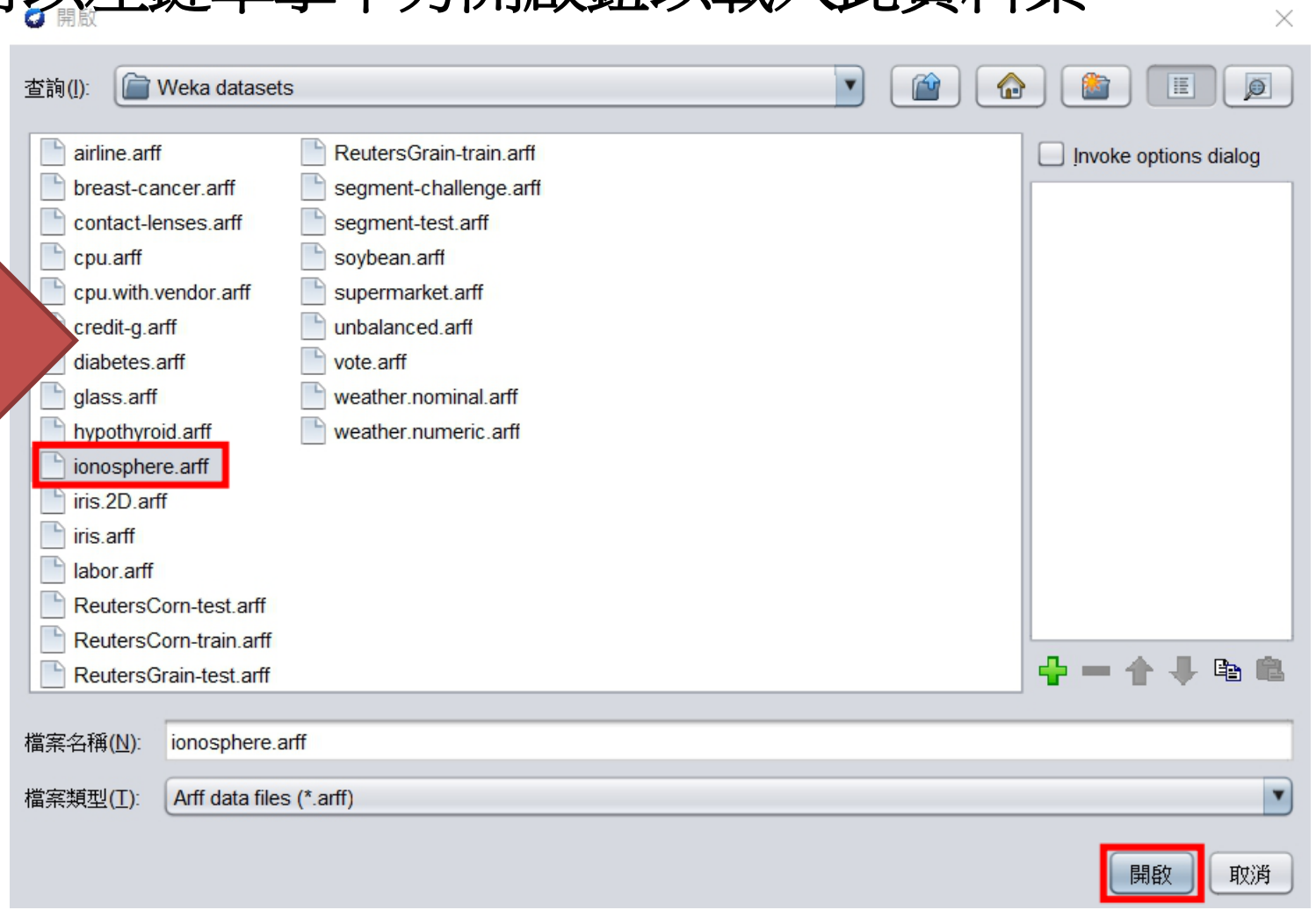




Lesson 1.3: 比較分類器

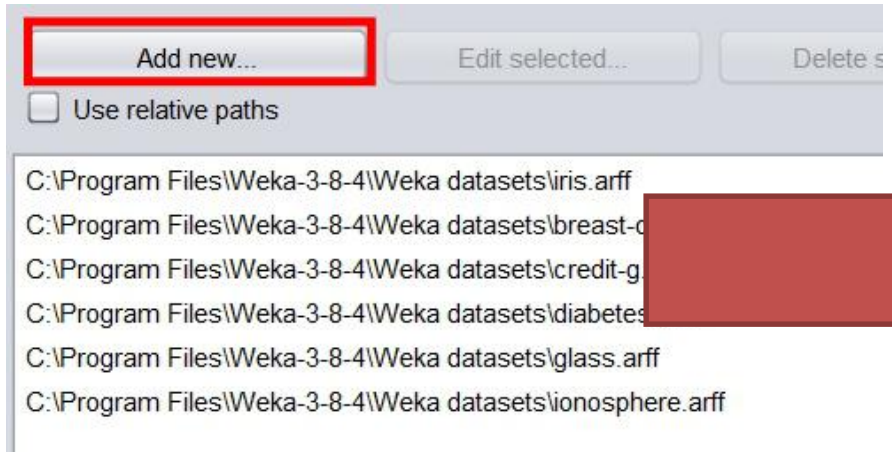
6.再次以左鍵單擊Datasets區域內的Add new...按鈕，左鍵單擊 ionosphere.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集

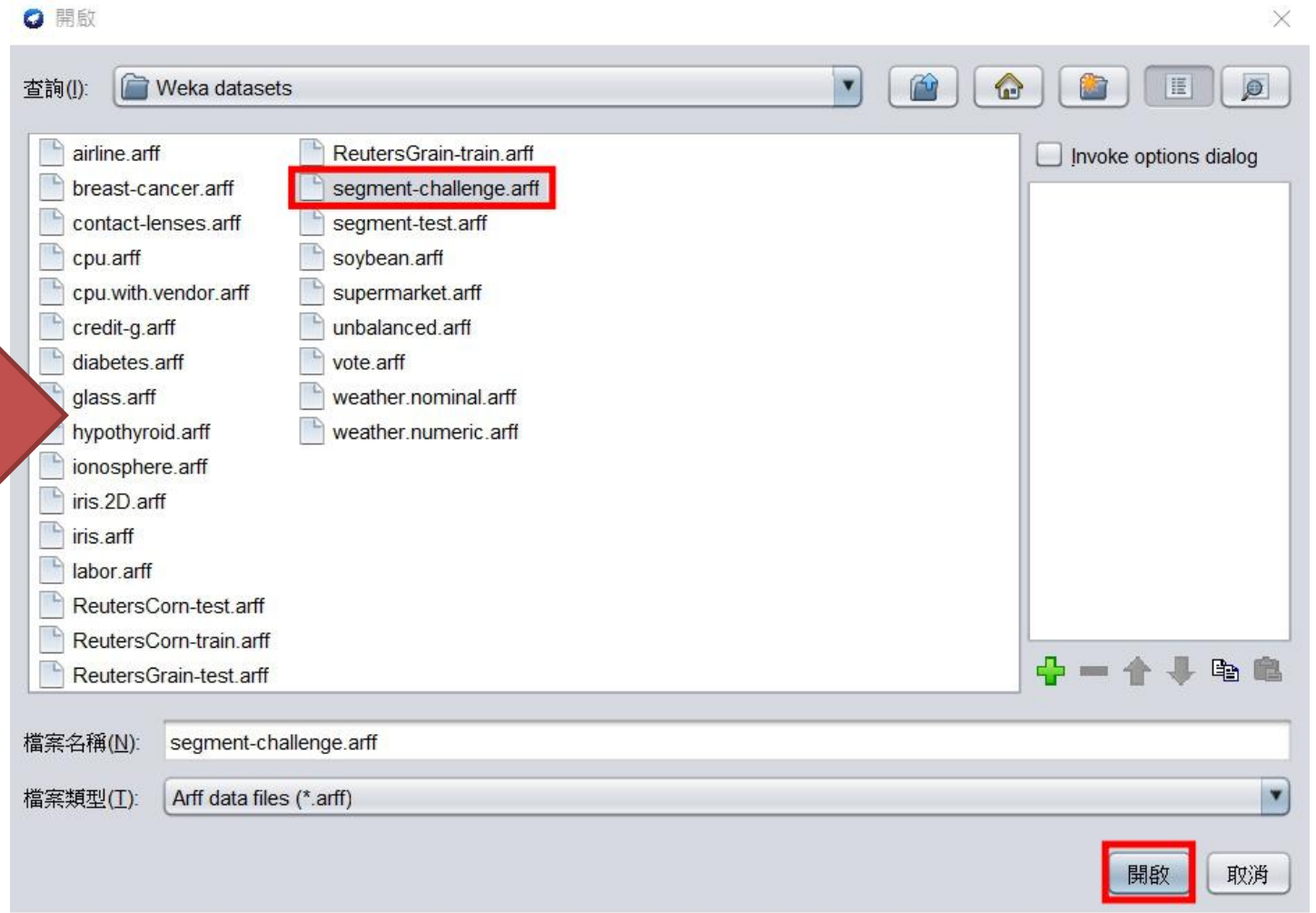




Lesson 1.3: 比較分類器

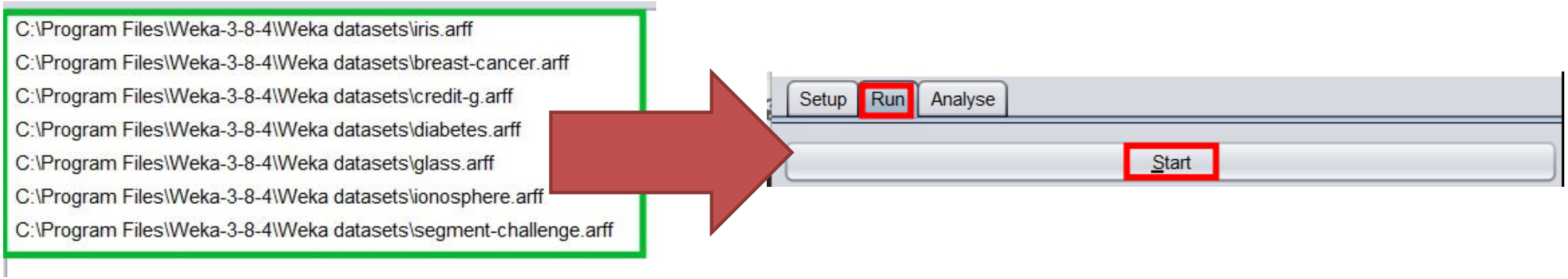
7.再次以左鍵單擊Datasets區域內的Add new...按鈕，左鍵單擊segment-challenge.arff的檔案後，再以左鍵單擊下方開啟鈕以載入此資料集





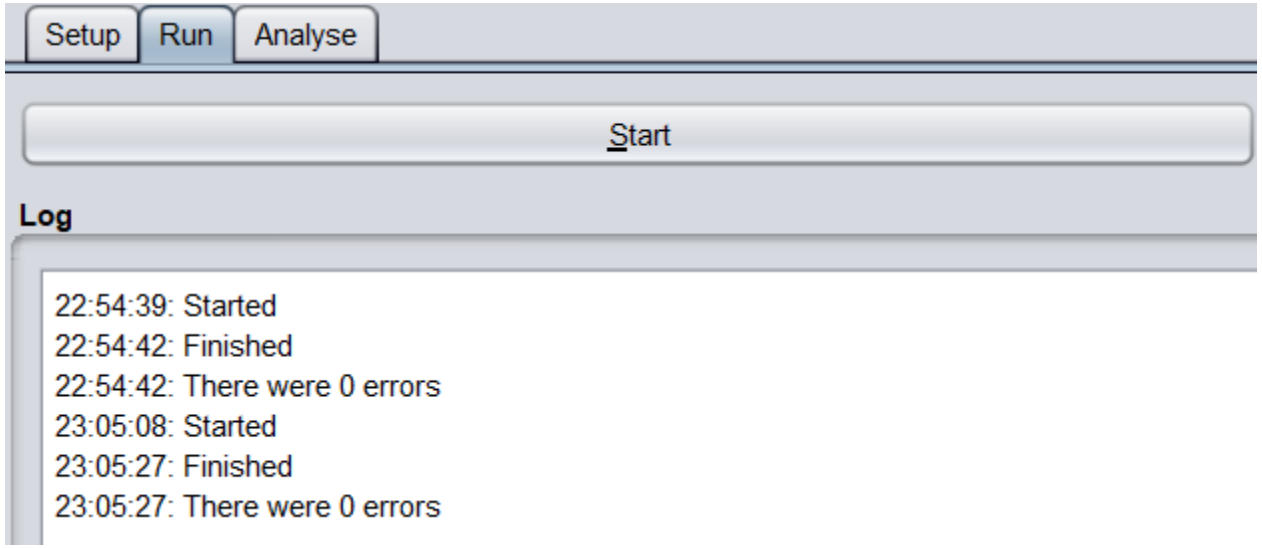
Lesson 1.3: 比較分類器

8.確認載入下圖所有資料集後，切換到Run面板左鍵單擊Start按鈕，等待結果執行完畢



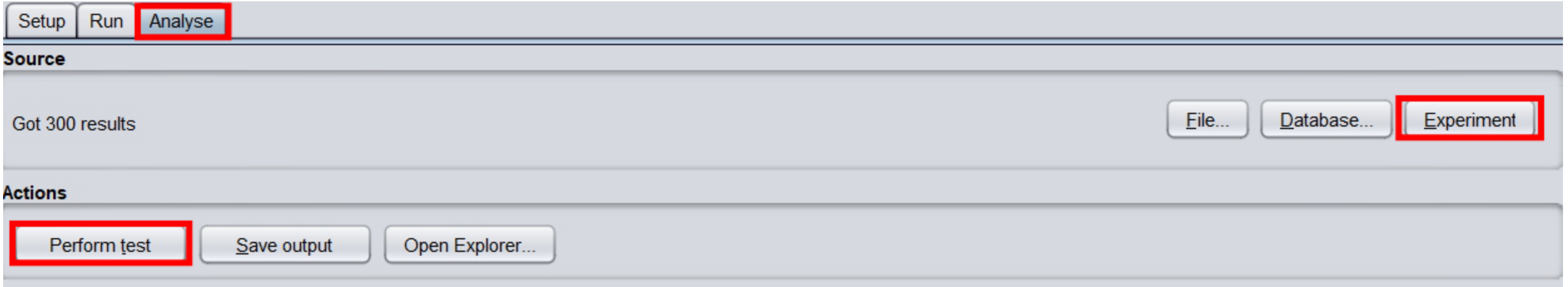
Lesson 1.3: 比較分類器

▼執行結果



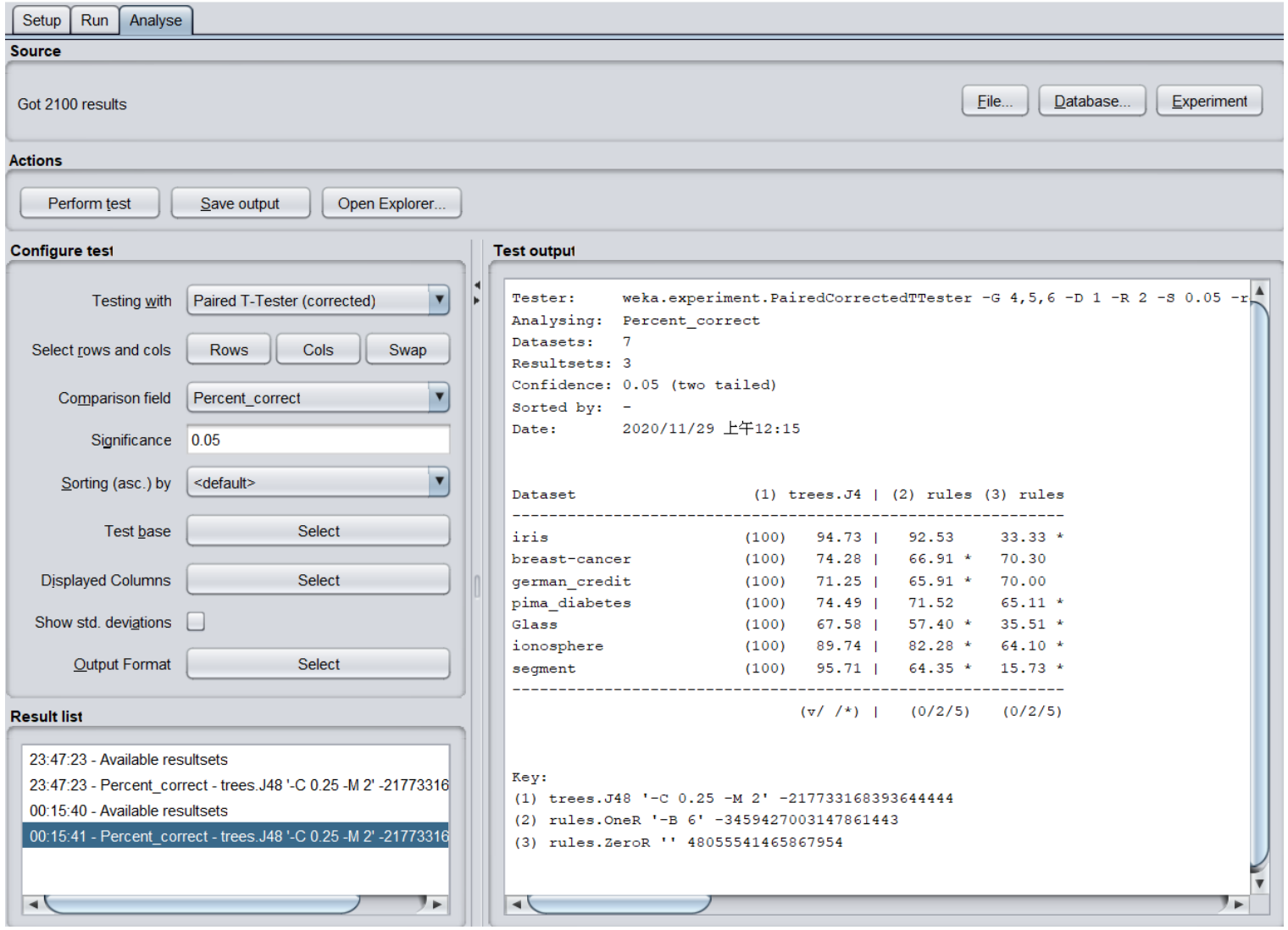
Lesson 1.3: 比較分類器

9.切換到Analyse面板，左鍵單擊Experiment按鈕，在左鍵單擊Perform test按鈕



Lesson 1.3: 比較分類器

▼執行結果



Lesson 1.3: 比較分類器

Dataset	(1) trees.J4	(2) rules	(3) rules
iris	(100) 94.73	92.53	33.33 *
breast-cancer	(100) 74.28	66.91 *	70.30
german_credit	(100) 71.25	65.91 *	70.00
pima_diabetes	(100) 74.49	71.52	65.11 *
Glass	(100) 67.63	57.40 *	35.51 *
ionosphere	(100) 89.74	82.28 *	64.10 *
segment	(100) 95.71	64.35 *	15.73 *
	(v/ /*)	(0/2/5)	(0/2/5)

Key:
(1) trees.J48 '
(2) rules.OneR
(3) rules.ZeroR

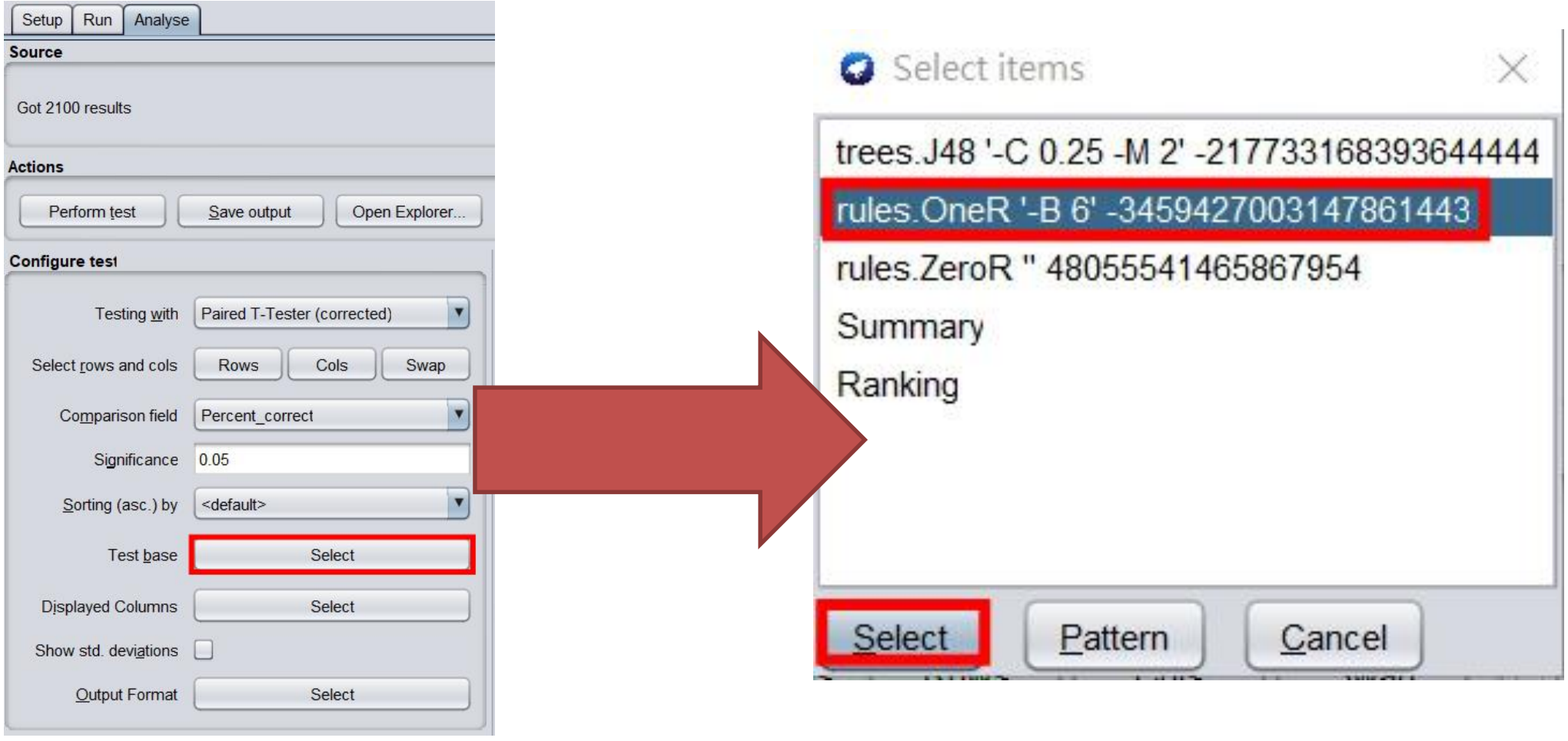
J48 遠優於（根據統計意義上的5% level）

- OneR 和 ZeroR在Glass, ionosphere, segment的表現
- OneR 在breast-cancer, german_credit的表現
- ZeroR 在iris, pima_diabetes的表現

Lesson 1.3: 比較分類器

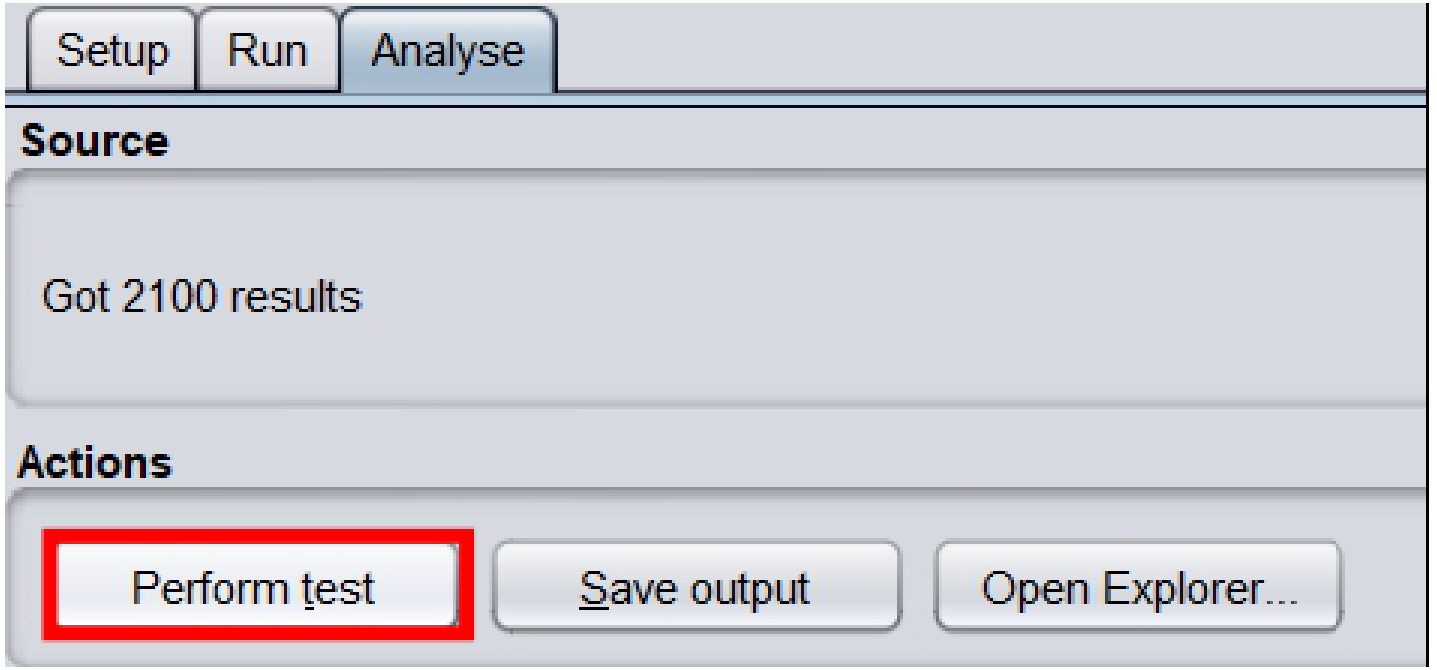
接下來，我們選擇不同的基準方案。

1.回到Analyse面板左鍵單擊Select按鈕，並在出現的視窗中左鍵單擊OneR的選項後，左鍵單擊視窗左下方的Select按鈕



Lesson 1.3: 比較分類器

2.左鍵單擊Analyse面板中的Perform test按鈕



Lesson 1.3: 比較分類器

Dataset	(2) rules.On	(1) trees	(3) rules
iris	(100) 92.53	94.73	33.33 *
breast-cancer	(100) 66.91	74.28 v	70.30
german_credit	(100) 65.91	71.25 v	70.00 v
pima_diabetes	(100) 71.52	74.49	65.11 *
Glass	(100) 57.40	67.63 v	35.51 *
ionosphere	(100) 82.28	89.74 v	64.10 *
segment	(100) 64.35	95.71 v	15.73 *
	(v/ /*)	(5/2/0)	(1/1/5)

Key:
(1) trees.J48
(2) rules.OneR
(3) rules.ZeroR

用OneR 與 ZeroR相比

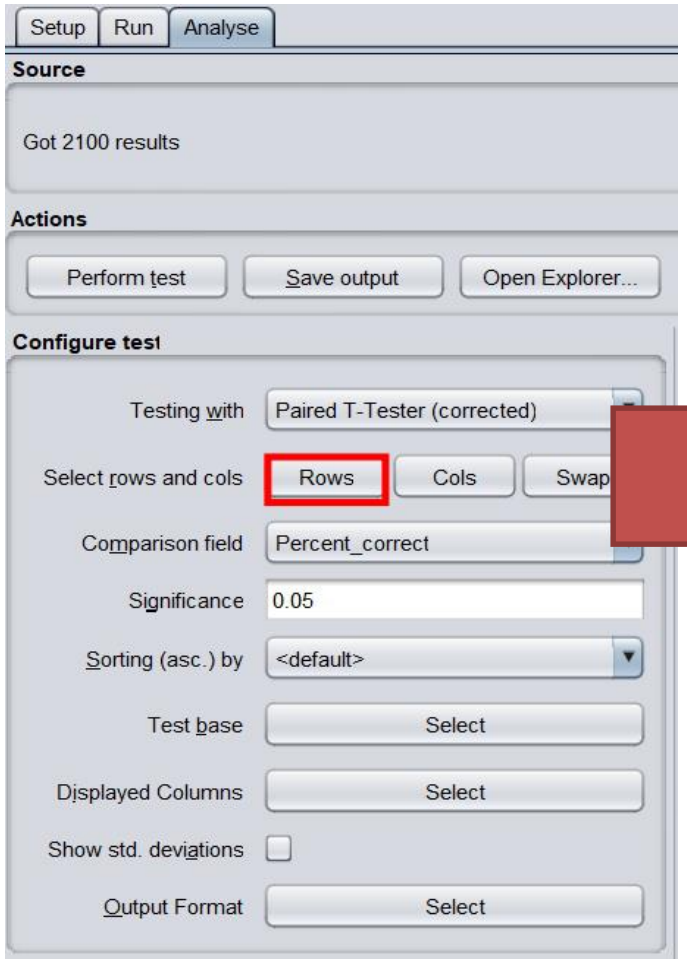
改變Analyse面板中的「Test base」選項

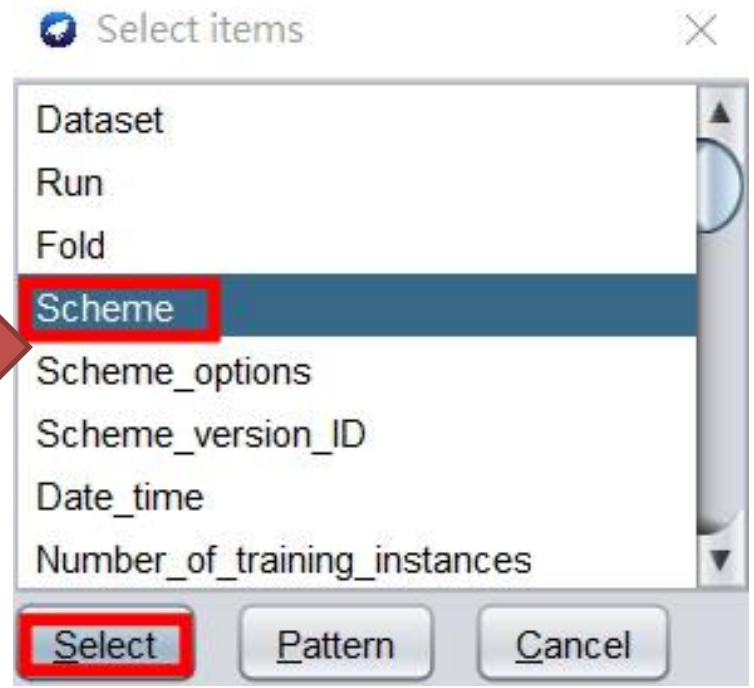
- OneR 在german-credit上的表現遠不如ZeroR
- OneR 在breast-cancer上的表現與ZeroR差不多
- OneR在所有剩下的資料集上的表現遠優於ZeroR

Lesson 1.3: 比較分類器

接下來，我們試著改變矩陣數列的順序。

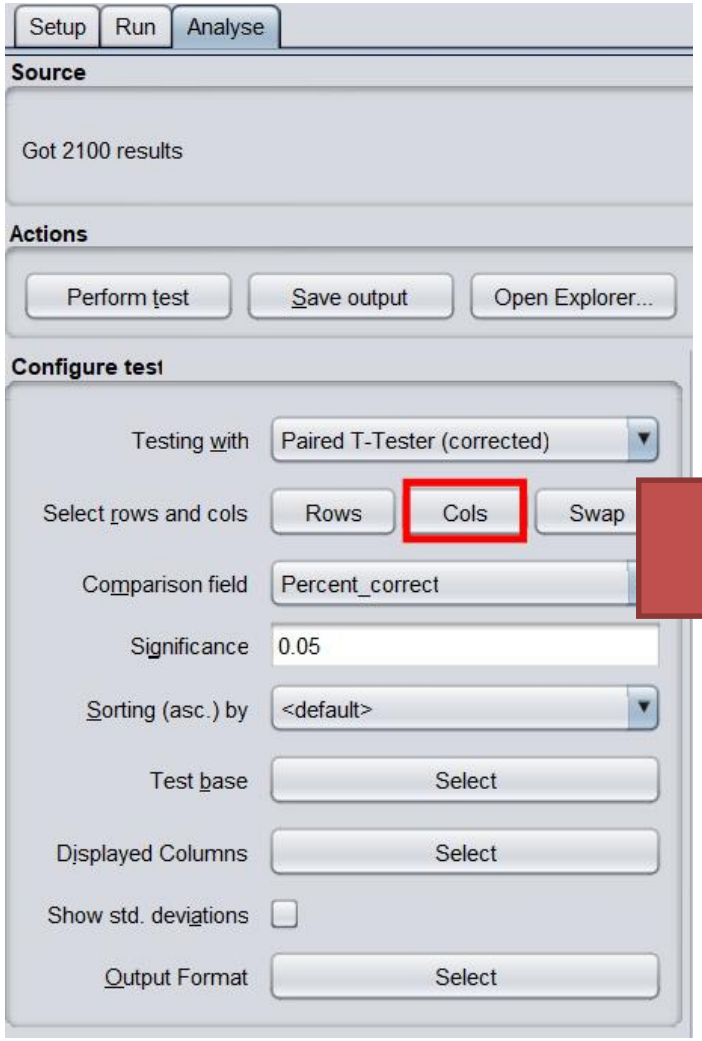
1.回到Analyse面板，左鍵單擊Rows按鈕，並在出現的視窗中左鍵單擊Scheme後，左鍵單擊視窗左下方的Select按鈕，選擇方案為行屬性。

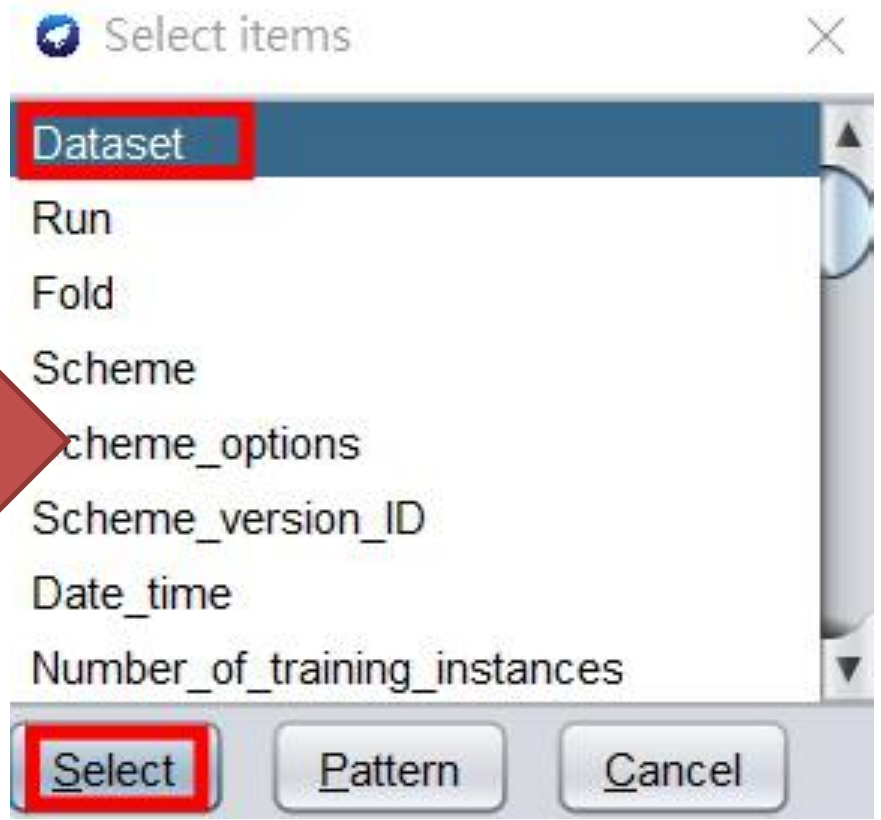




Lesson 1.3: 比較分類器

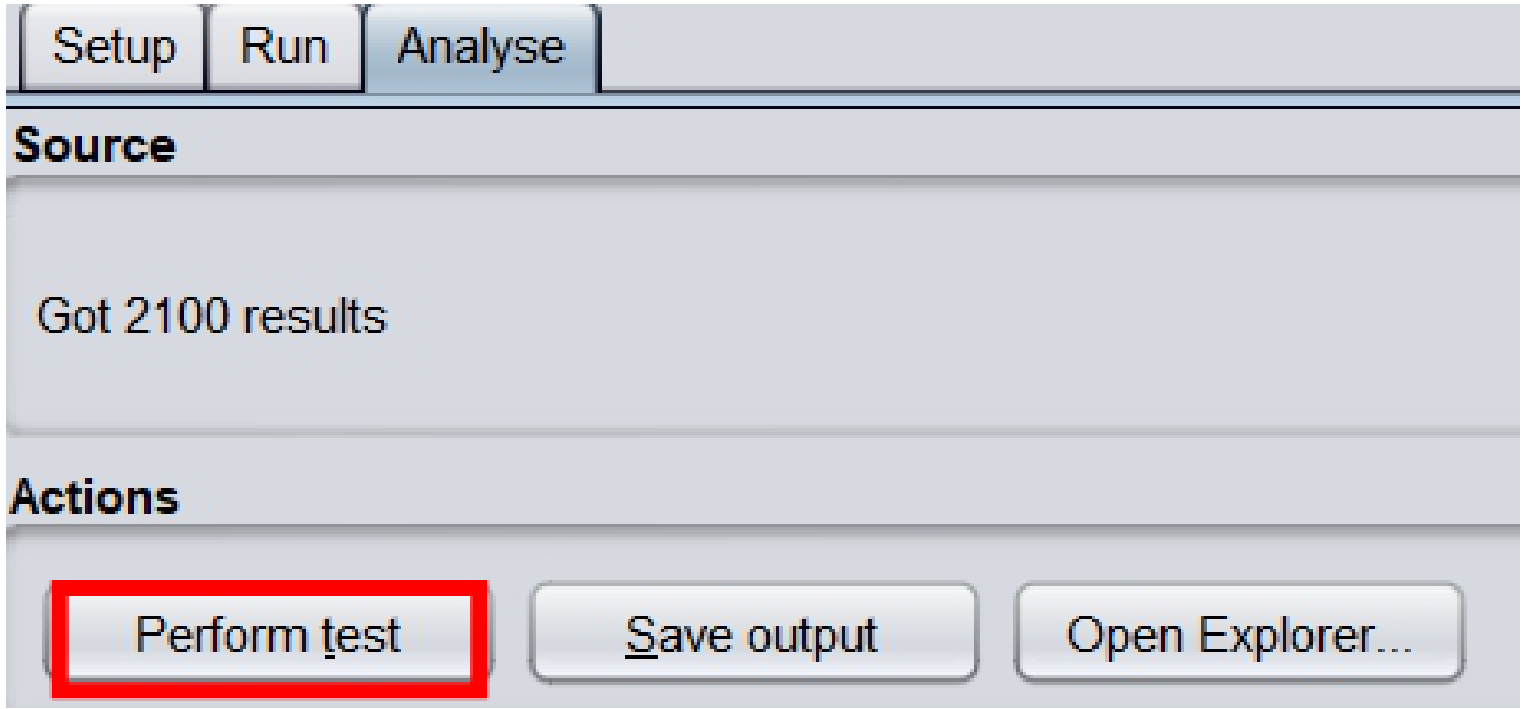
2.左鍵單擊Cols按鈕，並在出現的視窗中左鍵單擊Dataset後，左鍵單擊視窗左下方的Select按鈕，選擇資料集作為列屬性。





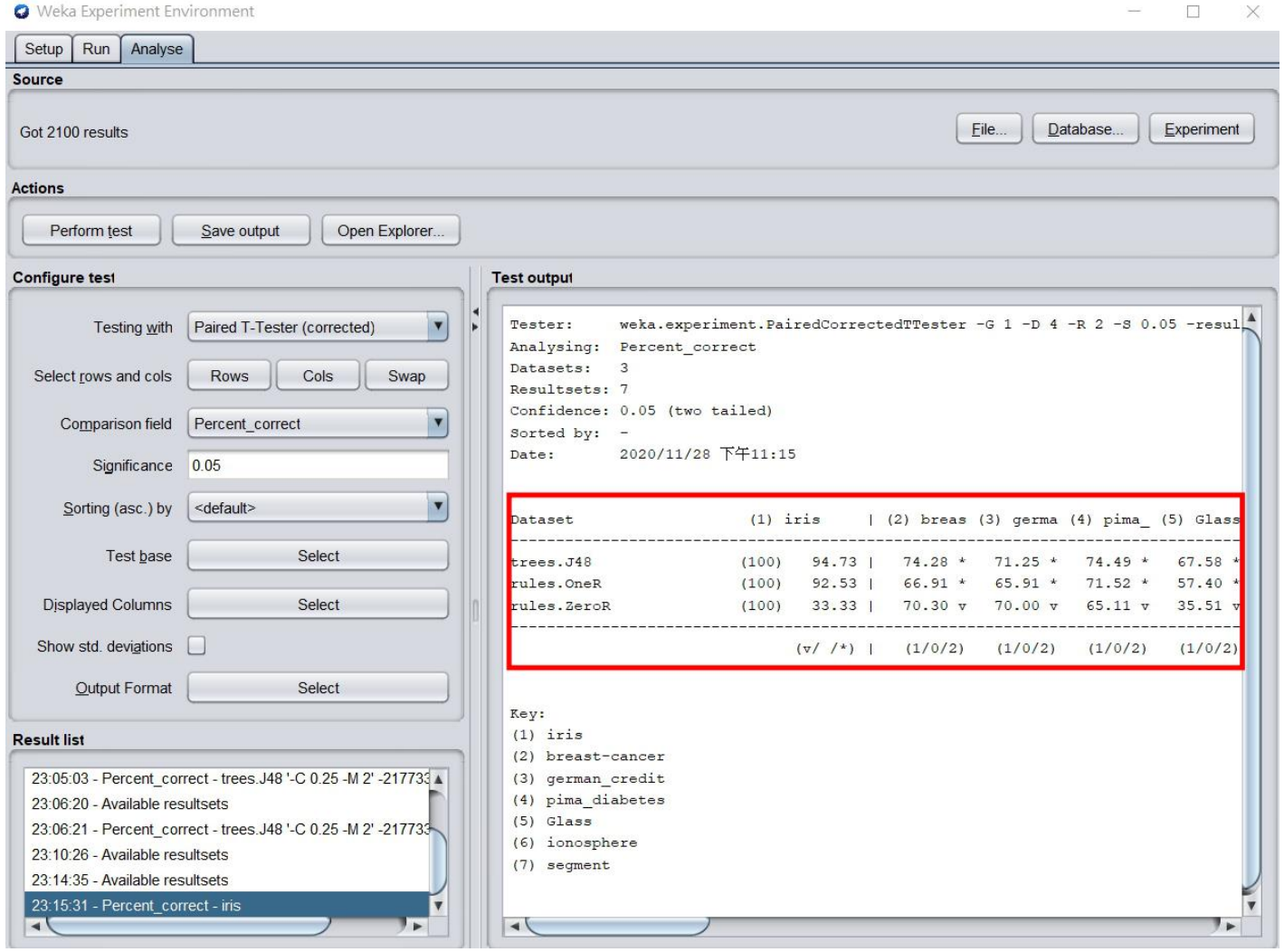
Lesson 1.3: 比較分類器

3.在Analyse面板中，左鍵單擊Perform test按鈕

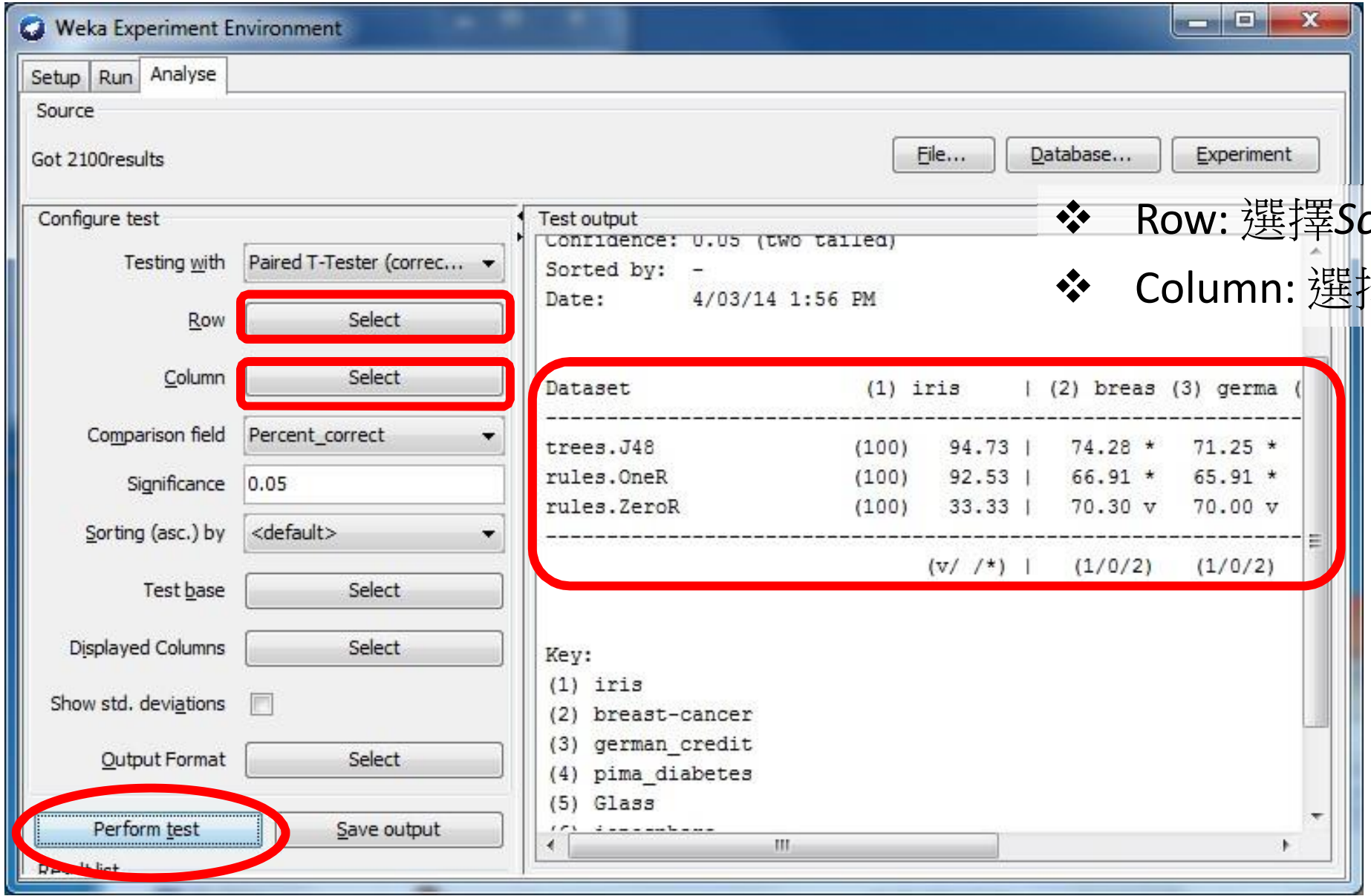


Lesson 1.3: 比較分類器

▼結果：可以看到資料集橫向排列，算法縱向排列。我們也可以觀察各分類器在各資料集上的表現是否優於在**breast-cancer**資料集上的表現。



Lesson 1.3: 比較分類器



- Row: 選擇*Scheme* (而非*Dataset*)
- Column: 選擇Dataset (而非 *Scheme*)

Lesson 1.3: 比較分類器

- 從統計學的角度來看，就是「零假設(null hypothesis)」，即某個分類器的準確率等於另一個分類器的準確率。
- 我們觀察的結果表明零假設幾乎不成立

  「*我們在統計意義上有95%的把握推翻零假設*」

  "A 的表現以5% level來看會遠優於B"
- 可以改變significance level (5% 和1%最常見)
- 可以改變比較的內容(我們比較了百分比準確率)
- 通常使用多個資料集比較

  「*在這些資料集中, 方法A贏了方法B幾次、輸了方法B幾次*」
- 多次比較的問題

  如果你做大量的測試，一些顯著差異可能會是偶然的！

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Class 1 – Lesson 4

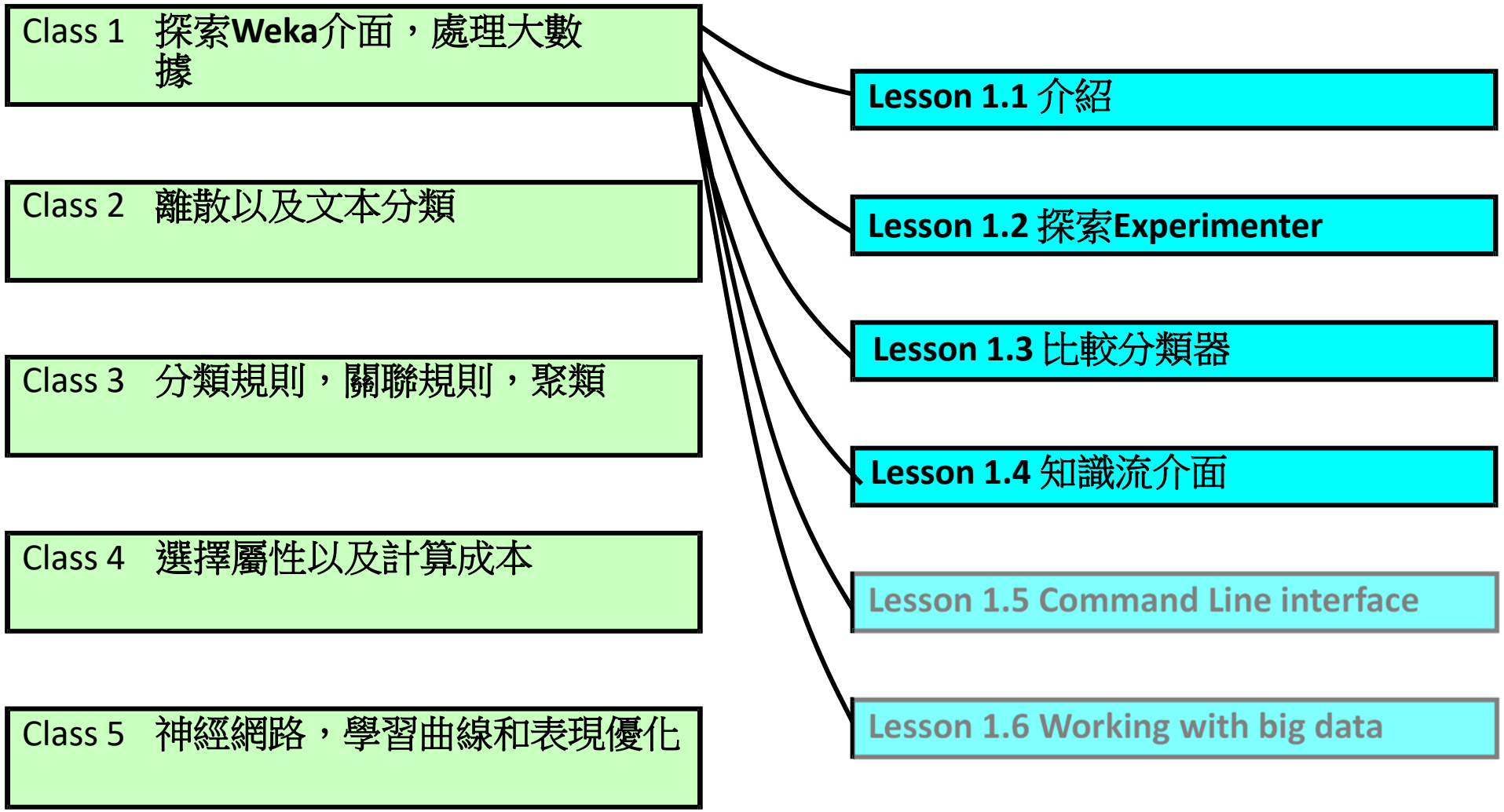
知識流介面(The Knowledge Flow interface)

Ian H. Witten

Department of Computer Science University of Waikato
New Zealand

weka.waikato.ac.nz

Lesson 1.4: 知識流介面



Lesson 1.4: 知識流介面

知識流介面可以替代Explorer

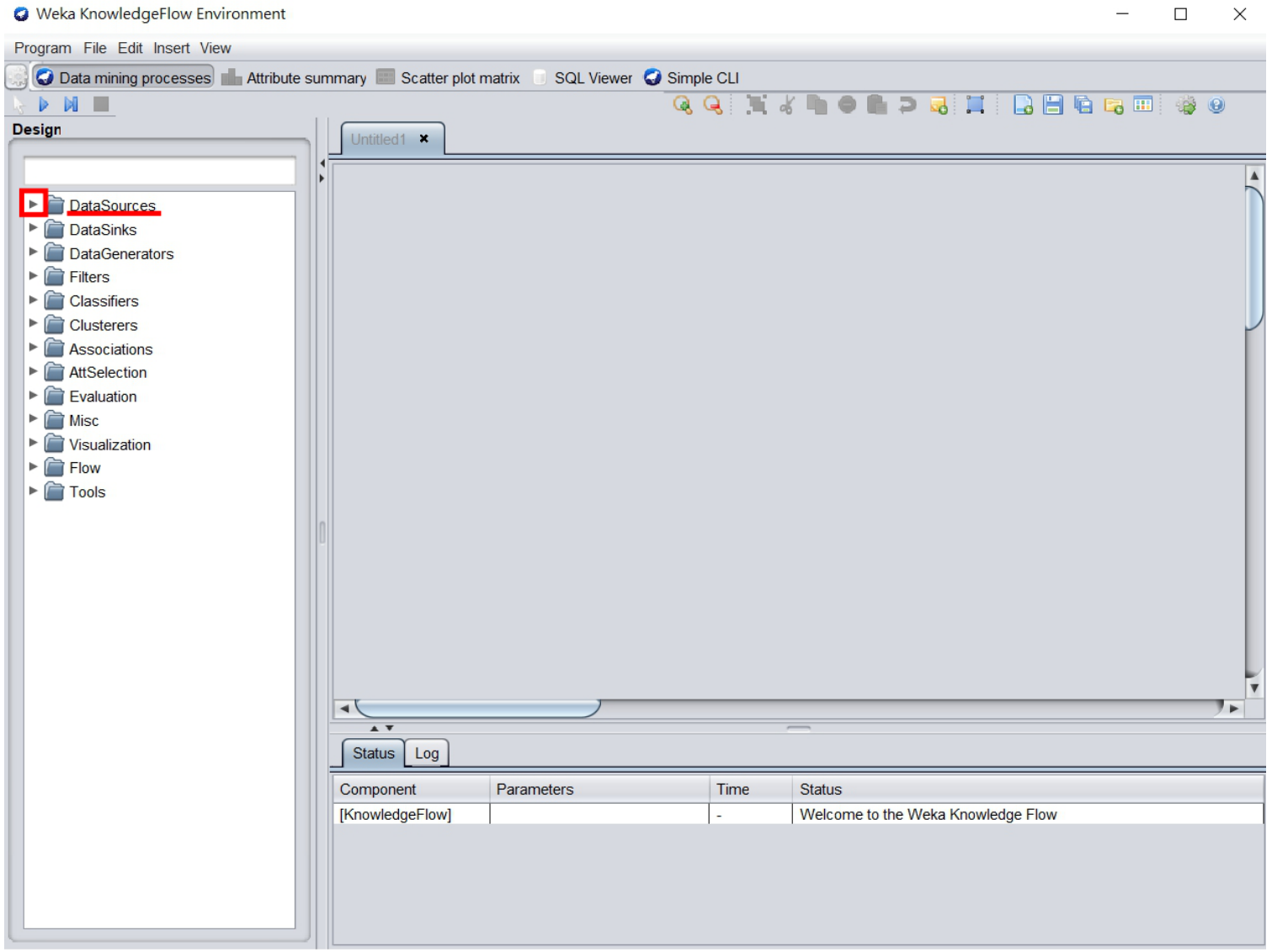
- 可以在2D設計畫布上交互式地安排過濾器、分類器和評估器
- 其他可用的組件包含資料來源、資料槽(data sinks)、評估、可視化
- 我們可以通過不同的方式連接各組件
  - *實例或資料集*
  - *測試集、訓練集*
  - 分類器
  - *輸出，文字或圖表*
- 可以遞增地處理可能的無限資料流
- 可以在個別建置好的模型上看到交叉驗證的內容

Lesson 1.4: 知識流介面

1. 開啟Weka程式，於Weka GUI Chooser介面左鍵單擊KnowledgeFlow按鈕

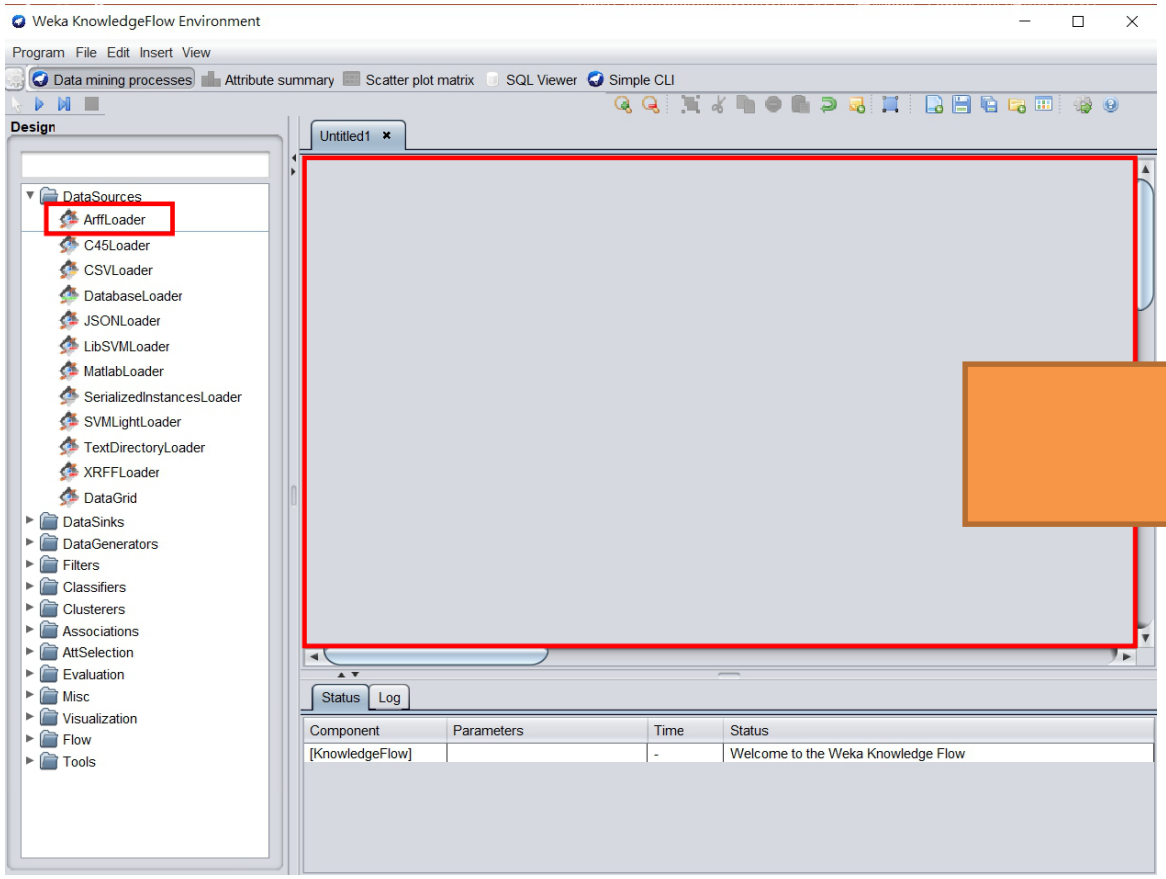
Lesson 1.4: 知識流介面

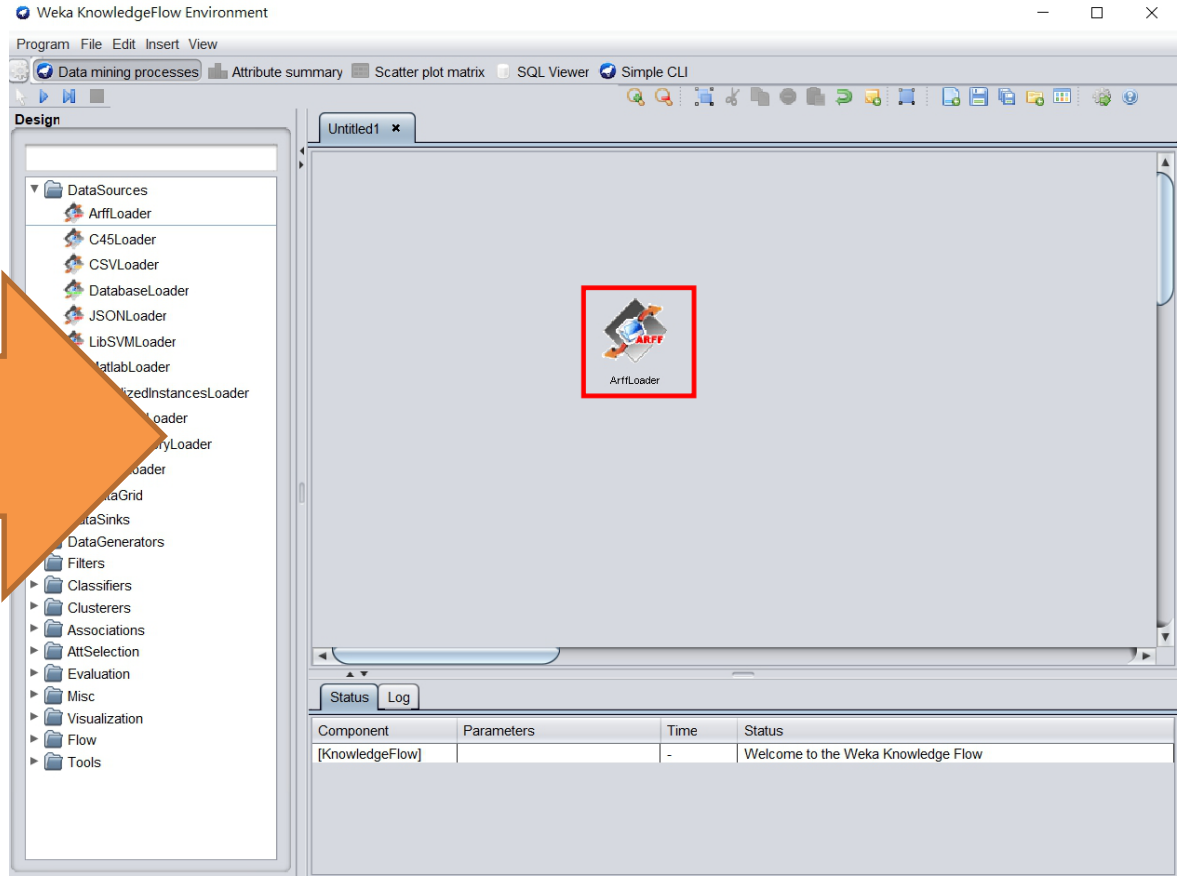
2.左鍵單擊DataSources資料夾前方的展開圖示



Lesson 1.4: 知識流介面

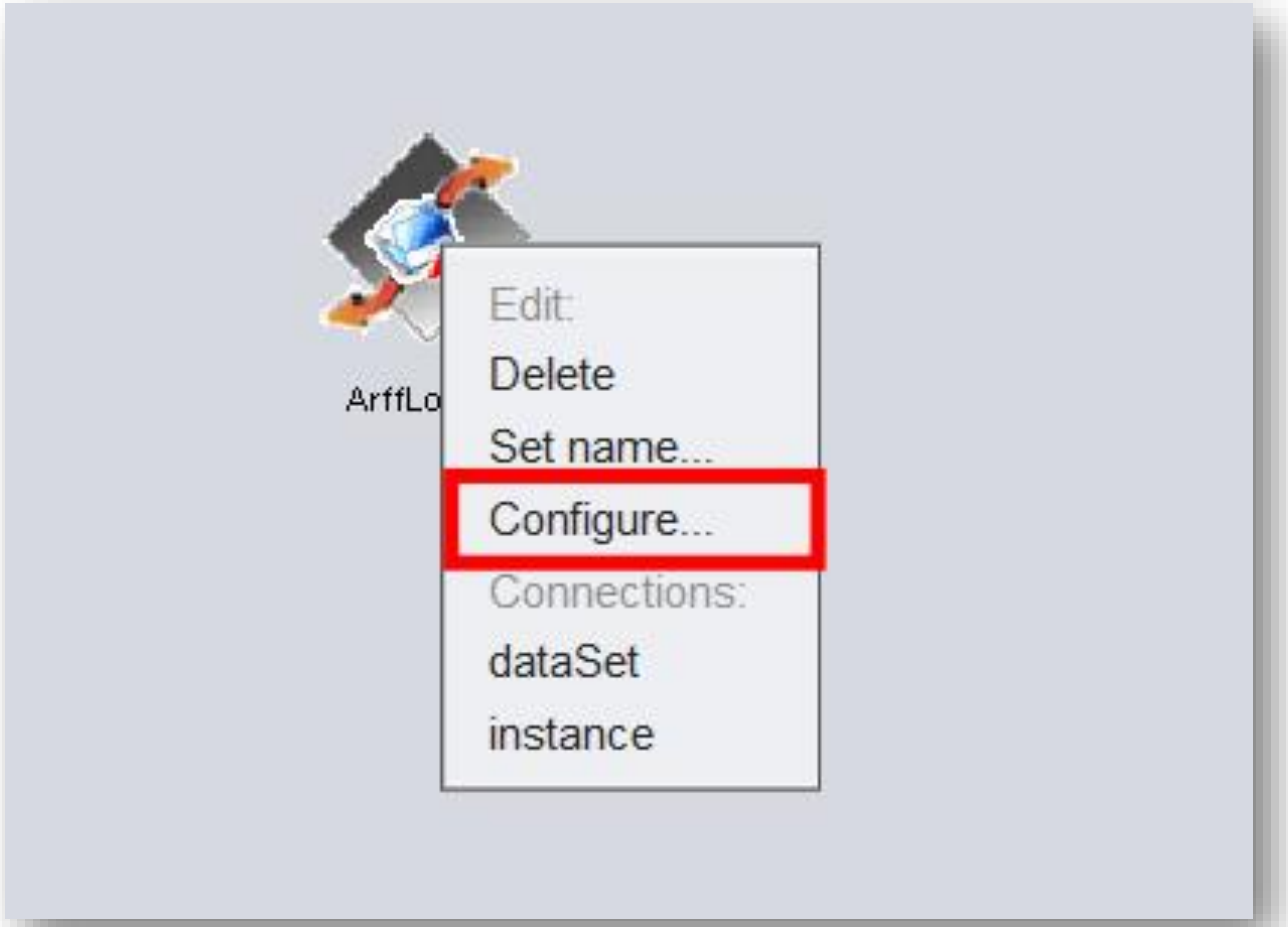
3.左鍵單擊ArffLoader元件，於右方畫布中單擊左鍵將其置入。





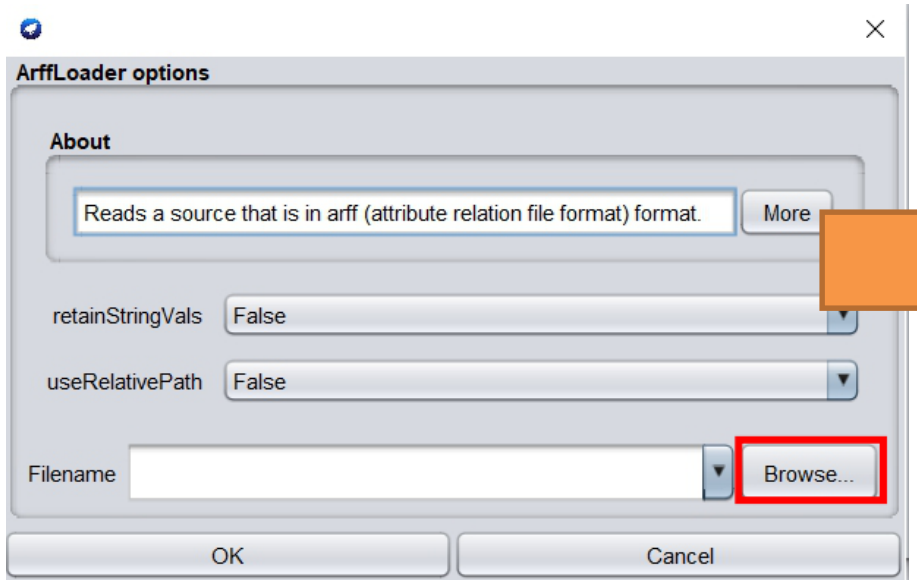
Lesson 1.4: 知識流介面

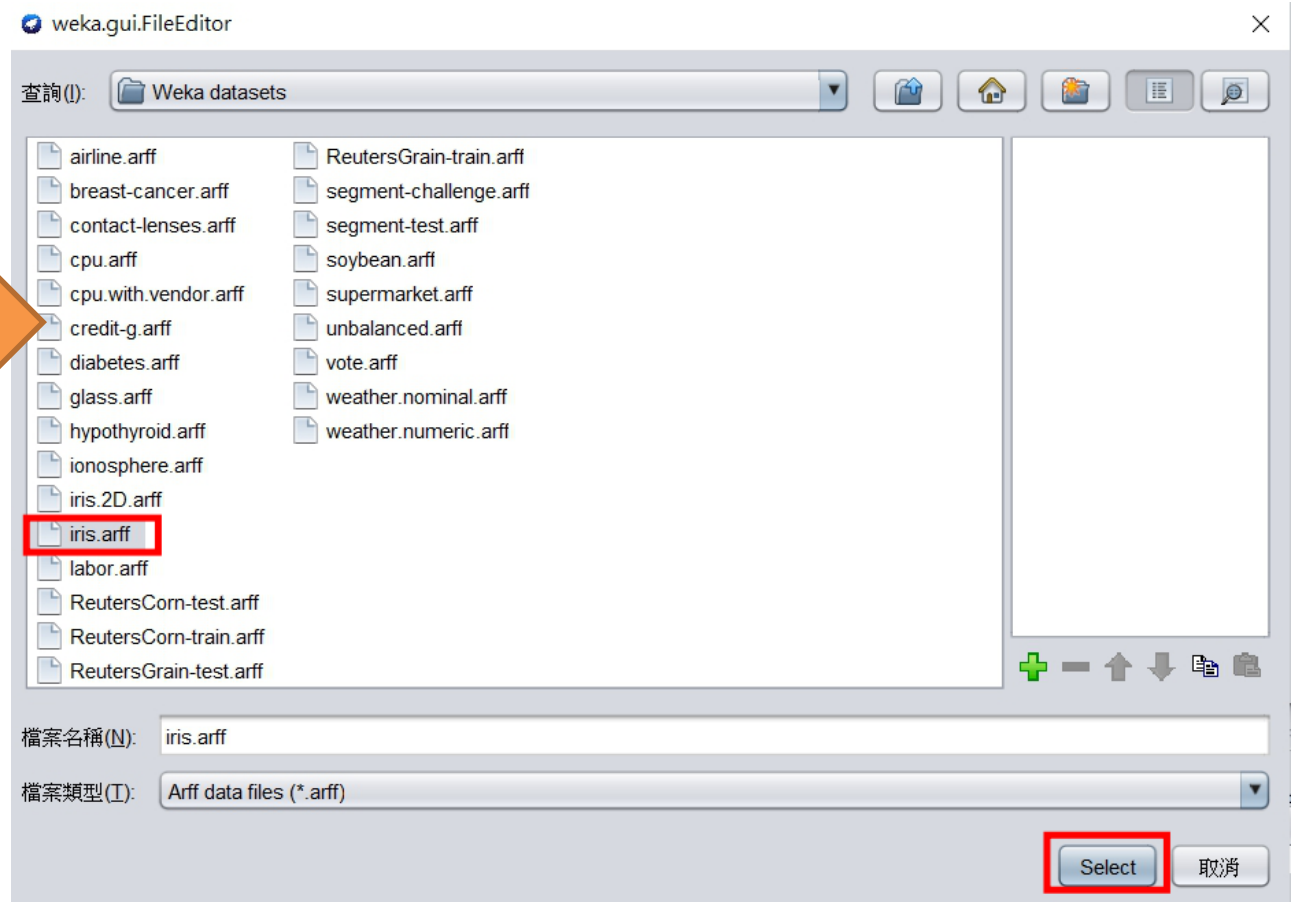
4. 右鍵單擊ArffLoader元件，在選單中左鍵單擊Configure。



Lesson 1.4: 知識流介面

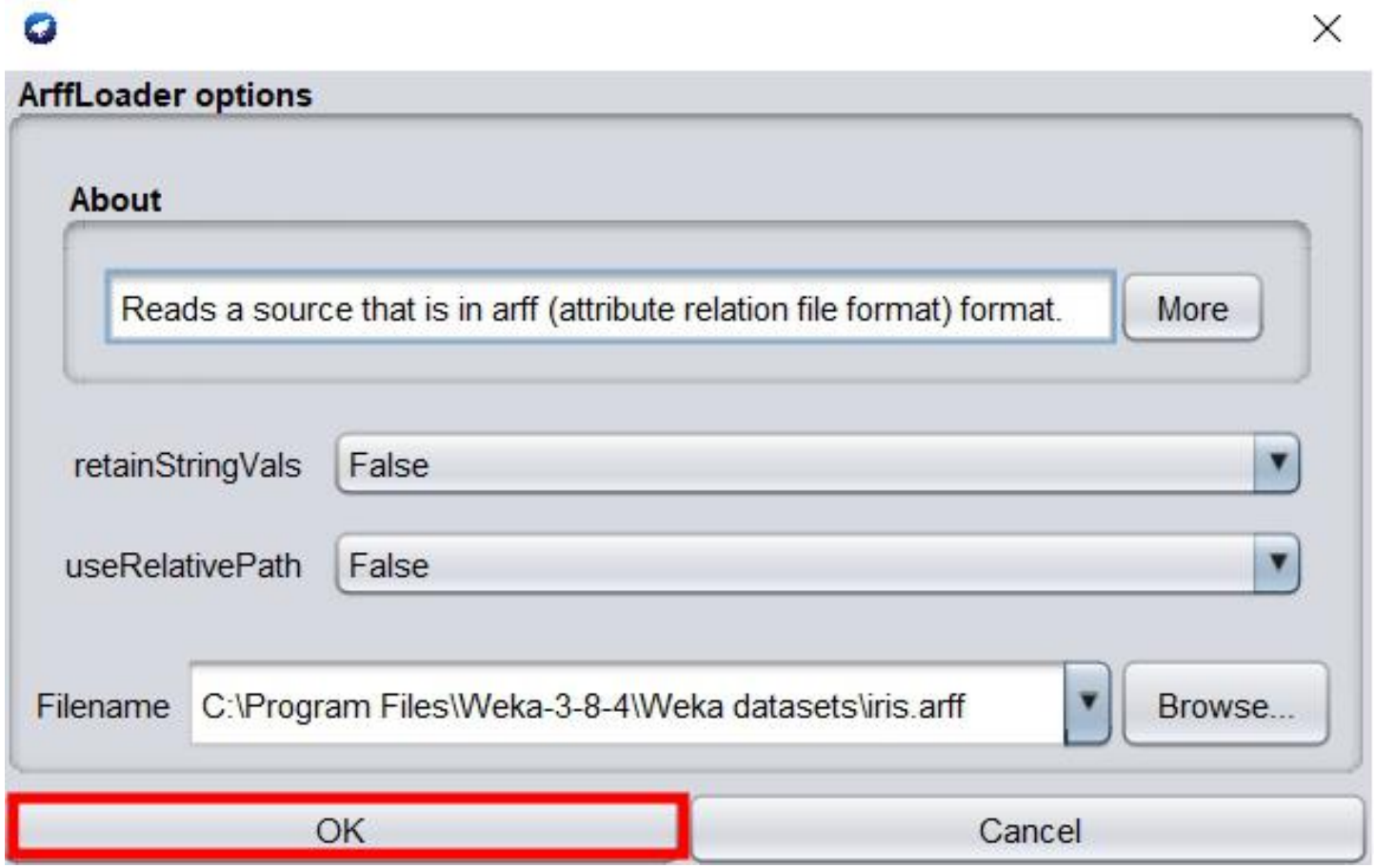
5.在出現的視窗中左鍵單擊Browse按鈕，進入自行複製的Weka datasets資料夾 → 左鍵單擊iris.arff檔案 → 左鍵單擊視窗右下方的Select按鈕。





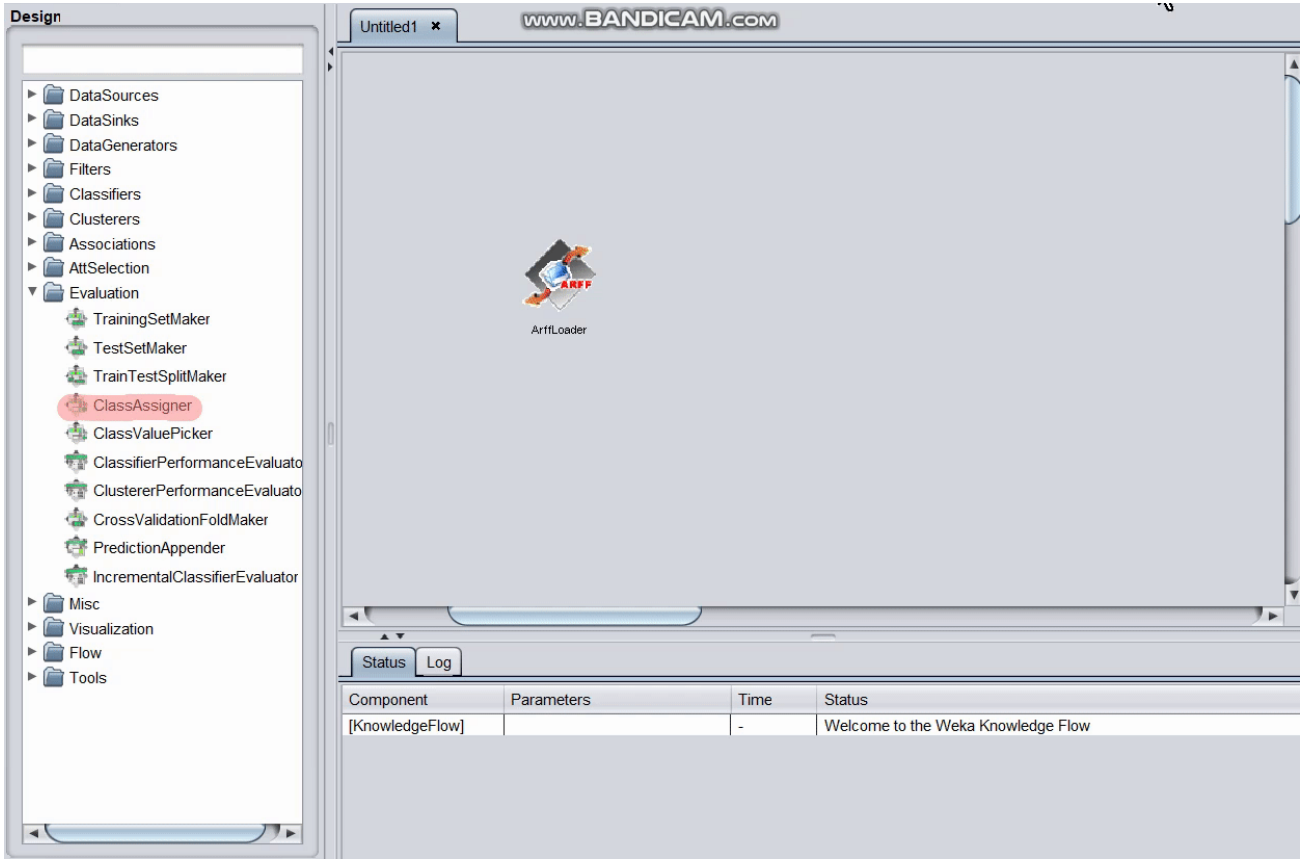
Lesson 1.4: 知識流介面

6.左鍵單擊OK按鈕。



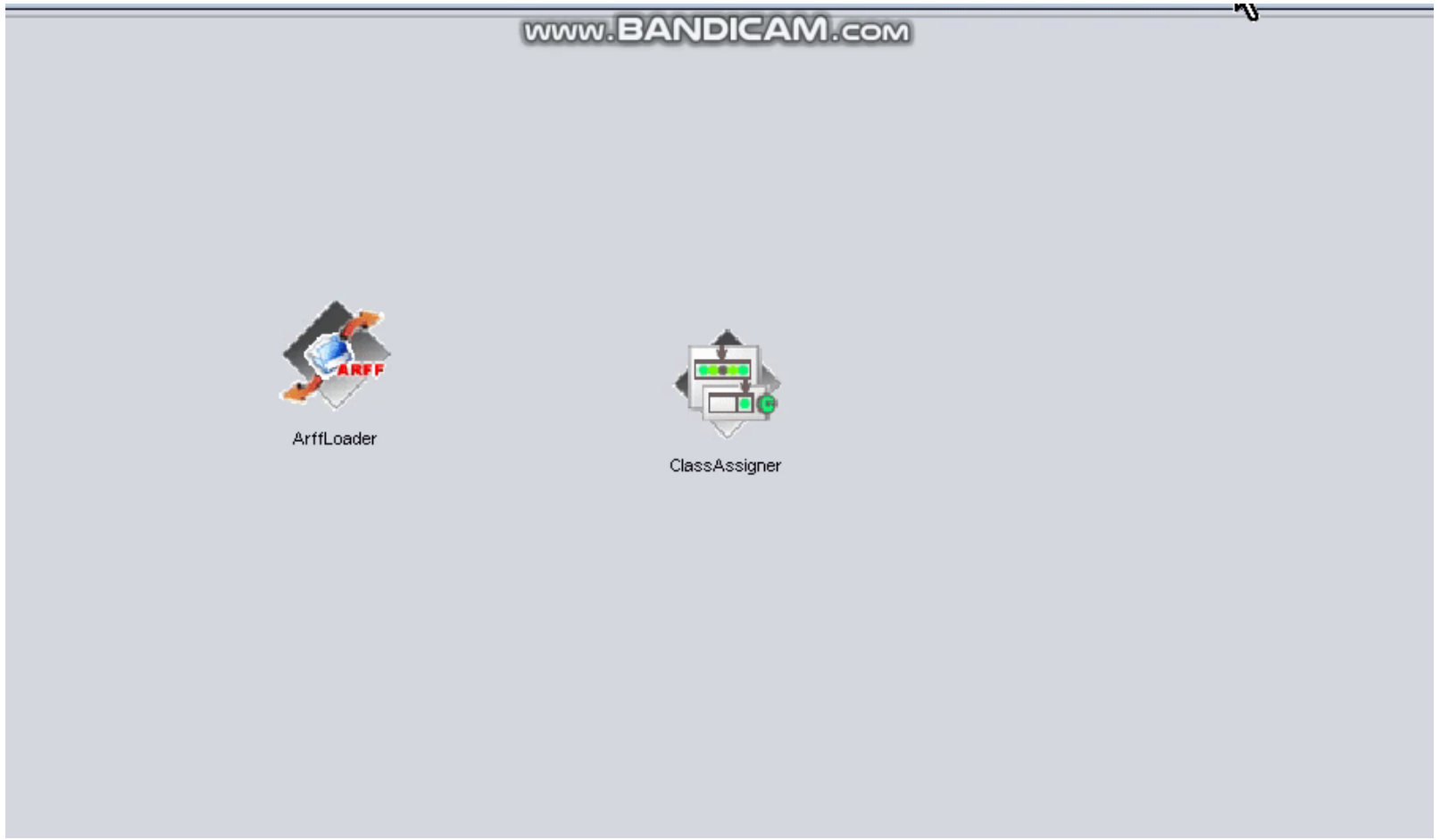
Lesson 1.4: 知識流介面

7.左鍵單擊Evaluation資料夾下的ClassAssigner元件，並放入右側畫布。



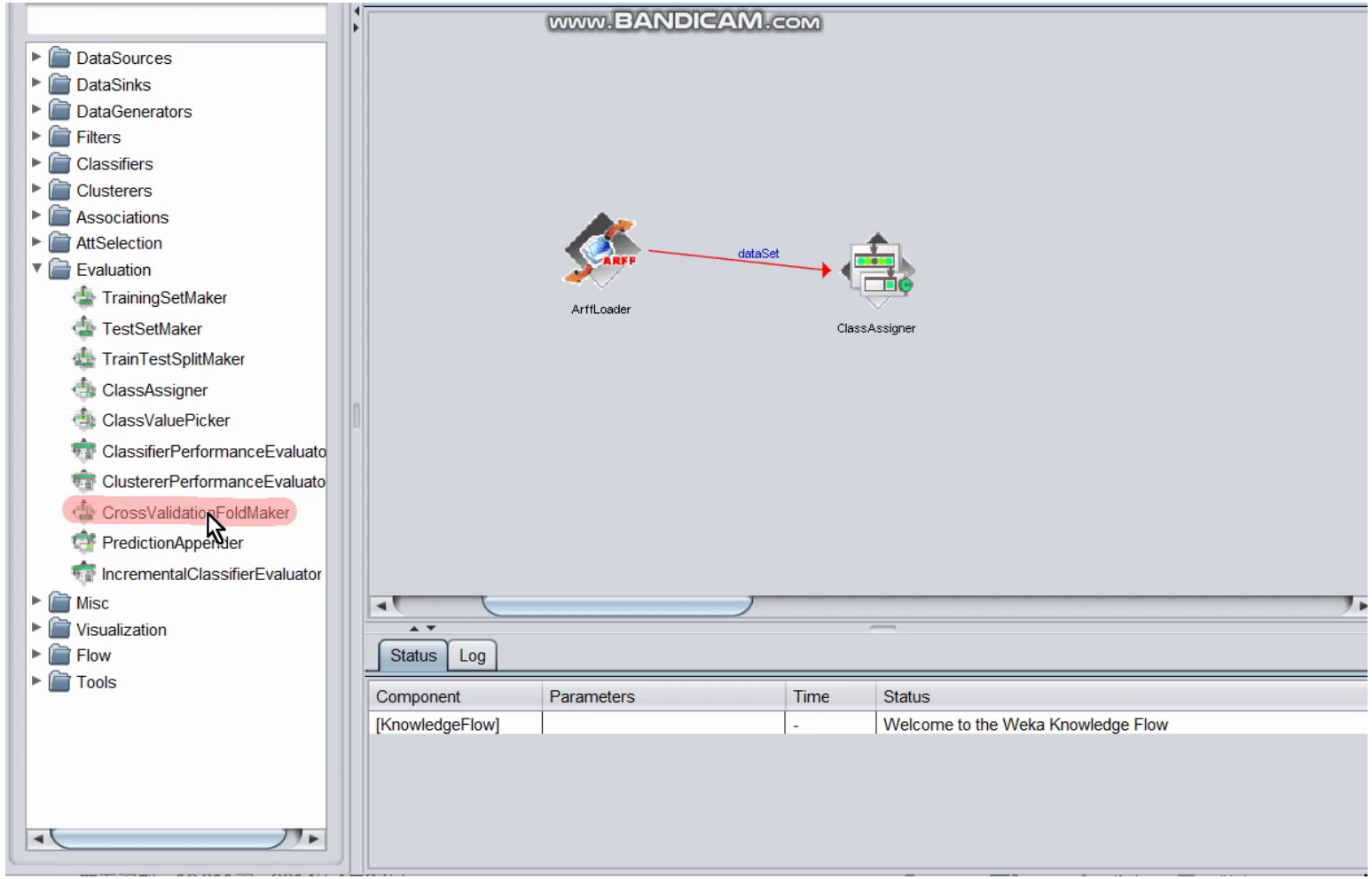
Lesson 1.4: 知識流介面

8.對ArffLoader元件單擊右鍵 → 在出現的選單中點選dataSet → 左鍵單擊ClassAssigner元件四周的圓點進行連接



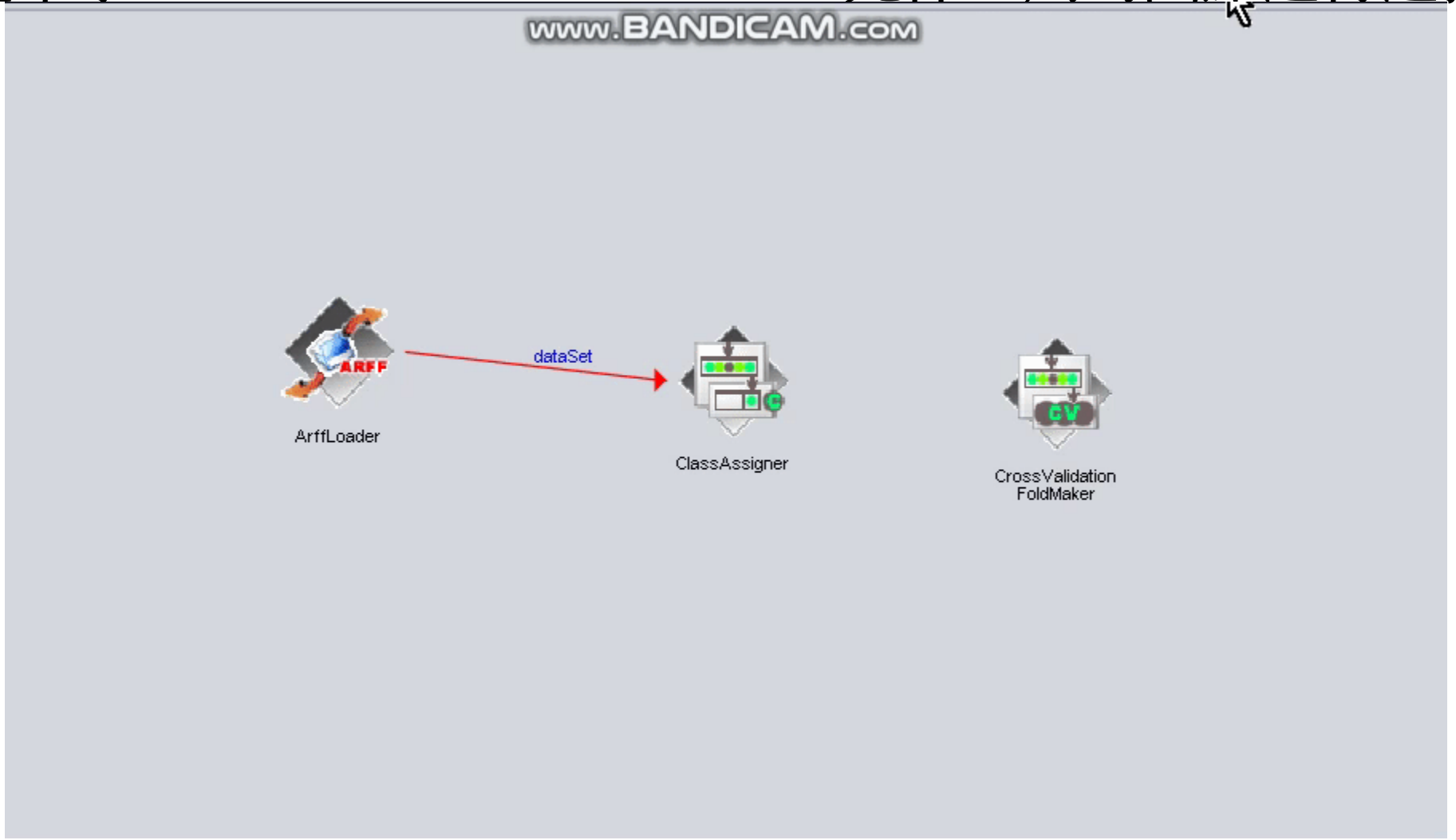
Lesson 1.4: 知識流介面

9.左鍵單擊Evaluation資料夾下的CrossValidationFoldMaker元件，接著放入右側畫布。我們將使用它來進行交叉驗證。



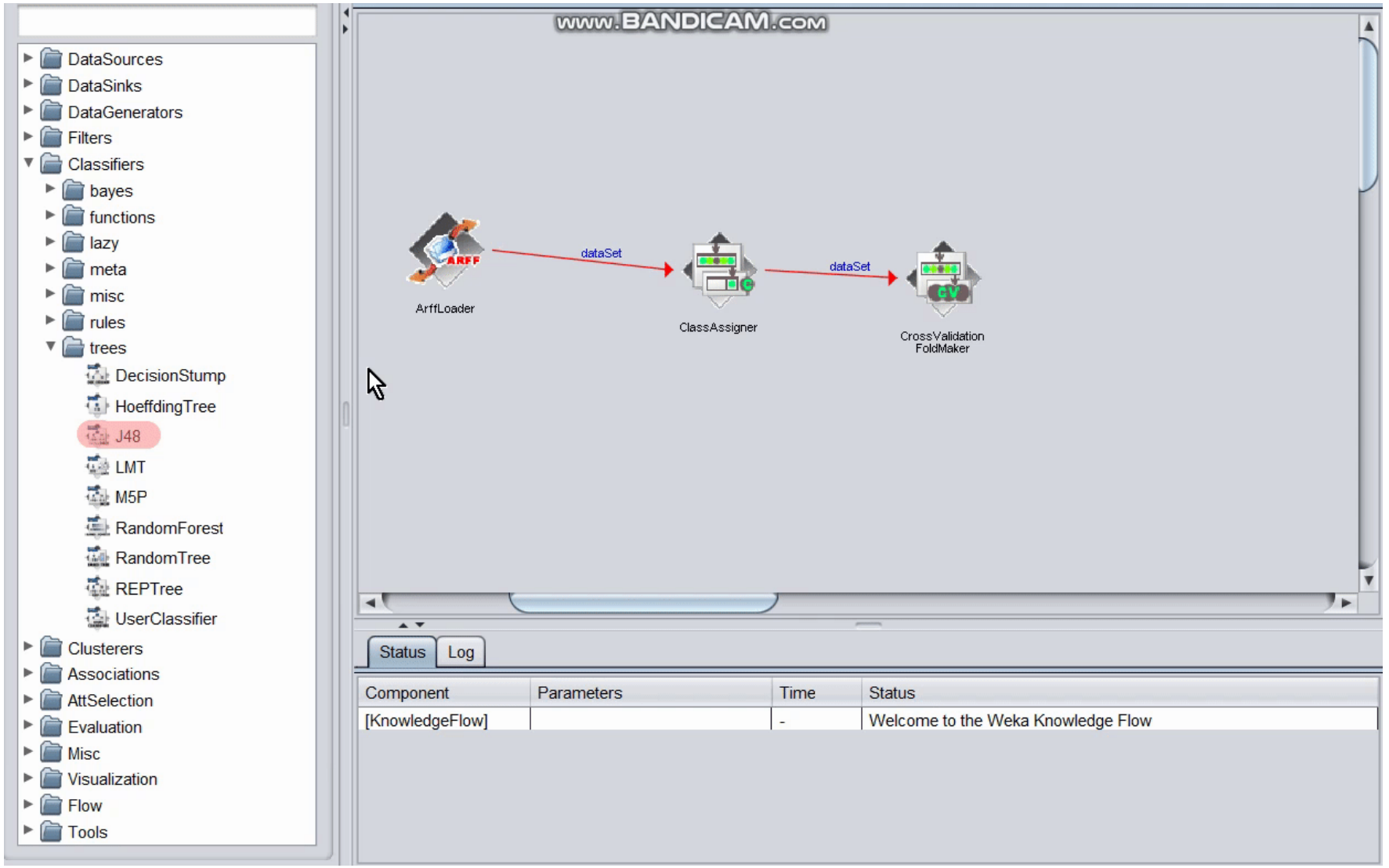
Lesson 1.4: 知識流介面

10.對ClassAssigner元件單擊右鍵 → 在出現的選單中點選dataSet → 左鍵單擊CrossValidationFoldMaker元件四周的圓點進行連接



Lesson 1.4: 知識流介面

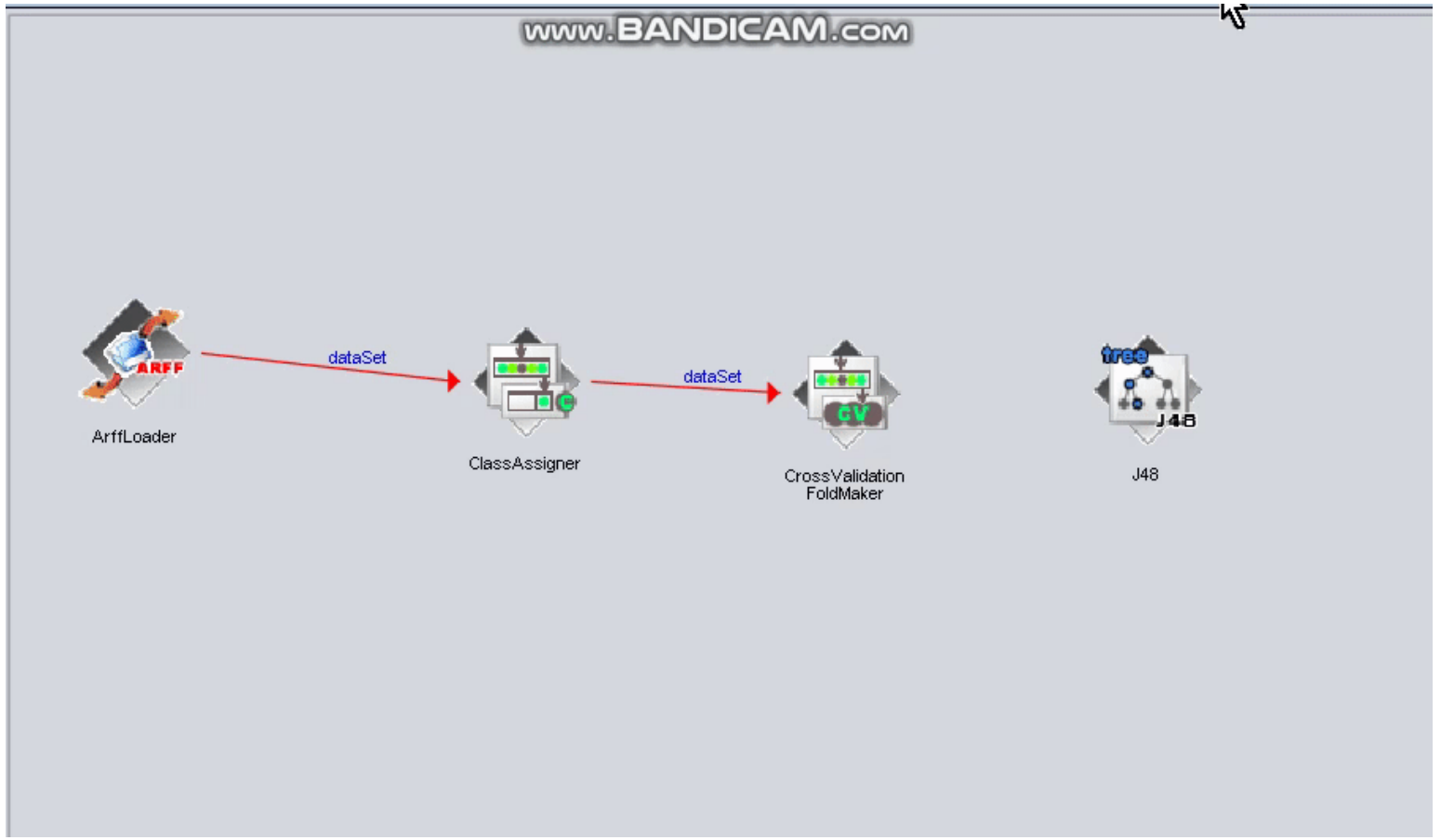
11.左鍵單擊Classifiers資料夾下的trees資料夾下的J48分類器元件，並放入右側畫布。



Lesson 1.4: 知識流介面

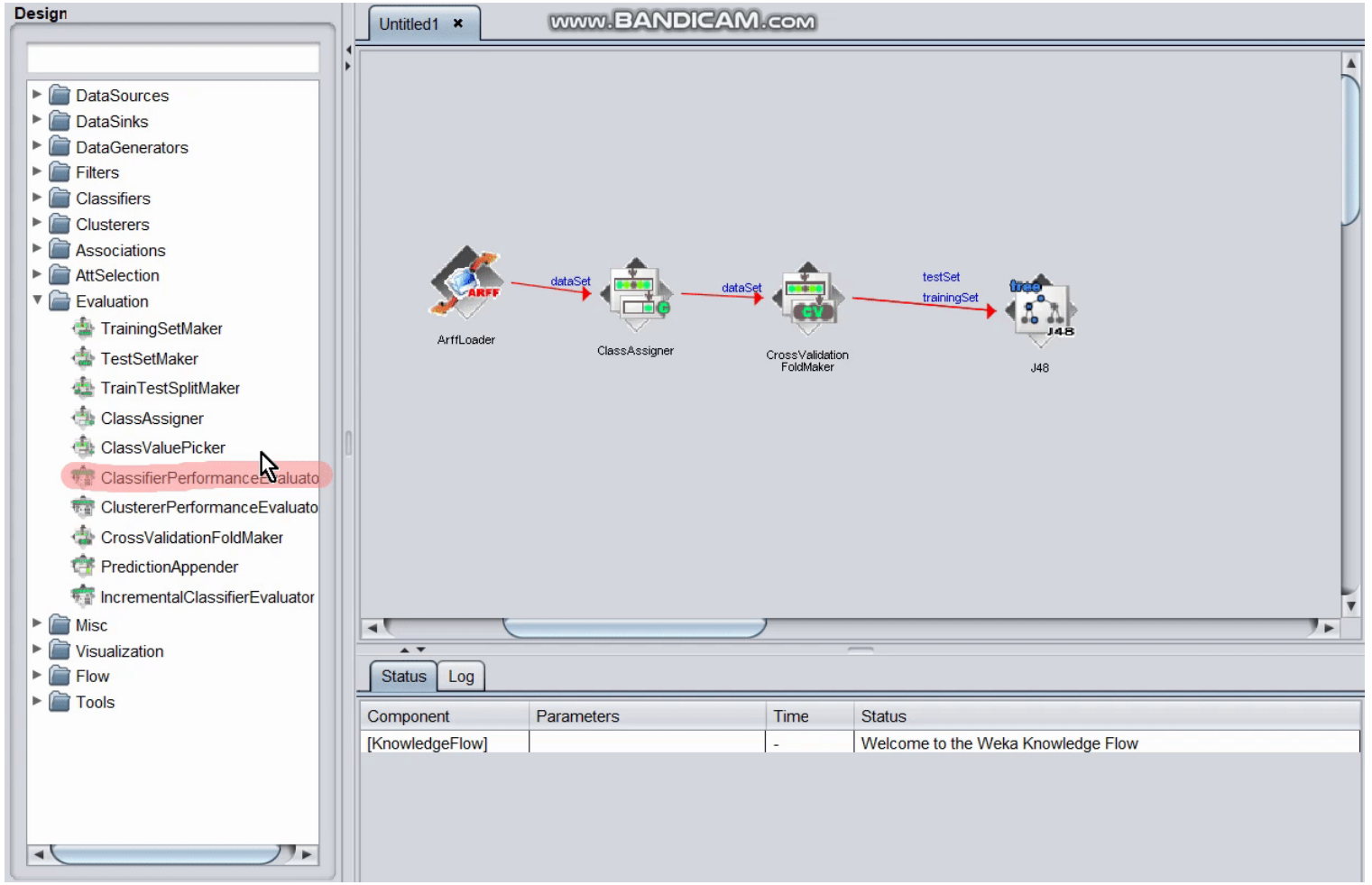
12.對CrossValisationFoldMaker元件單擊右鍵 → 在出現的選單中點選trainingSet → 左鍵單擊J48分類器元件四周的圓點進行連接

再次對CrossValisationFoldMaker元件單擊右鍵 → 在出現的選單中點選testSet → 左鍵單擊J48分類器元件四周的圓點進行連接



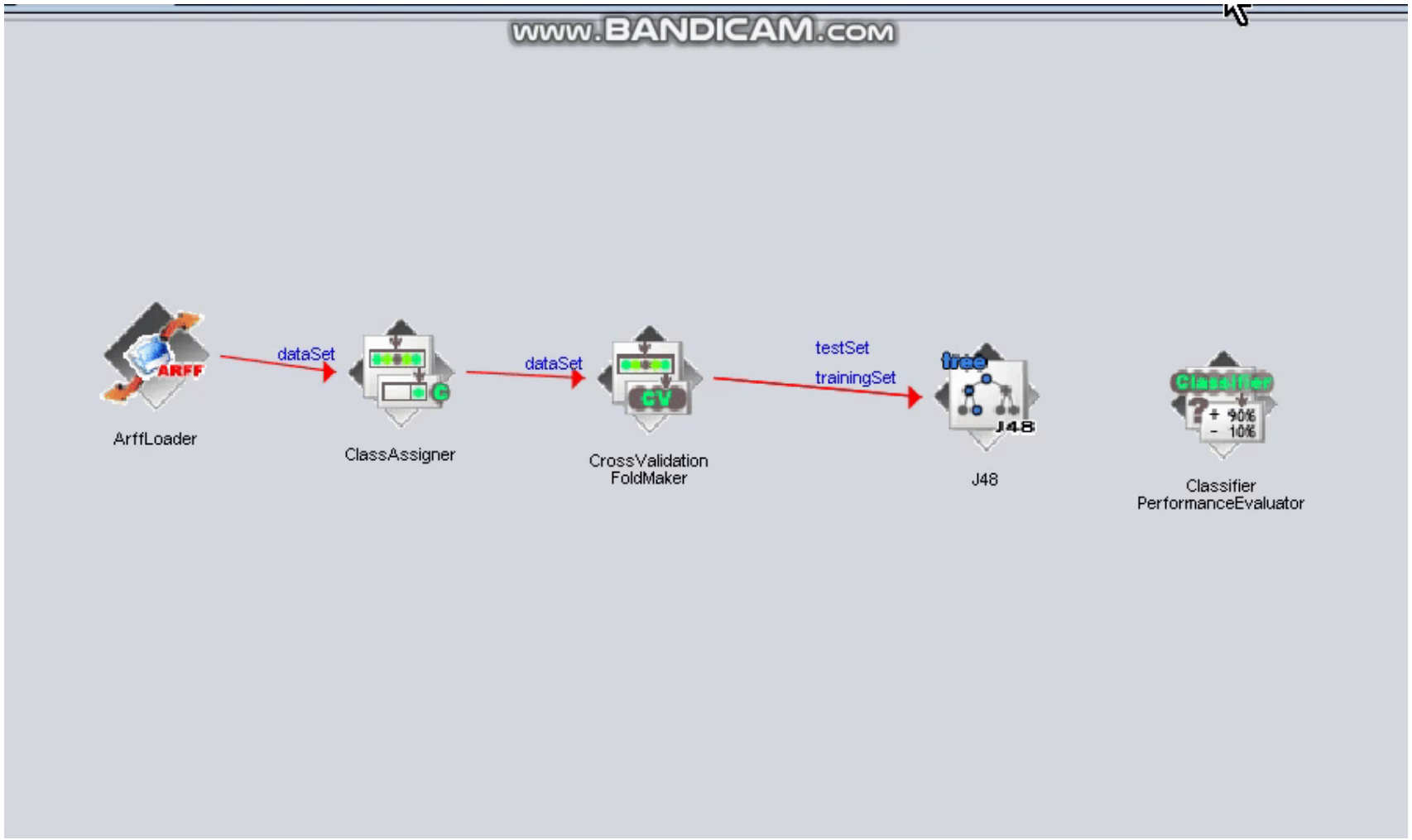
Lesson 1.4: 知識流介面

13.左鍵單擊Evaluation資料夾下的ClassifierPerformanceEvaluator元件，並放入右側畫布。



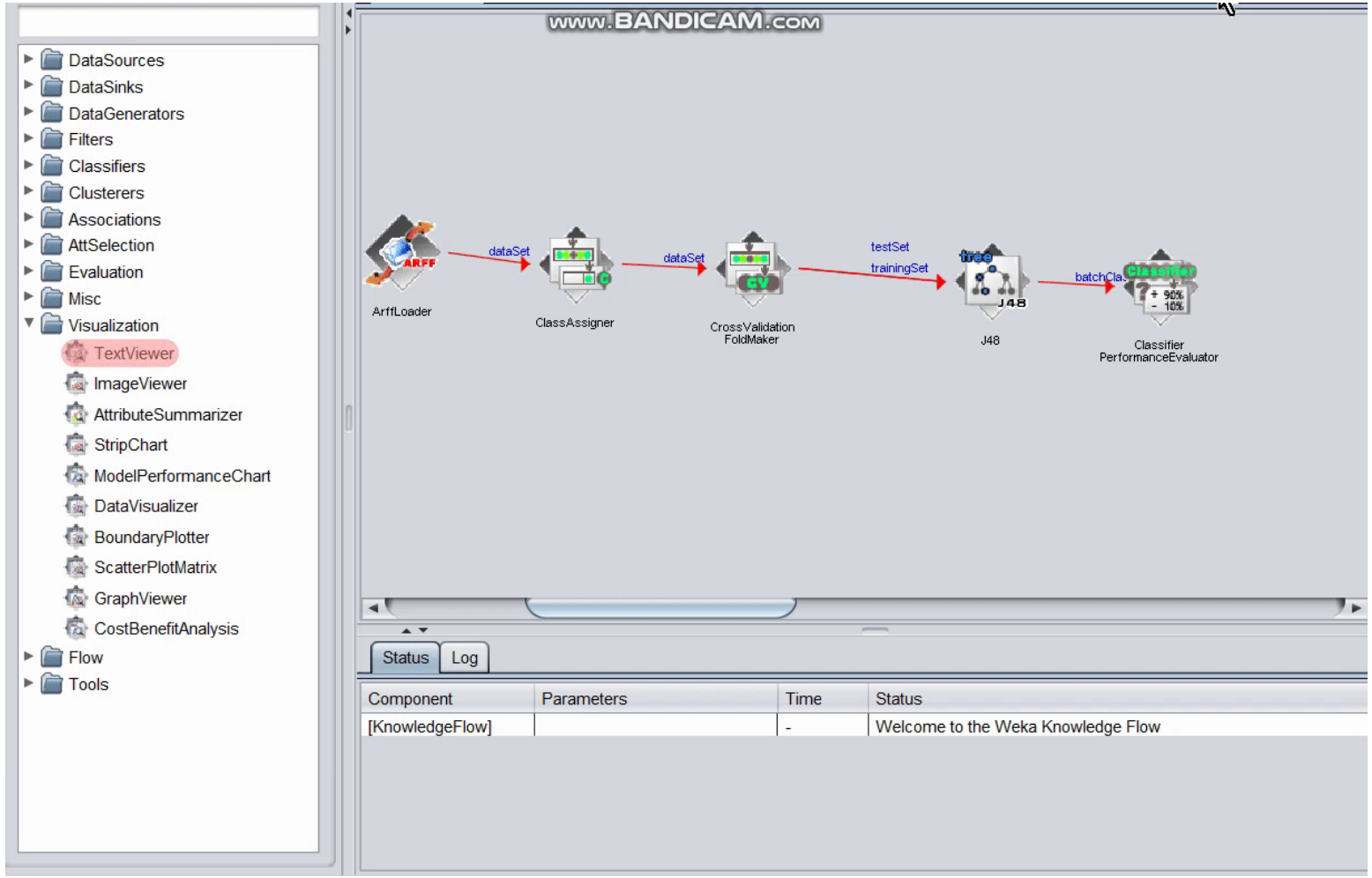
Lesson 1.4: 知識流介面

14.對J48分類器元件單擊右鍵 → 在出現的選單中點選batchClassifier → 左鍵單擊ClassifierPerformanceEvaluator元件四周的圓點進行連接



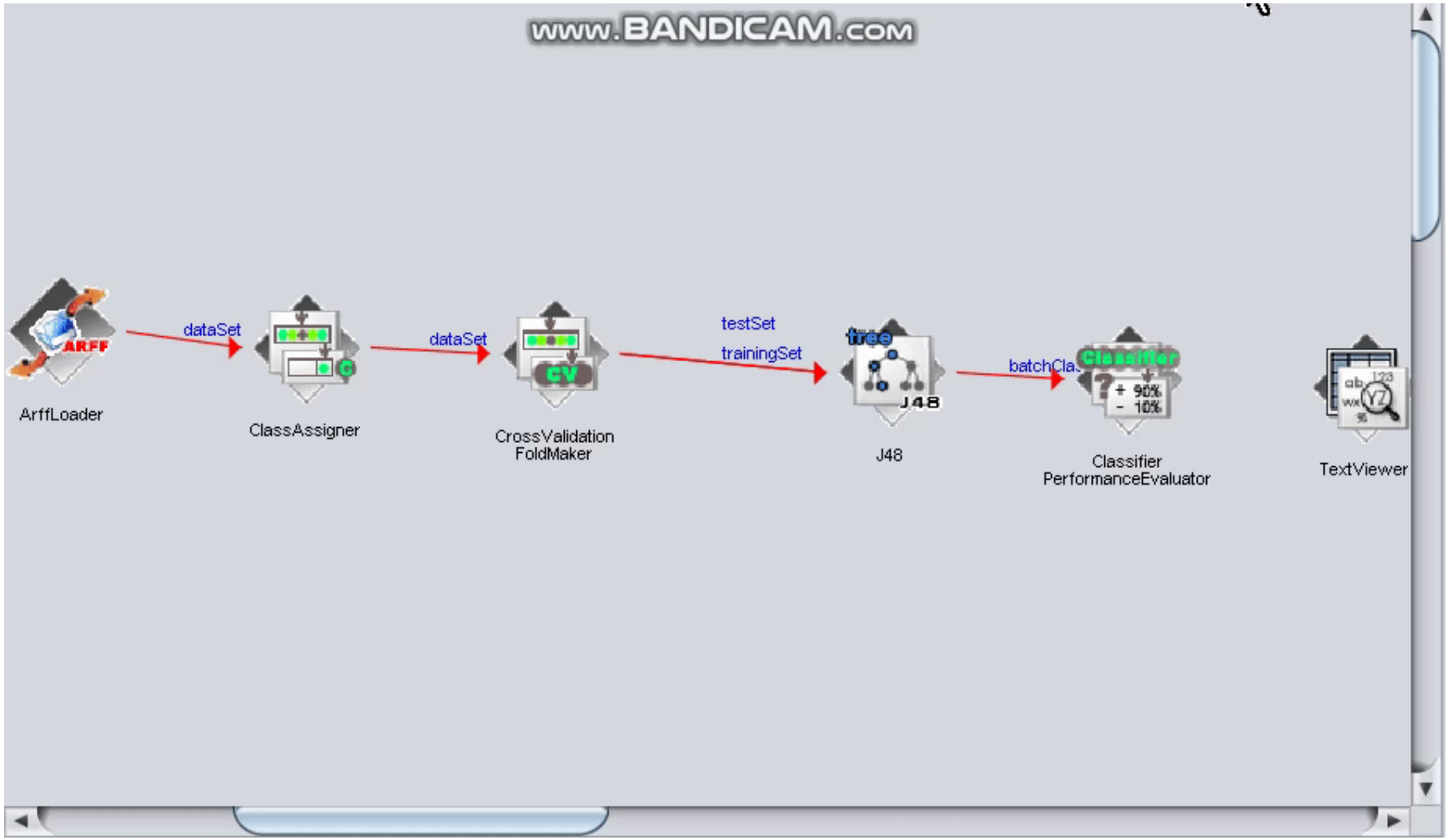
Lesson 1.4: 知識流介面

15.左鍵單擊Visualization資料夾下的TextViewer元件，並放入右側畫布，我們用它來做文本輸出。



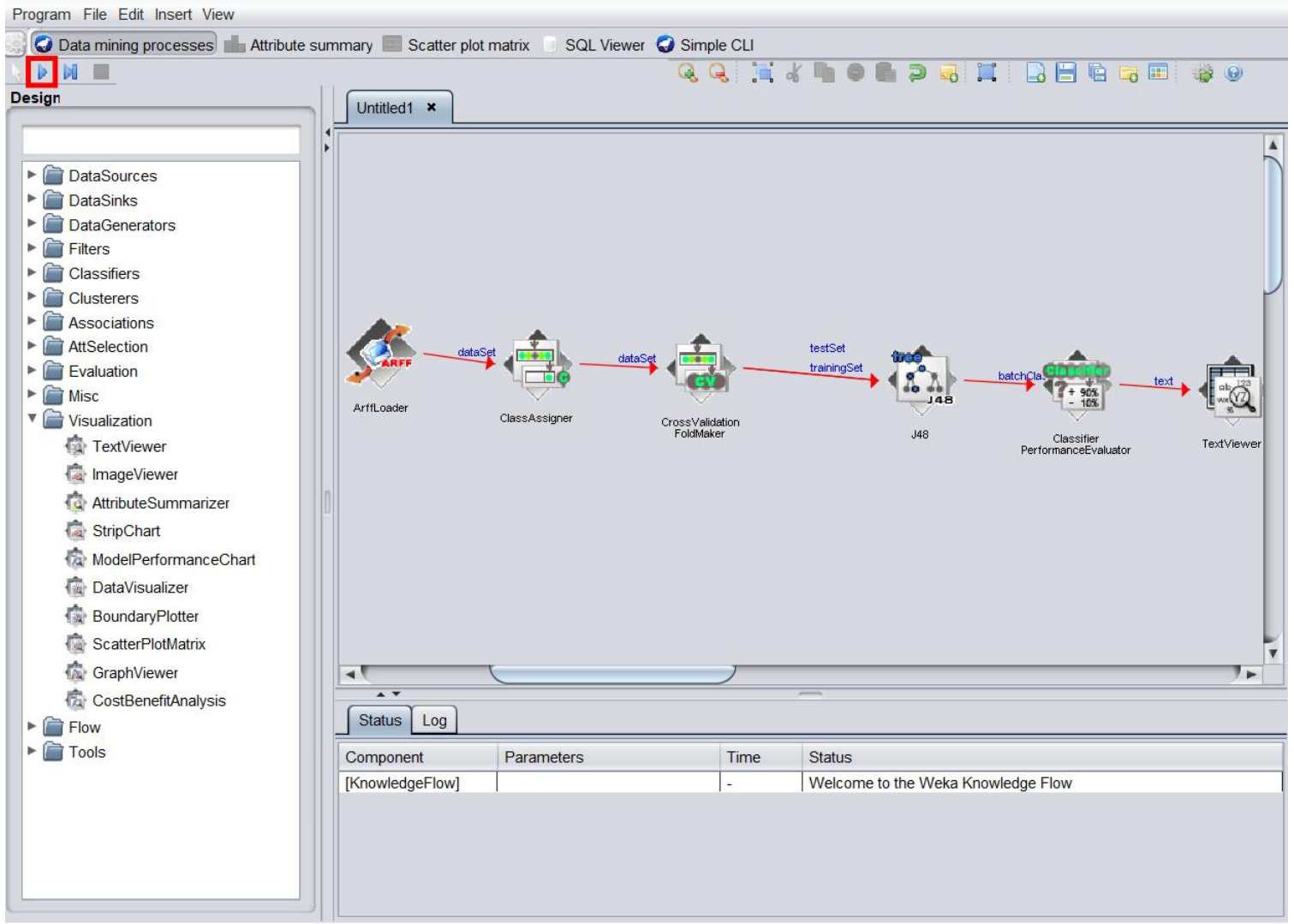
Lesson 1.4: 知識流介面

16.對ClassifierPerformanceEvaluator元件單擊右鍵 → 在出現的選單中點選text → 左鍵單擊TextViewer元件四周的圓點進行連接



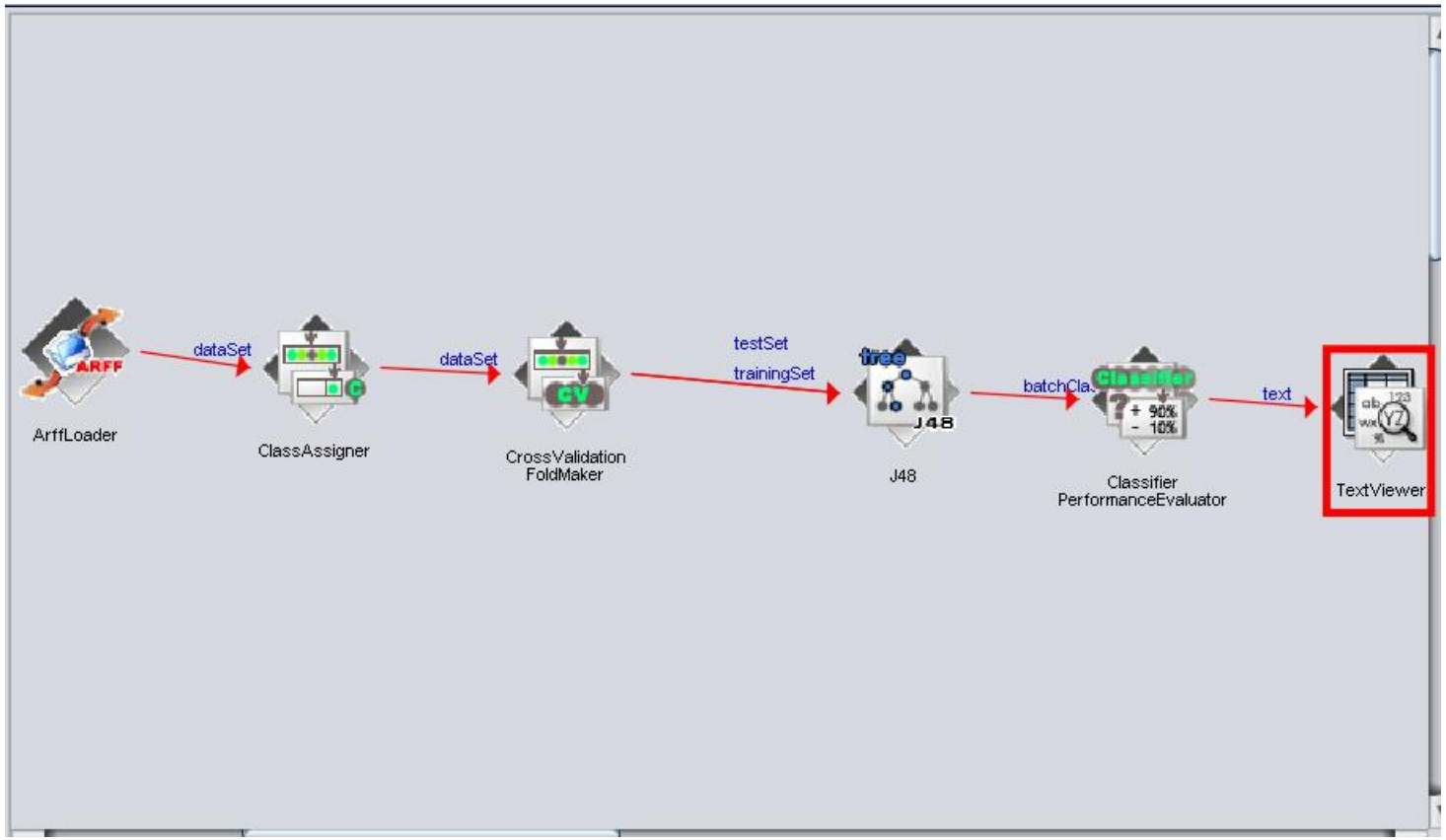
Lesson 1.4: 知識流介面

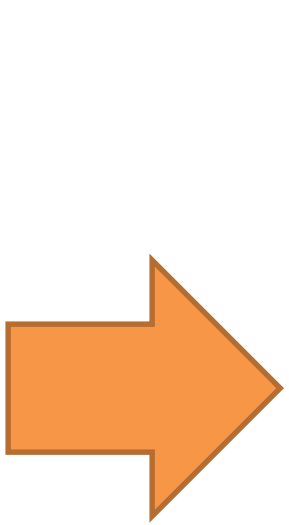
17.左鍵單擊視窗左上角的運行圖示 運行。

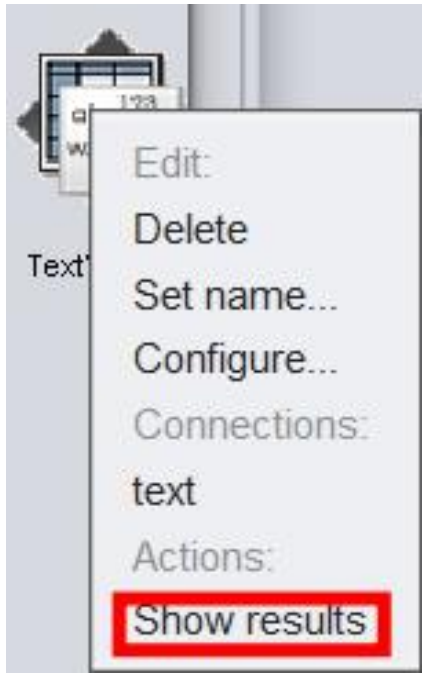


Lesson 1.4: 知識流介面

18. 右鍵單擊TextViewer元件，在出現的選單中左鍵單擊Show results

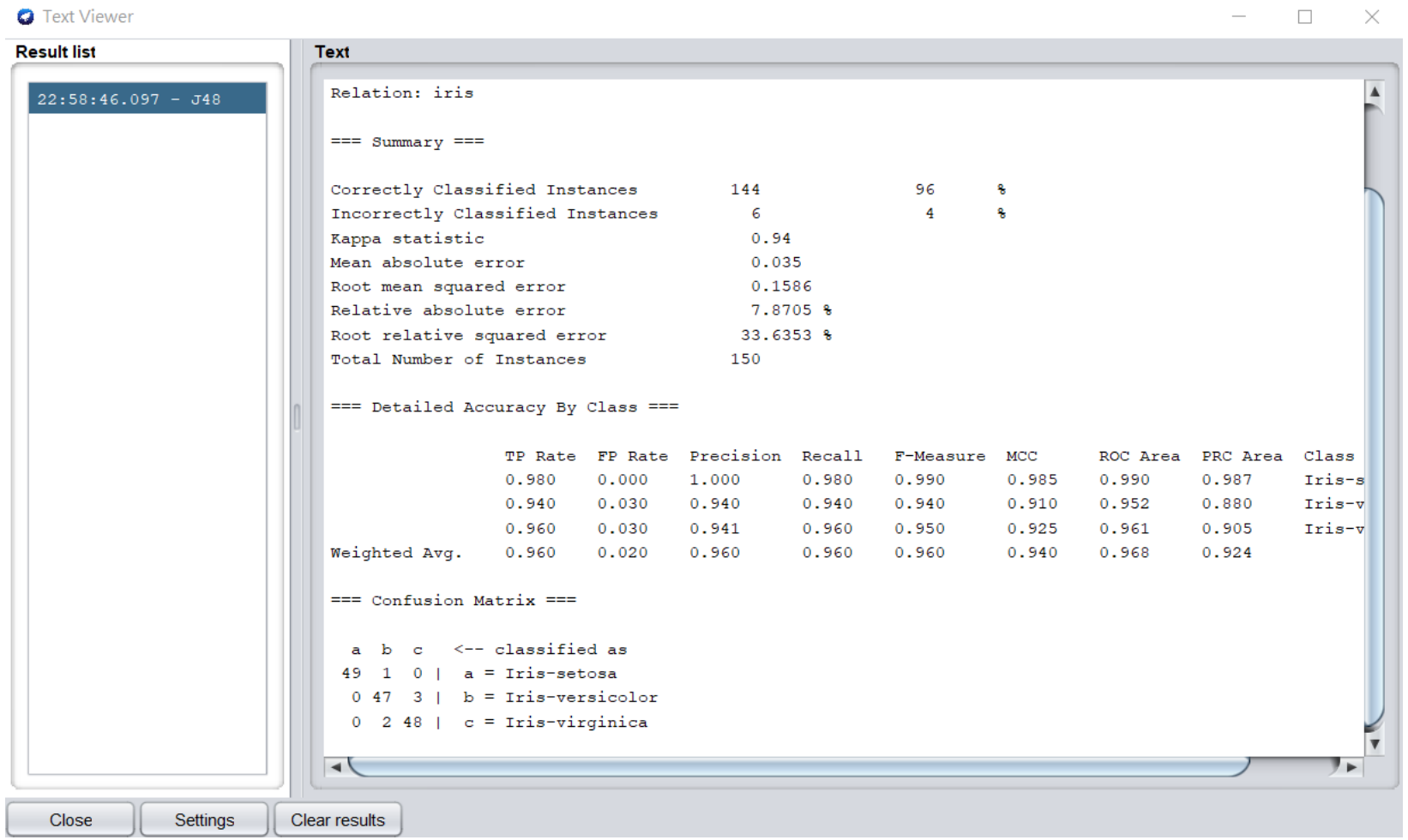






Lesson 1.4: 知識流介面

▼執行結果



Lesson 1.4: 知識流介面

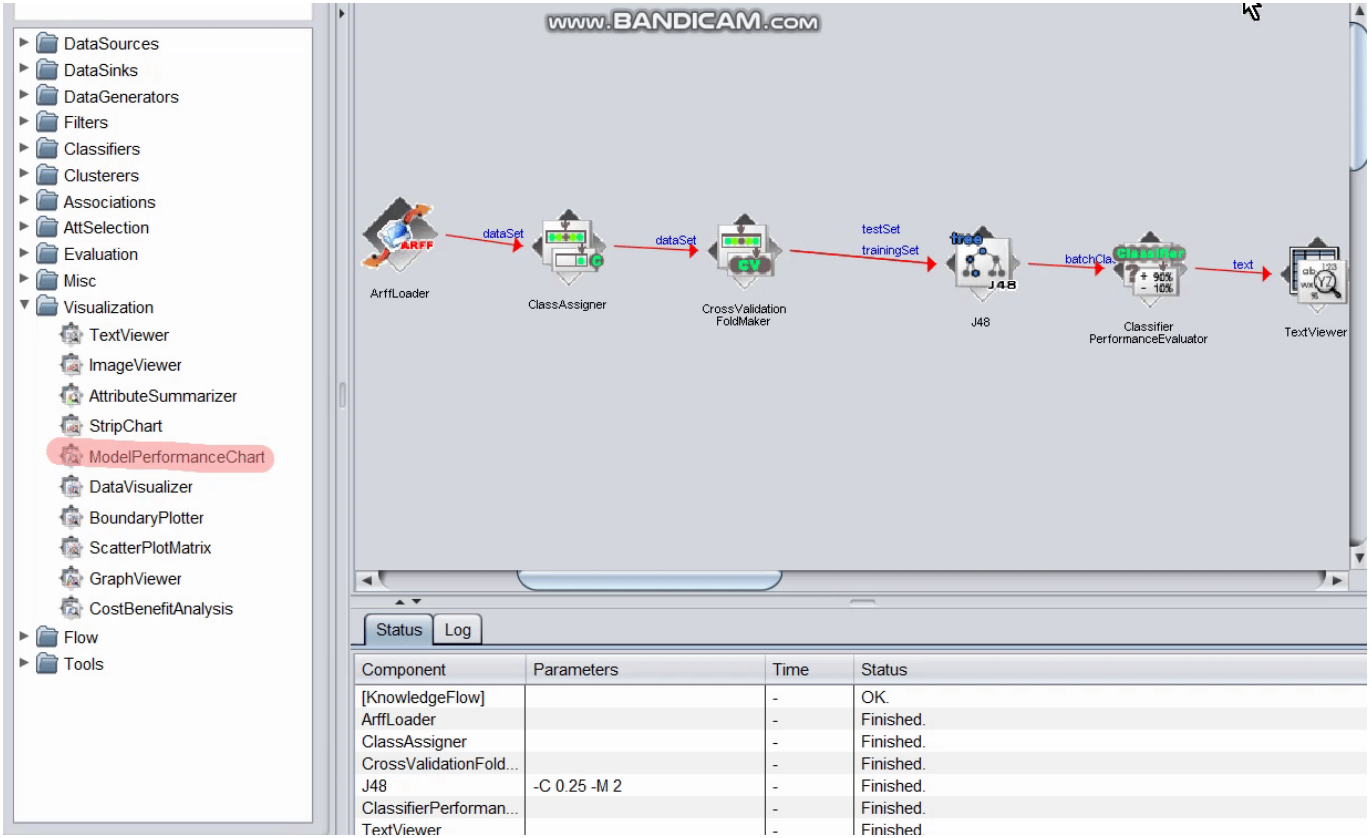
載入ARFF檔案, 選擇J48, 使用交叉驗證來評估

	工具箱
❖ 選擇一個*ArffLoader元件*; 使用Configure來設定*iris.arff檔案*	*DataSources*
❖ 連結*ClassAssigner，用來選擇類別*	*Evaluation*
❖ 將結果連結到*CrossValidationFoldMaker*	*Evaluation*
❖ 再連結到*J48*	*Classifiers*
❖ 操作兩個連結, 一個連結*trainingSet，另一個連結testSet*	
❖ 連結J48到*ClassifierPerformanceEvaluator*	*Evaluation*
❖ 再連結到*TextViewer*	*Visualization*

接著，執行！ (使用  來運行)

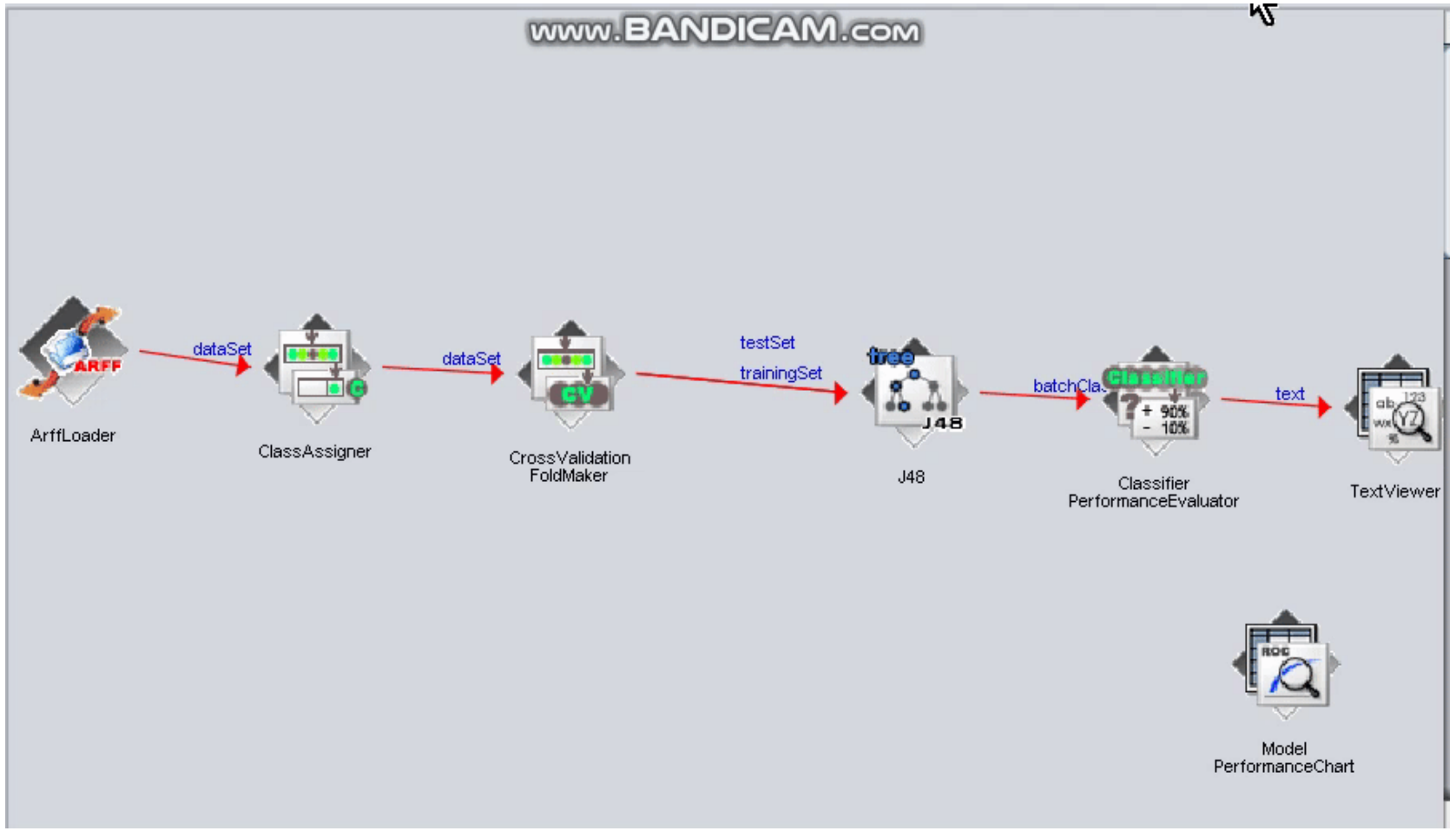
Lesson 1.4: 知識流介面

1.左鍵單擊Visualization資料夾下的ModelPerformanceChart元件，並放入右側畫布。



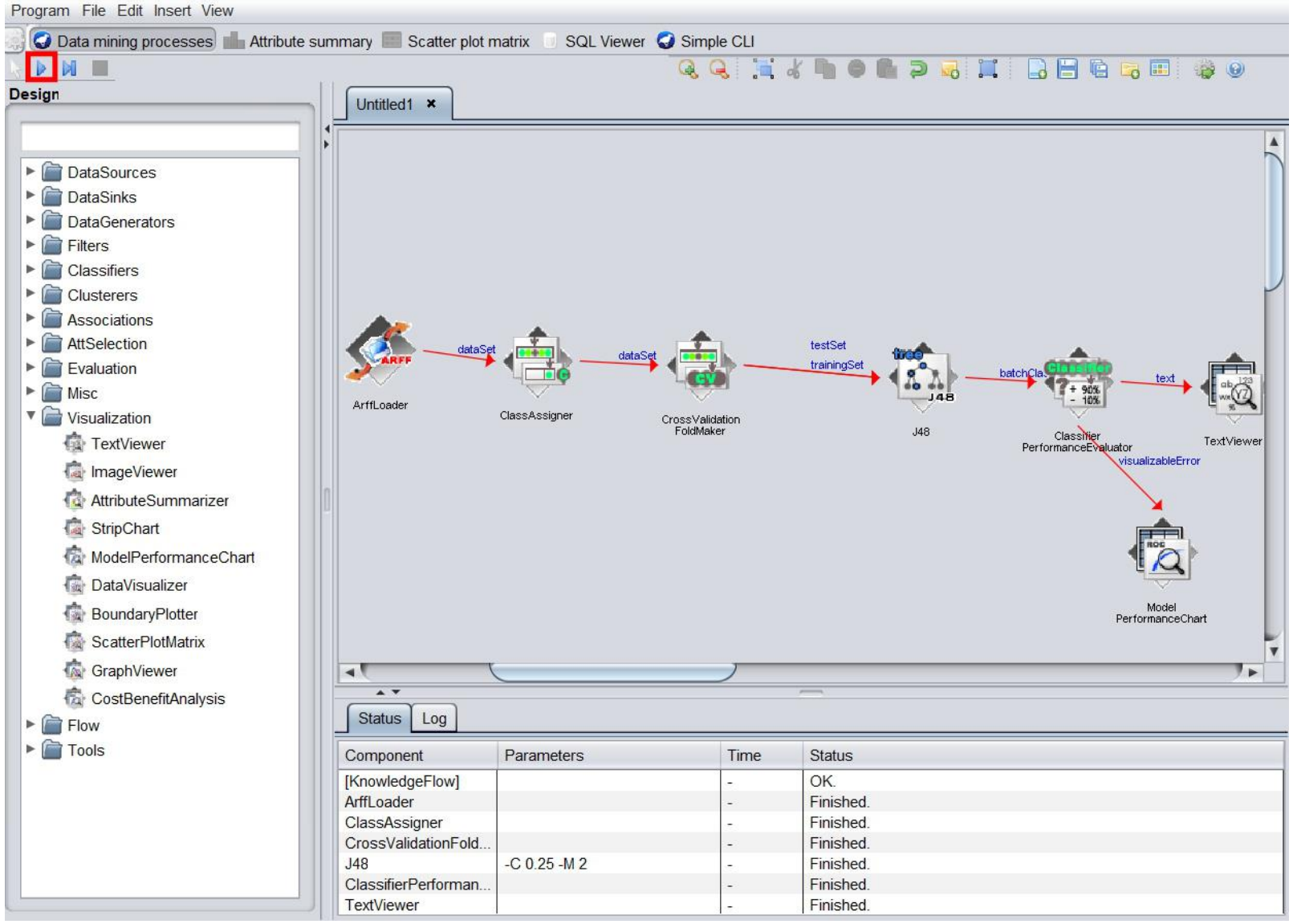
Lesson 1.4: 知識流介面

2.對ClassifierPerformanceEvaluator元件單擊右鍵 → 在出現的選單中點選visualizableError → 左鍵單擊ModelPerformanceChart元件四周的圓點進行連接



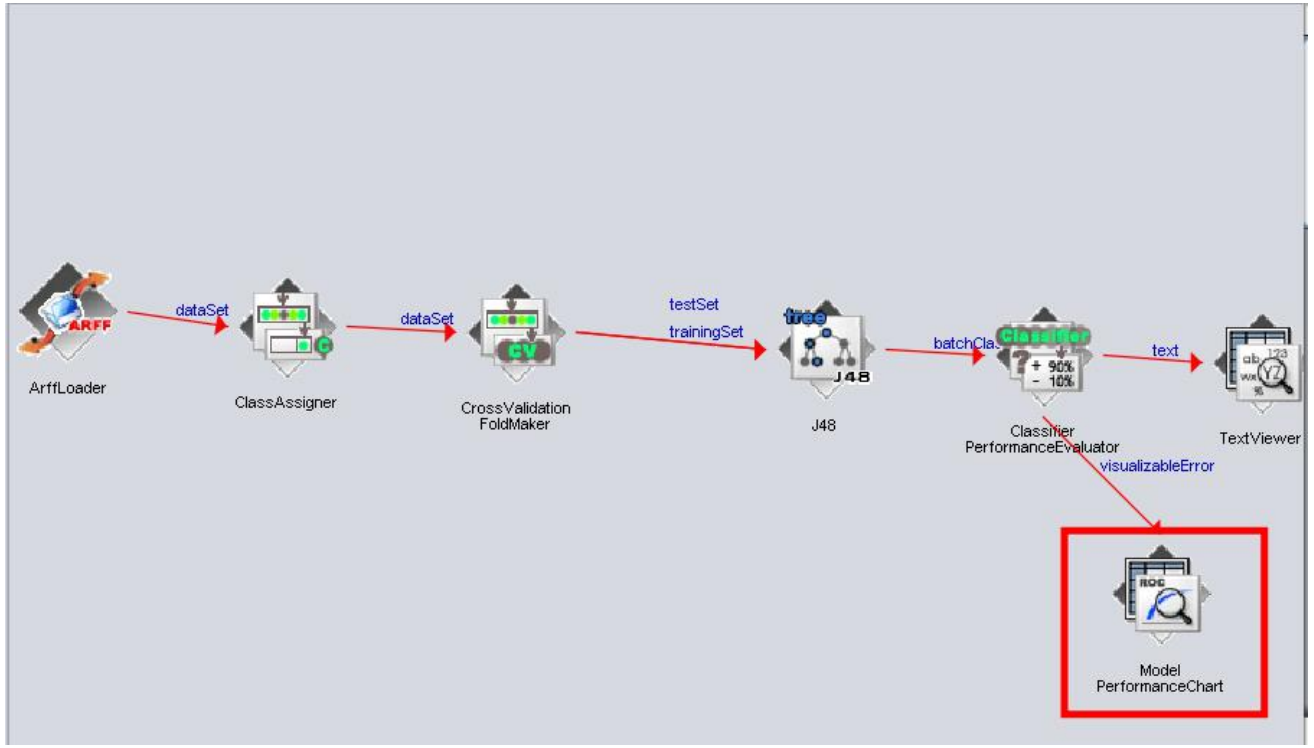
Lesson 1.4: 知識流介面

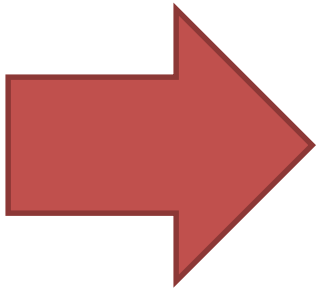
3.左鍵單擊視窗左上角的運行圖示 運行。

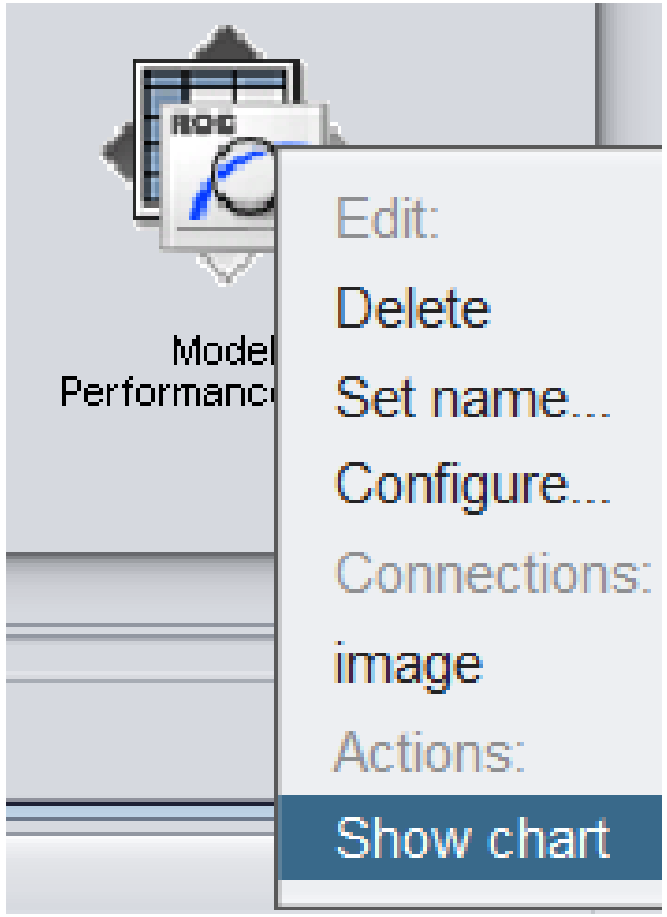


Lesson 1.4: 知識流介面

4.右鍵單擊ModelPerformanceChart元件 → 在選單中左鍵單擊Show chart選項

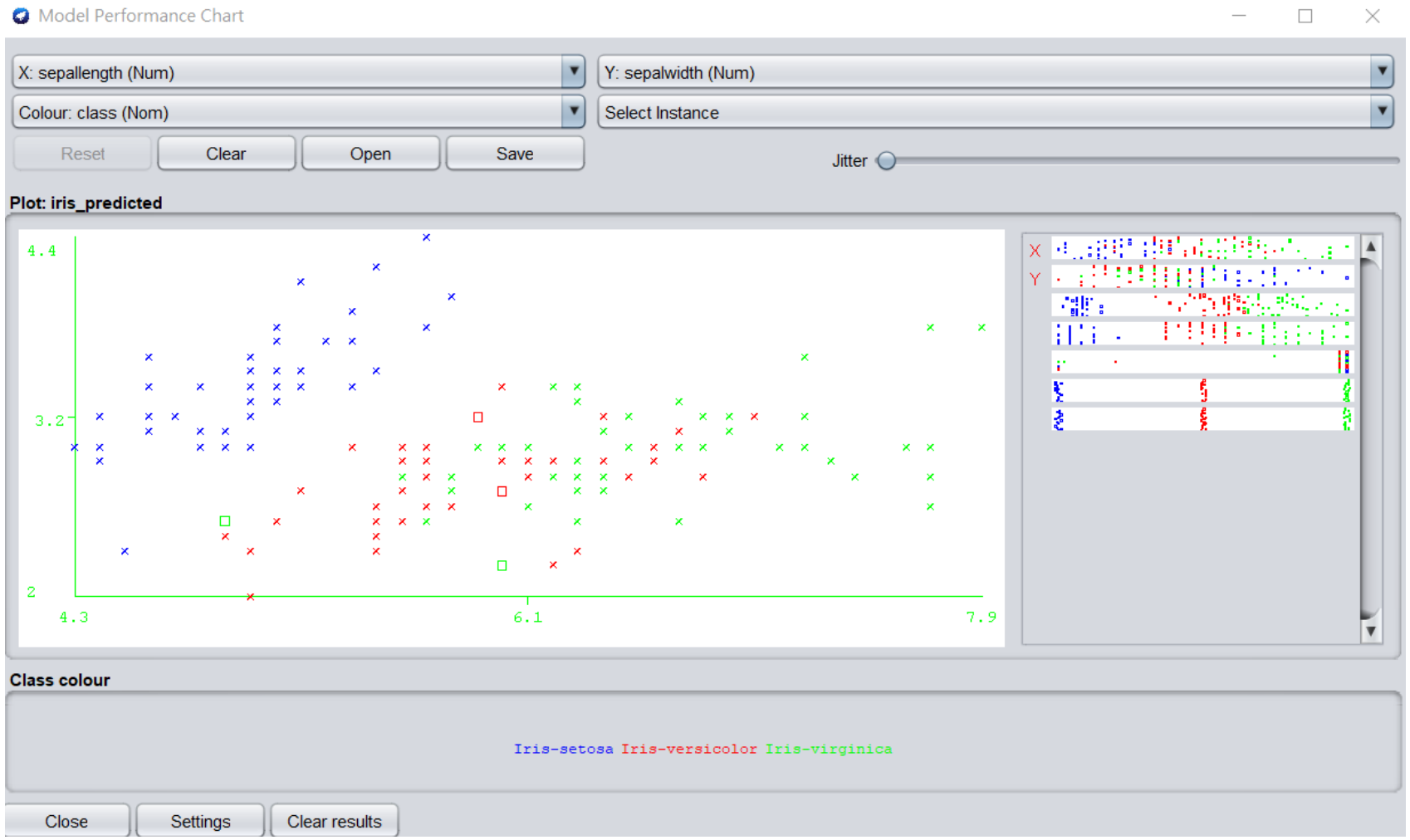






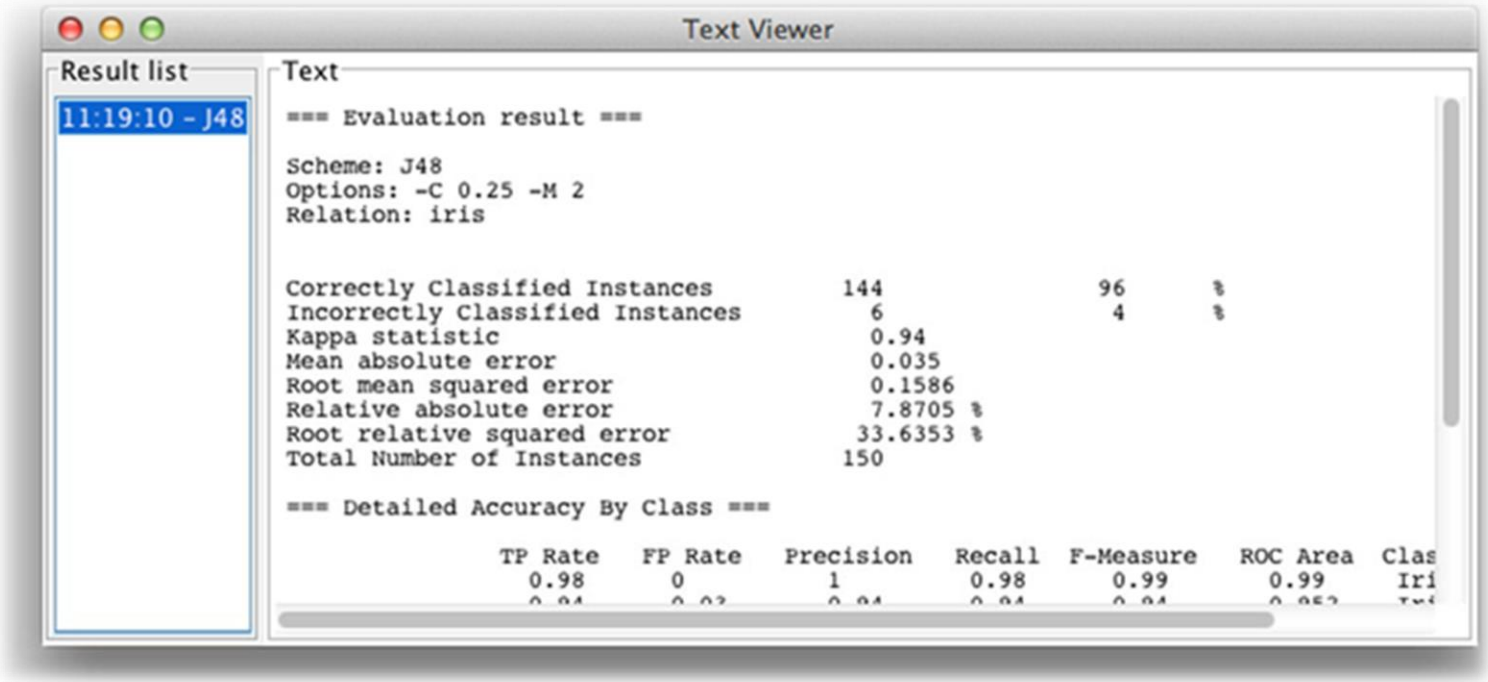
Lesson 1.4: 知識流介面

▼執行結果



Lesson 1.4: 知識流介面

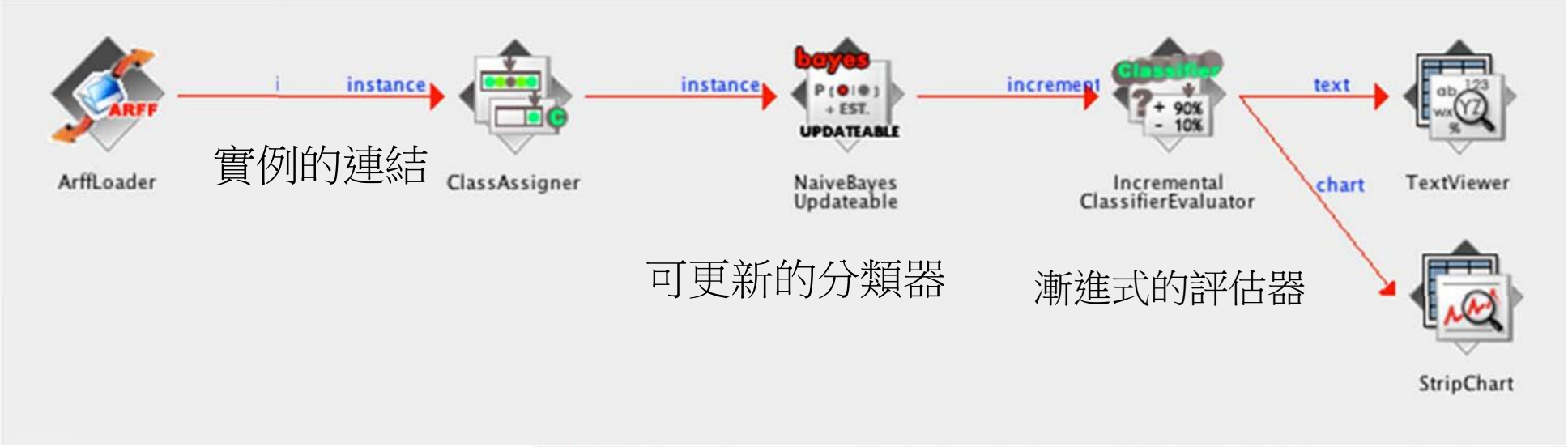
- *TextViewer*: 顯示結果

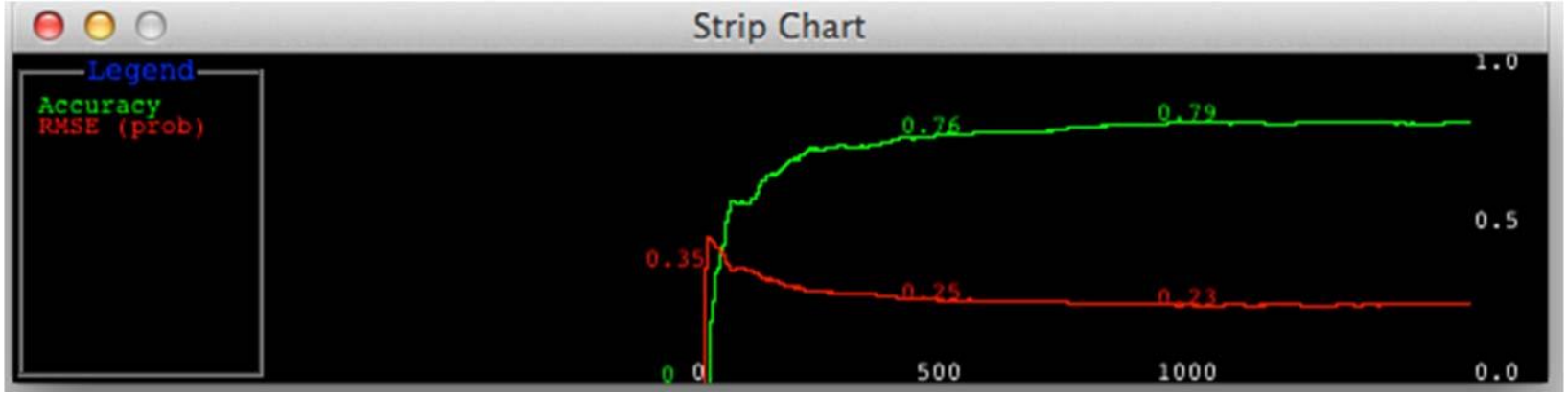


- *加入一個ModelPerformanceChart*
- *將ClassifierPerformanceEvaluator的輸出連結到visualizableError*
- 顯示圖表(需要再次執行)

Lesson 1.4: 知識流介面

接著，我們試著處理串流資料



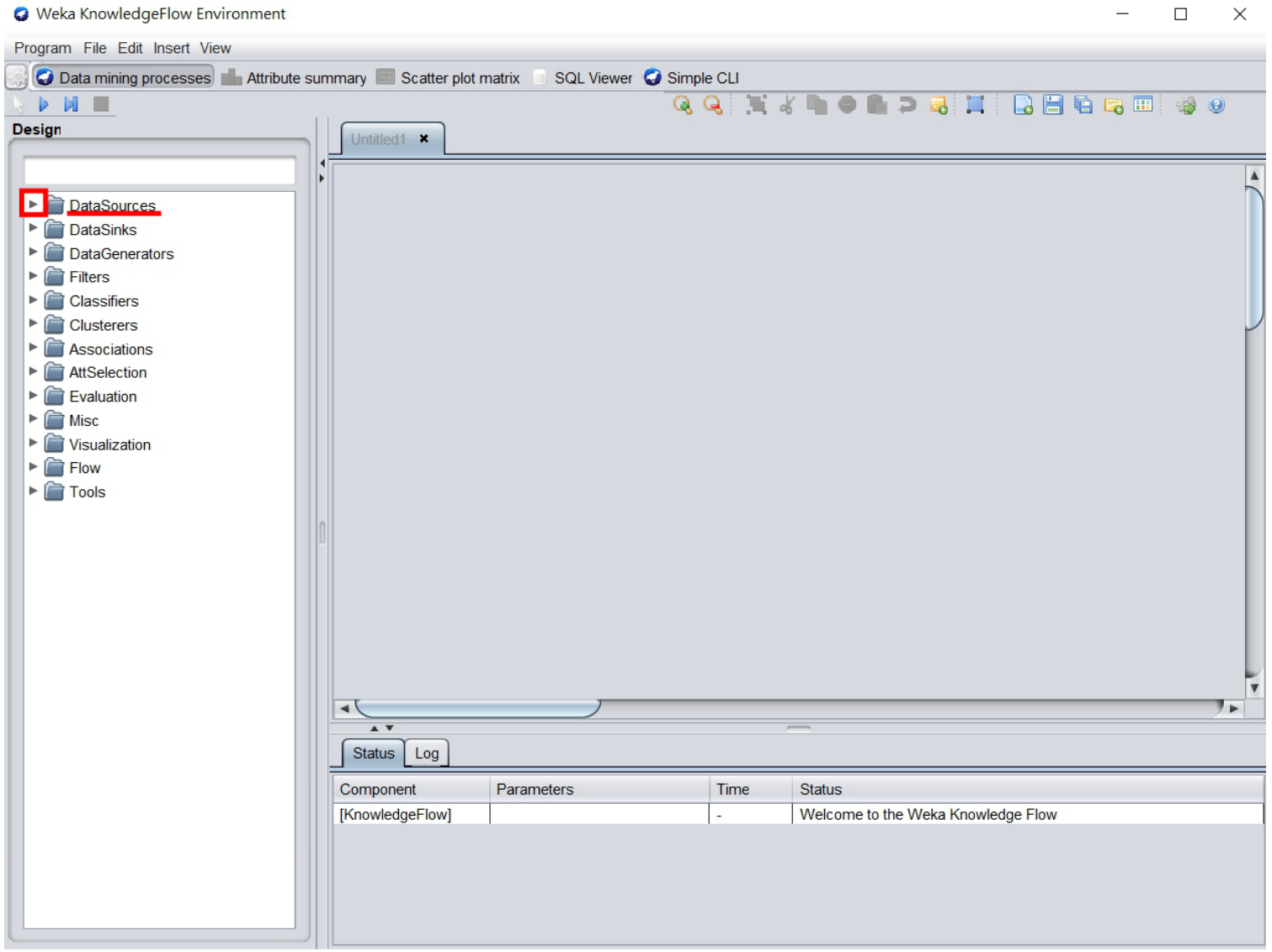


Lesson 1.4: 知識流介面

1. 開啟Weka程式，於Weka GUI Chooser介面左鍵單擊KnowledgeFlow按鈕

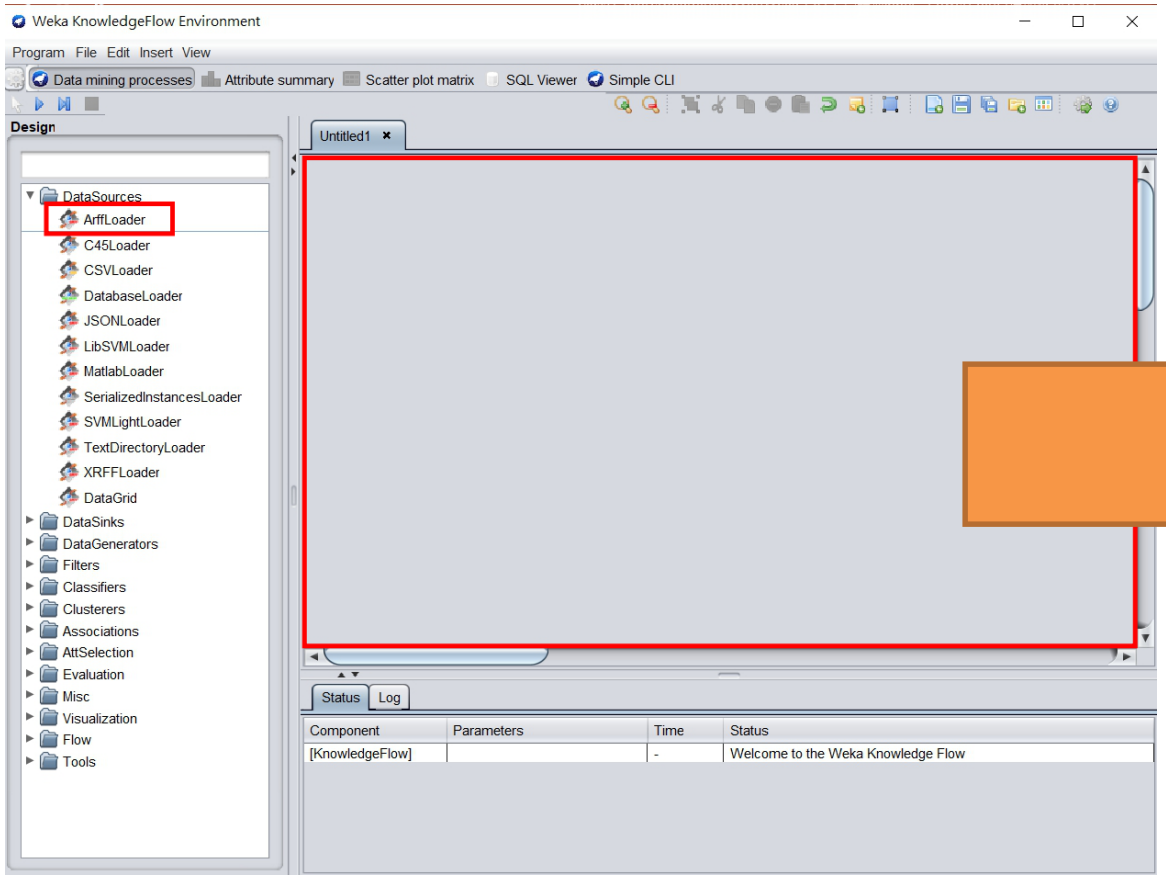
Lesson 1.4: 知識流介面

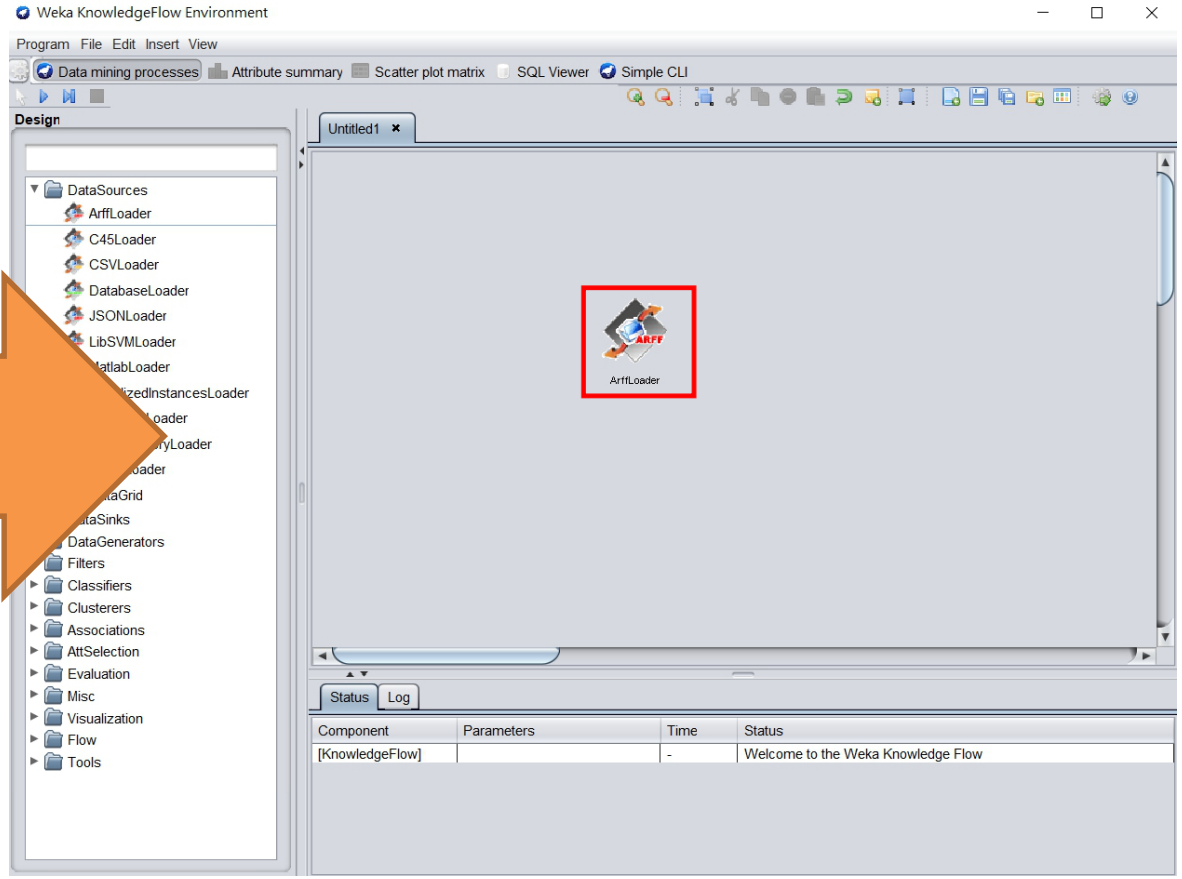
2.左鍵單擊DataSources資料夾前方的展開圖示



Lesson 1.4: 知識流介面

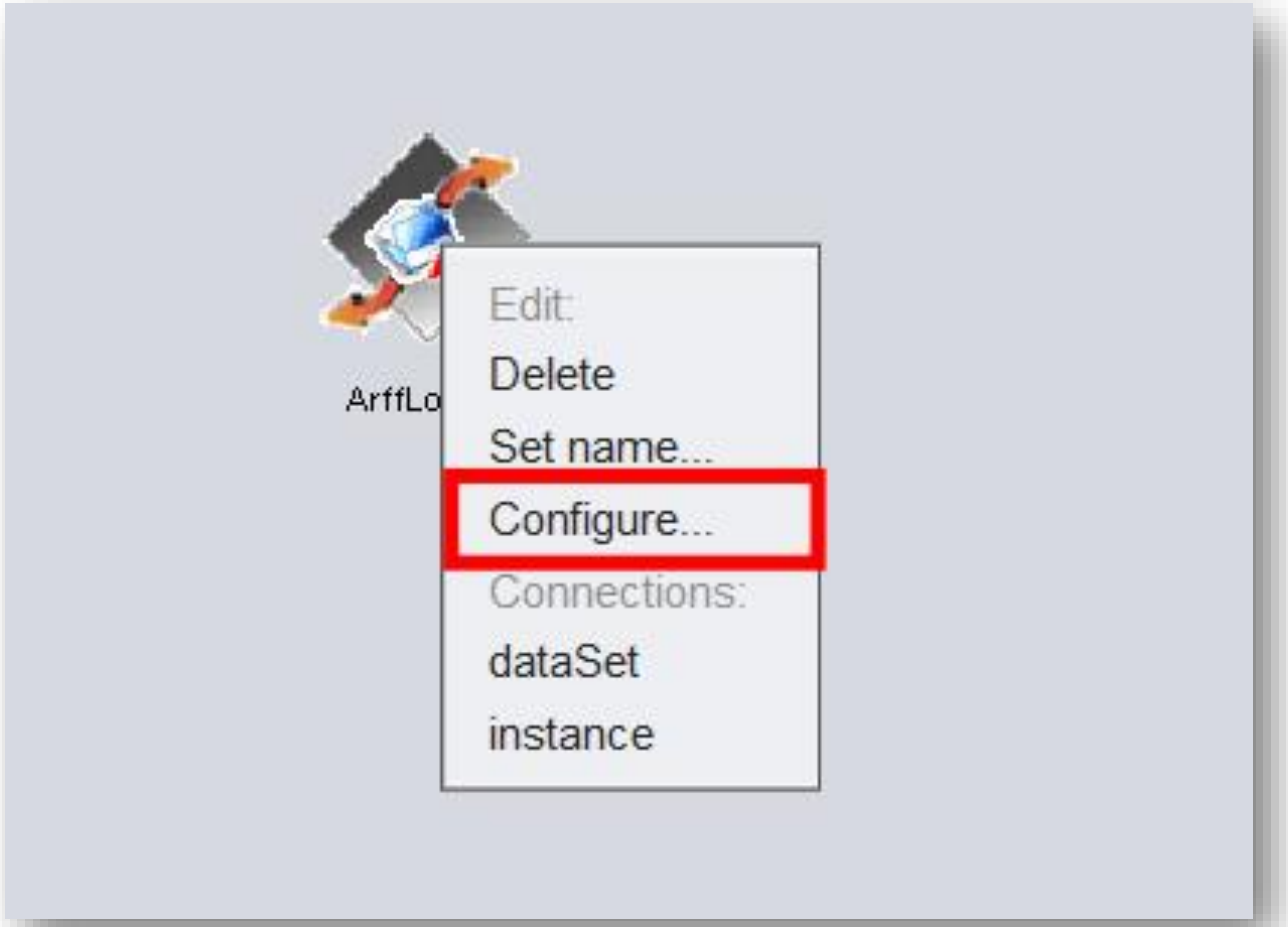
3.左鍵單擊ArffLoader元件，於右方畫布中單擊左鍵將其置入。





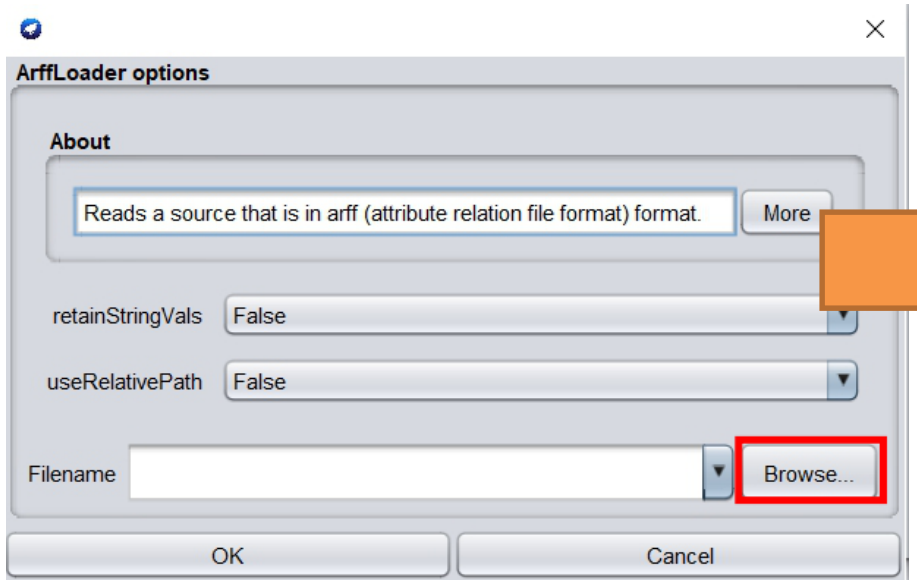
Lesson 1.4: 知識流介面

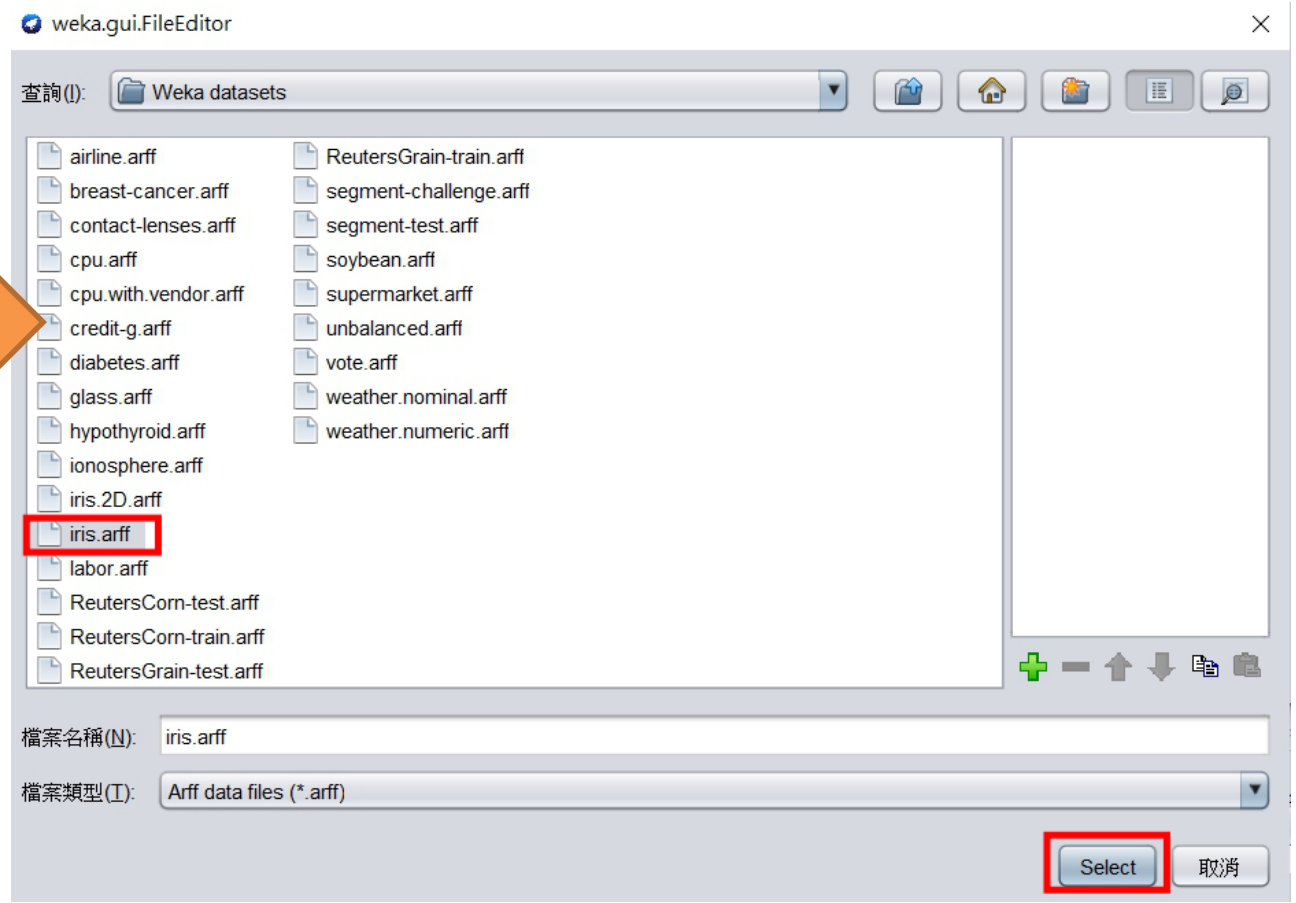
4. 右鍵單擊ArffLoader元件，在選單中左鍵單擊Configure。



Lesson 1.4: 知識流介面

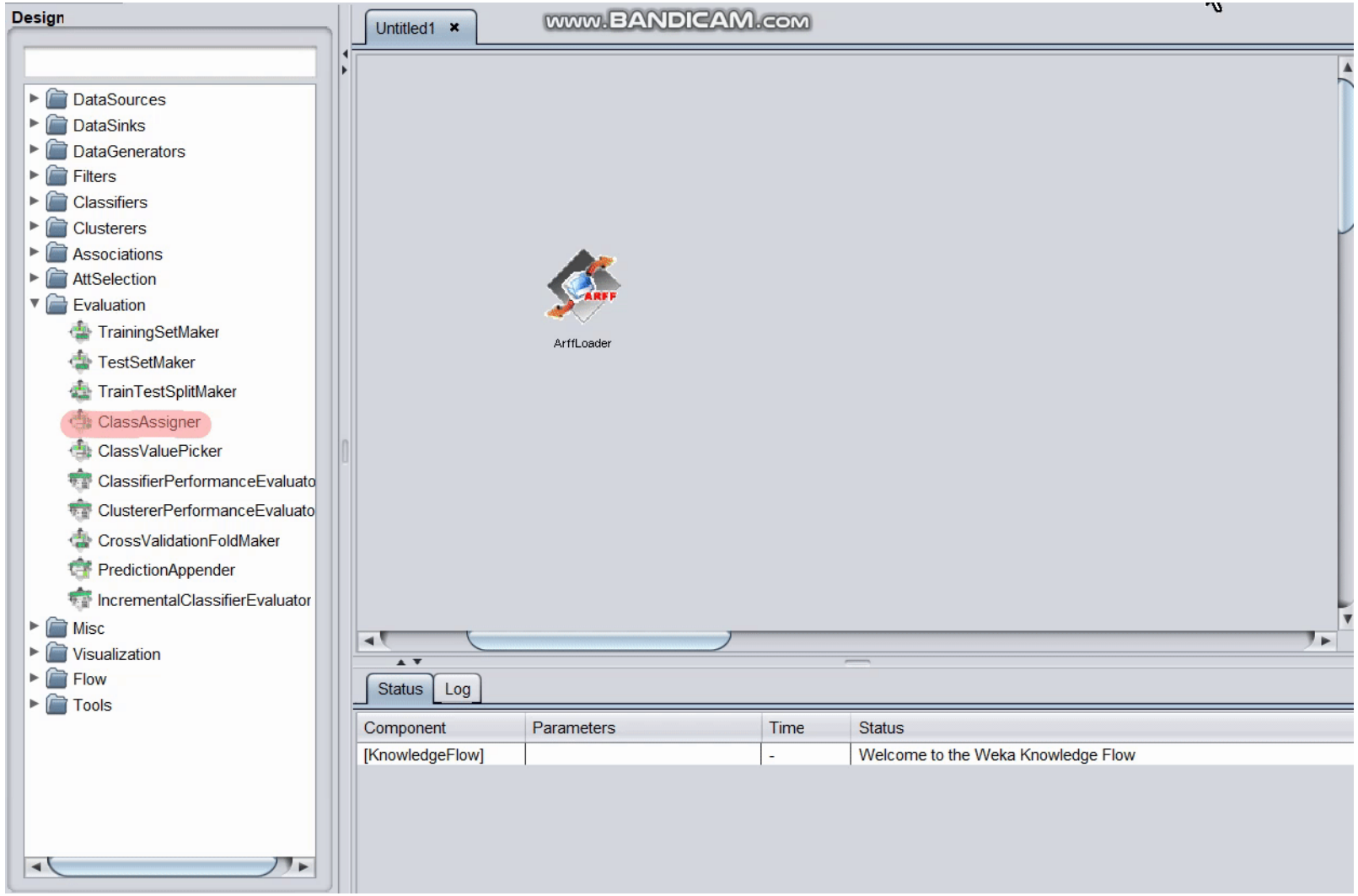
5.在出現的視窗中左鍵單擊Browse按鈕，進入自行複製的Weka datasets資料夾 → 左鍵單擊iris.arff檔案 → 左鍵單擊視窗右下方的Select按鈕。





Lesson 1.4: 知識流介面

6.左鍵單擊Evaluation資料夾下的ClassAssigner元件，並放入右側畫布。

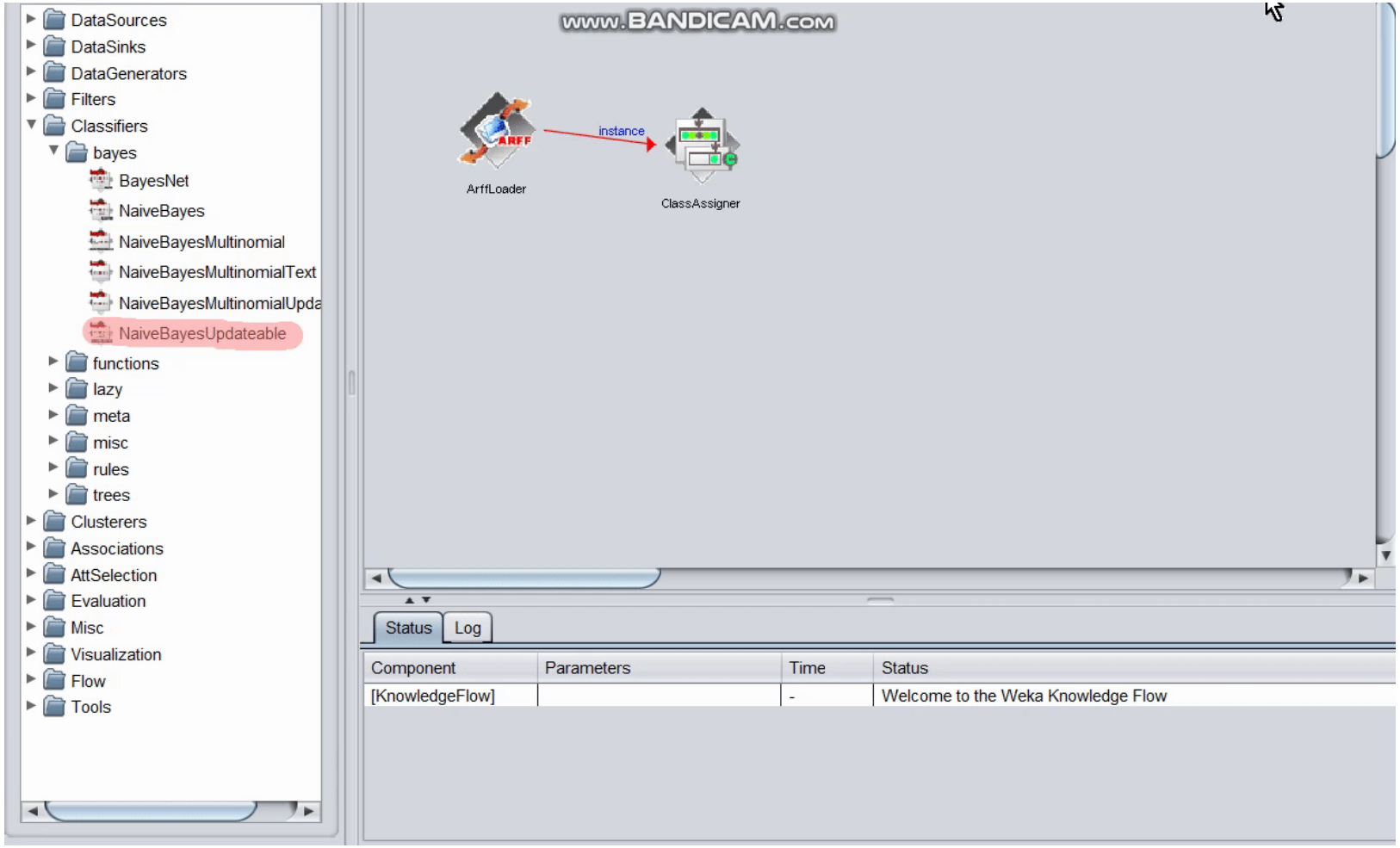


Lesson 1.4: 知識流介面

7.對ArffLoader元件單擊右鍵 → 在出現的選單中點選instance → 左鍵單擊ClassAssigner元件四周的圓點進行連接

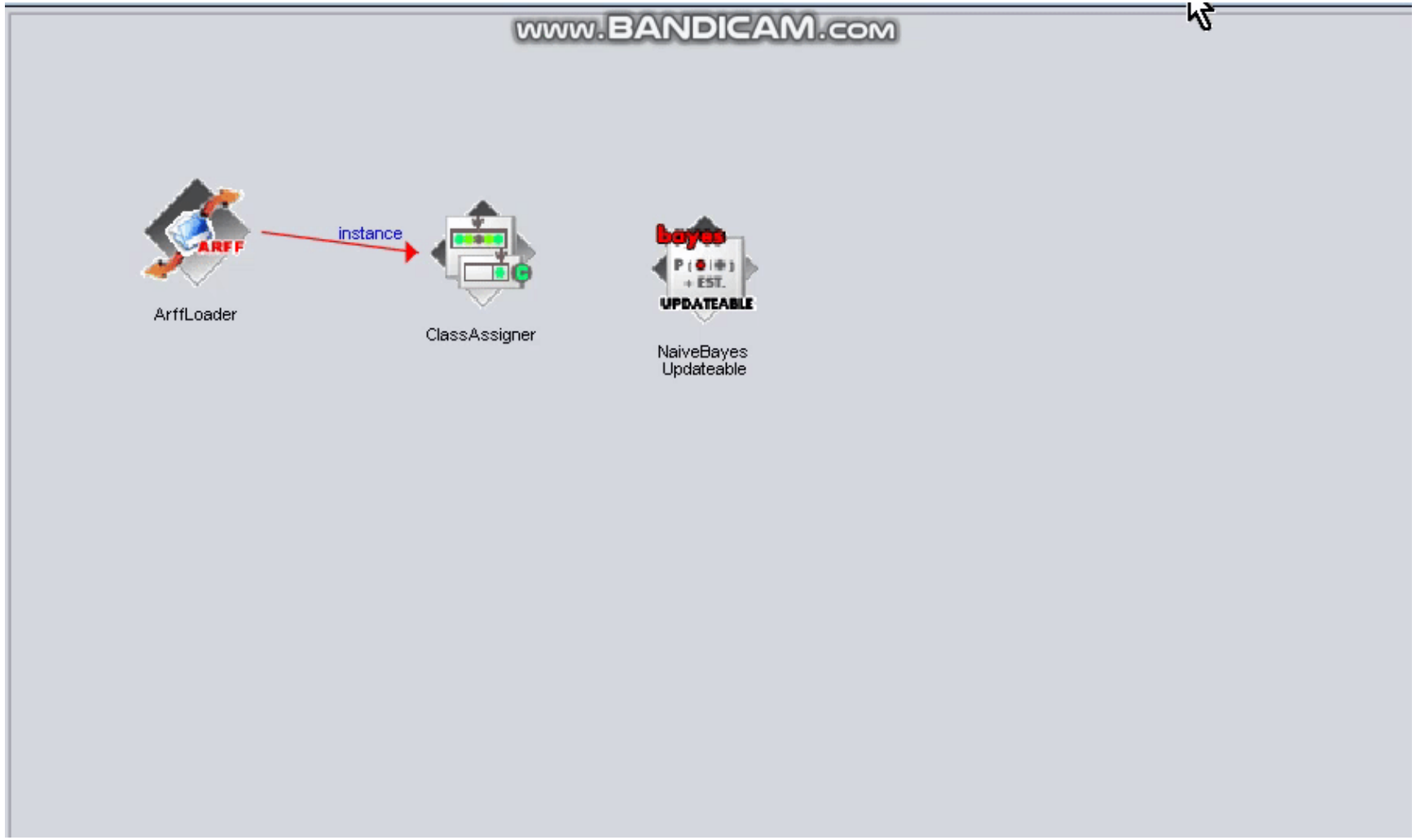
Lesson 1.4: 知識流介面

9.左鍵單擊Classifier資料夾下bayes資料夾中的NaiveBayesUpdateable分類器元件，並放入右側畫布。



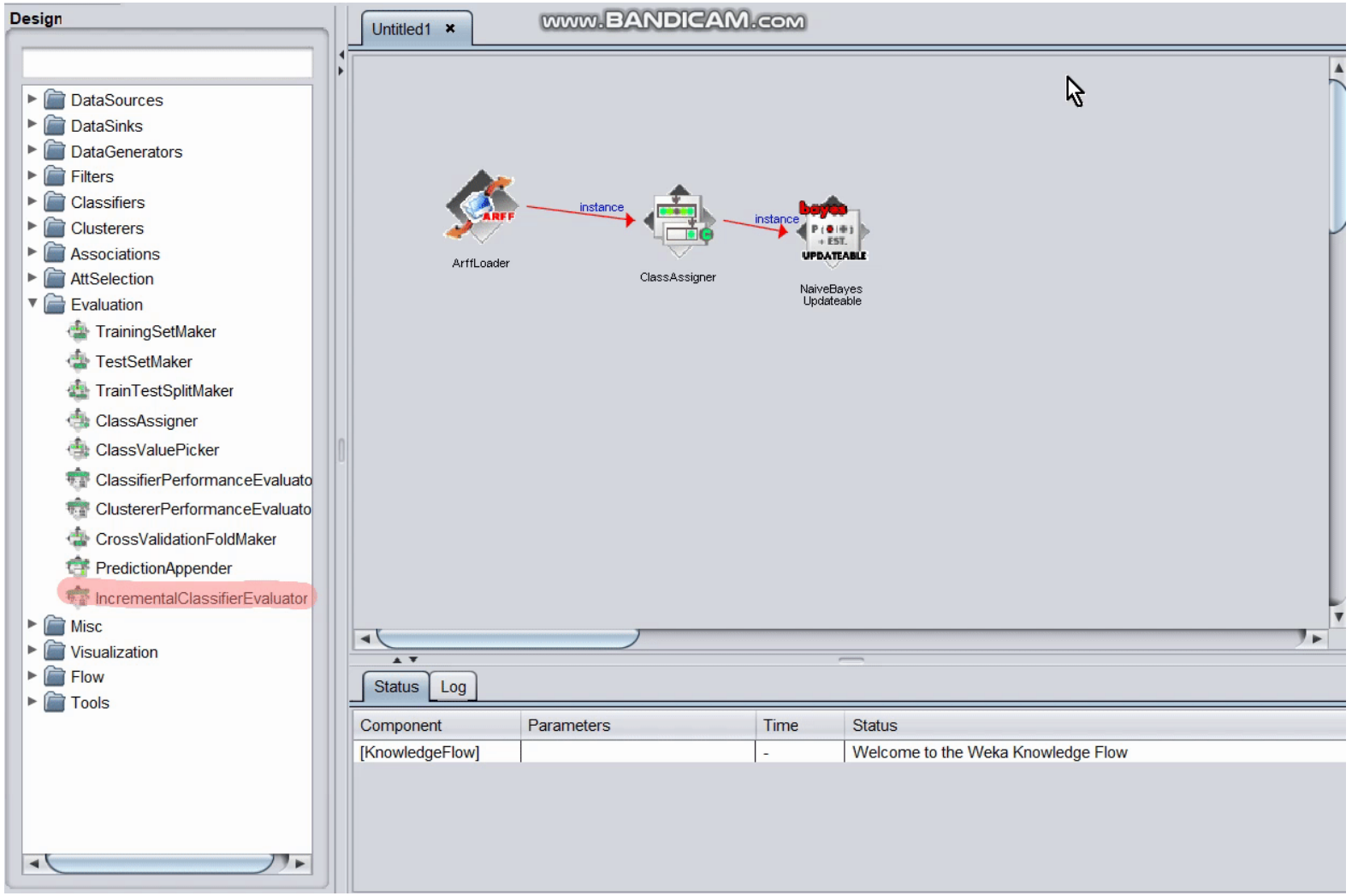
Lesson 1.4: 知識流介面

10.對ClassAssigner元件單擊右鍵 → 在出現的選單中點選instance → 左鍵單擊NaiveBayesUpdateable元件四周的圓點進行連接



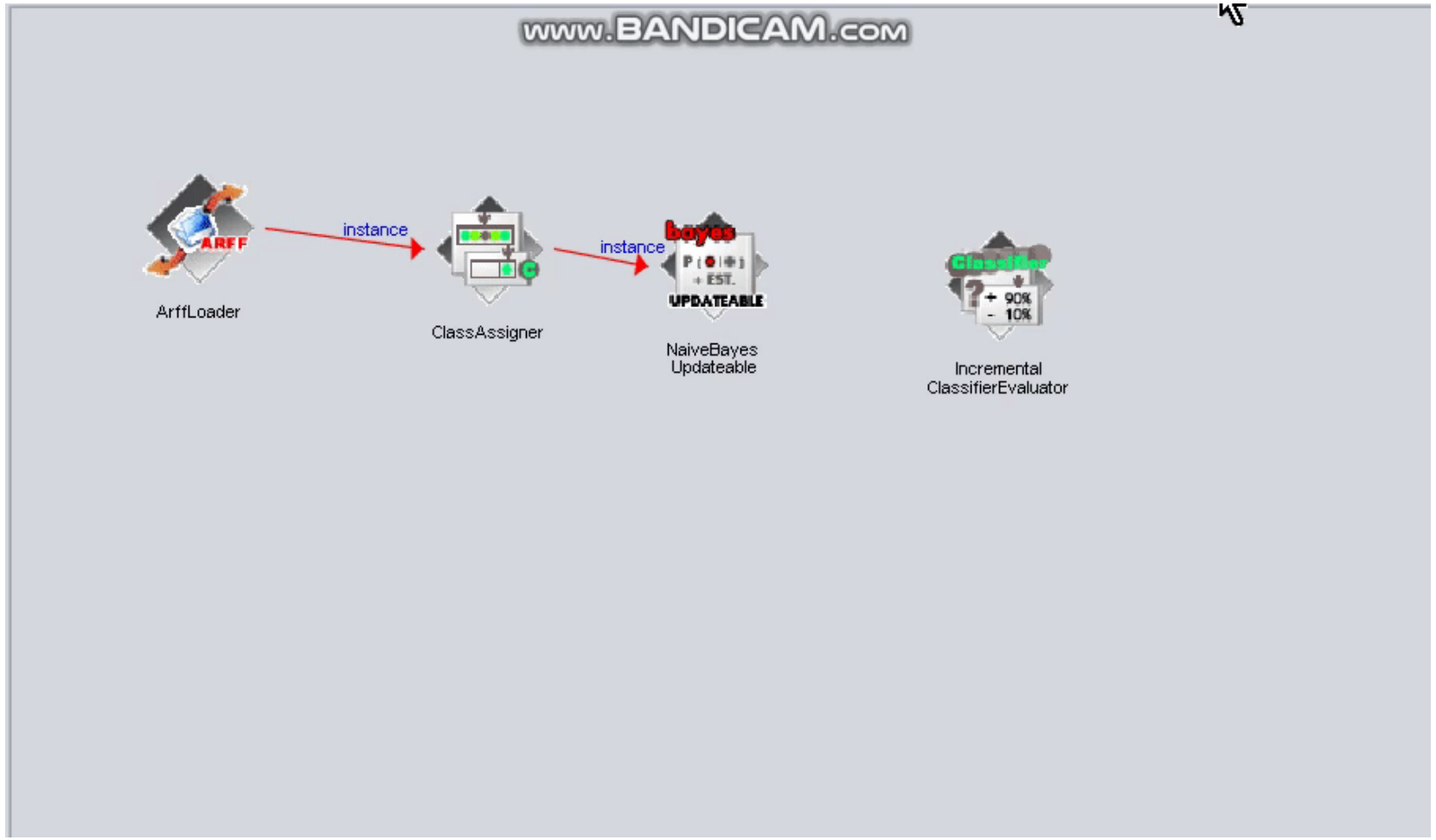
Lesson 1.4: 知識流介面

11.左鍵單擊Evaluation資料夾下的IncrementalClassifierEvalutor元件，並放入右側畫布。



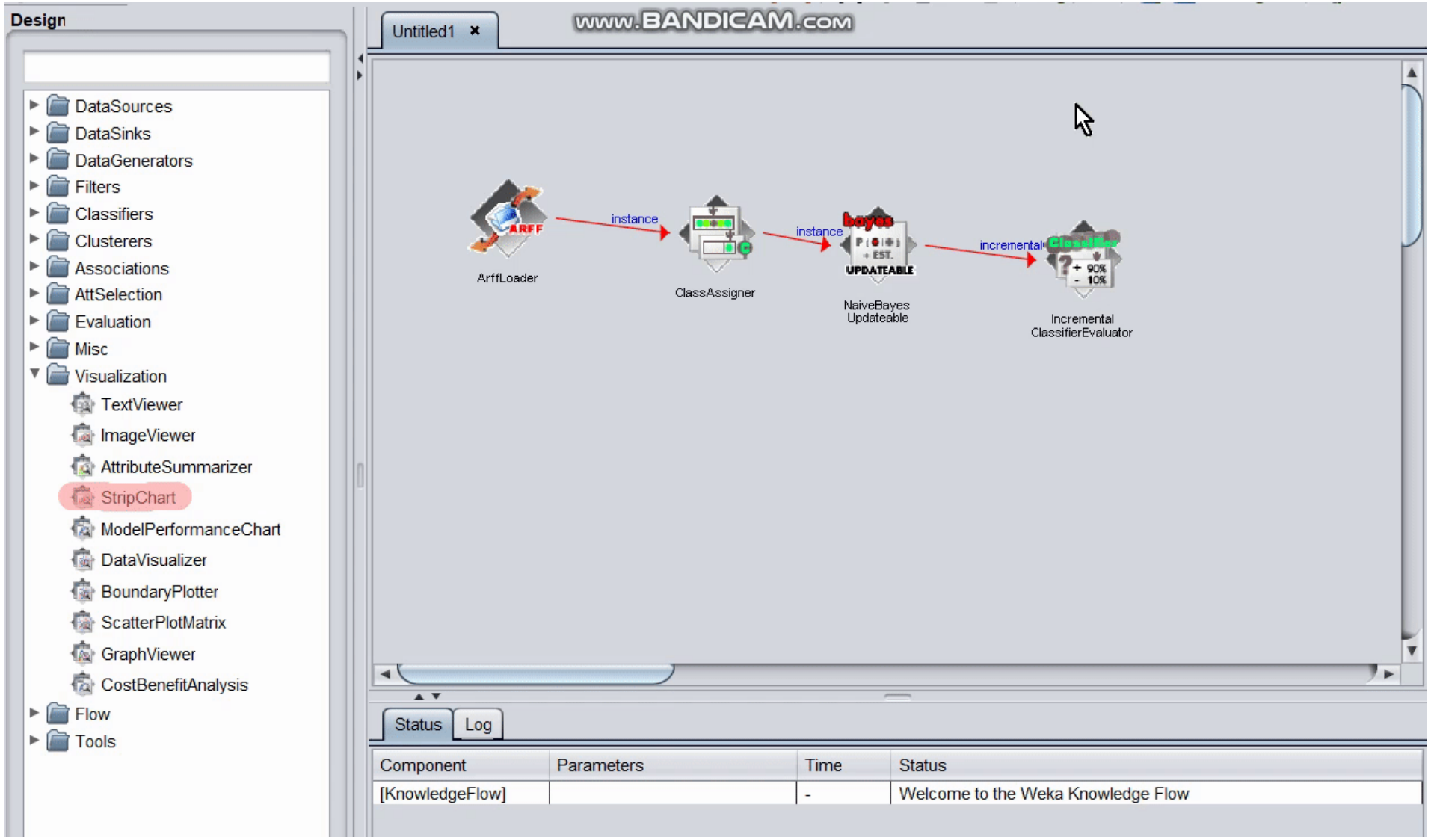
Lesson 1.4: 知識流介面

12. 對NaiveBayesUpdateable元件單擊右鍵 → 在出現的選單中點選incrementalClassifer → 左鍵單擊IncrementalClassifierEvaluator元件四周的圓點進行連接



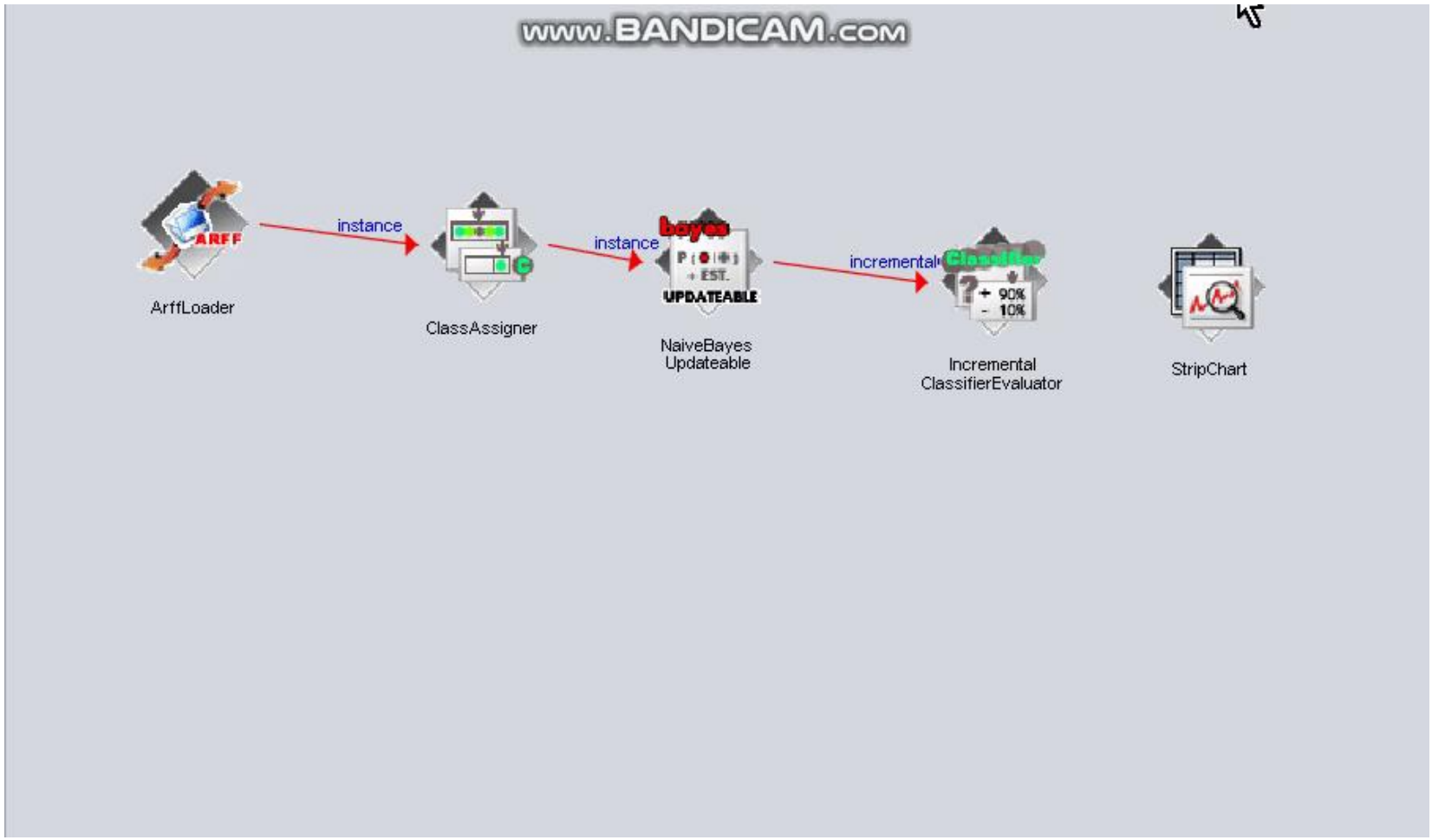
Lesson 1.4: 知識流介面

13.左鍵單擊Visualization資料夾下的StripChart元件，並放入右側畫布。



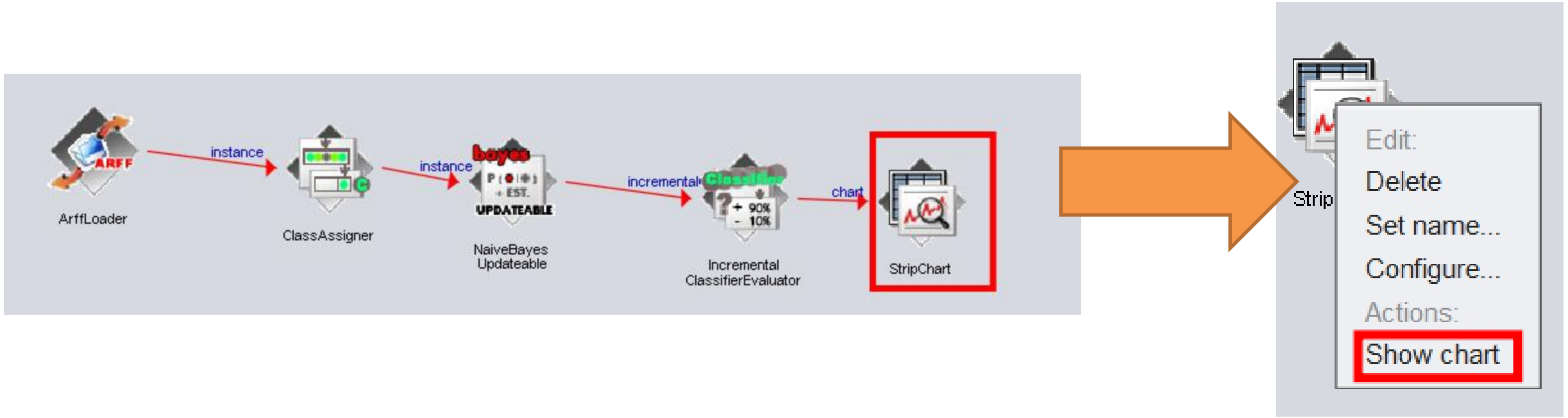
Lesson 1.4: 知識流介面

14. 對IncrementalClassifierEvaluator元件單擊右鍵 → 在出現的選單中點選chart → 左鍵單擊StripChart元件四周的圓點進行連接



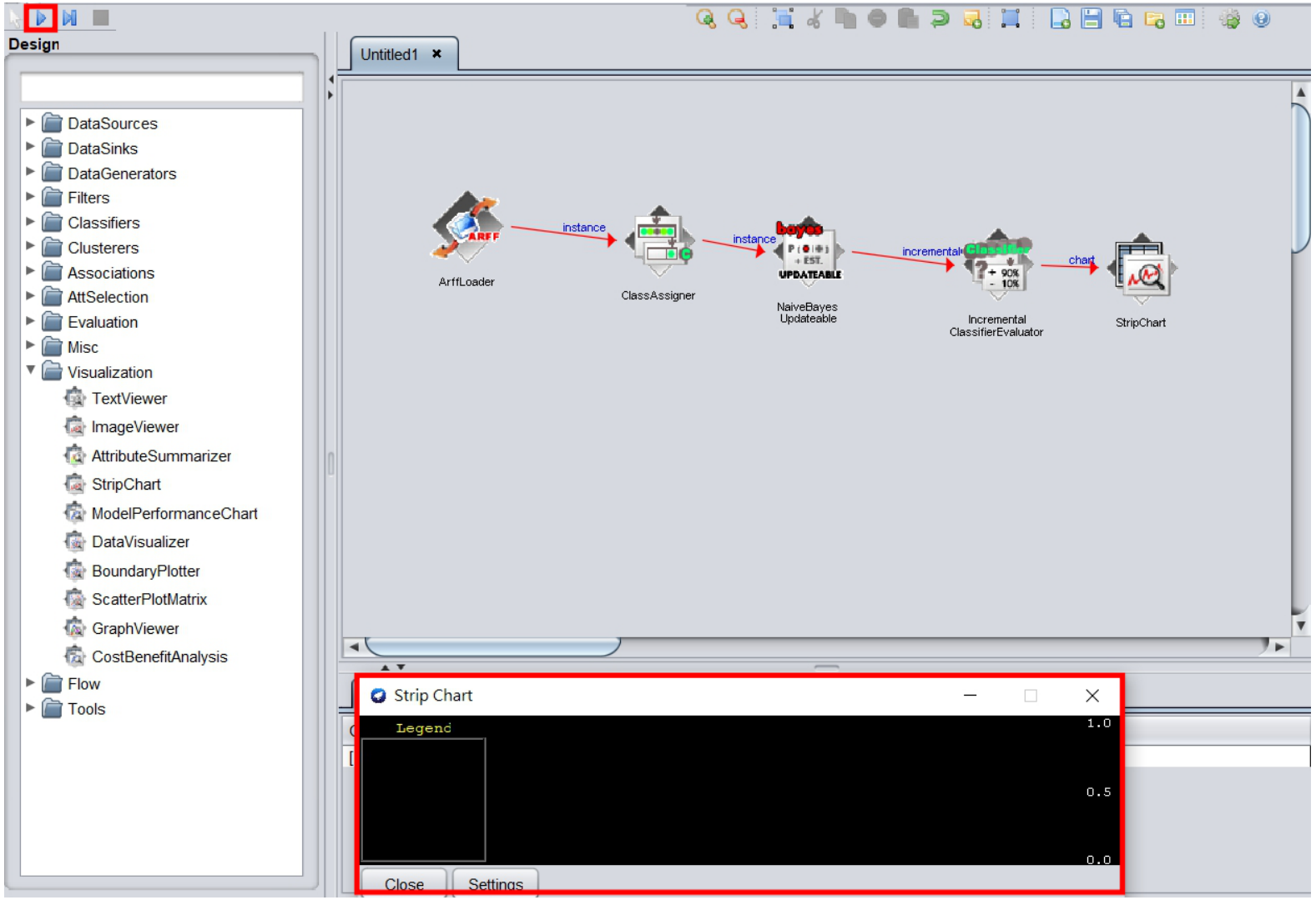
Lesson 1.4: 知識流介面

15.右鍵單擊StripChart元件，在出現的選單中左鍵單擊Show chart



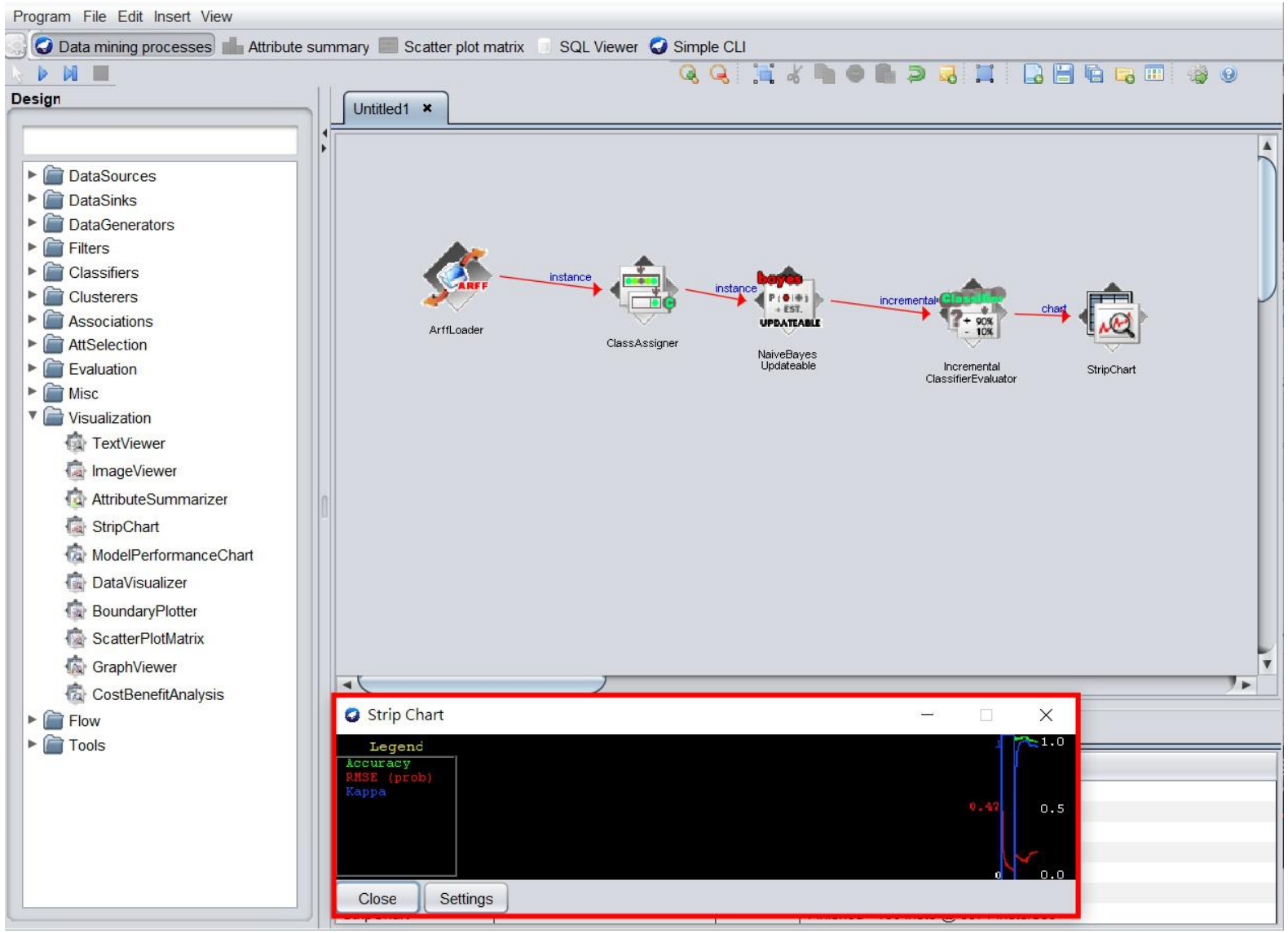
Lesson 1.4: 知識流介面

16.出現Strip Chart的視窗後，左鍵單擊視窗左上角的運行圖示



Lesson 1.4: 知識流介面

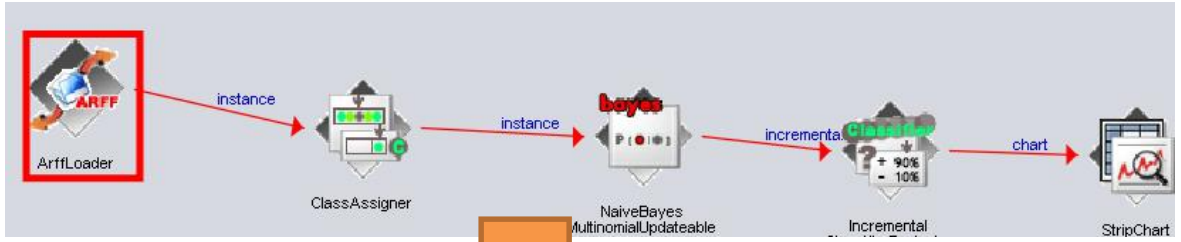
▼運行結果

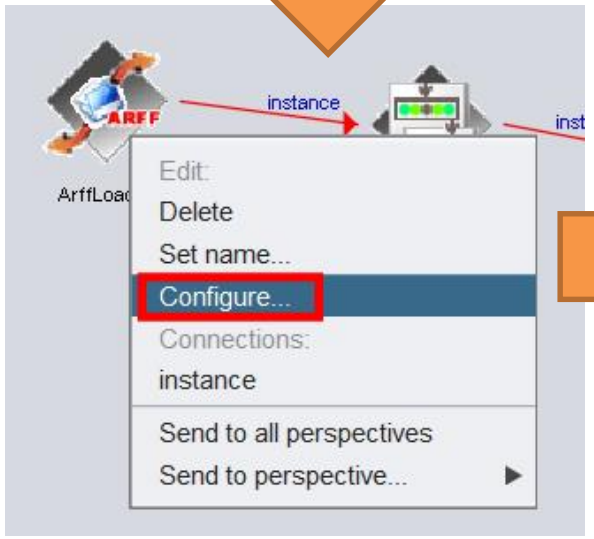


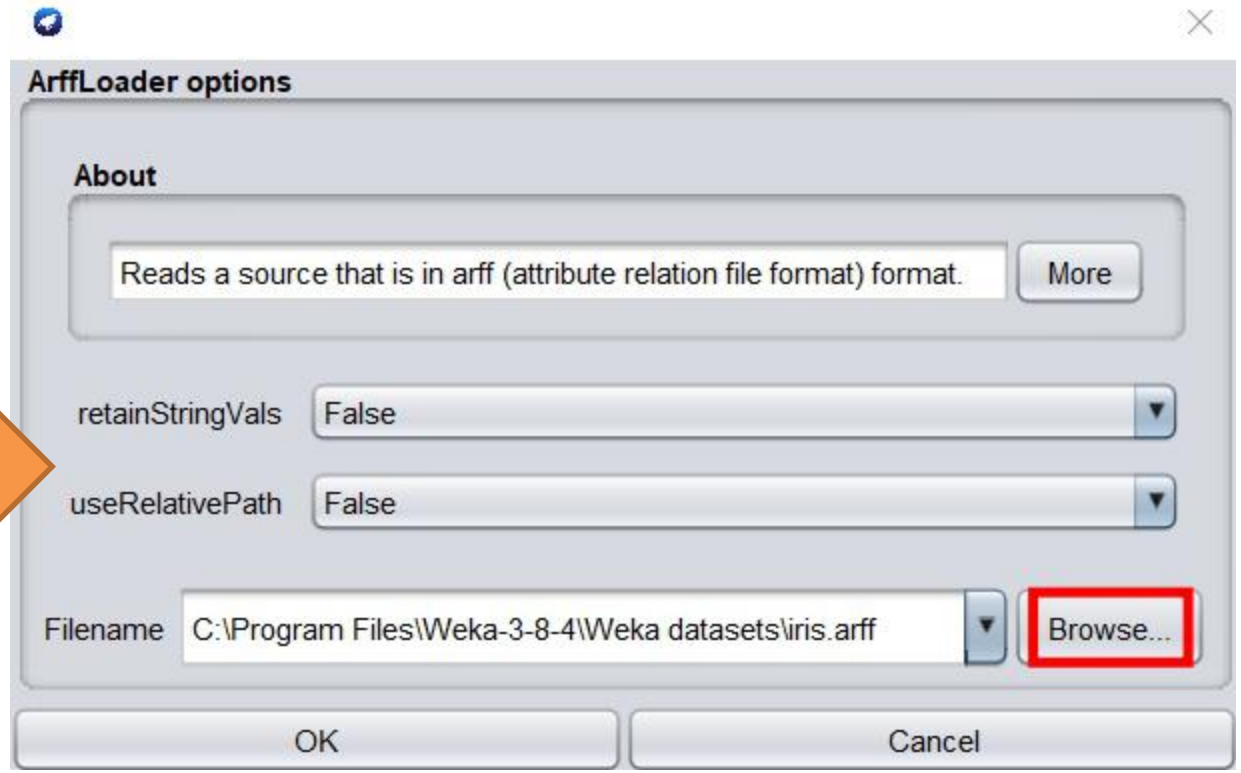
Lesson 1.4: 知識流介面

我們換個資料集運行看看

1.左鍵單擊ArffLoader元件，在出現的選單中選擇Configure，並在出現的視窗中左鍵單擊Browse...按鈕

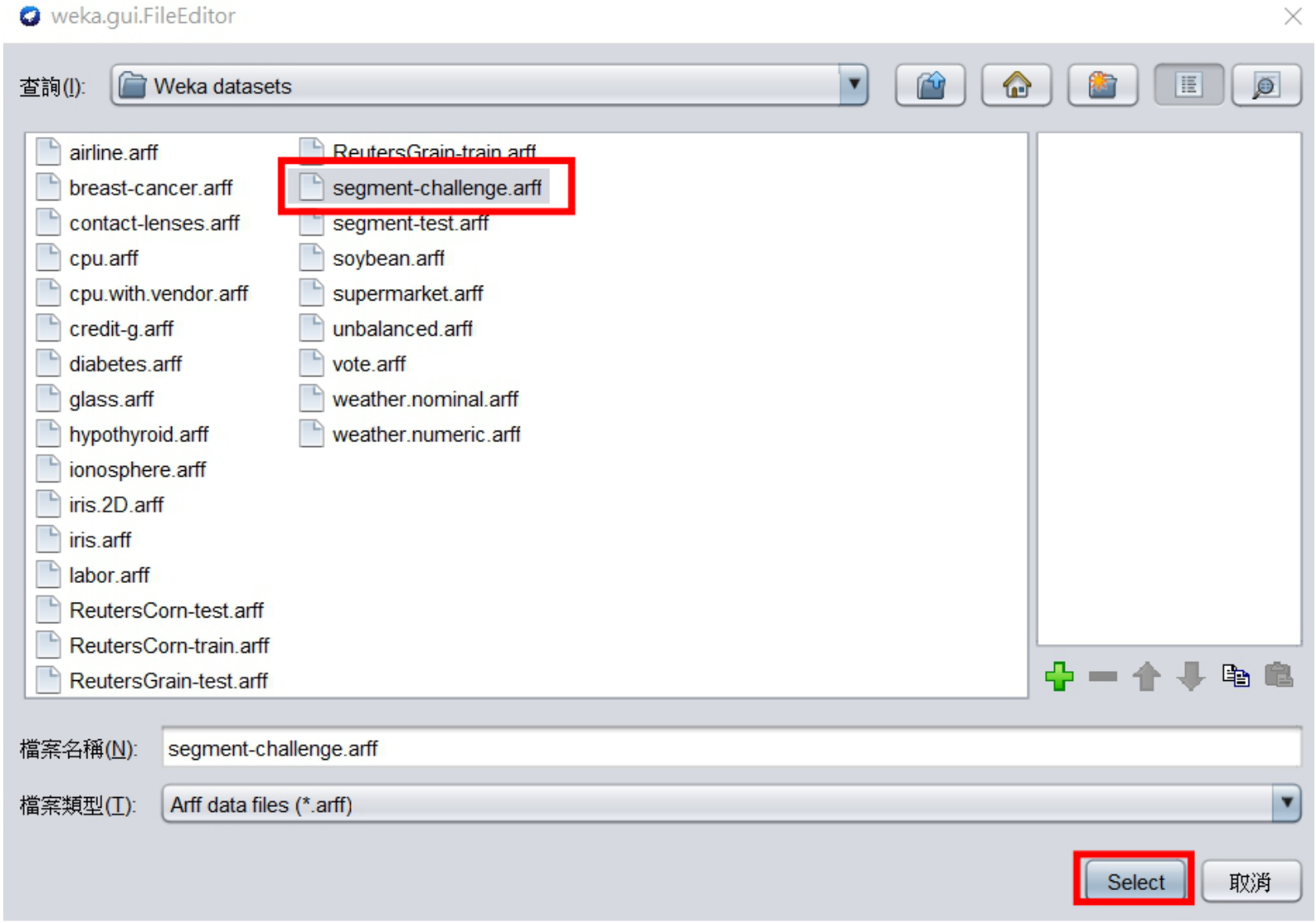






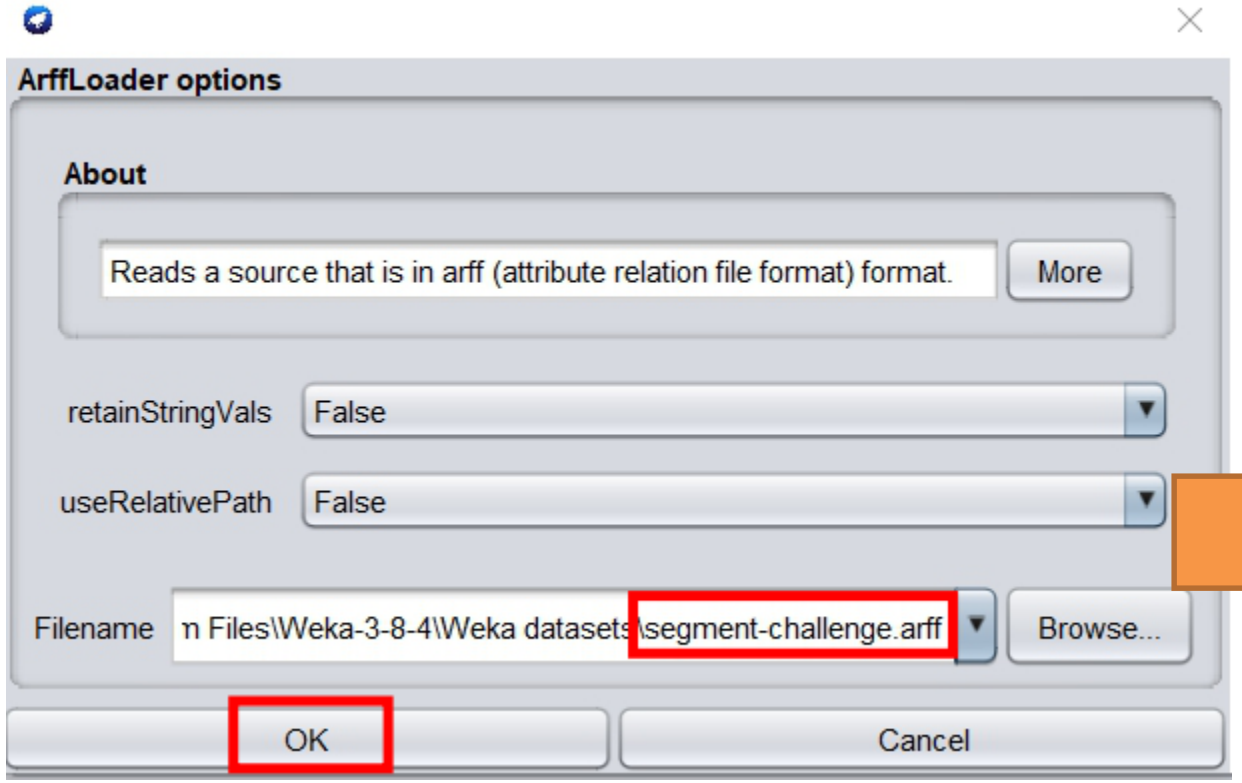
Lesson 1.4: 知識流介面

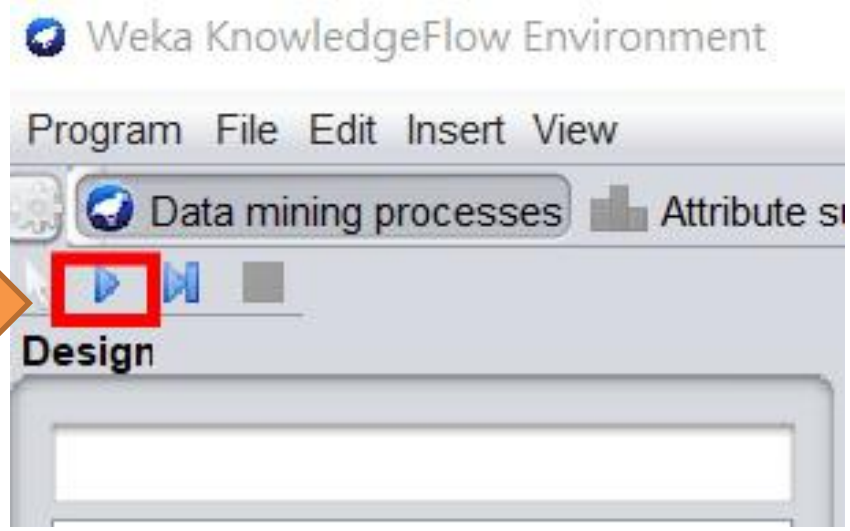
2.左鍵單擊segment-challenge.arff檔案，在以左鍵單擊右下方Select按鈕



Lesson 1.4: 知識流介面

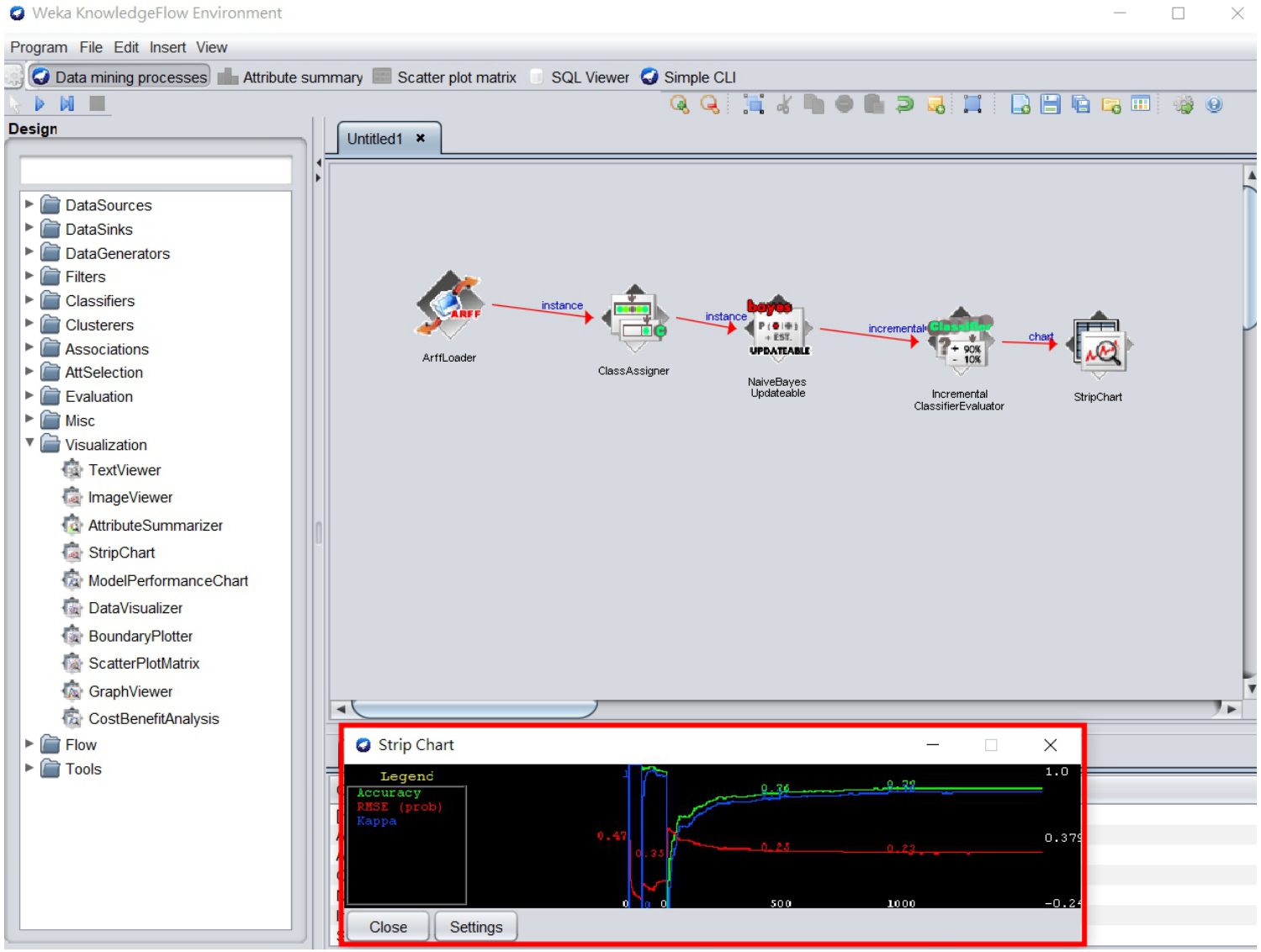
3.確認選擇好segment-challenge.arff檔案後，左鍵單擊OK按鈕回到KnowledgeFlow視窗，左鍵單擊視窗左上方的運行按鈕。





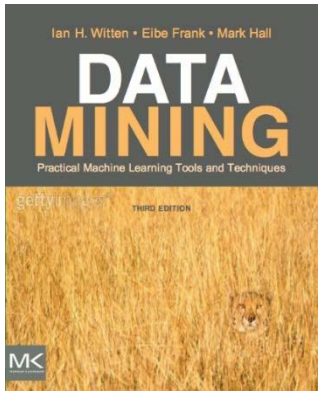
Lesson 1.4: 知識流介面

▼運行結果



Lesson 1.4: 知識流介面

- 知識流面板和Explorer的相似, 除了…
  - *DataSources 侍從Filters分離出來的*
  - 我們使用*DataSinks來寫入資料或模型到檔案中*
  - *Evaluation* 是獨立的面板
- 功能也差不多, 除了…
  - 我們在知識流面板可以遞增地處理可能的無限的資料集
  - 我們在知識流面板可以看到交叉驗證每個層的結果
- 有些人喜歡圖形介面



課程文本

Chapter 12 *The Knowledge Flow Interface*

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Class 1 – Lesson 5

命令行介面

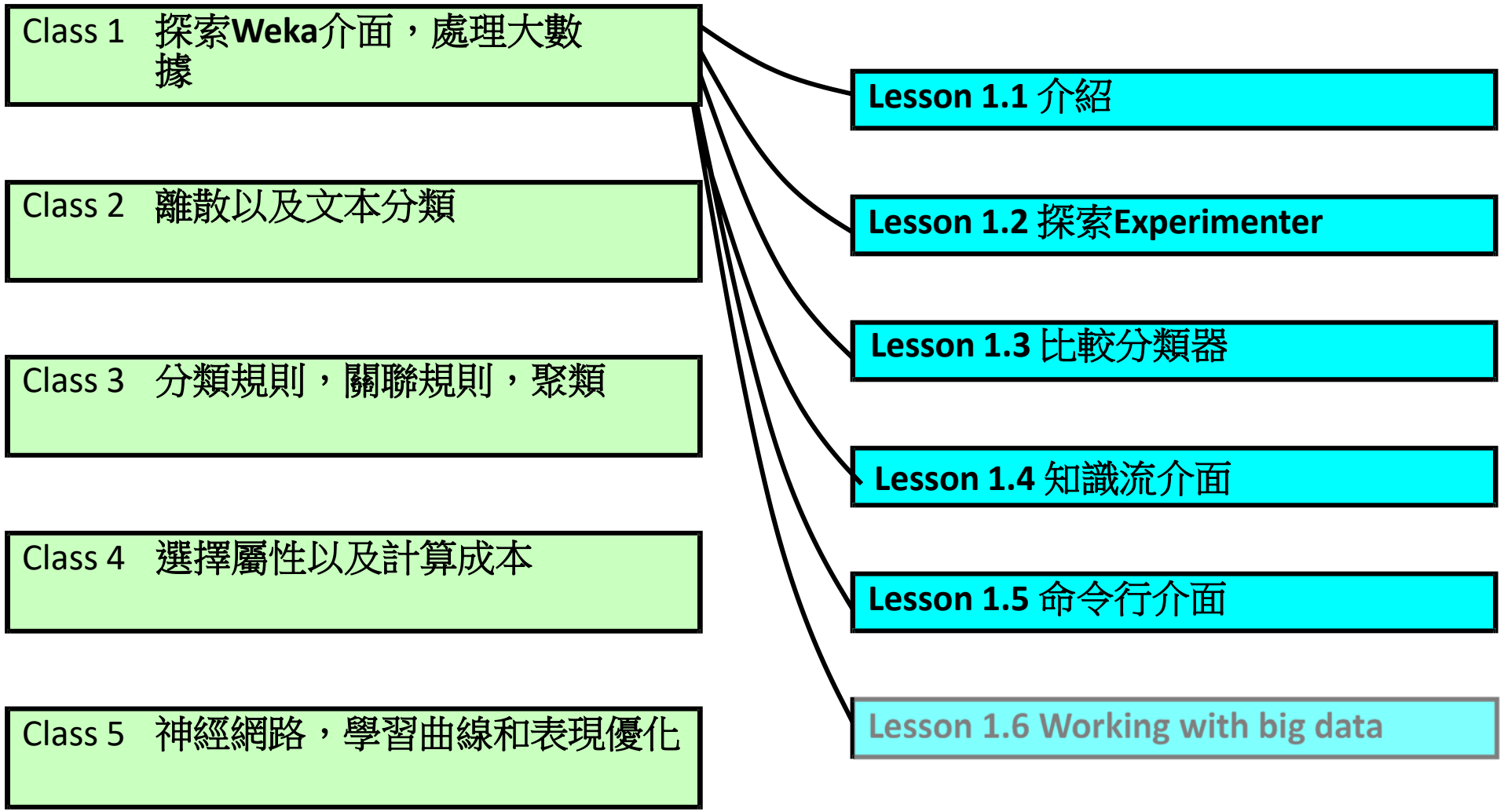
(The Command Line interface)

Ian H. Witten

Department of Computer Science
University of Waikato
New Zealand

weka.waikato.ac.nz

Lesson 1.5: 命令行介面



Lesson 1.5: 命令行介面

在CLI中運行分類器

- 將J48的選項印出:

  java weka.classifiers.trees.J48

- 一般通用的選項

  –h print help info
  –t <name of training file>　　[絕對路徑名稱…]
  –T <name of test file>

- J48特定的選項 (來自Explorer 組態面板)
- 運行J48:

  java weka.classifiers.trees.J48 –C 0.25 –M 2　　← 從Explorer複製

  –t "C:\Users\ihw\My Documents\Weka datasets\iris.arff"　　← 訓練集

Lesson 1.5: 命令行介面

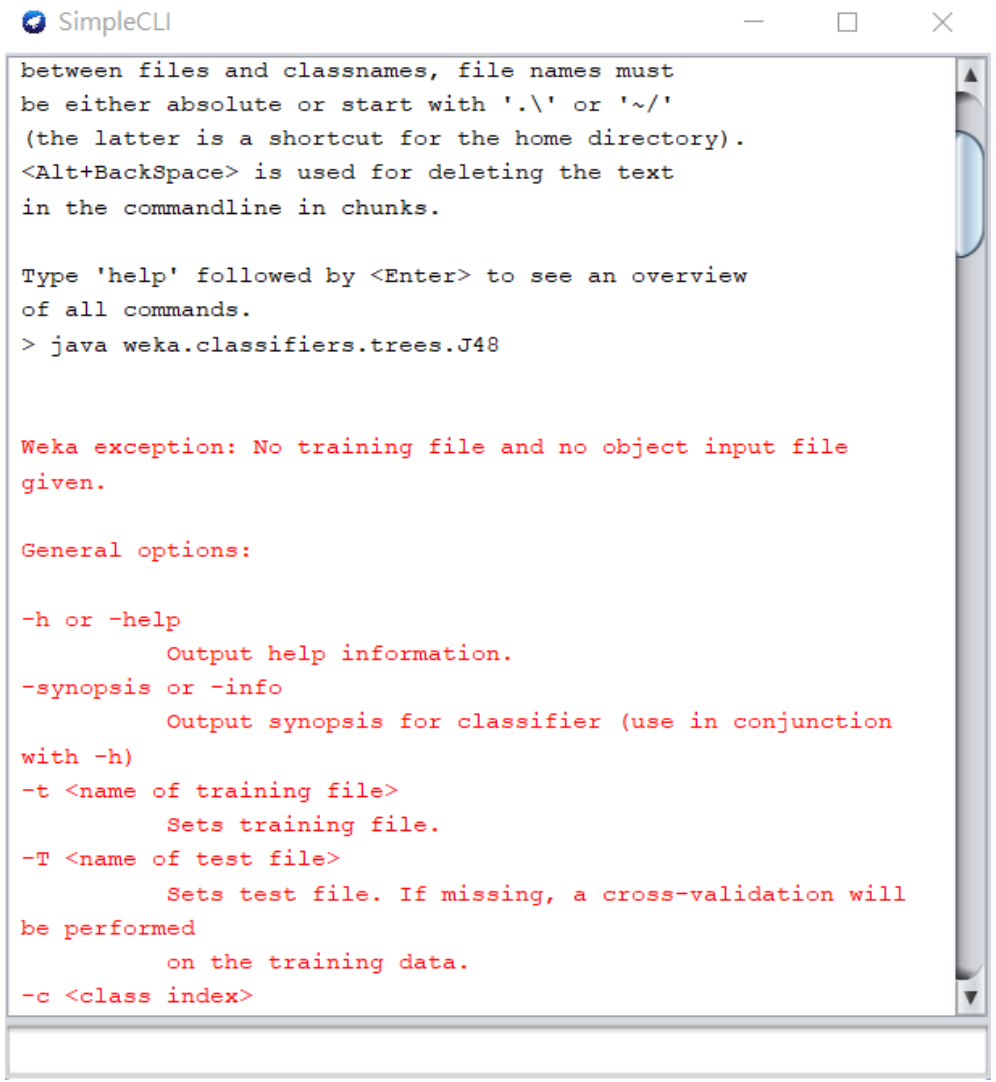
1. 開啟Weka程式，於Weka GUI Chooser介面左鍵單擊Simple CLI按鈕

Lesson 1.5: 命令行介面

2.在輸入框中輸入 java weka.classifiers.trees.J48並按下enter鍵

Lesson 1.5: 命令行介面

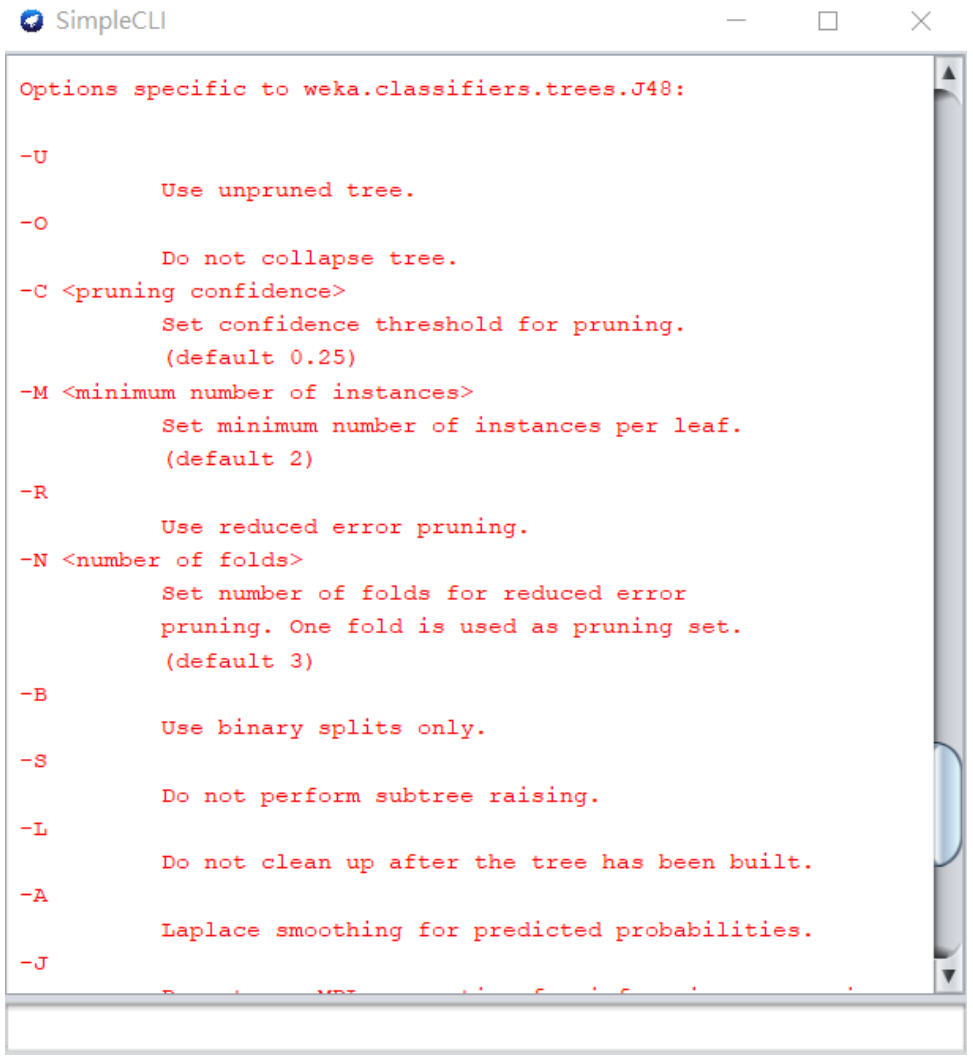
▼執行結果事實上為錯誤信息「WEKA 錯誤：沒有訓練文件並且輸入文件為空。」
因為沒有指定訓練文件，所以Weka不能理解這行語句，並且顯示了J48的參數選項。



Lesson 1.5: 命令行介面

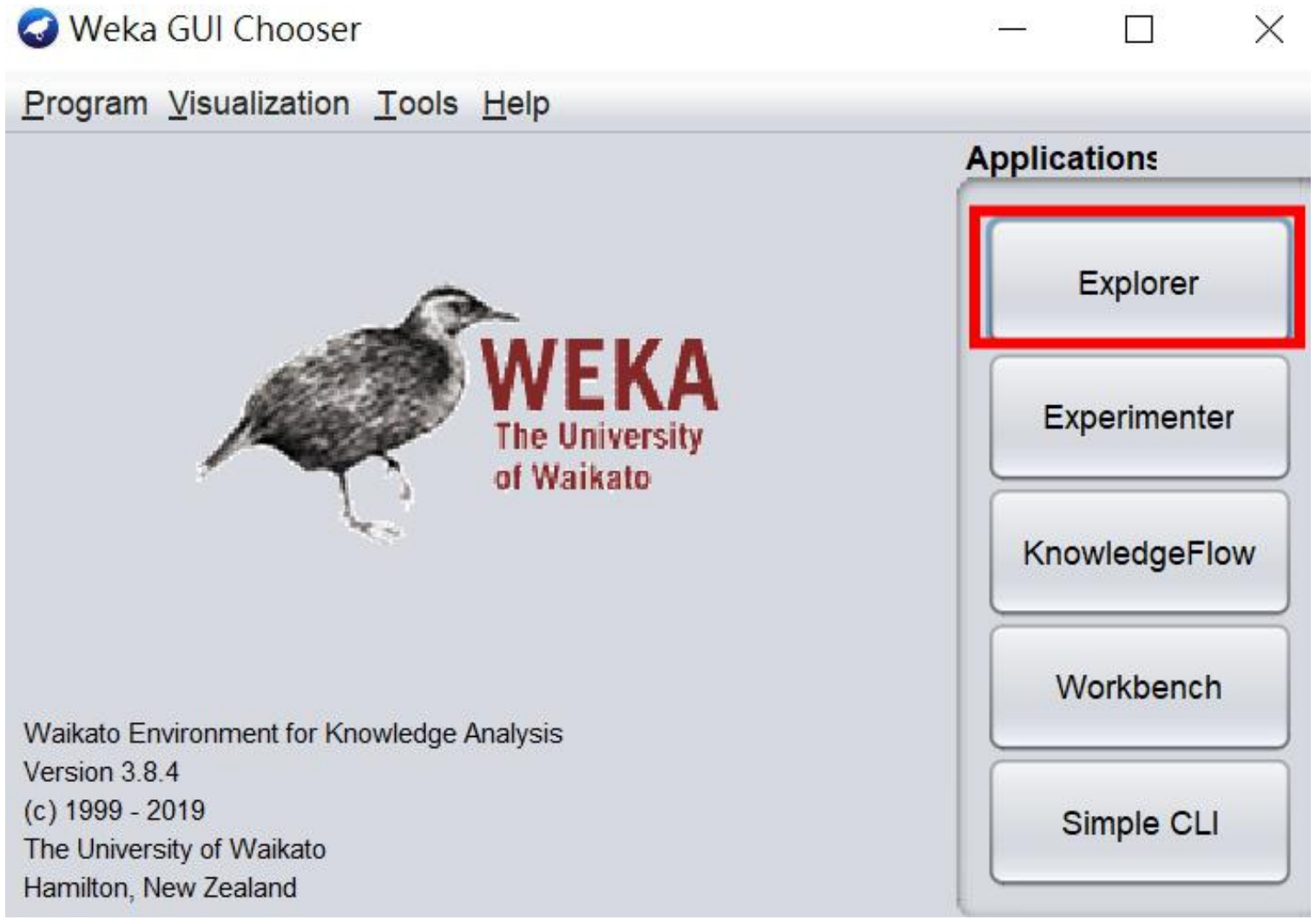
通用選項：「-h」代表幫助;「-t」代表指定訓練文件;「-T」代表指定測試文件。

J48特有選項：「-C」選項和「-M」選項。



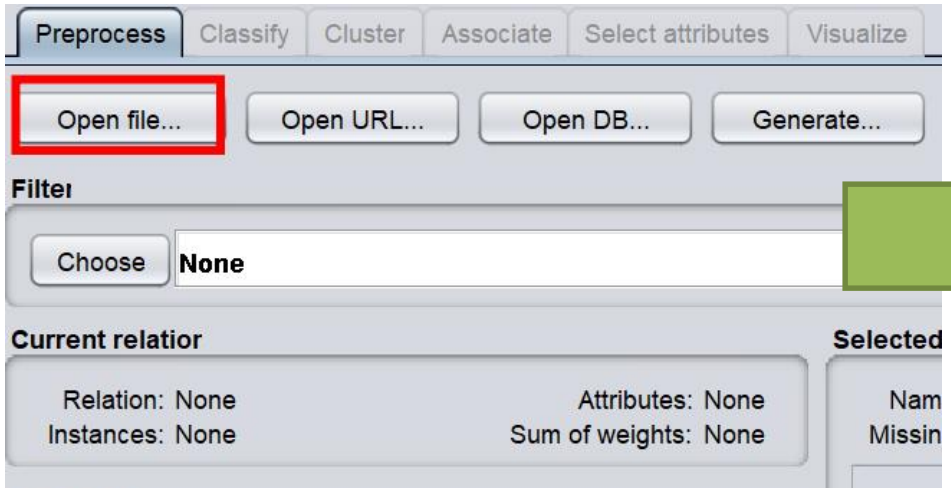
Lesson 1.5: 命令行介面

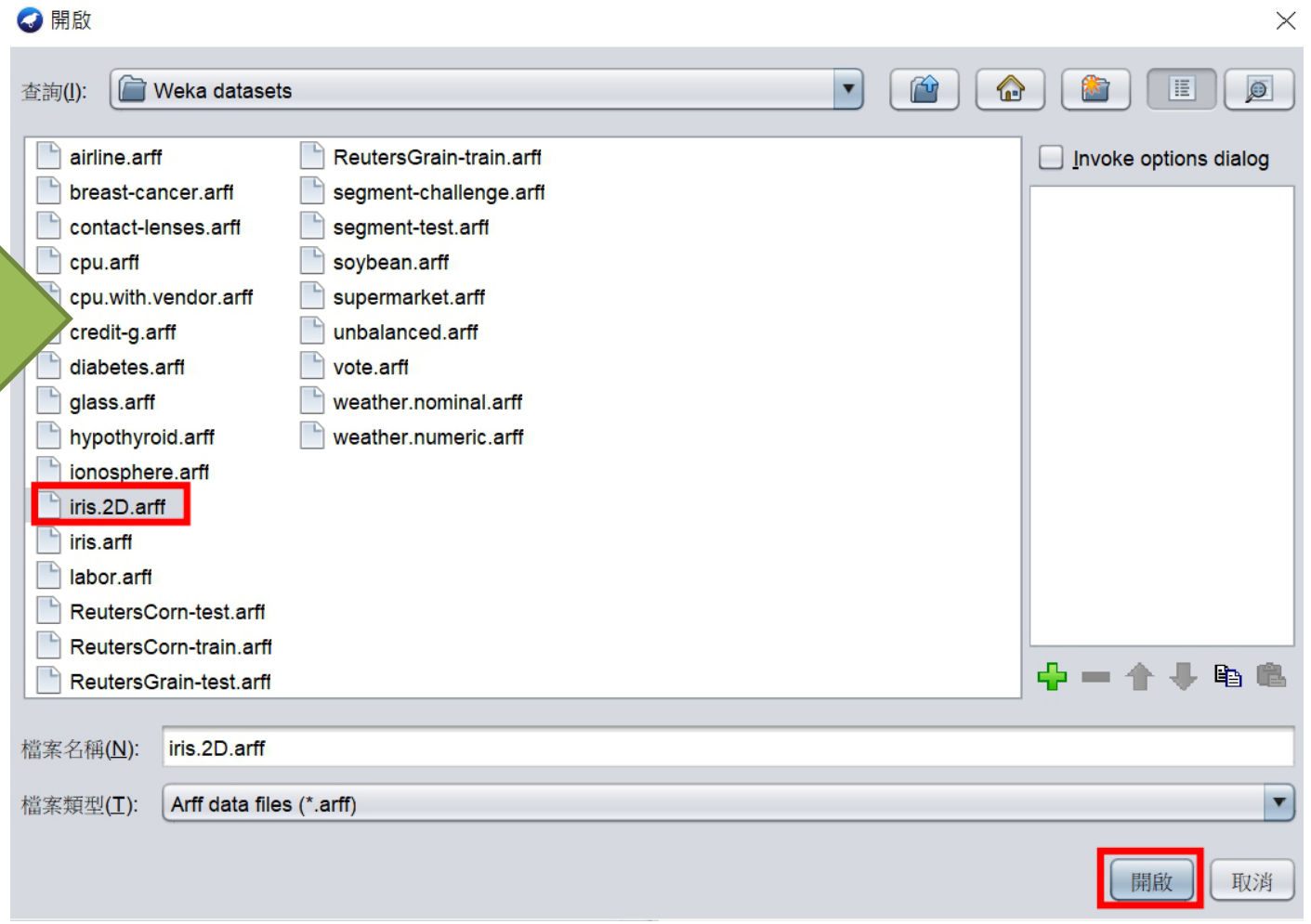
3. 回到Weka GUI Chooser介面開啟Weka的Explorer



Lesson 1.5: 命令行介面

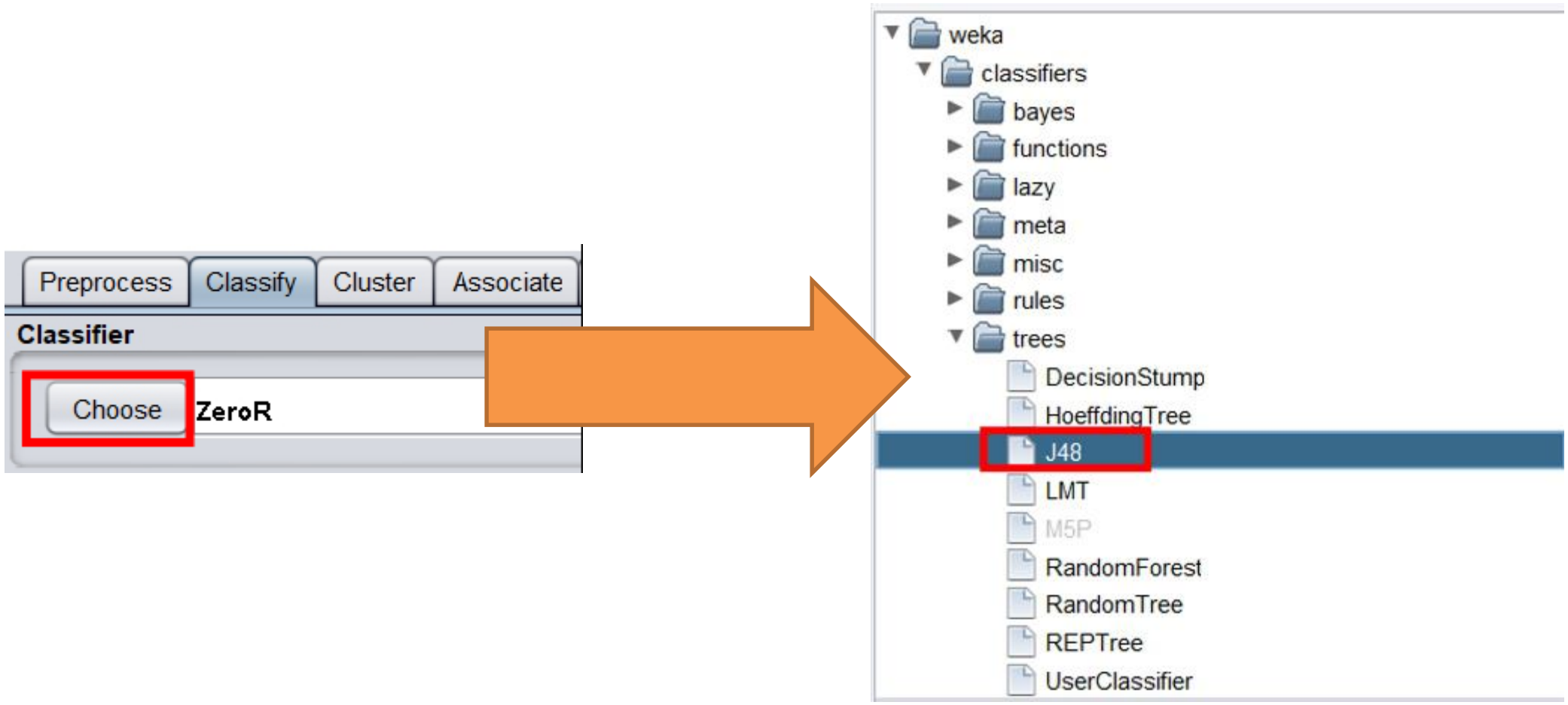
4.左鍵點擊Open file...開啟右圖視窗，進入自行複製的Weka datasets，左鍵單擊iris.2D.arff的檔案後，再以左鍵單擊下方”開啟”以載入此檔案





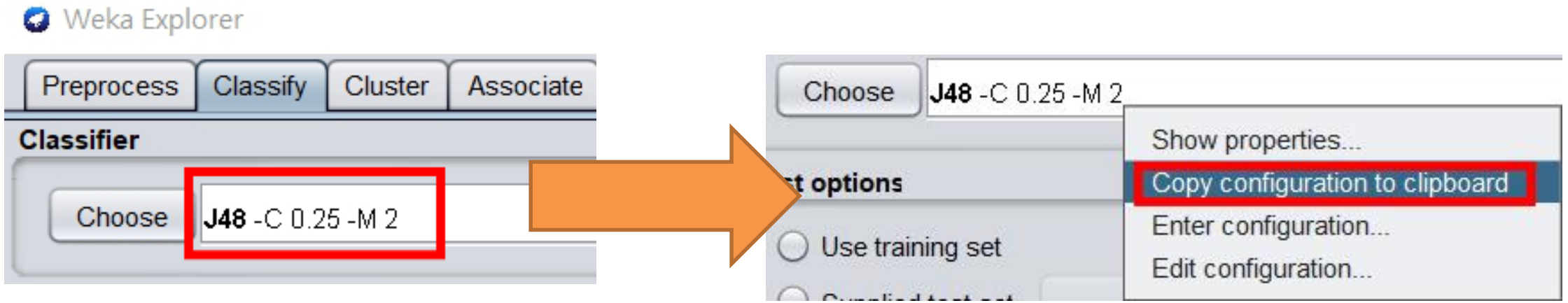
Lesson 1.5: 命令行介面

5.切換到Classify介面，左鍵單擊Choose鈕，並在出現的選單中左鍵單擊trees資料夾下的J48分類器。



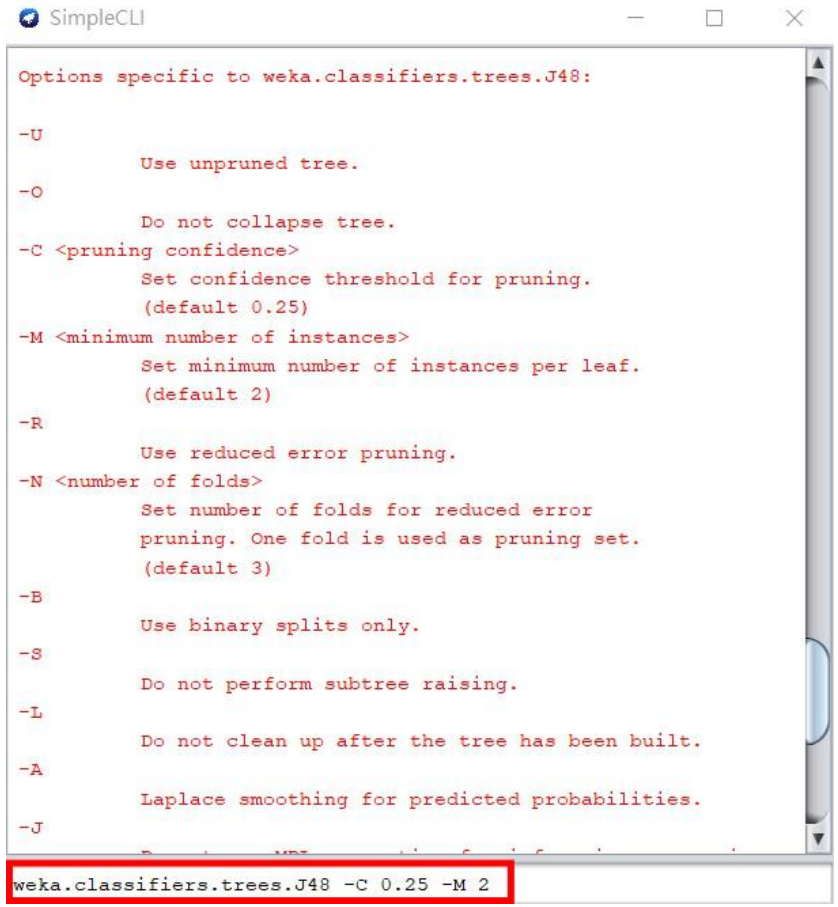
Lesson 1.5: 命令行介面

6.對紅色方框中的J48敘述單擊右鍵，並在出現的選單中單擊左鍵選擇Copy configuration to clipboard選項



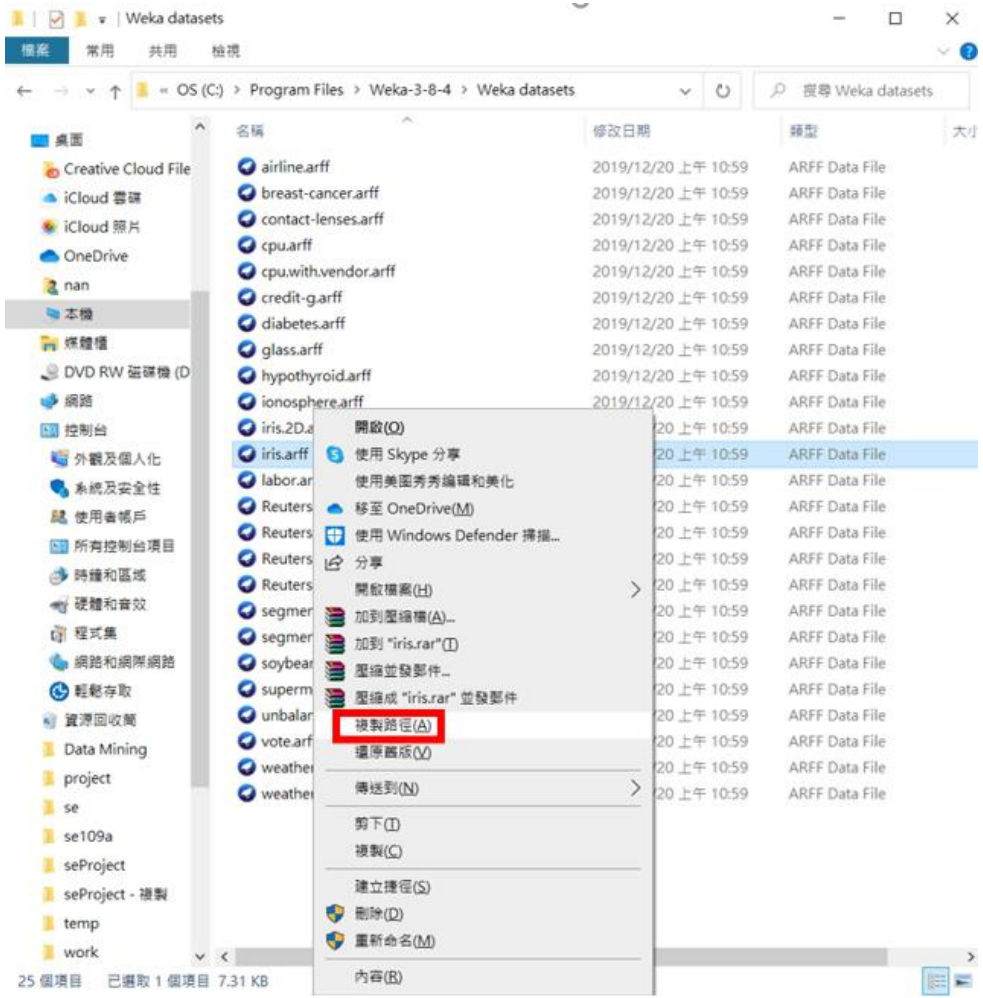
Lesson 1.5: 命令行介面

7.回到SimpleCLI視窗，在輸入框中使用Ctrl+V的方式將剛才複製的J48敘述貼上



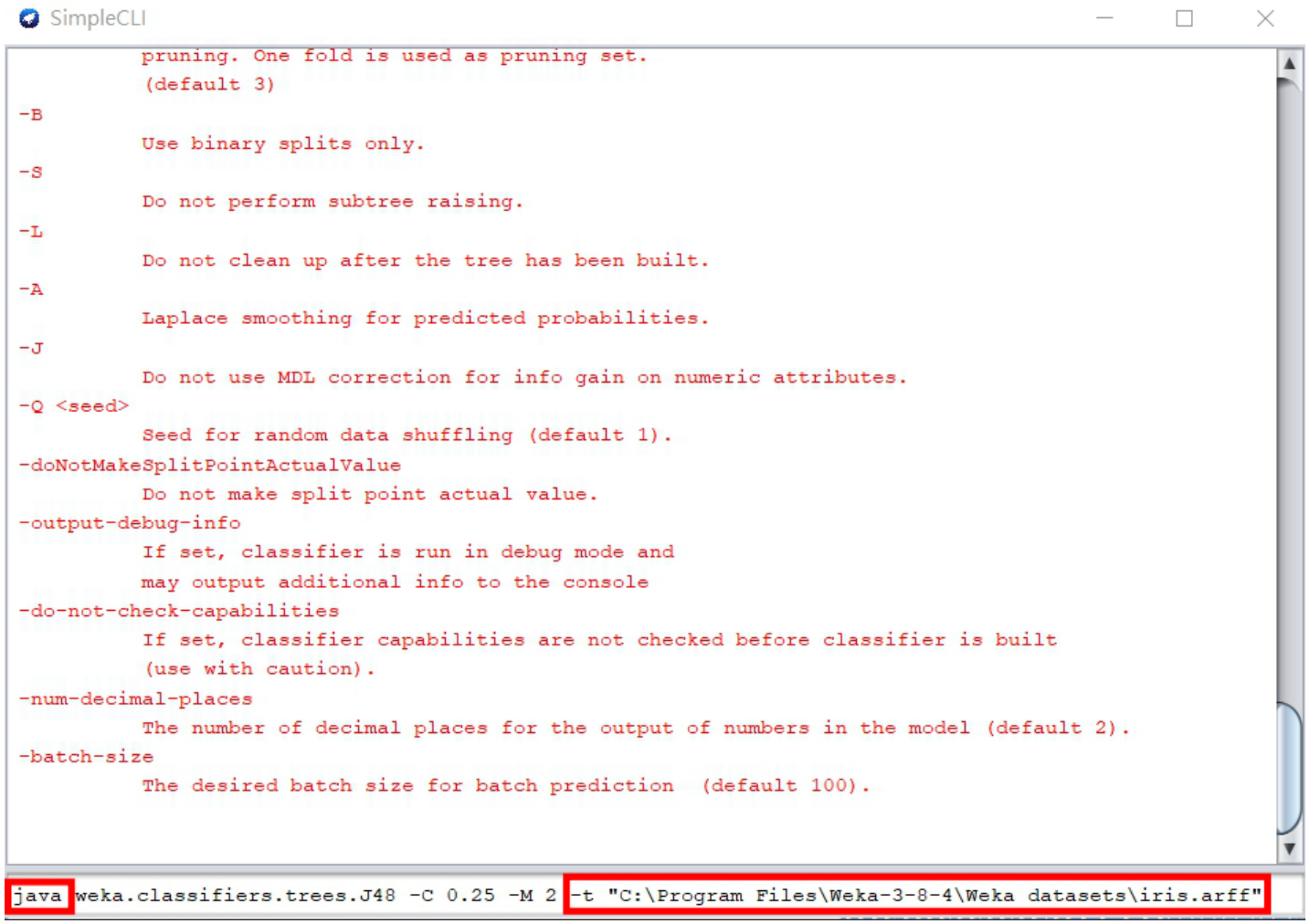
Lesson 1.5: 命令行介面

8. 找到本地的iris.arff檔案。接著於鍵盤按住shift鍵，同時以滑鼠右鍵單擊iris.arff檔，在出現的選單中左鍵單擊複製路徑(A)的選項



Lesson 1.5: 命令行介面

9.回到SimpleCLI視窗，在剛剛貼上的J48敘述前方加上「java」、後方加上「 -t 」並空一格以Ctrl+V的方式貼上剛才複製的路徑，接著按下enter鍵運行



注意：文件名要用引號括起來

Lesson 1.5: 命令行介面

▼運行結果，我們已經多次見過這種的輸出結果。

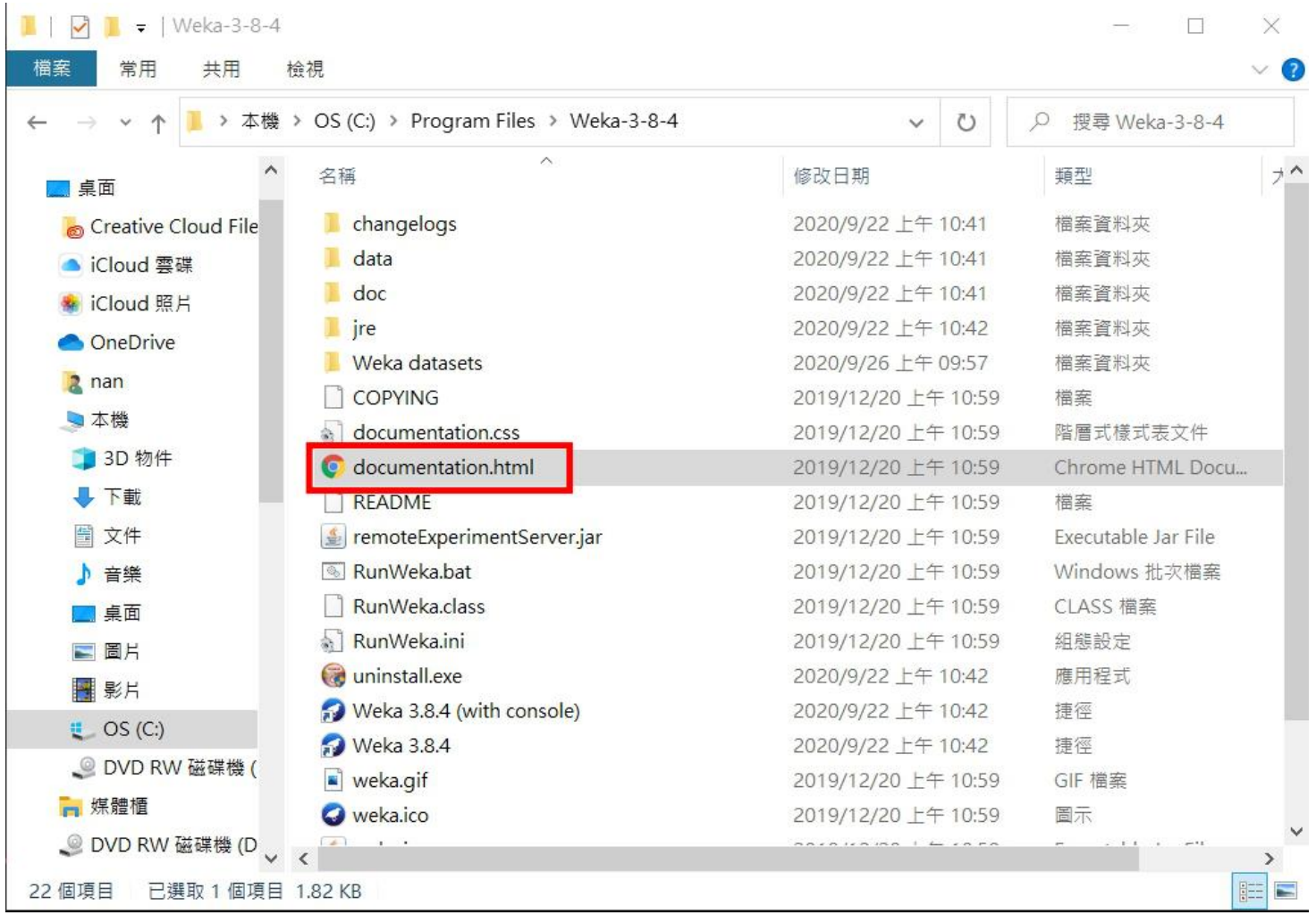
SimpleCLI

```
=== Stratified cross-validation ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
Total Number of Instances              150


=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0.980     0.000     1.000      0.980    0.990      0.985    0.990     0.987
Iris-setosa
                 0.940     0.030     0.940      0.940    0.940      0.910    0.952     0.880
Iris-versicolor
                 0.960     0.030     0.941      0.960    0.950      0.925    0.961     0.905
Iris-virginica
Weighted Avg.    0.960     0.020     0.960      0.960    0.960      0.940    0.968     0.924


=== Confusion Matrix ===

  a  b  c   <-- classified as
 49  1  0 |  a = Iris-setosa
  0 47  3 |  b = Iris-versicolor
  0  2 48 |  c = Iris-virginica
```

Lesson 1.5: 命令行介面

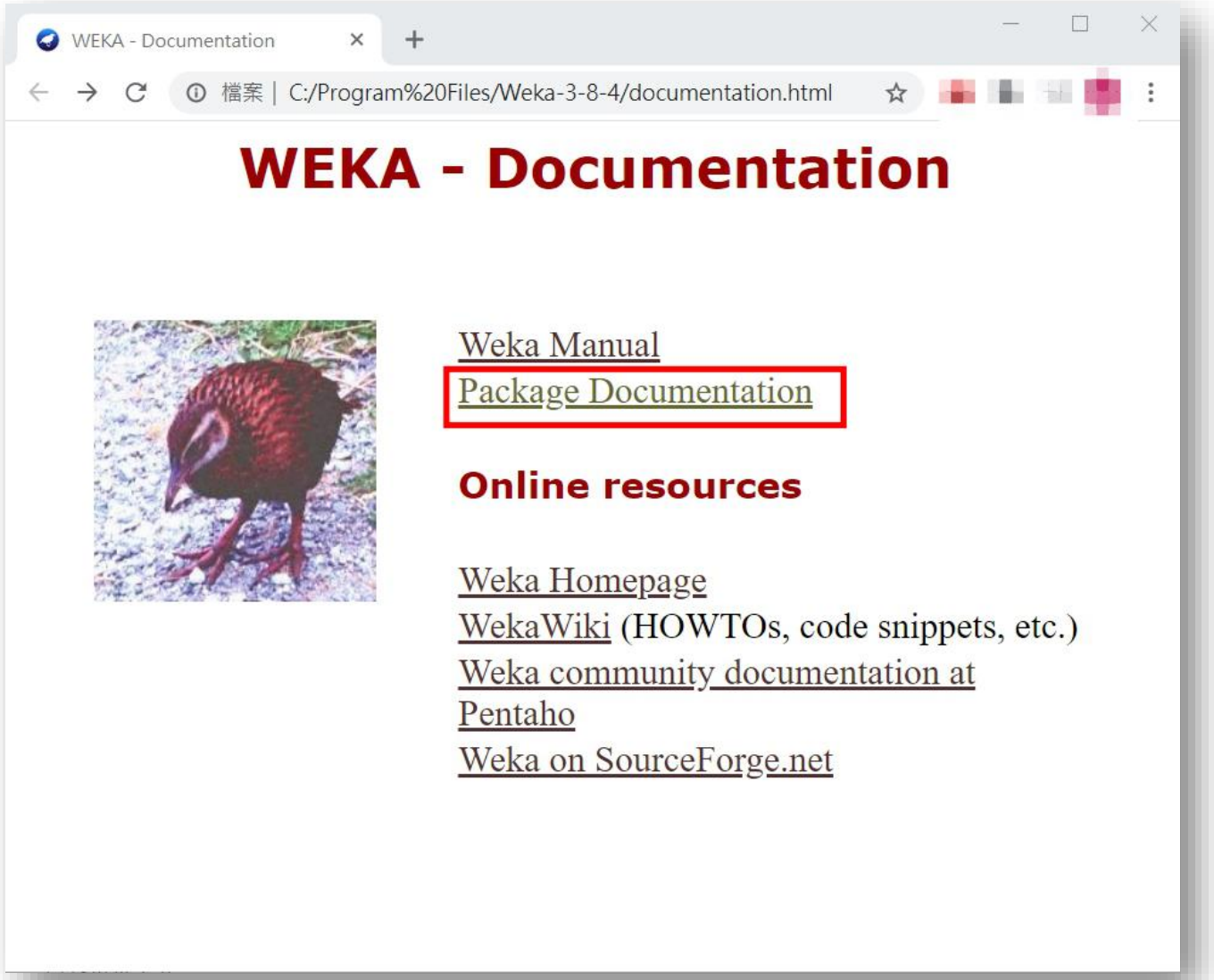
我們可在Javadoc中查看更詳細的說明文檔。

1.進入本地的Weka-3-8-4資料夾，左鍵雙擊documentation.html檔案。



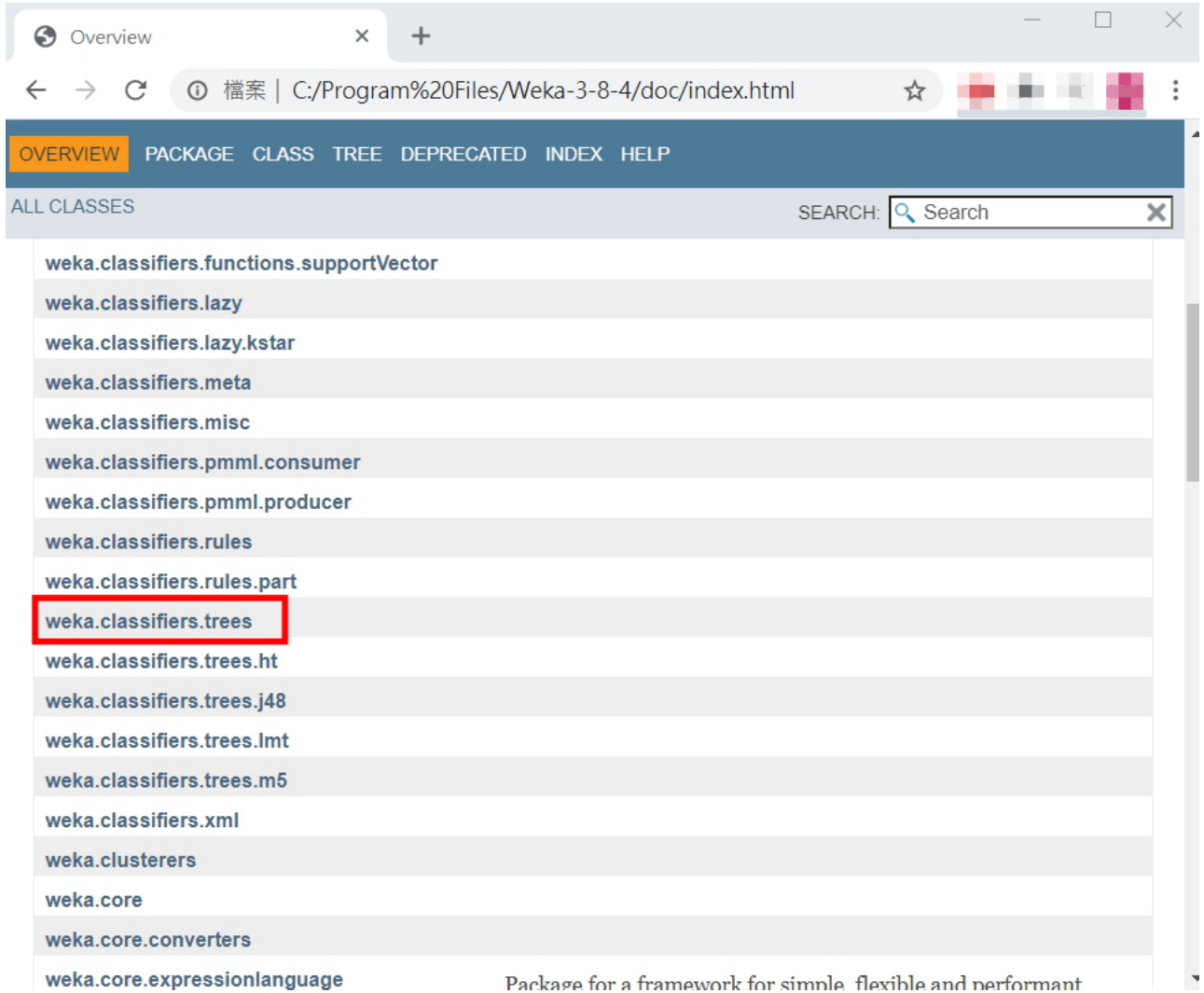
Lesson 1.5: 命令行介面

2.開啟Weka程式，於Weka GUI Chooser介面左鍵單擊Experimenter按鈕



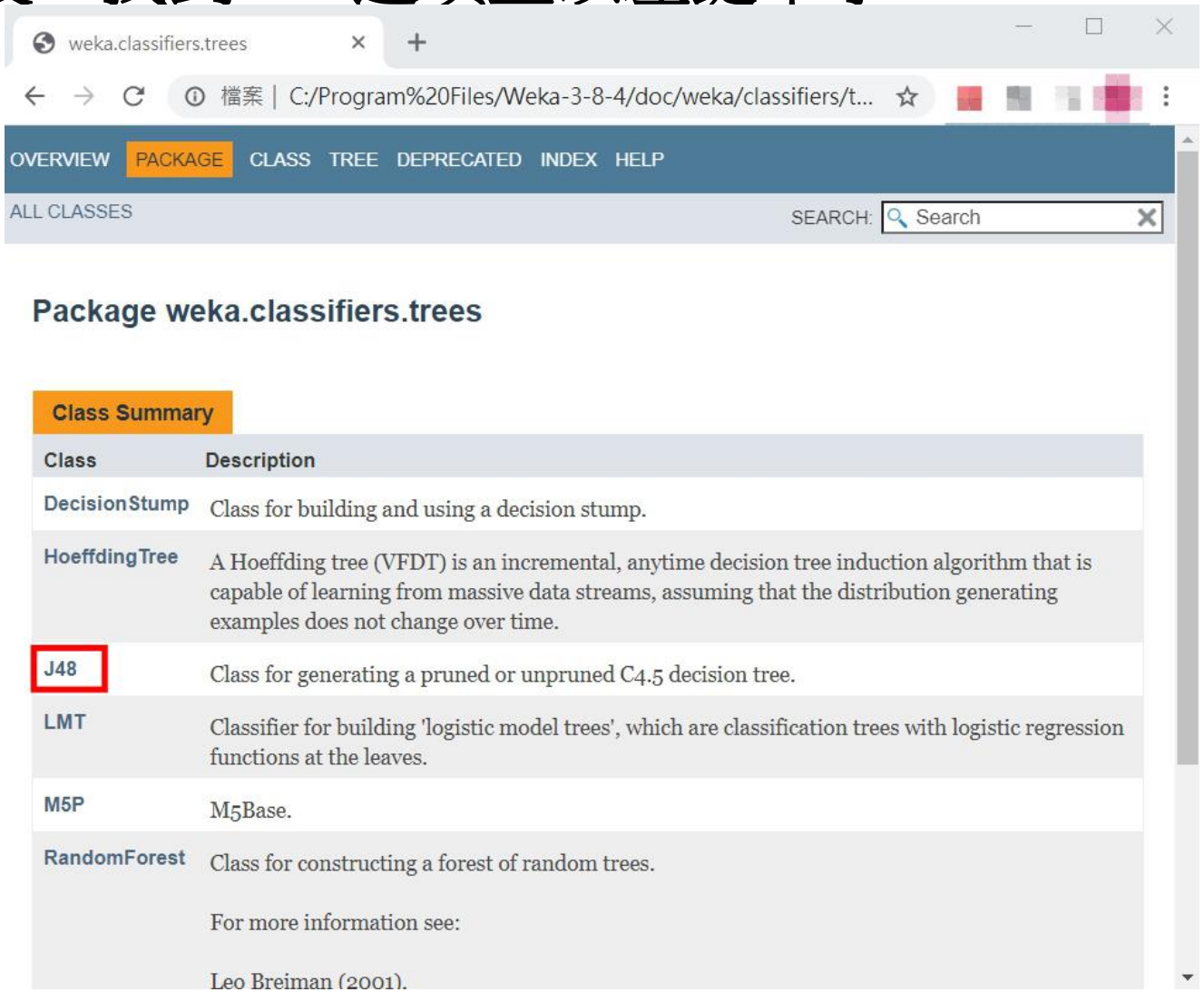
Lesson 1.5: 命令行介面

3.找到weka.classigiers.trees文件，並以左鍵單擊



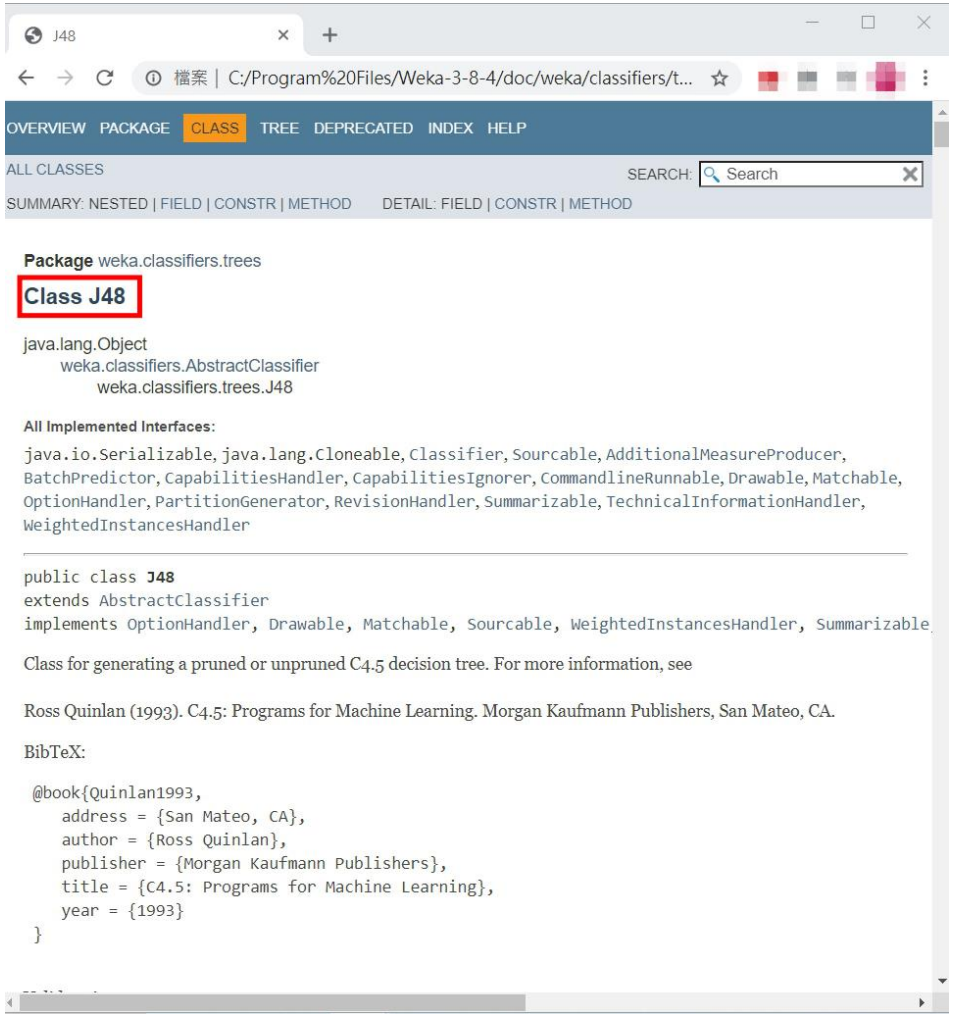
Lesson 1.5: 命令行介面

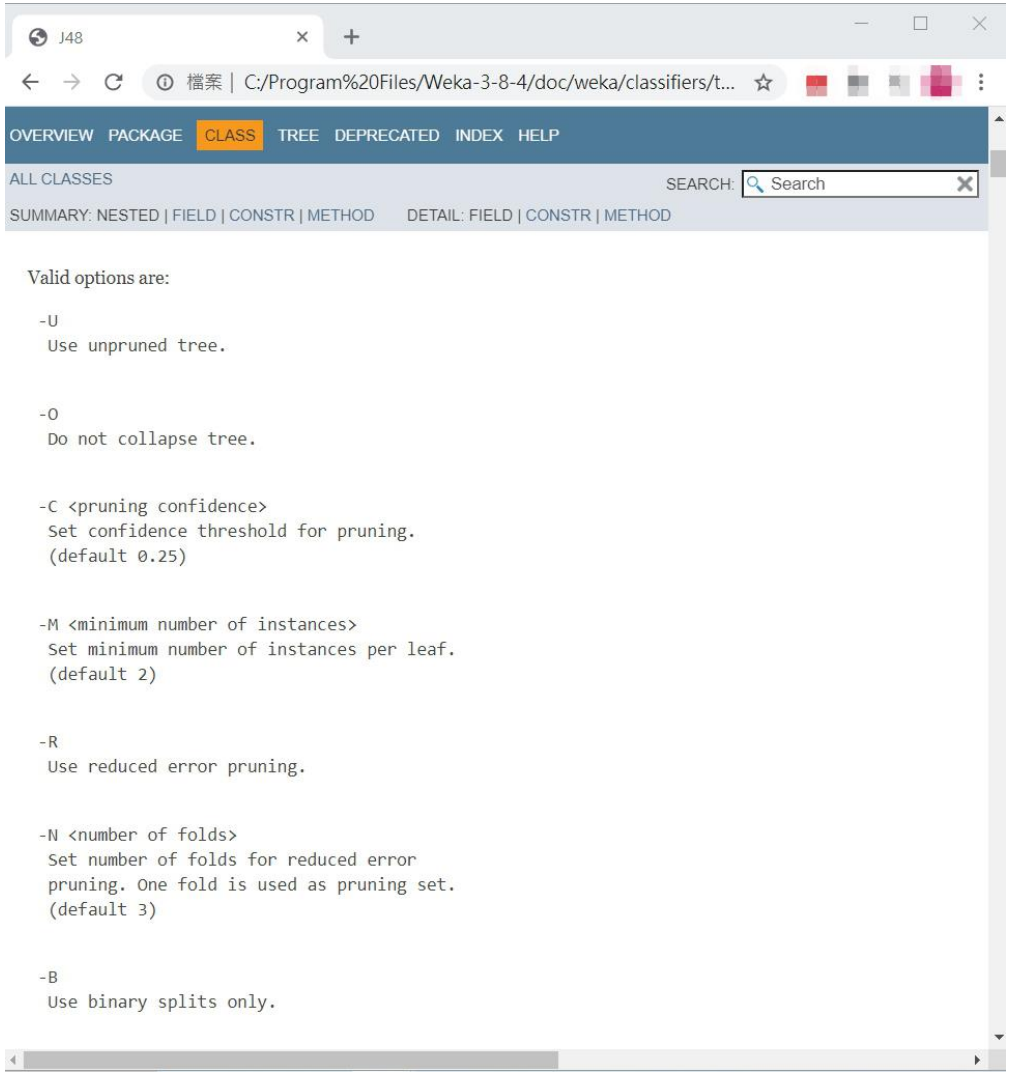
4.進入頁面後，找到J48選項並以左鍵單擊



Lesson 1.5: 命令行介面

▼在這份說明文檔中，我們可以看到J48的詳細說明





Lesson 1.5: 命令行介面

類別以及文件包

- J48 是一個「類別」
  - 它是變數的集合，伴隨著一些可以在其上操作的方法
- 「Package」是一個包含相關類別的目錄

weka.classifiers.trees.J48

packages　類別

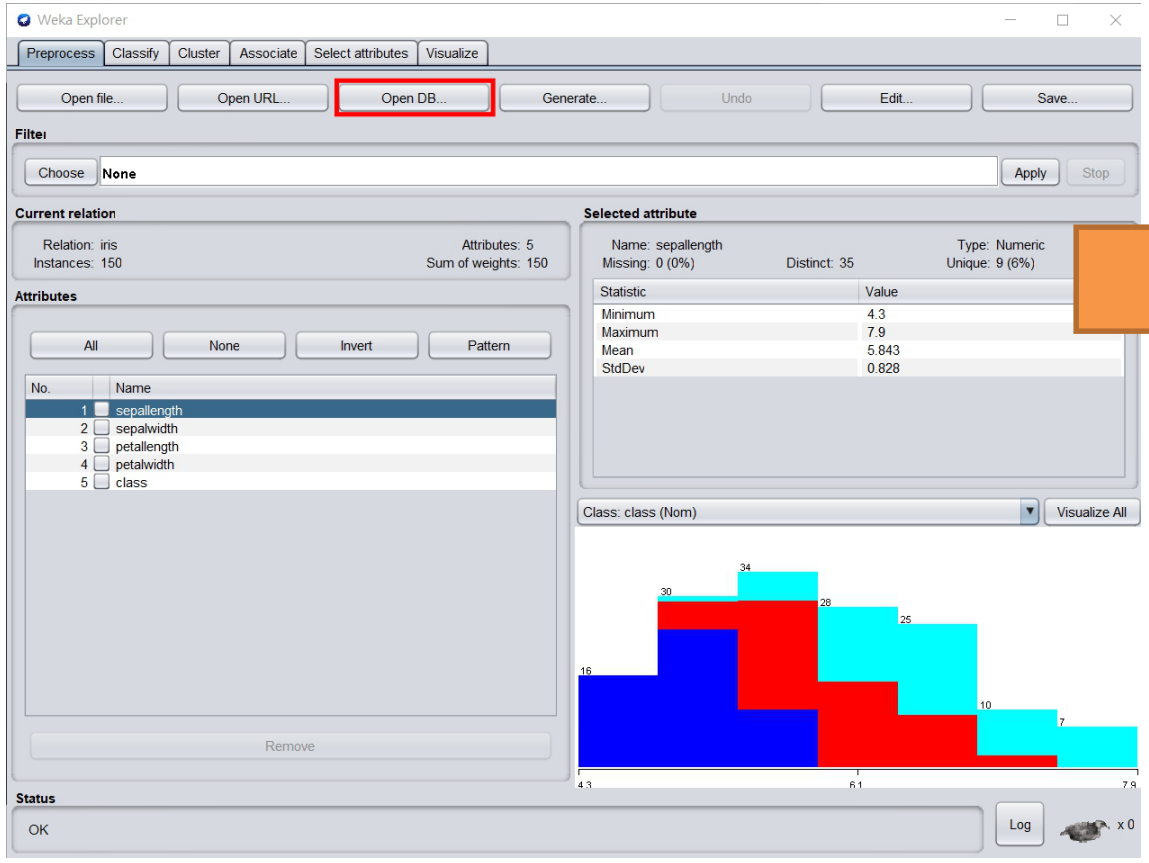
- Javadoc: Weka官方的說明文檔

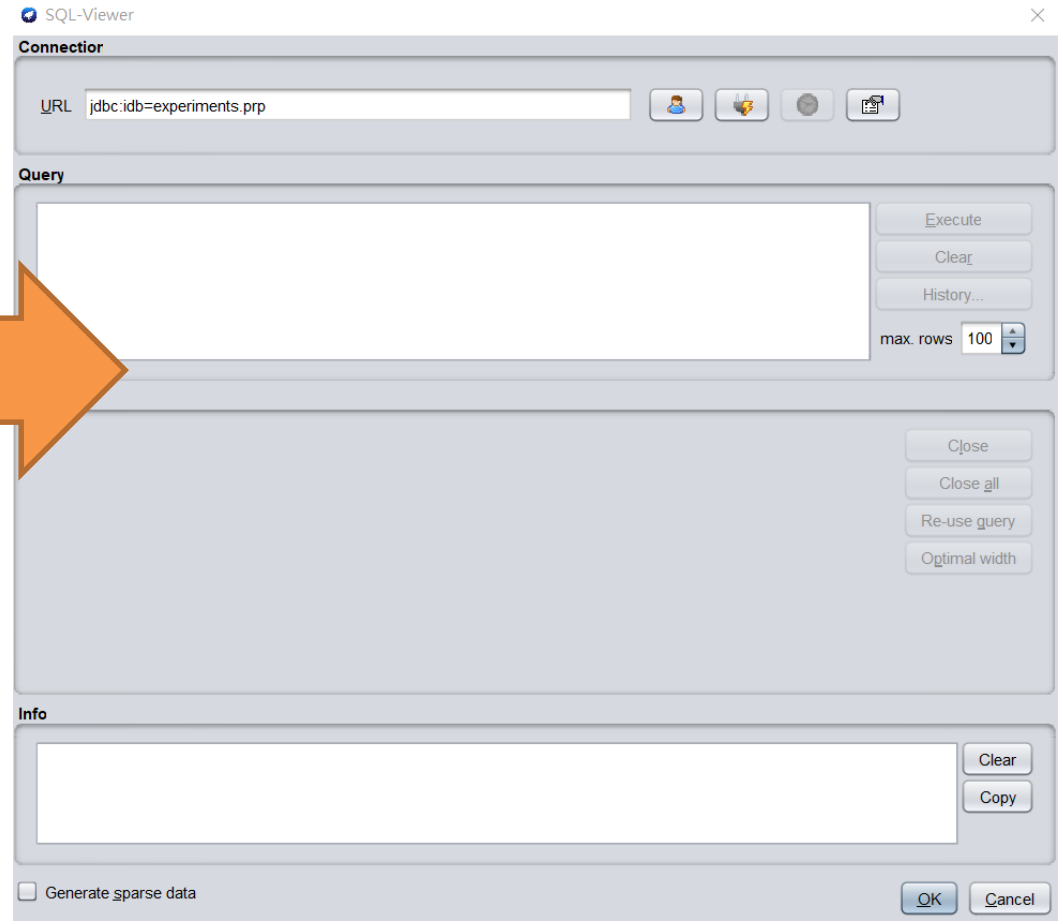
Weka-3-6\documentation.html

- … 在「All classes」列表中找到J48

Lesson 1.5: 命令行介面

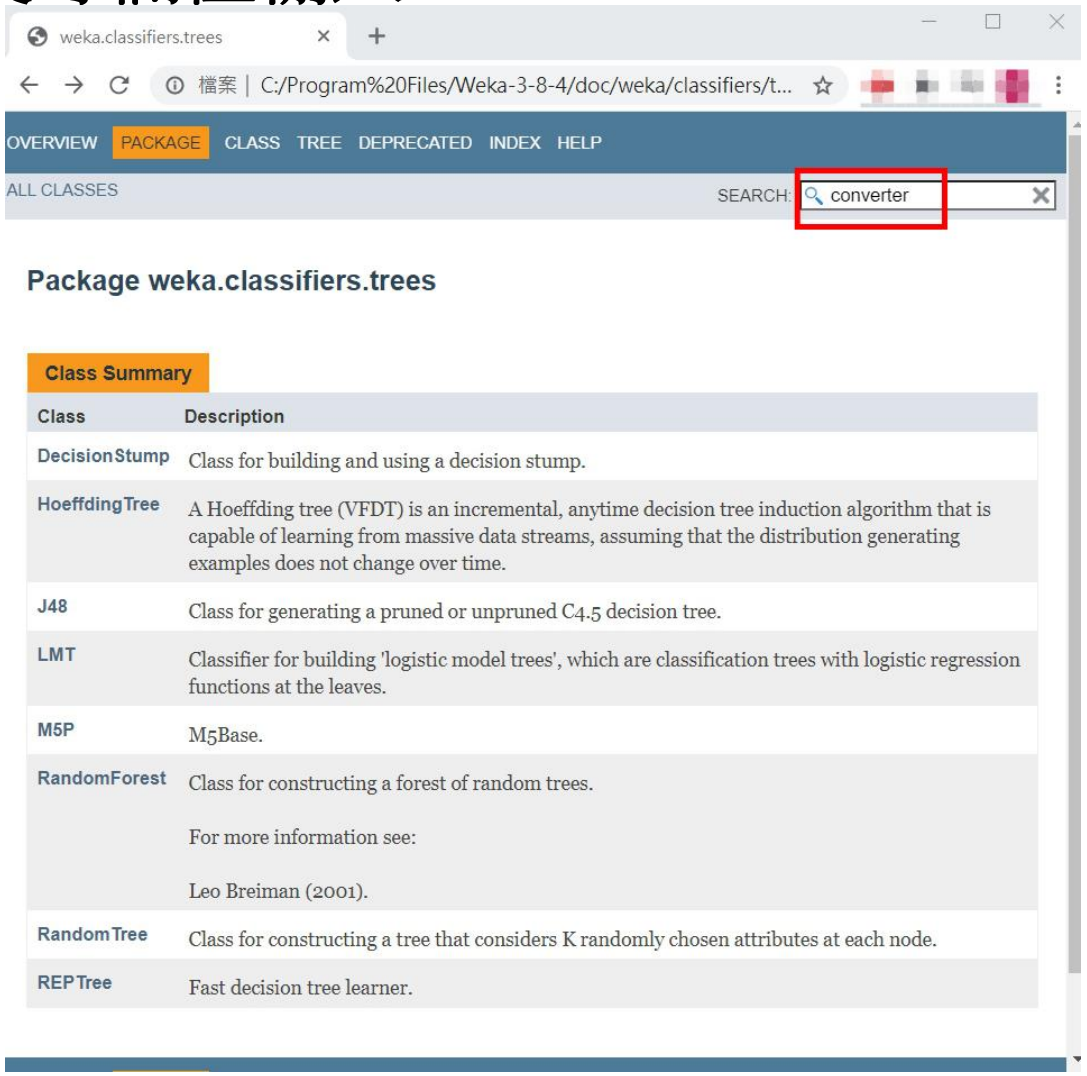
我們還可以在Javadoc找到一些想知道的資訊。
1.回到Explorer介面的Preprocess面板，可以看到Open DB...的按鈕。
我們左鍵單擊它，可以開啟右圖的視窗。事實上，它是一個轉換器(converter)。





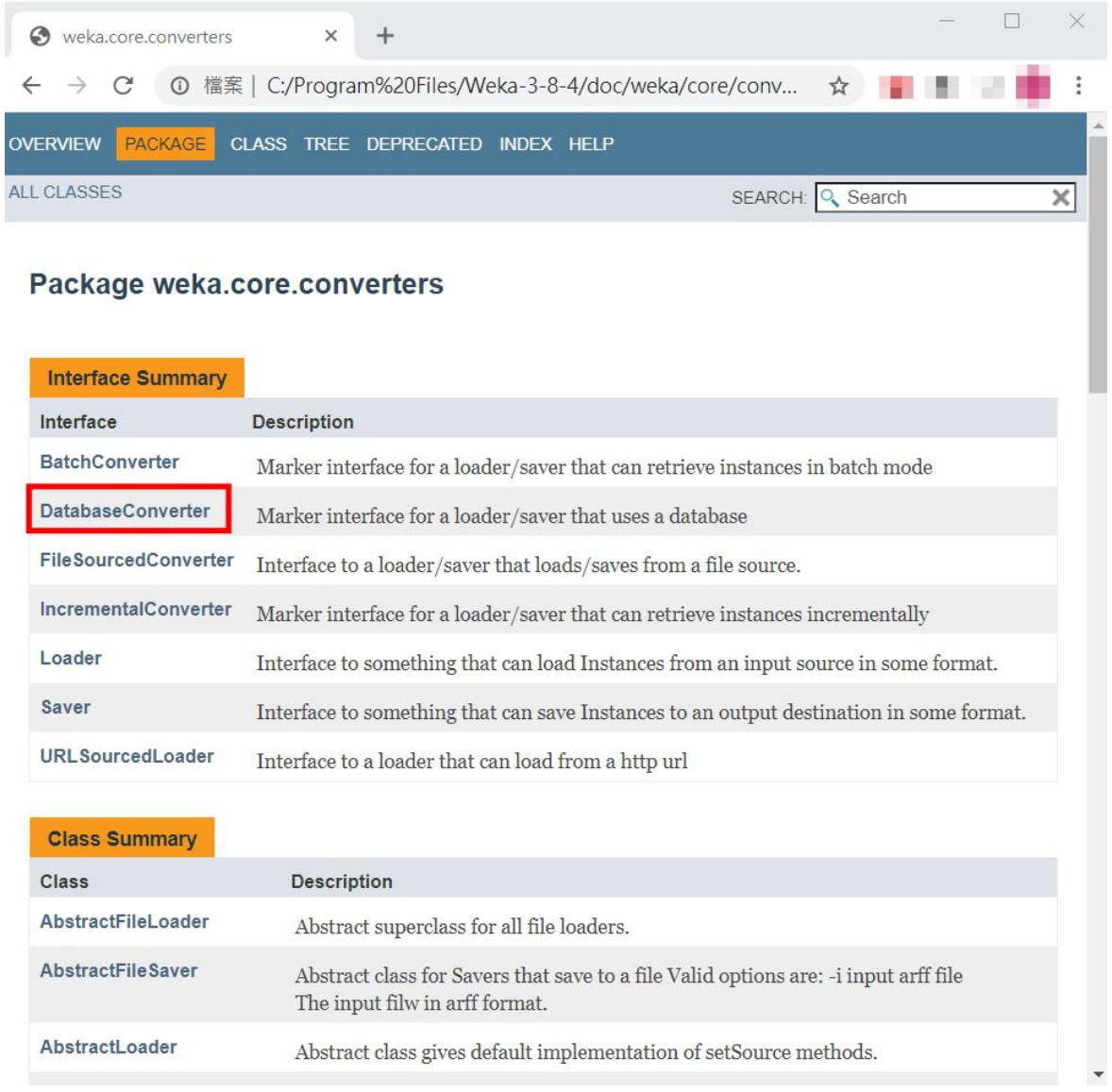
Lesson 1.5: 命令行介面

2. 我們想知道關於這個Open DB...按鈕的功能，可以在Javadoc視窗右上方的SEARCH搜尋欄位輸入converter



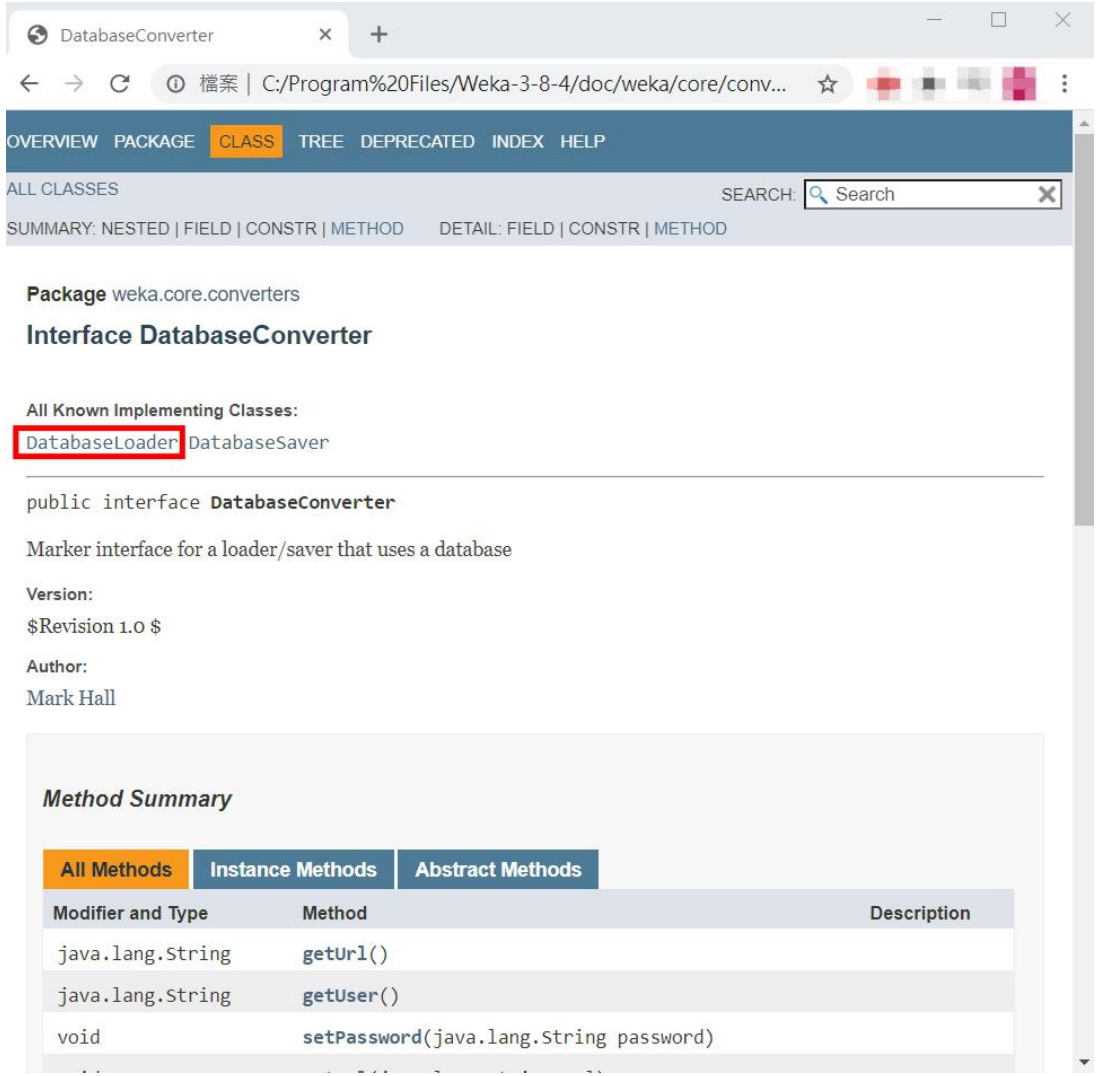
Lesson 1.5: 命令行介面

3.在出現的結果中，以左鍵單擊DatabaseConverter



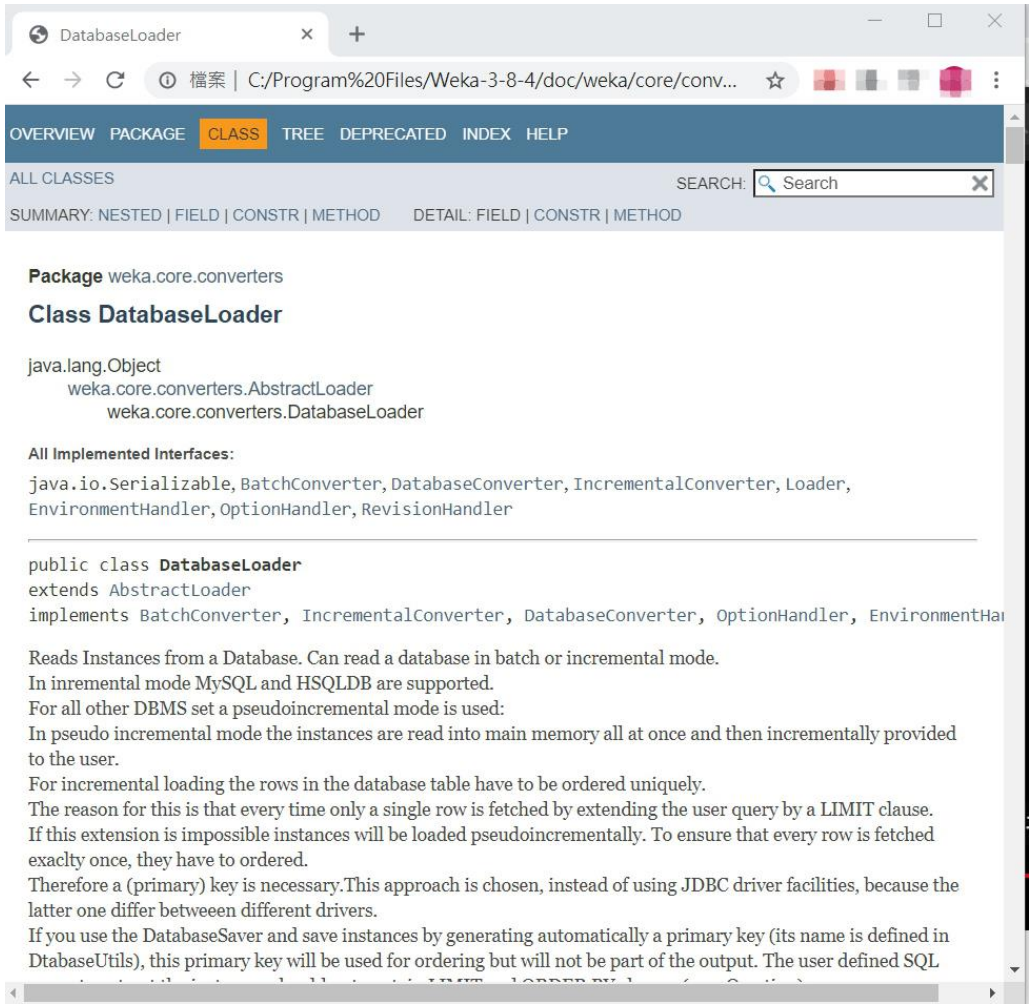
Lesson 1.5: 命令行介面

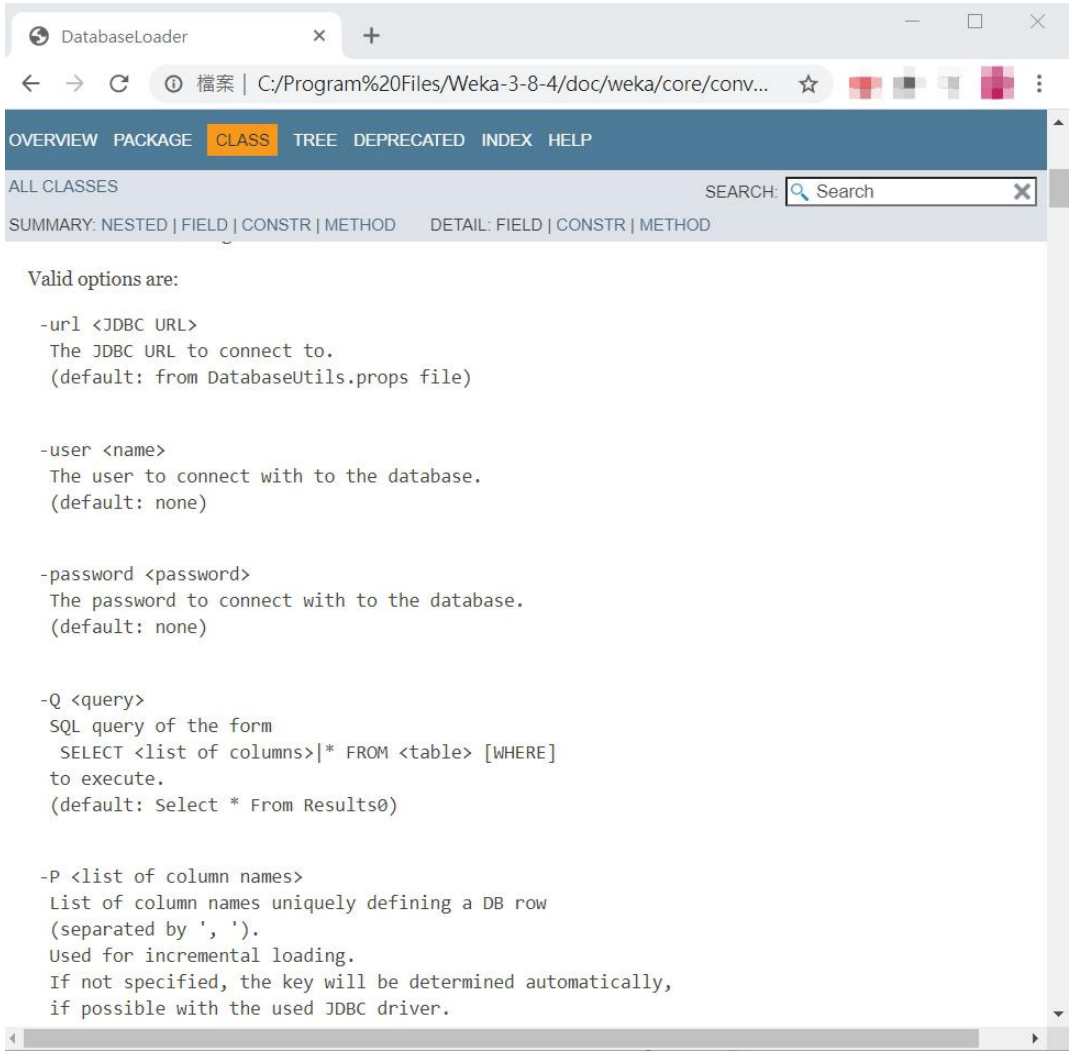
4.進入頁面後，以左鍵單擊DatabaseLoader



Lesson 1.5: 命令行介面

5. 以下就是關於Database converter的詳細說明。





Lesson 1.5: 命令行介面

使用**Javadoc**

- 找到converter的package

  weka.core.converters

- 找到databaseLoader的類別(class)

  weka.core.converters.DatabaseLoader

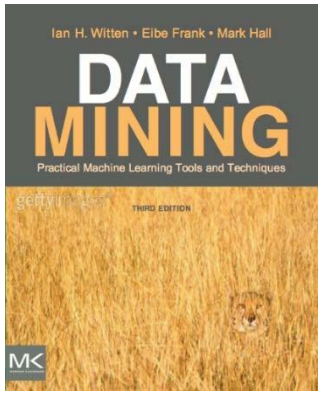
- 它可以從任何JDBC資料庫載入資料

  specify URL, password, SQL query

- 它就在Explorer的Preprocess面板, 但是說明文件卻在**Javadoc**中

Lesson 1.5: 命令行介面

- 可以從命令行做任何Explorer可以做到的事情
- 然而人們通常開啟一個終端機視窗
  - 然後進行編寫(如果你知道怎麼編寫的話)
  - … 但是你需要正確地建置你的環境
- 可以從Explorer複製及貼上已配置的分類器
- 優點: 更能掌控記憶體的使用(下一堂課會提到)
- Javadoc是關於Weka的精確說明文檔



課程文本

- Chapter 14 *The Command-Line Interface*

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Class 1 – Lesson 6

處理大數據

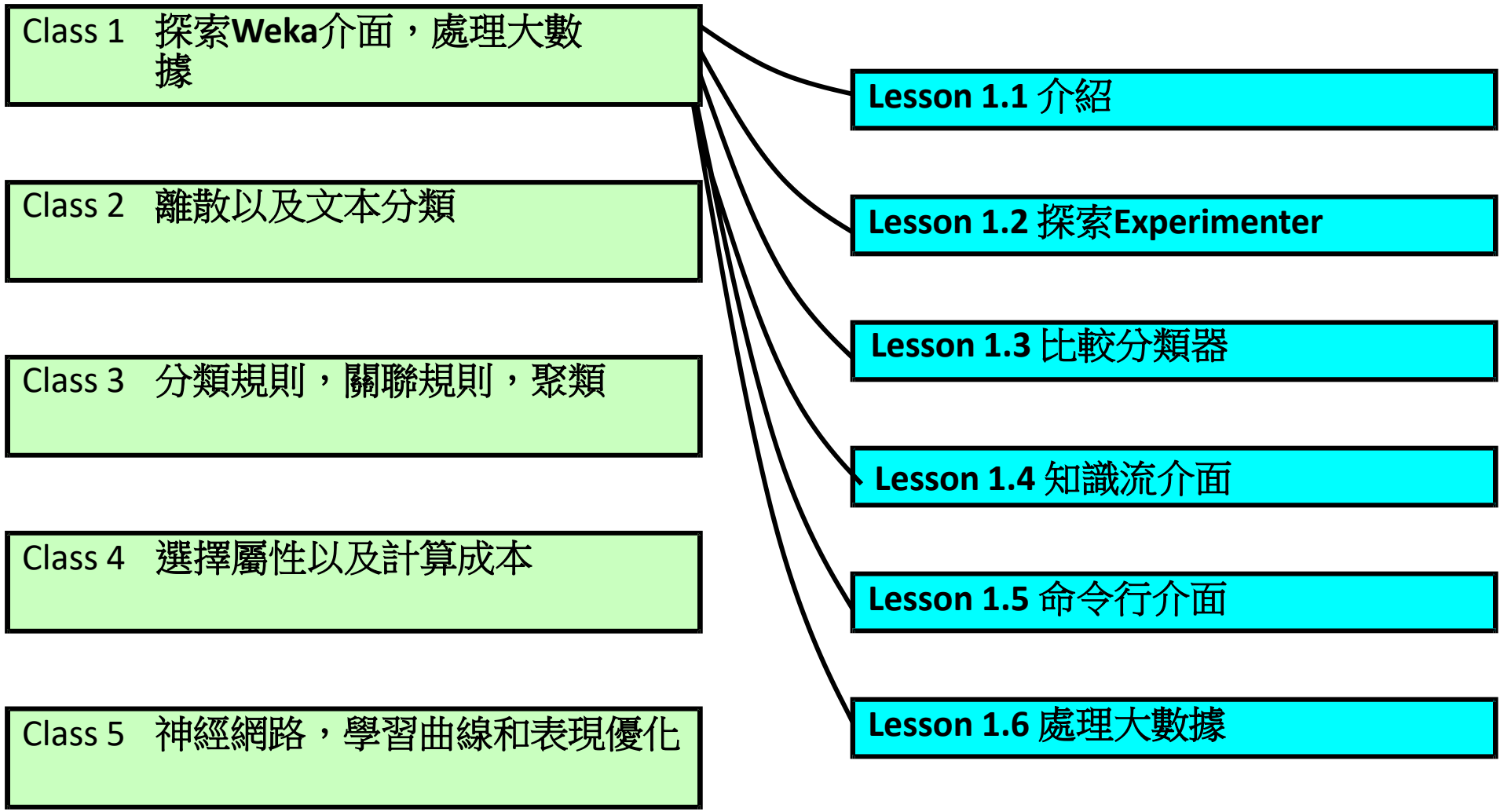
(Working with big data)

Ian H. Witten

Department of Computer Science
University of Waikato
New Zealand

weka.waikato.ac.nz

Lesson 1.6: 處理大數據



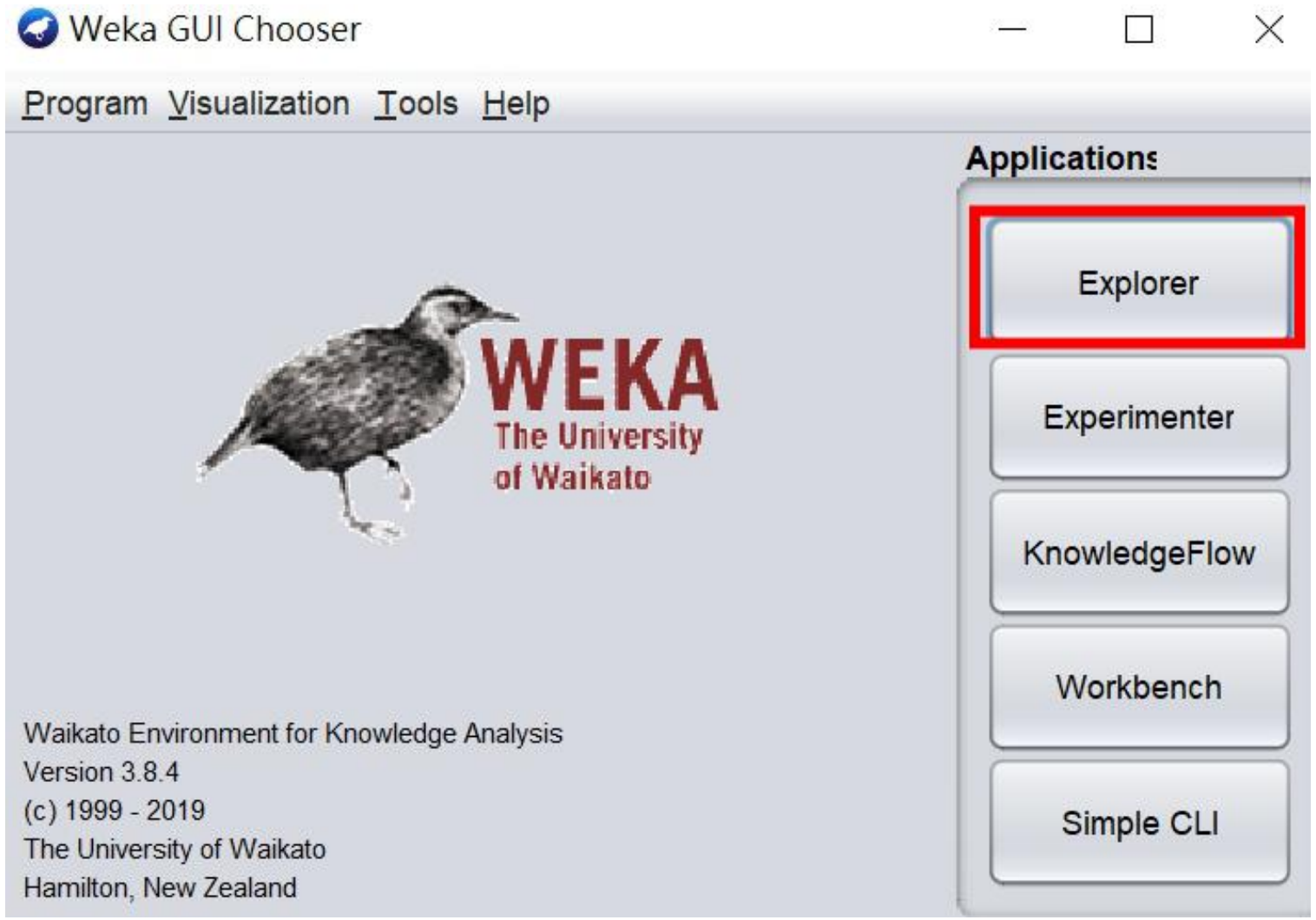
Lesson 1.6: 處理大數據

Explorer可以處理多大的資料？ (~ 1M 個實例, 25 個屬性)

- 記憶體資訊: Explorer, 右鍵單擊Status
  - *Free/total/max: 226,366,616 / 236,453,888 / 954,728,448 (bytes) [1 GB]*
  - *那是什麼意思？ 查看Java的freeMemory(), totalMemory(), maxMemory() 指令*
- 讓它崩潰看看！
- 下載一個巨大的資料集？
  - *在相關的課後練習中使用covertype資料集*
  - *580,000 個實例, 54 個屬性(0.75 GB 未壓縮的)*
- Weka 資料產生器
  - *Preprocess 面板, Generate, 選擇LED24; 顯示文字: 100 個實例, 25 個屬性*
  - *100,000 examples (使用spilt切割!) NaiveBayes 74% J48 73%*
  - *1,000,000 examples NaiveBayes 74% J48 記憶體用光了*
  - *2,000,000 examples Generate 的處理會停掉*
- (執行Weka的console版本)

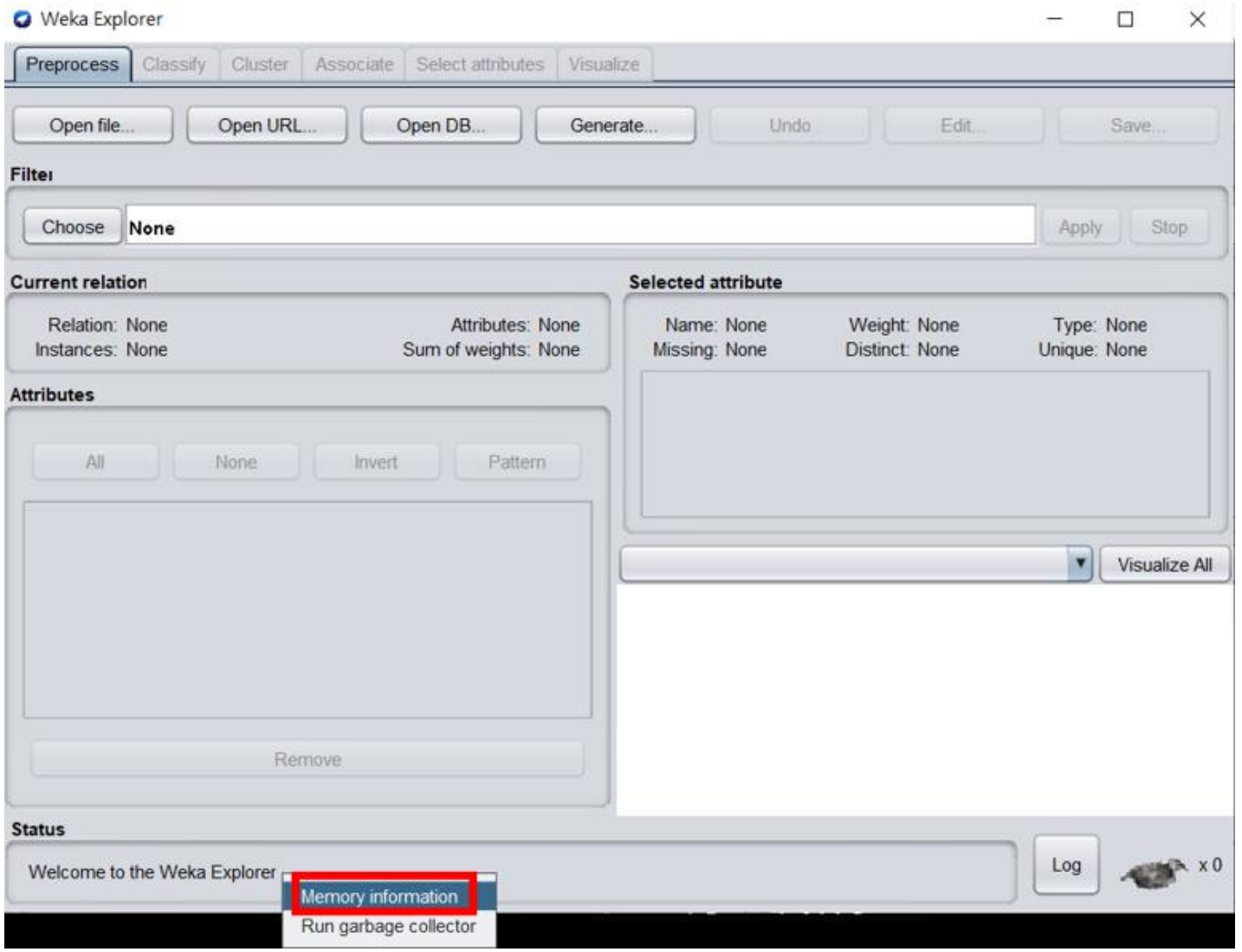
Lesson 1.6: 處理大數據

1.開啟Weka的Explorer



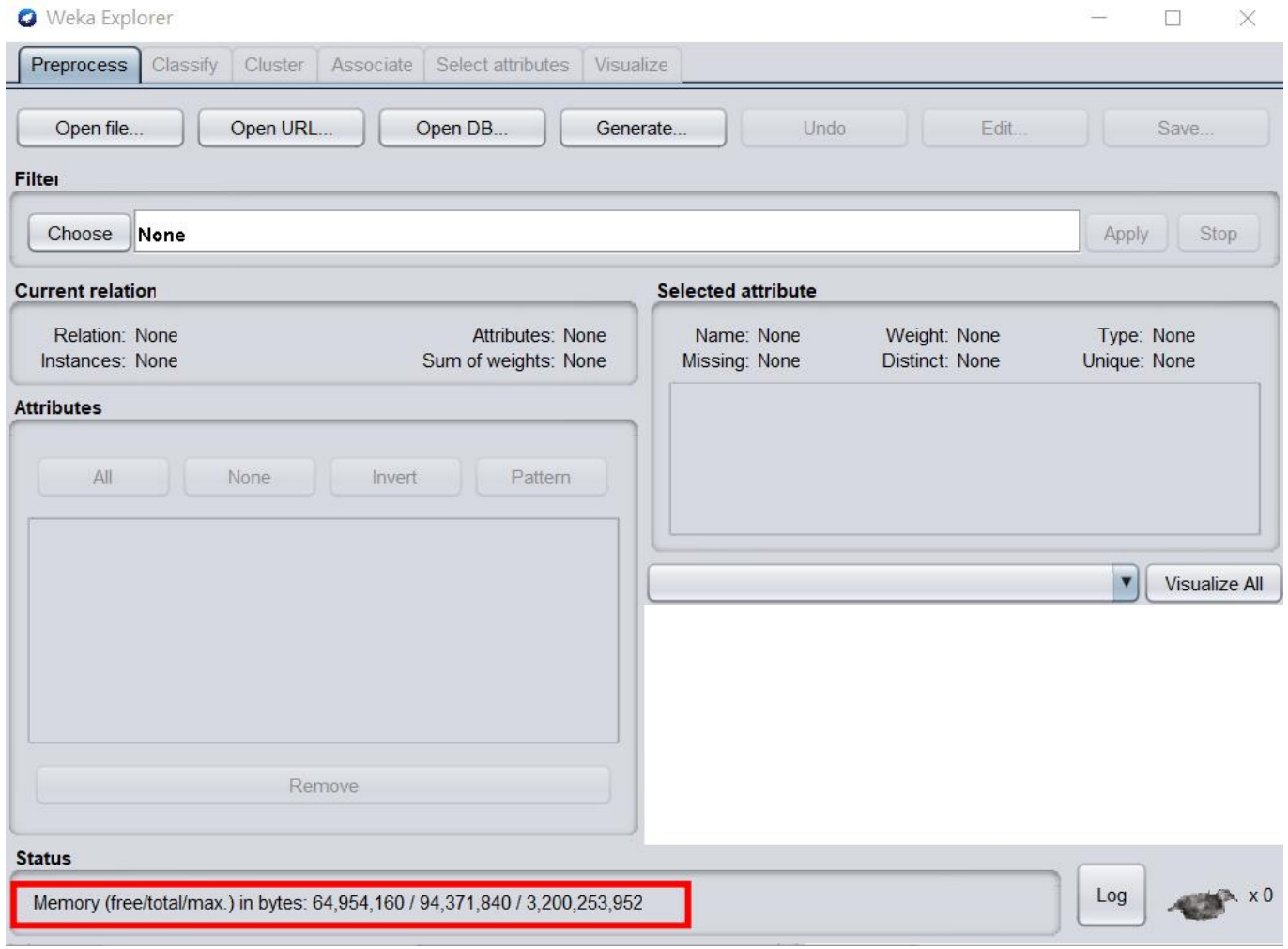
Lesson 1.6: 處理大數據

查看記憶體資訊：
在Explorer介面的下方，
右鍵單擊Status的區域，
在出現的選單中左鍵單擊
Memory information選項



Lesson 1.6: 處理大數據

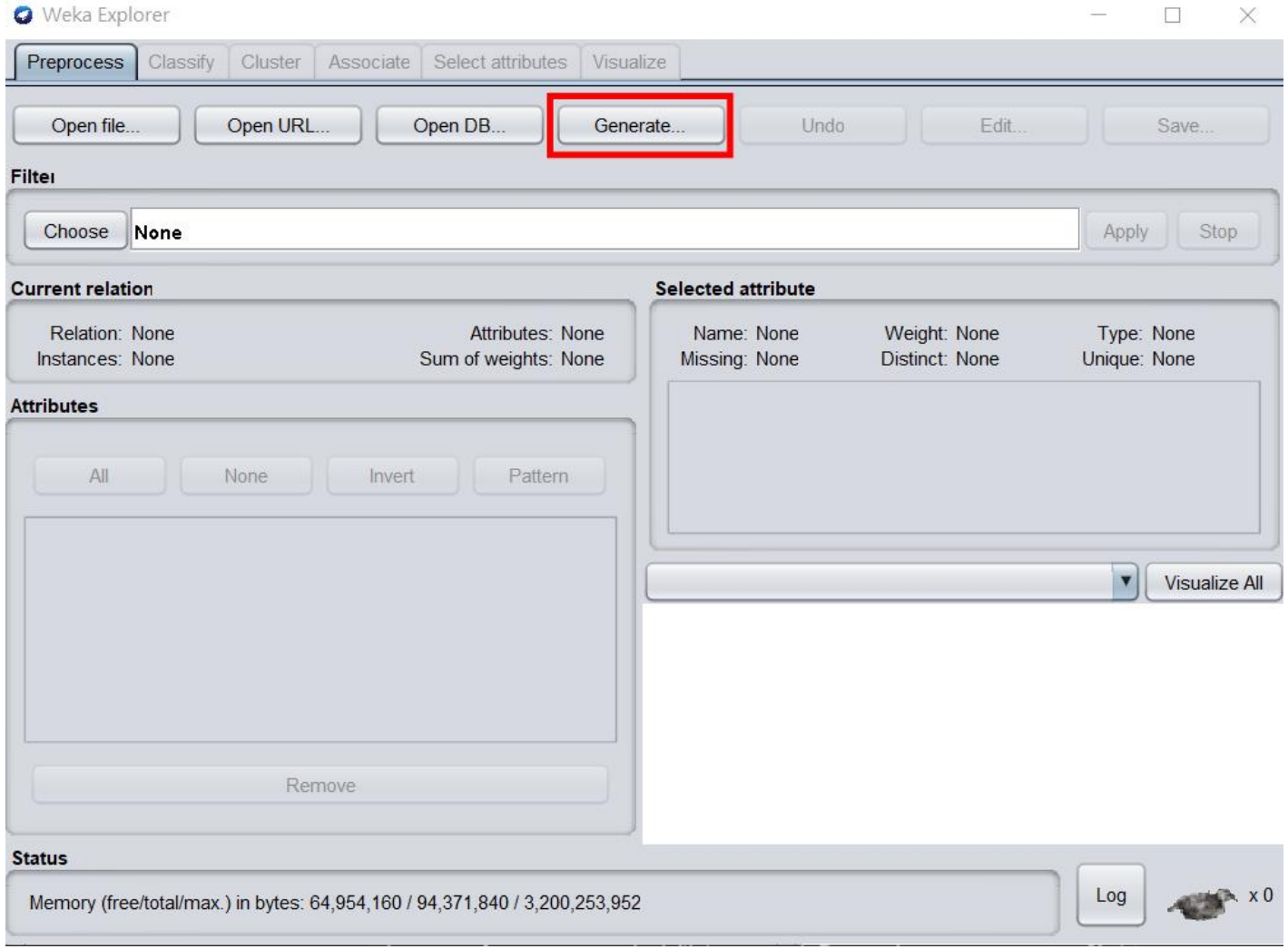
▼可以看到三個數字。最後一個數字是分配給Weka的內存總量。另外兩個數字有點複雜。如果想了解更多，可以看看Java函數freeMemory()和totalMemory()。



Lesson 1.6: 處理大數據

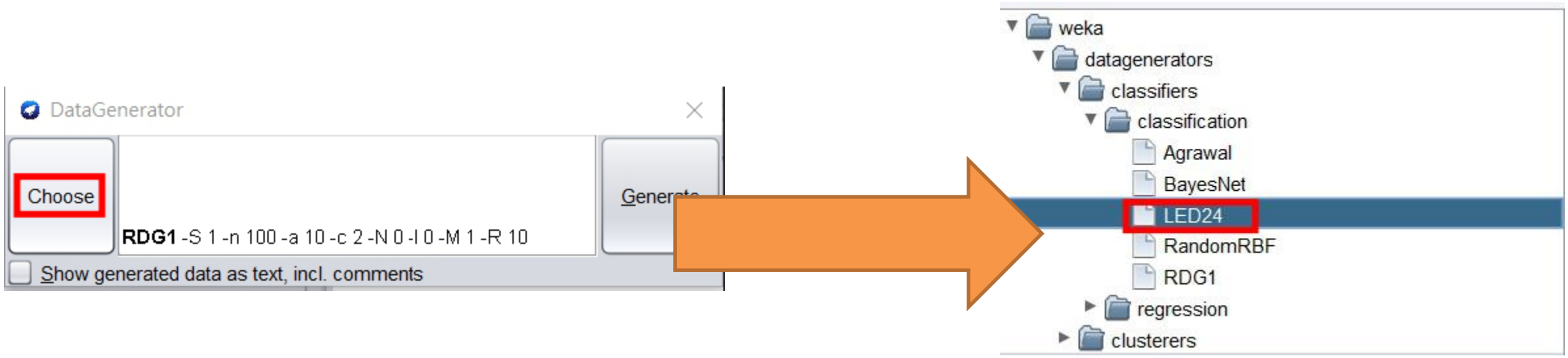
接著，我們來嘗試記憶體的極限。

1.在Explorer介面左鍵單擊上方Generate按鈕。



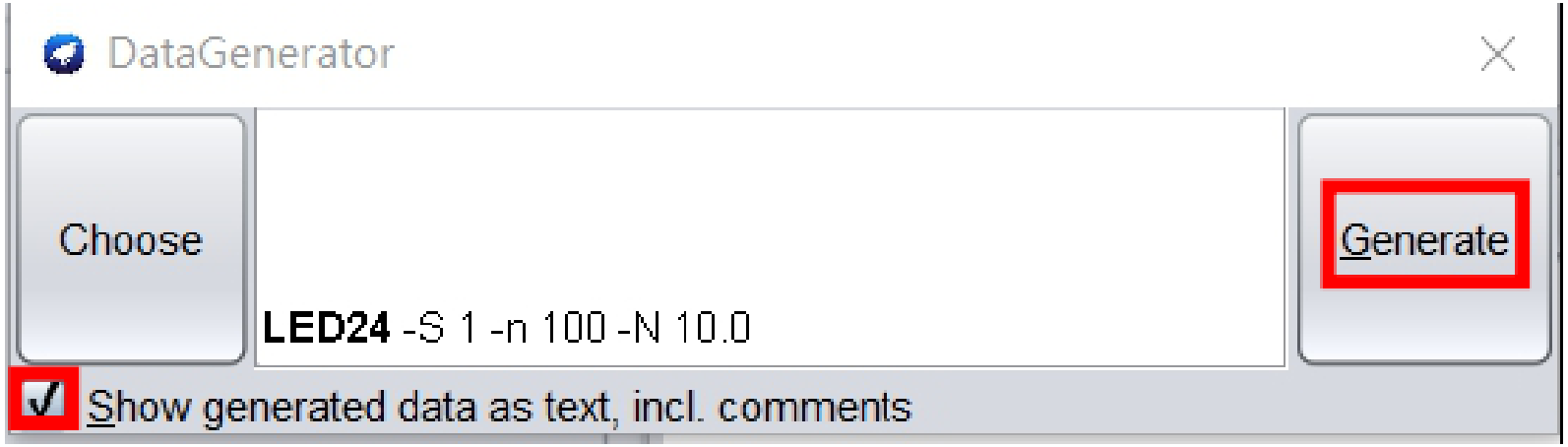
Lesson 1.6: 處理大數據

2.在出現的視窗中左鍵單擊Choose按鈕，再以左鍵單擊選單中的LED24



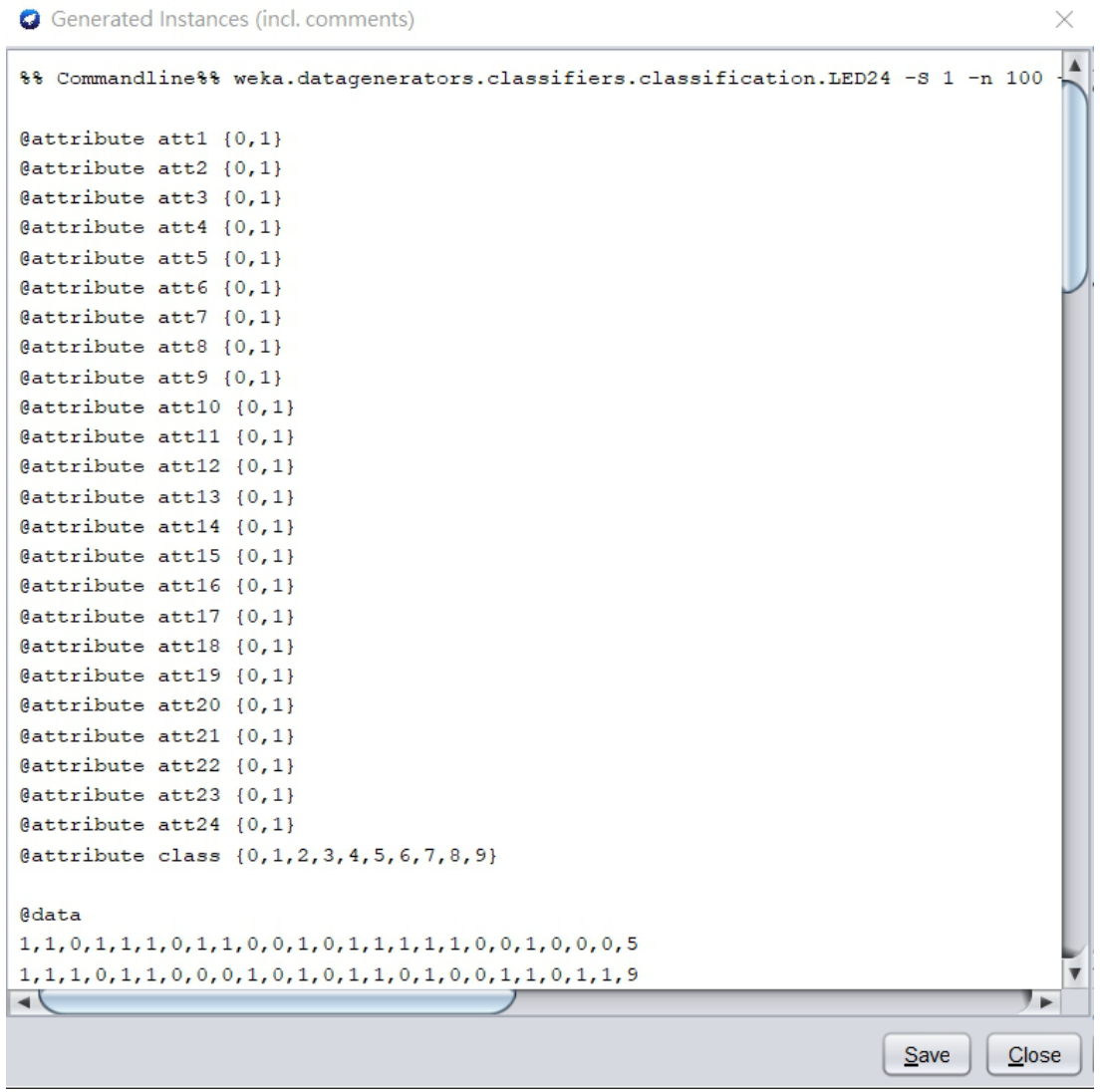
Lesson 1.6: 處理大數據

3. 勾選視窗左下角的方框，然後左鍵單Generate按鈕



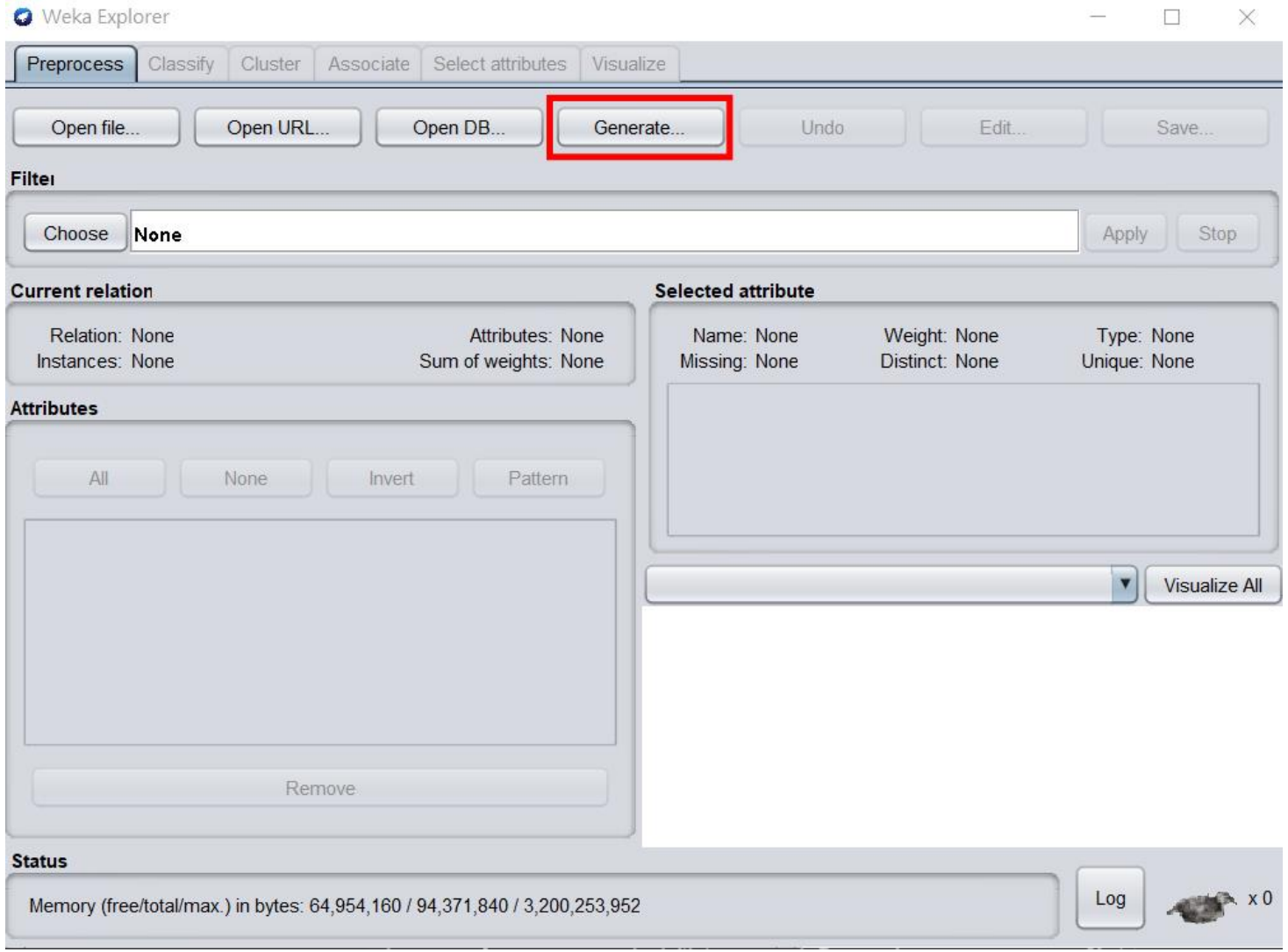
Lesson 1.6: 處理大數據

▼運行結果



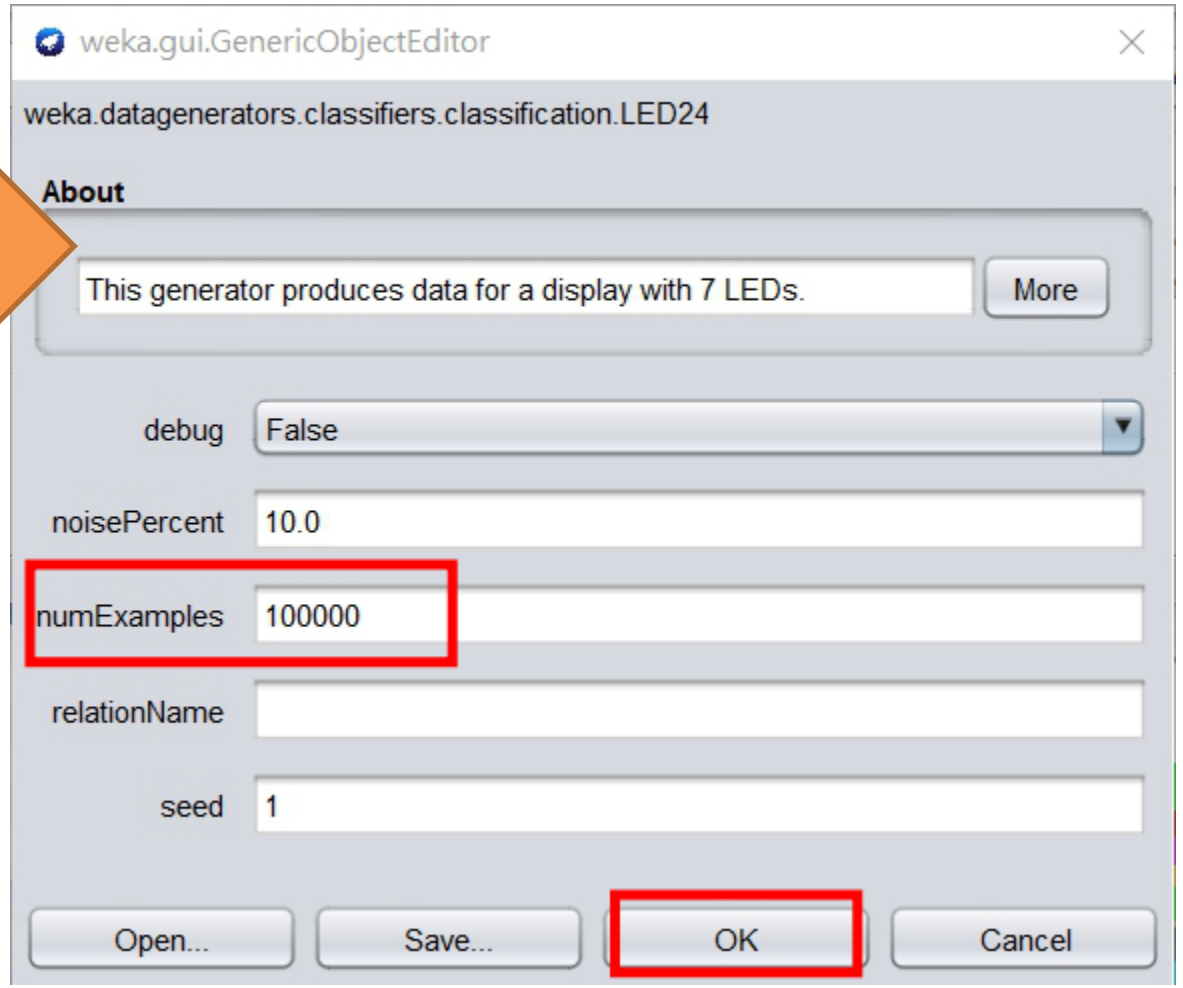
Lesson 1.6: 處理大數據

4.接著我們回到Explorer，左鍵單擊Generate按鈕



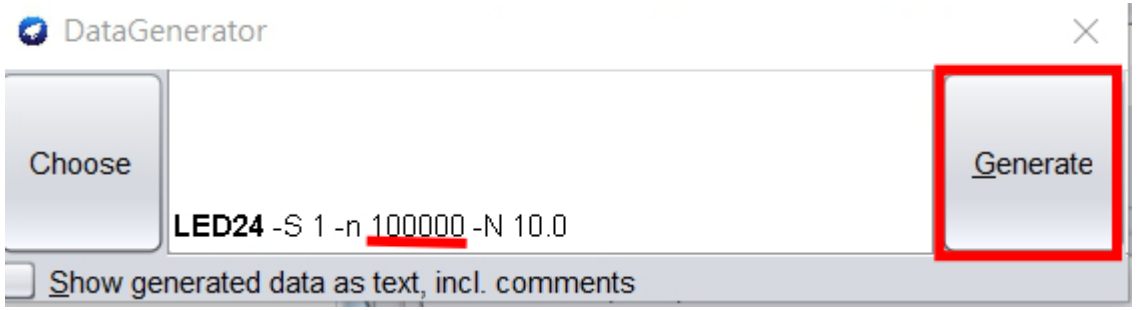
Lesson 1.6: 處理大數據

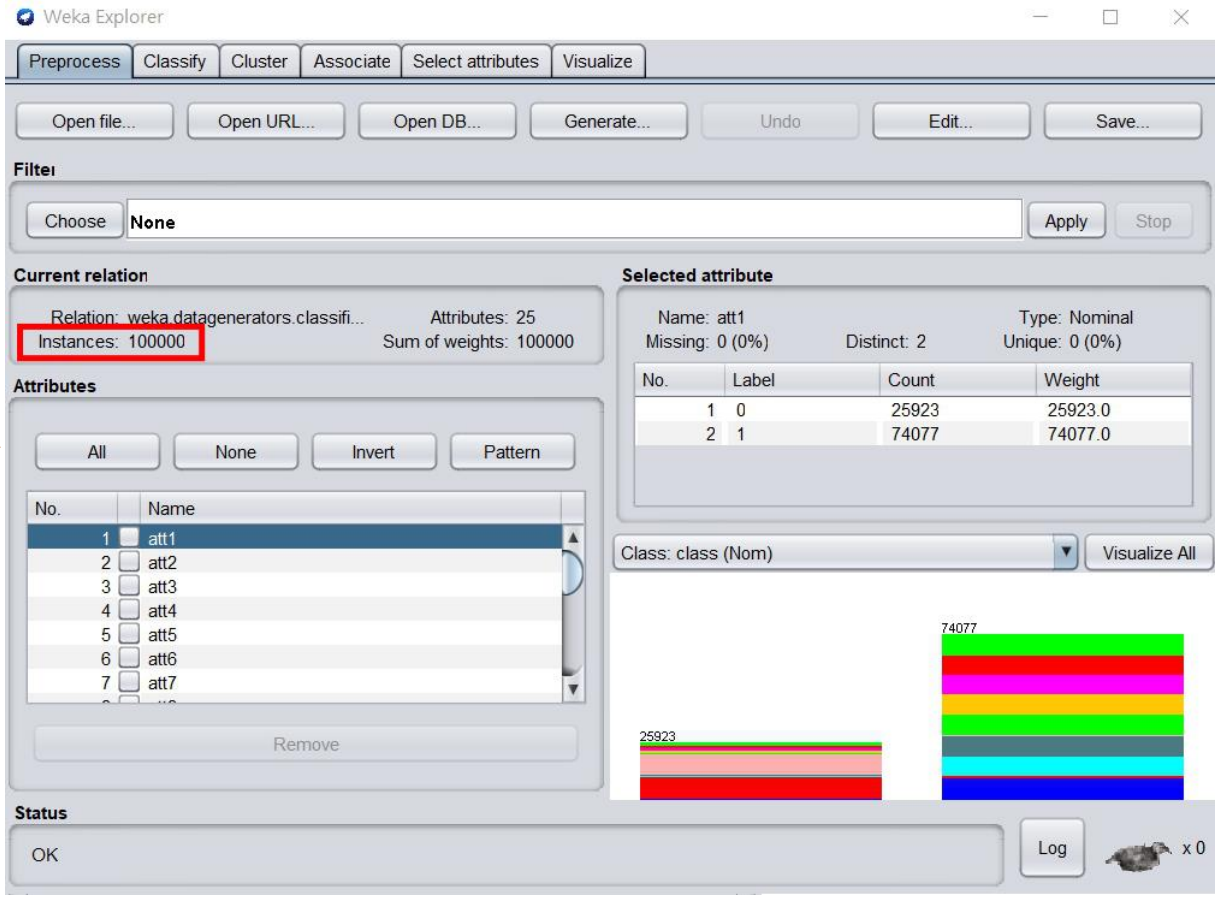
5.左鍵單擊圖中紅色方框中的LED24，開啟右圖配置視窗，將numExamples參數改為100,000，接著左鍵單擊下方OK按鈕



Lesson 1.6: 處理大數據

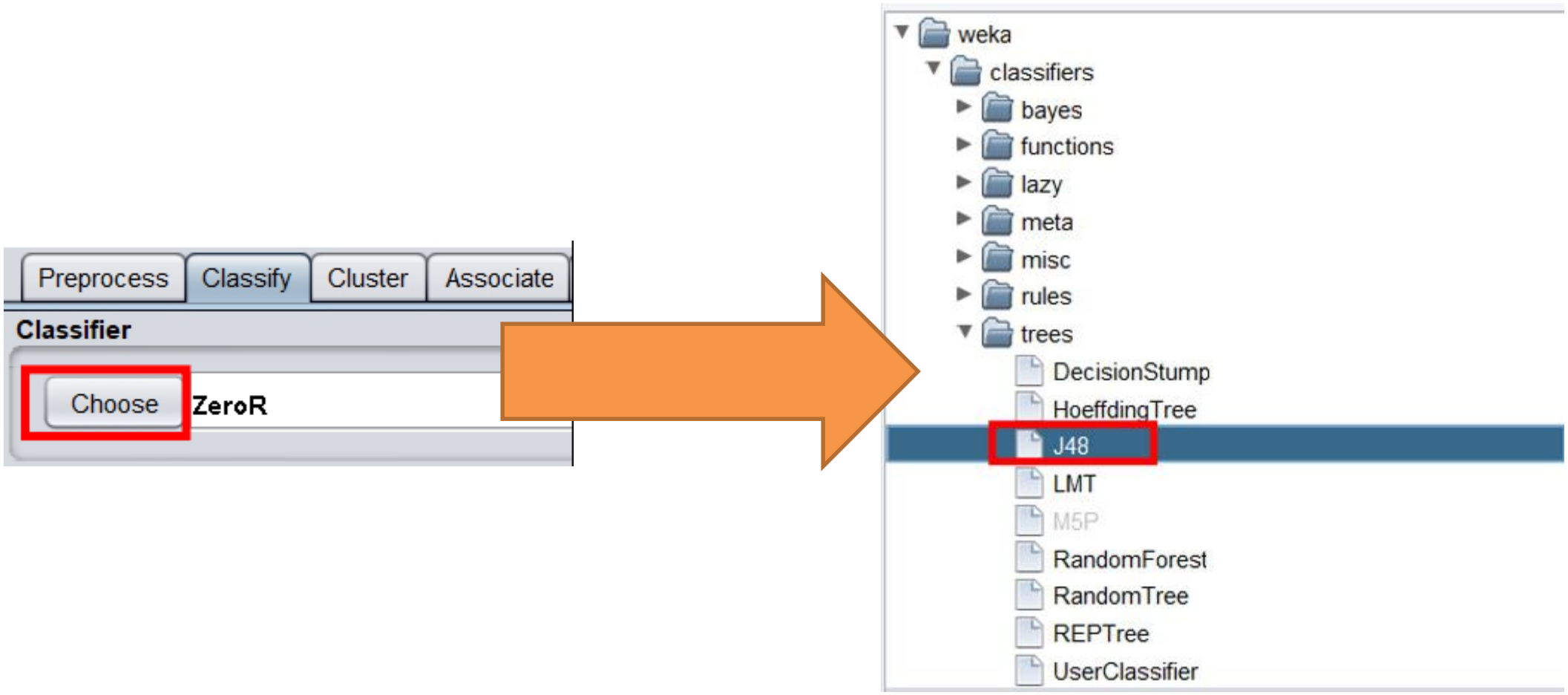
6.確認我們已經將參數n設定為100,000後，按下Generate按鈕，回到Preprocessor介面，可以看到instances數量變成100,000





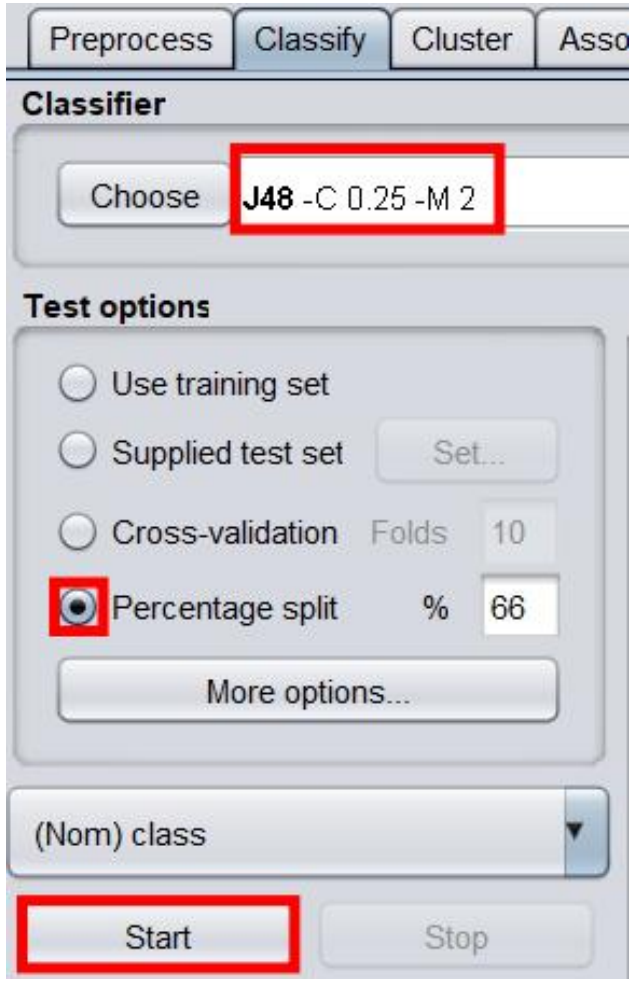
Lesson 1.6: 處理大數據

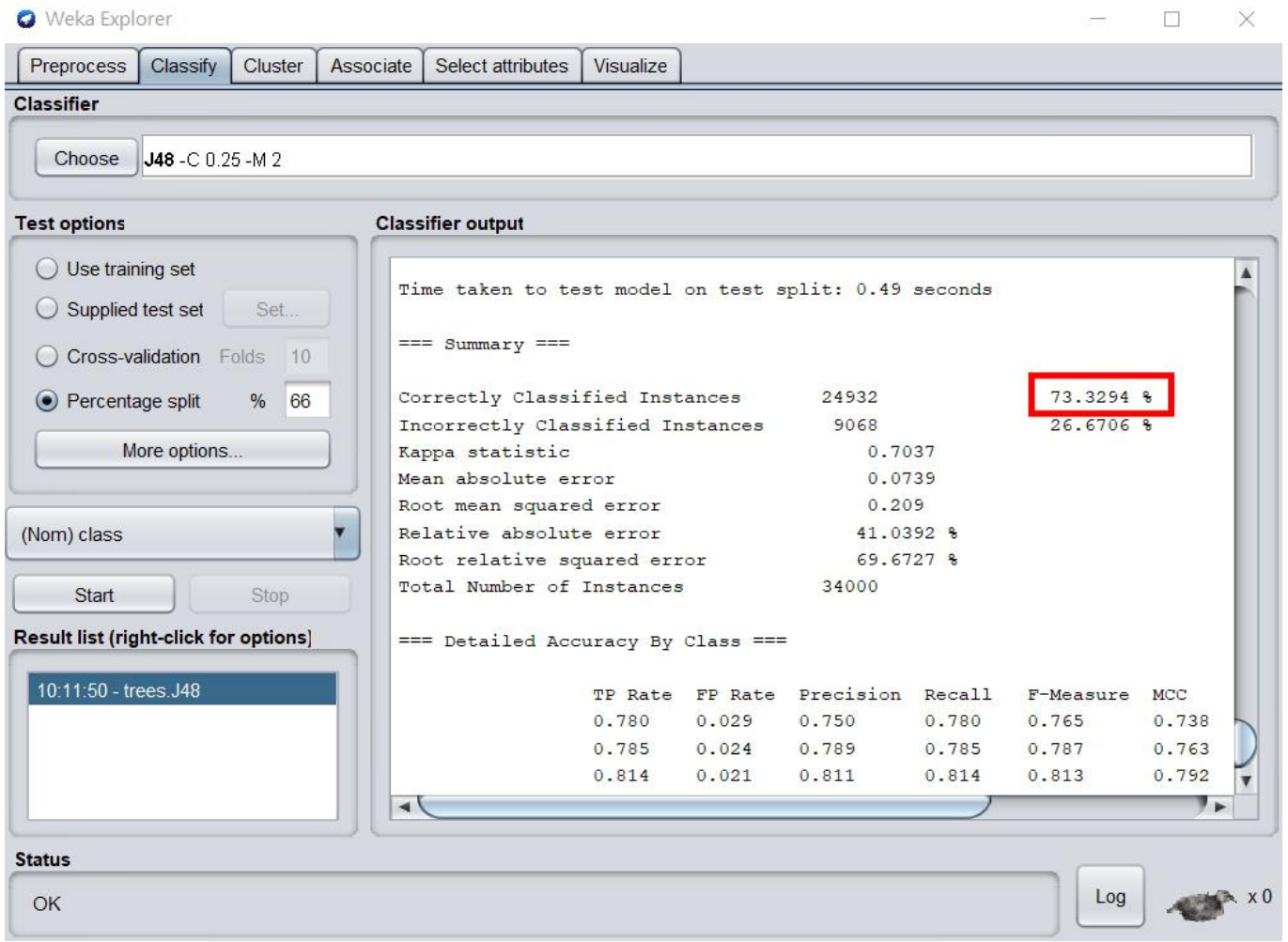
7.切換到Classify介面，左鍵單擊Choose鈕，並在出現的選單中左鍵單擊trees資料夾下的J48分類器。



Lesson 1.6: 處理大數據

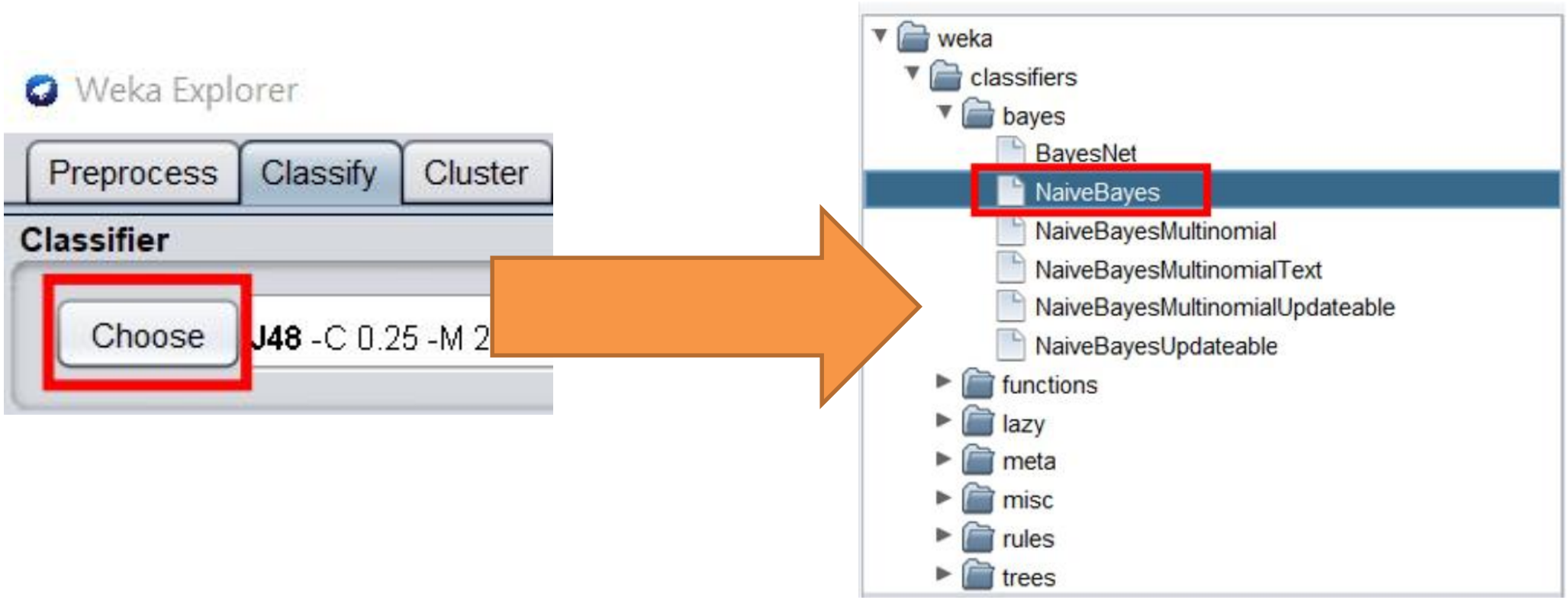
8.確認選擇好J48分類器後，左鍵單擊Percentage split前方圓圈、確定後方為66%後，左鍵單擊Start運行。運行結果：正確率為73.3294%





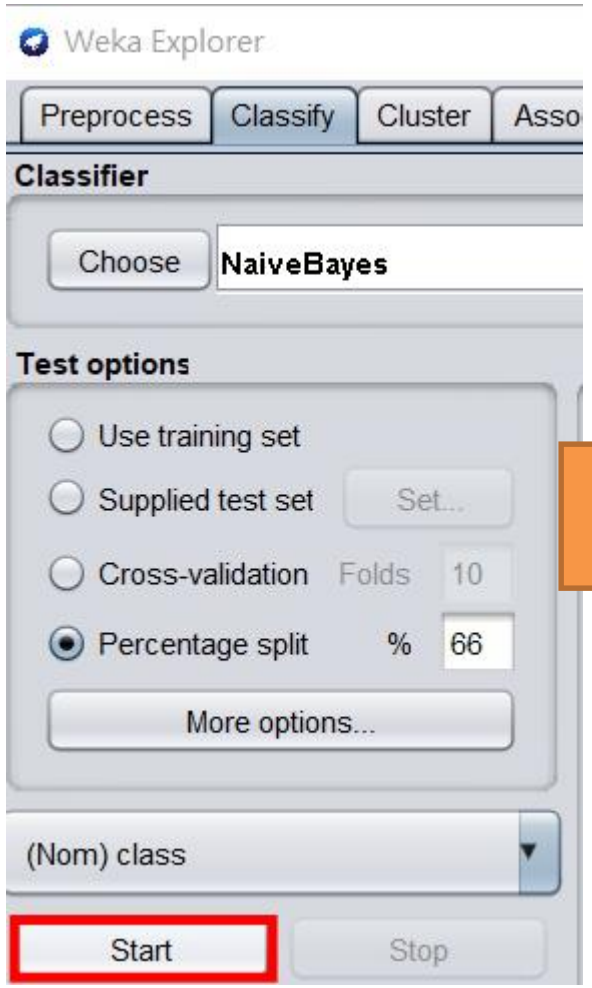
Lesson 1.6: 處理大數據

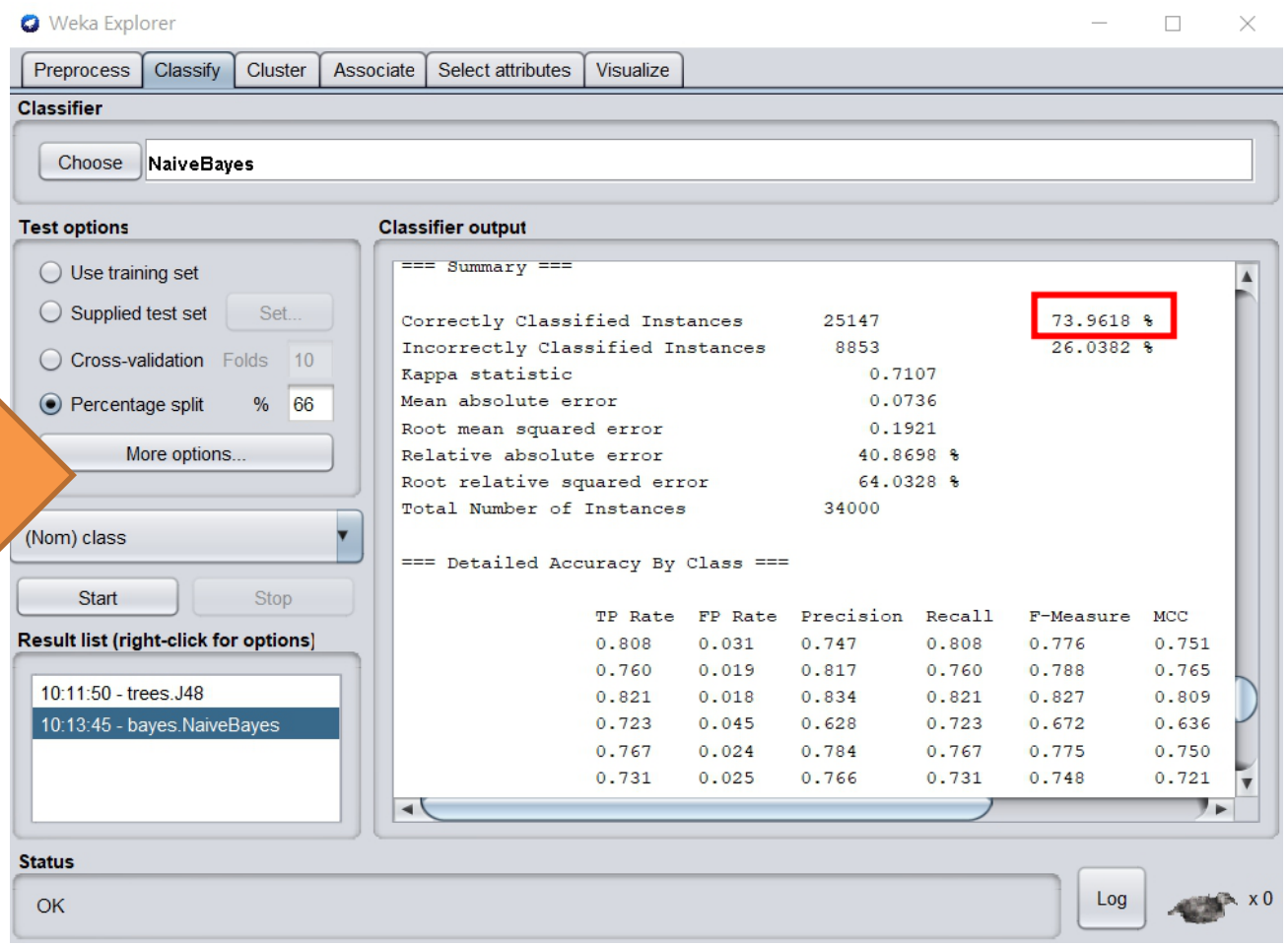
9.左鍵單擊Classify介面中的Choose按鈕，在出現的選單中左鍵單擊NaiveBayes分類器



Lesson 1.6: 處理大數據

10. 左鍵單擊Start按鈕運行分類器。運行結果：正確率為73.9618%。

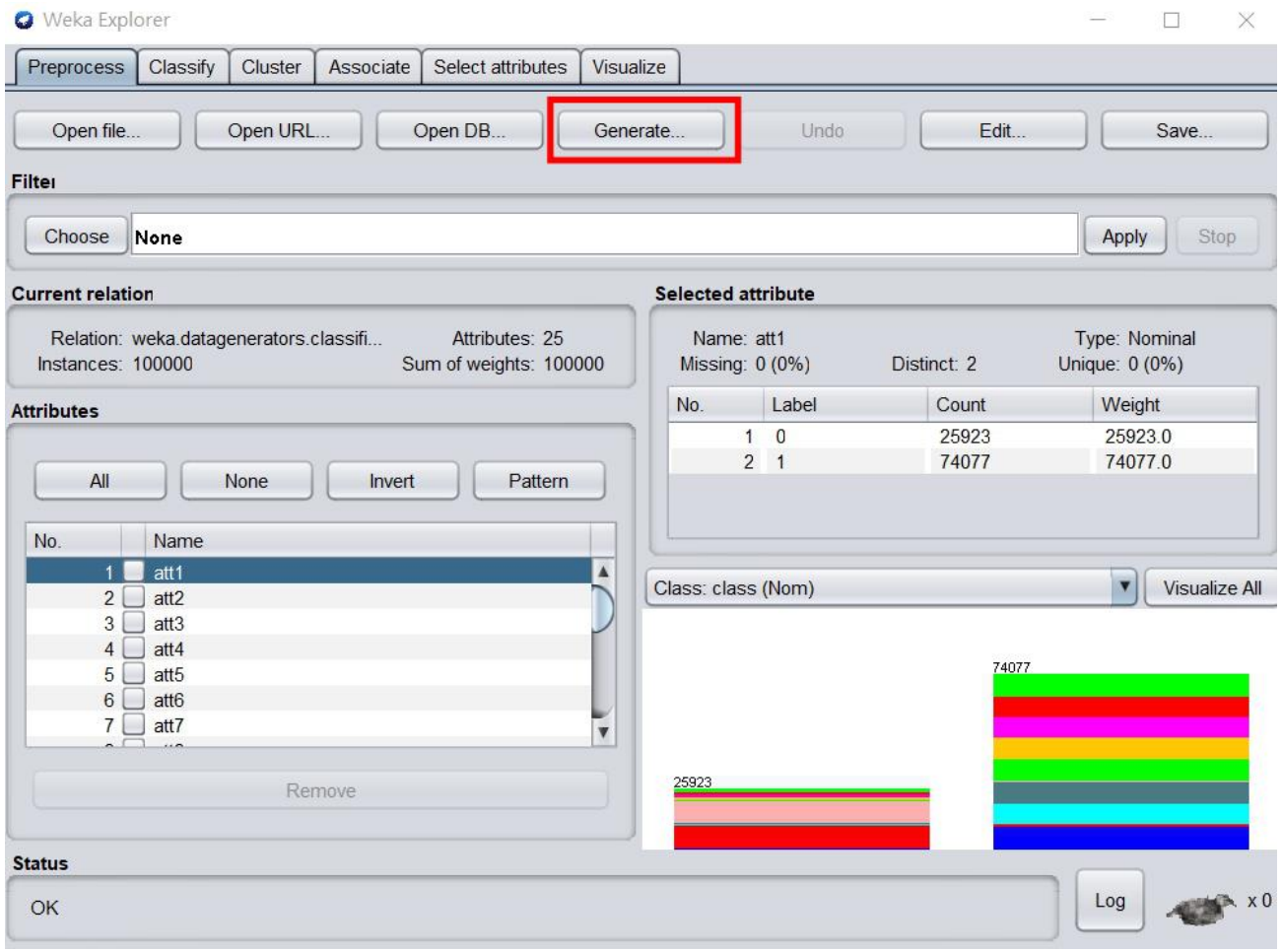




Lesson 1.6: 處理大數據

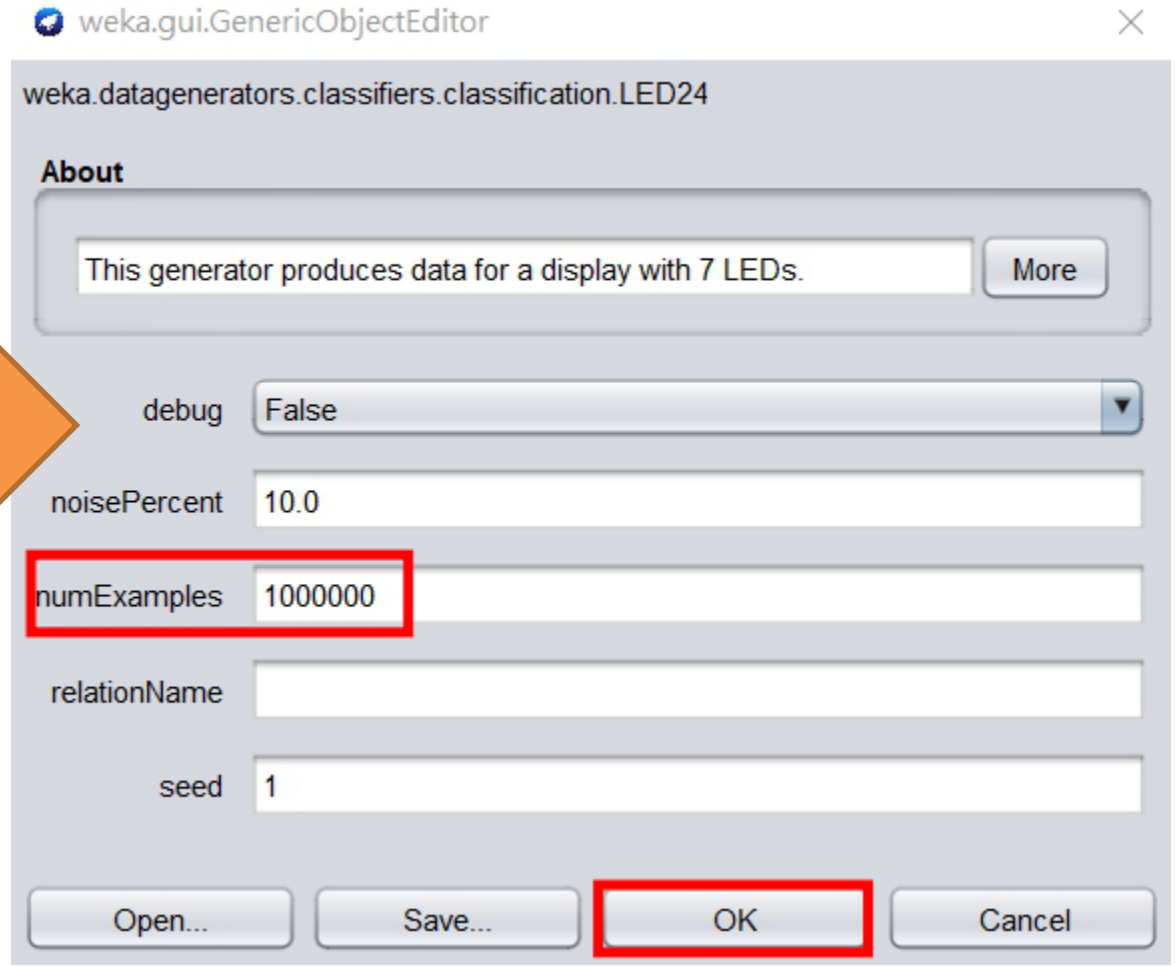
接著測試1,000,000實例數。

11.回到Preprocess介面，左鍵單擊Generate按鈕。



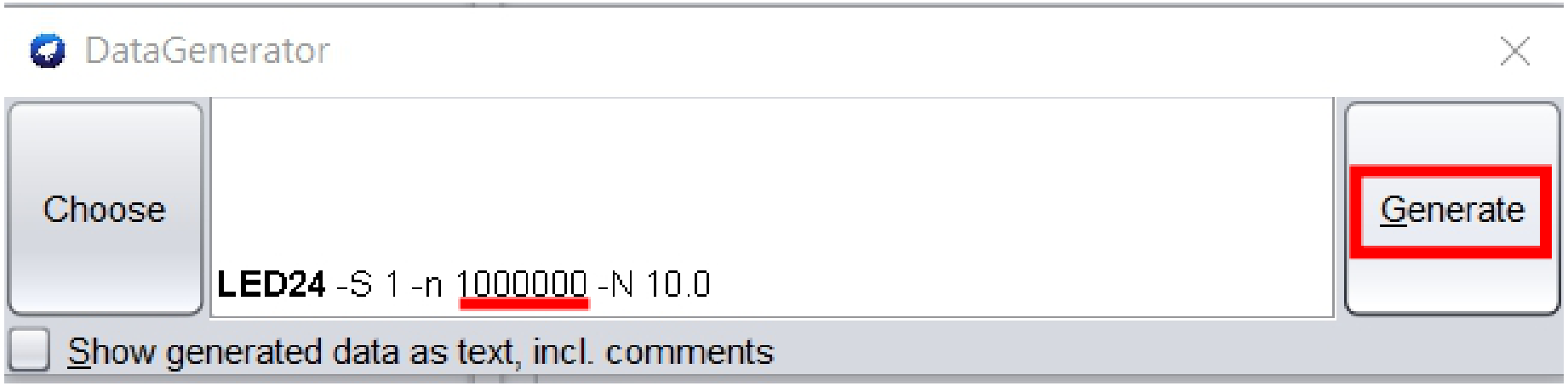
Lesson 1.6: 處理大數據

12.在DataGenerator視窗中，左鍵單擊圖中紅色方框內的LED24，在出現的配置視窗中將numExamples參數改為1,000,000，並按下視窗下方的OK按鈕。



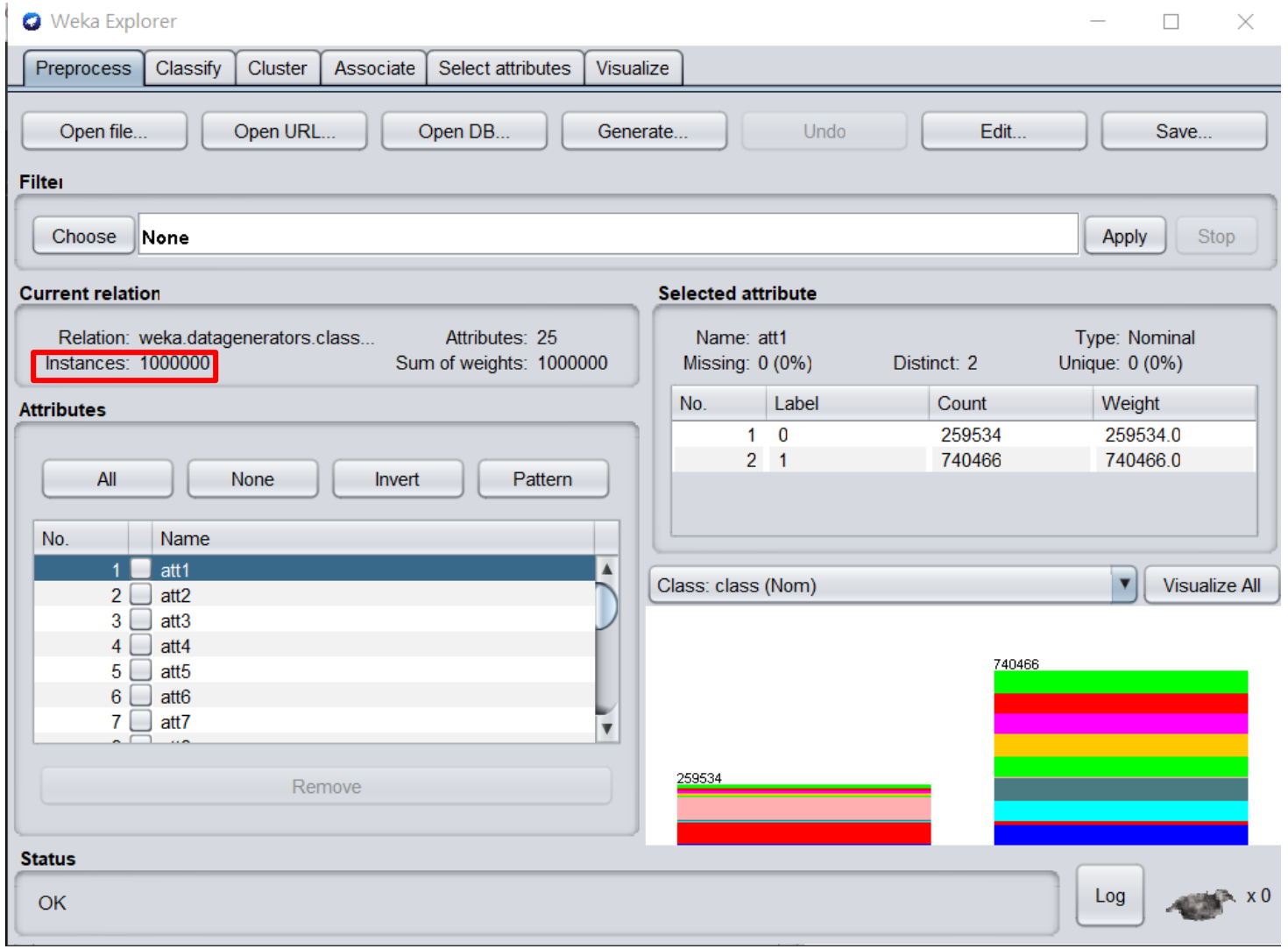
Lesson 1.6: 處理大數據

13.確認參數n已經改為1,000,000後，左鍵單擊Generate按鈕。



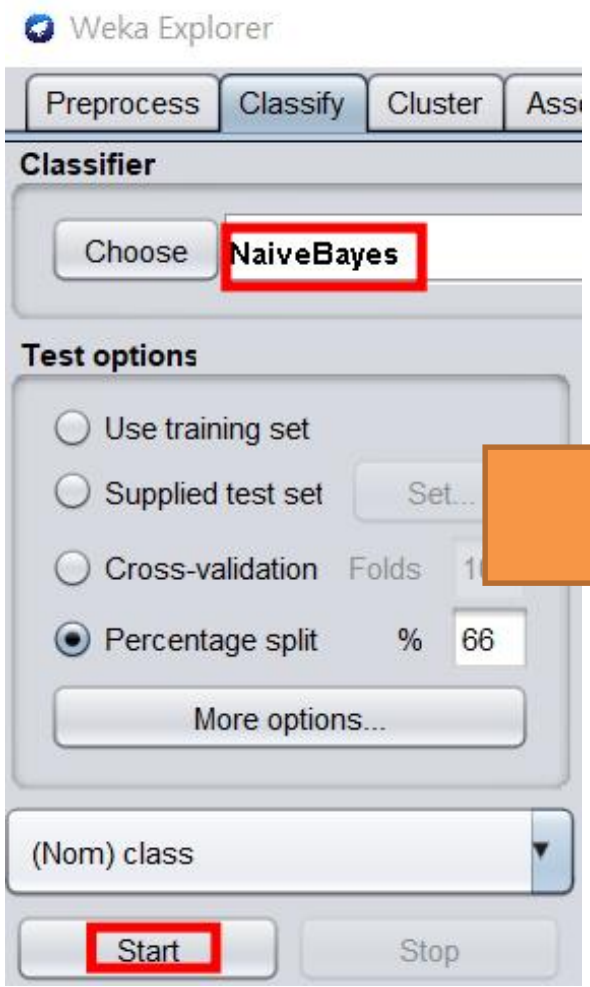
Lesson 1.6: 處理大數據

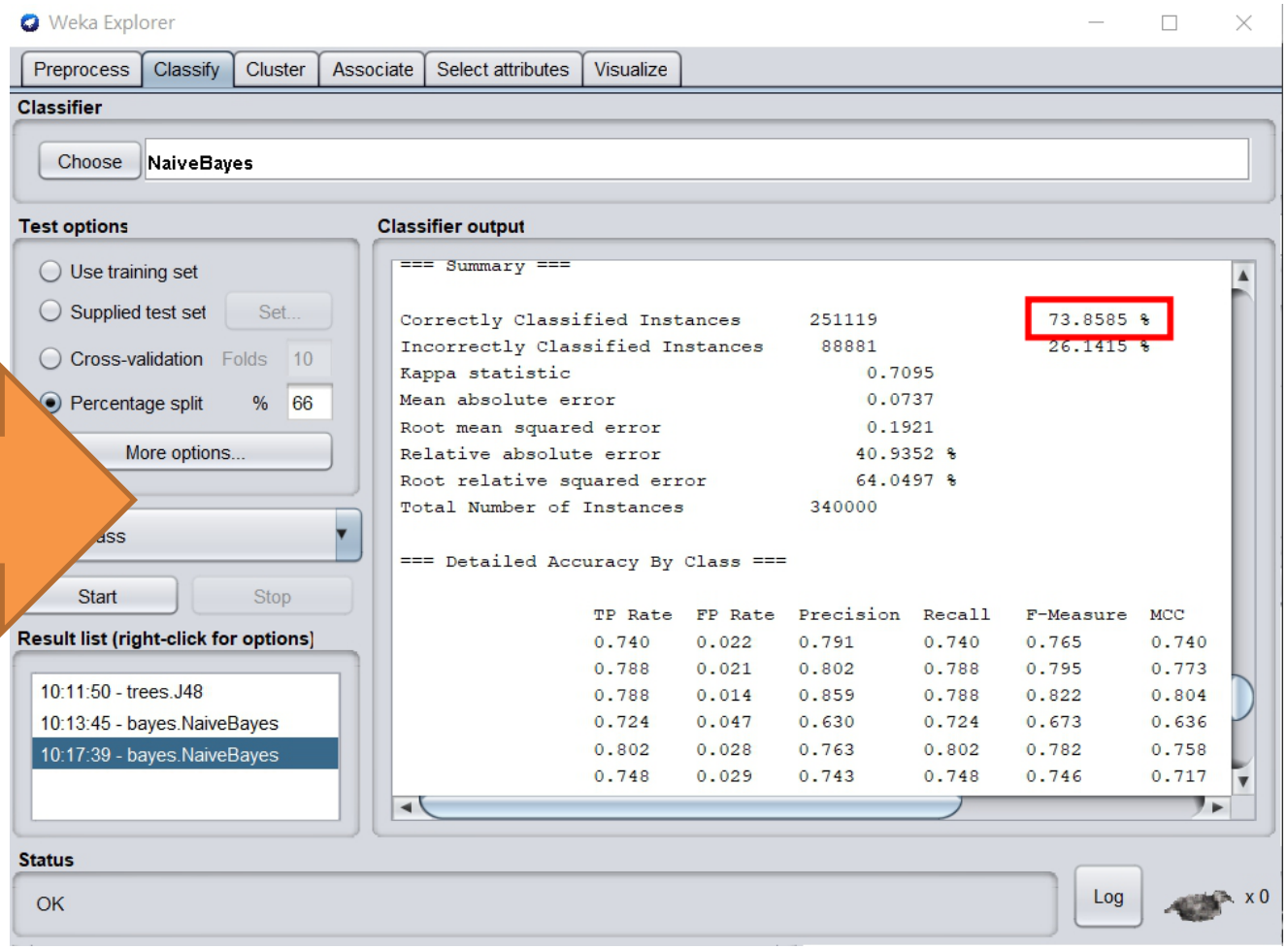
14.回到Preprocess介面可以看到instances數量變為1,000,000



Lesson 1.6: 處理大數據

15.切換到Classify介面，確定分類器為NaiveBayes後按下Start按鈕。運行結果為73.8585%。

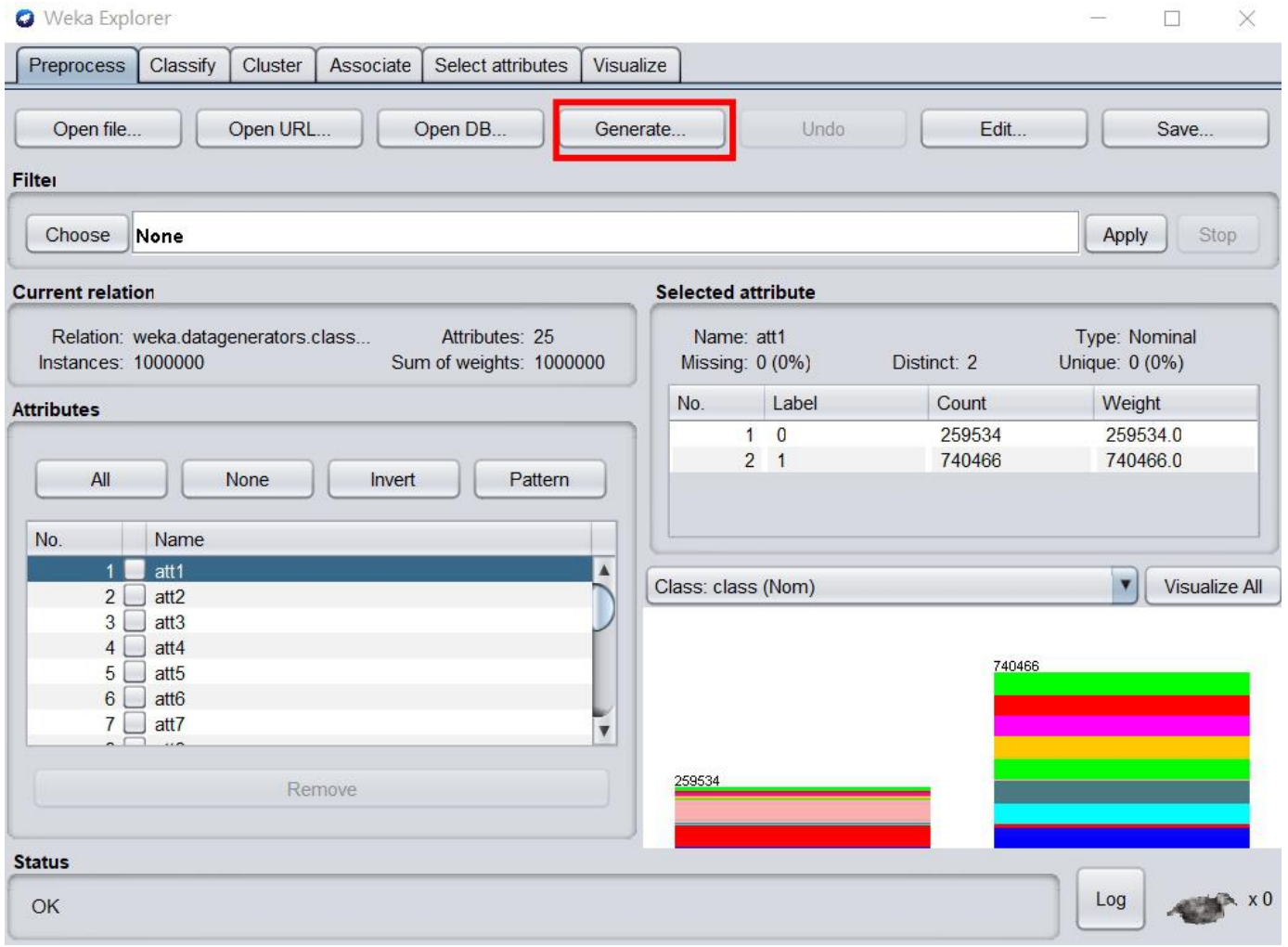




Lesson 1.6: 處理大數據

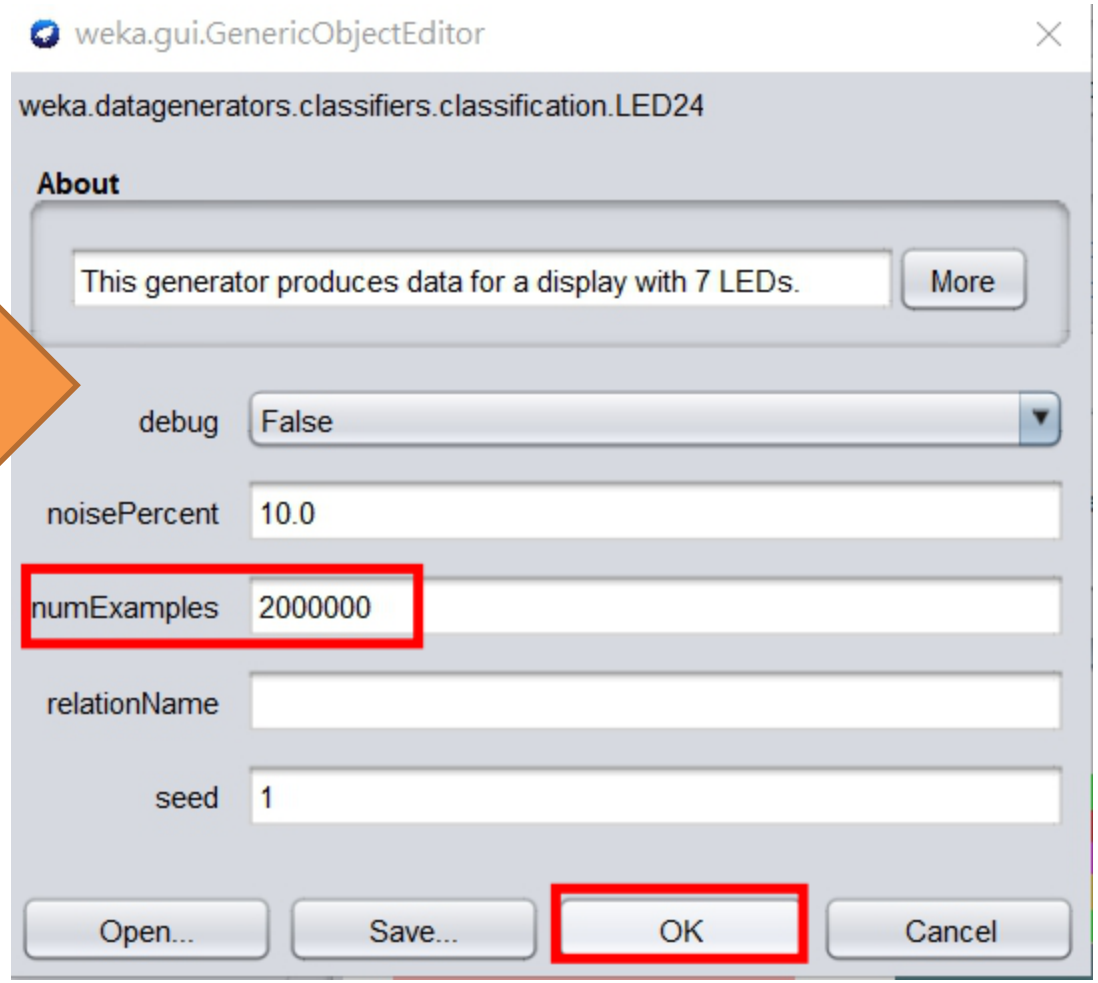
接著測試2,000,000實例數。

16.回到Preprocess介面，左鍵單擊Generate按鈕



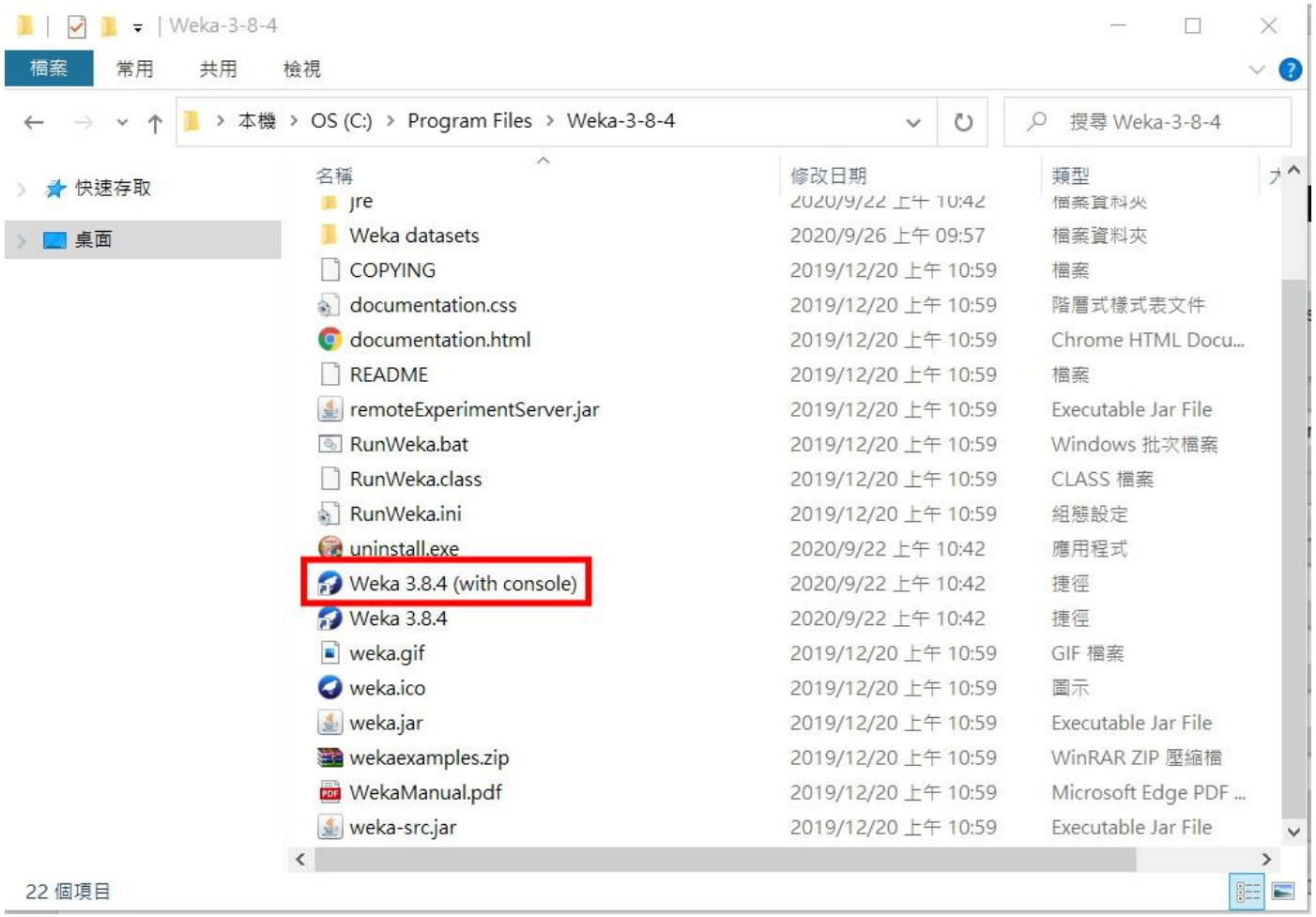
Lesson 1.6: 處理大數據

17.在DataGenerator視窗中，左鍵單擊圖中紅色方框內的LED24，在出現的配置視窗中將numExamples參數改為2,000,000，並按下視窗下方的OK按鈕。



Lesson 1.6: 處理大數據

但我們並不實際操作這2百萬個實例，它很可能使Explorer崩潰。
操作這種巨大實例數的資料時我們最好選用Weka控制台版本。



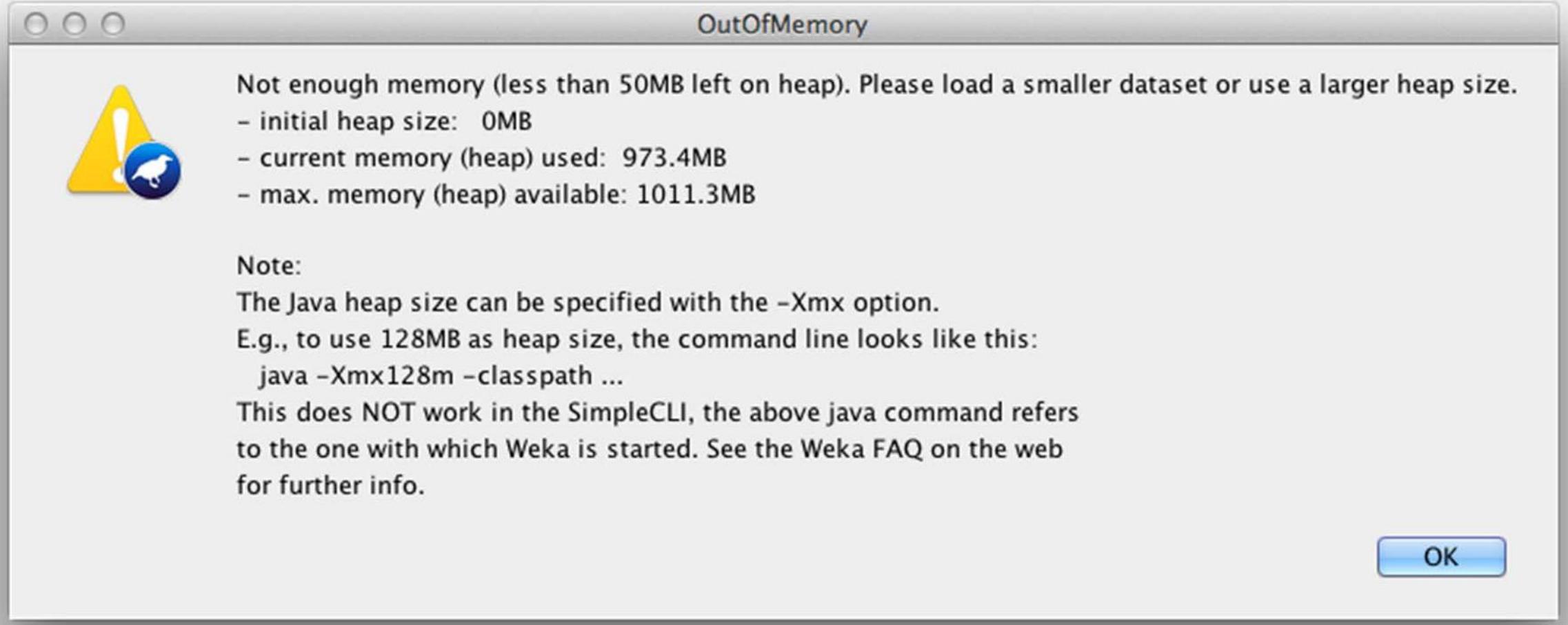
可以在Weka的資料夾中找到！

Lesson 1.6: 處理大數據

運行控制台版本後，我們可以看到它自動帶有一個控制台窗口，當出現程序崩潰、內存不足...等錯誤時，控制台窗口會報告錯誤。

Lesson 1.6: 處理大數據

這是J48崩潰時，你會得到的錯誤消息。



Lesson 1.6: 處理大數據

「可更新的」分類器

- 下列為遞增型分類模型：一次處理一個實例
  - *AODE, AODEsr, DMNBtext, IB1, IBk, KStar, LWL, NaiveBayesMultinomialUpdateable, NaiveBayesUpdateable, NNge, RacedIncrementalLogitBoost, SPegasos, Winnow*
- *NaiveBayesUpdateable*: 等同於*NaiveBayes，但可以更新*
- *NaiveBayesMultinomialUpdateable*: 可以在Text Mining的課程上看到
- *IB1*, *IBk* (但是用於測試可能會非常慢)
- *KStar*, *LWL* (局部權重學習法): 以實例為基礎的
- *SPegasos* (於*functions資料夾*)
  - *builds a linear classifier, SVM-style (restricted to numeric or binary class)*
- *RacedIncrementalLogitBoost*: 一種boosting的學習法

Lesson 1.6: 處理大數據

Weka 的Simple CLI可以處理多大的資料？ – 無可限量 (有條件的)

- 創建一個巨大資料集

  java weka.datagenerators.classifiers.classification.LED24 –n 100000 –o C:\Users\ihw\test.arff

  – *Test file with 100 K instances, 5 MB*

  java weka.datagenerators.classifiers.classification.LED24 –n 10000000 –o C:\Users\ihw\train.arff

  – *Training file with 10 M instances; 0.5 GB*

- 使用NaiveBayesUpdateable分類器

  java weka.classifiers.bayes.NaiveBayesUpdateable –t …train.arff –T …test.arff

  – *74%; 執行4分鐘*

  – *Note: 如果沒有特定的測試檔案, 將會做交叉驗證(但如果是遞增型分類器就會運行失敗)*

- 試著使用100 M 的實例 (5 GB的測試檔) – 沒有問題(但是要執行40分鐘)

Lesson 1.6: 處理大數據

- *Explorer* 可以支撐1M個實例, 25種屬性(50 MB的檔案)
- *Simple CLI 在遞增型分類器上的支撐無上限*
- 一些分類器的運行是可更新的
  - *可以在Javadoc中找到*
- 可更新的分類器處理任意大小的檔案(可以到好幾GB)
  - *但是不要嘗試使用交叉驗證*
- 處理大數據會是很艱難且令人灰心的任務

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

使用Weka進行更深入的資料探勘

Department of Computer Science
University of Waikato
New Zealand

weka.waikato.ac.nz