Đại học Quốc gia Thành phố Hồ Chí Minh

Đại học Khoa học Tự Nhiên

Môn Nhập Môn Trí Tuệ Nhân Tạo

BÁO CÁO LAB3 MÁY HỌC
TÌM HIỂU VỀ WEKA

Thông tin nhóm:

1612272 Trần Nhật Huy

1612282 Trần Đình Khải

BÁO CÁO TÌM HIỂU WEKA

1. Thành viên nhóm

1	**1612272**	**Trần Nhật Huy**
2	**1612282**	**Trần Đình Khải**

2. Giới thiệu:

Weka (viết tắt của Waikato Environment for Knowledge Analysis) là một bộ phần mềm học máy được Đại học Waikato, New Zealand phát triển bằng Java. Weka là phần mềm tự do phát hành theo Giấy phép Công cộng GNU.

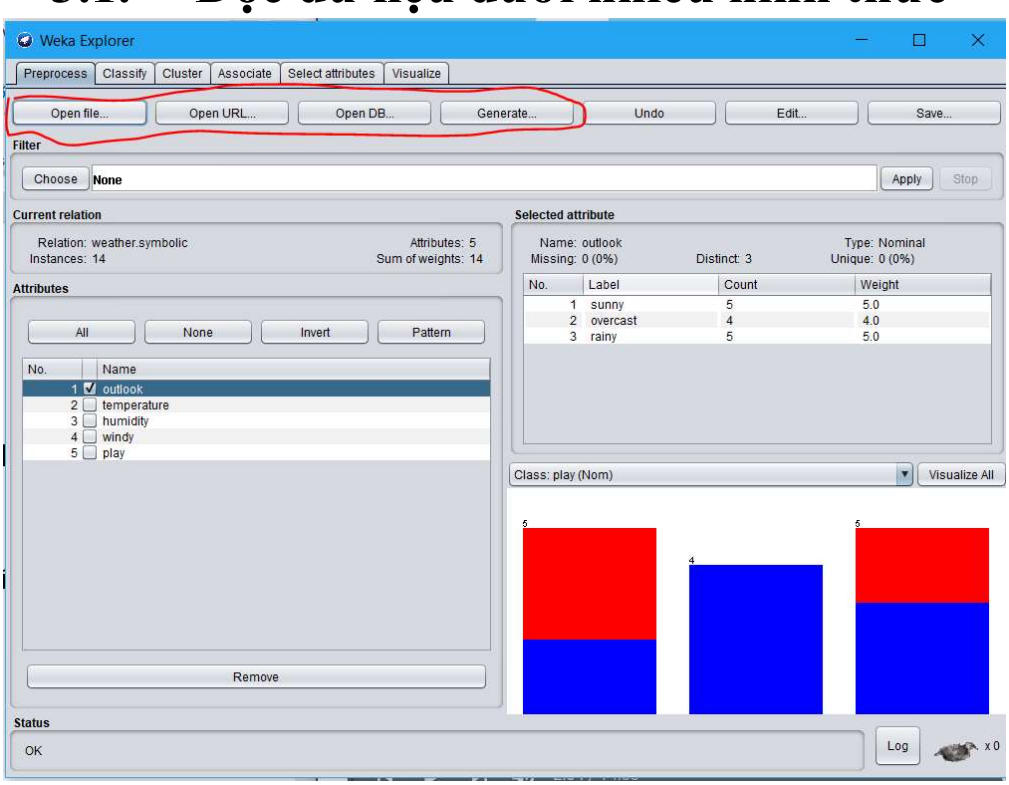
<nguồn: https://vi.wikipedia.org/wiki/Weka_(h%E1%BB%8Dc_m%C3%A1y)>

Các chức năng chính của WEKA:

- Preprocessing: Tiền xử lý.
- Associate: Khai thác luật kết hợp.
- Classtify: Phân lớp
- Cluster: Gom nhóm

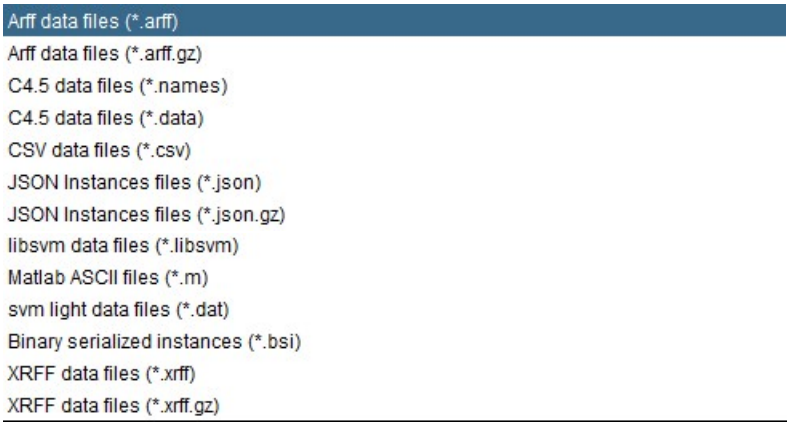
3. Preprocessing

3.1. Đọc dữ liệu dưới nhiều hình thức



Đây là 4 nút để ta đọc dữ liệu vào Weka:

OpenFile: Đọc dữ liệu bằng một file. Các dạng file có thể đọc được như sau: *.arff *.arff.gz *.names *.data *.csv *.libsvm *.bsi *.xrff



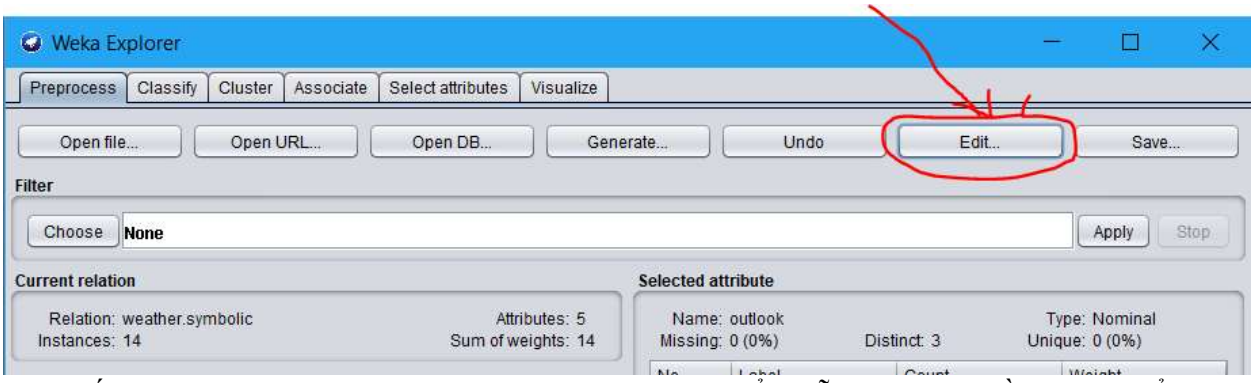
Open URL: Đọc dữ liệu từ một nơi lưu trữ bằng địa chỉ URL

Open DB: đọc dữ liệu từ các cơ sở dữ liệu MSSQL, MySQL,…

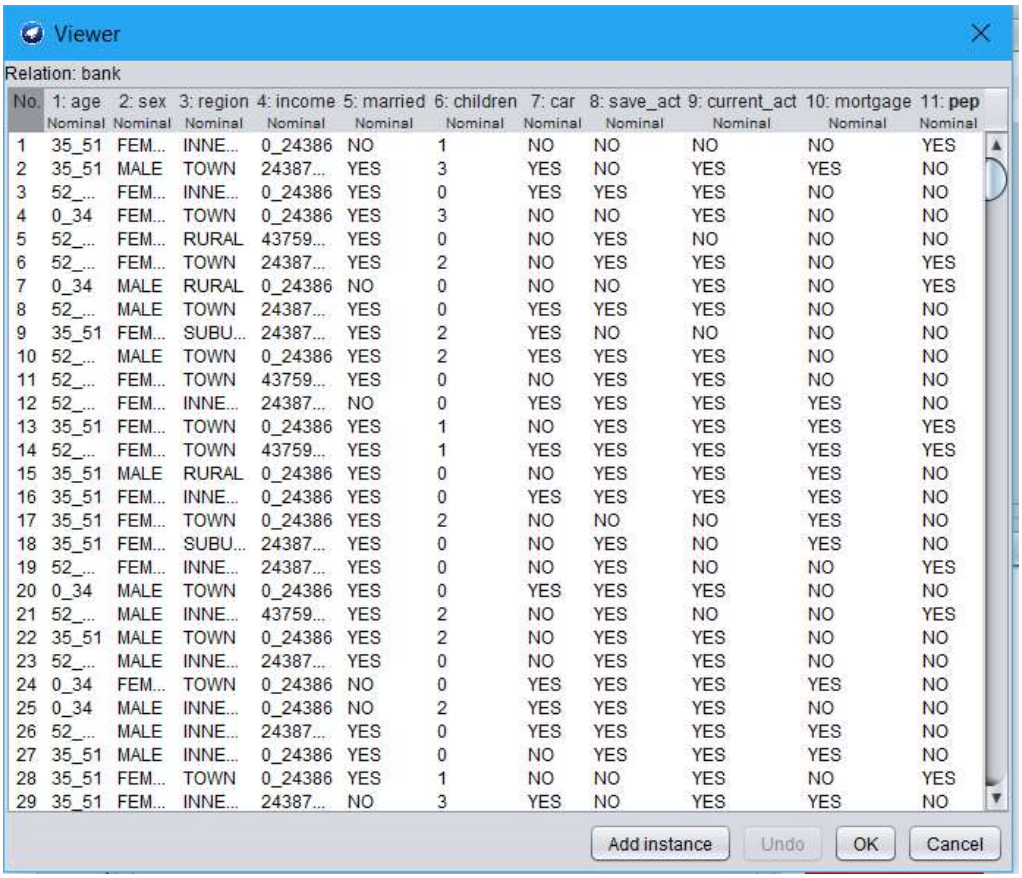
Generate: Phát sinh dữ liệu mới từ bộ phát sinh dữ liệu DataGenerators(do Weka cài đặt)…

3.2. Hiệu chỉnh dữ liệu

Đôi khi bộ dữ liệu của chúng ta không thật sự chuẩn, bị thiếu dữ liệu hay có các giá trị bất thường,… Ta sử dụng Chức năng Edit để hiệu chỉnh dữ liệu để được bộ dữ liệu như mong muốn.



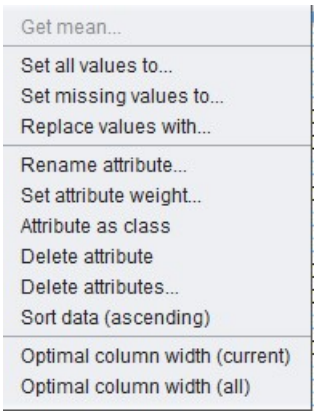
Khi nhấn vào nút Edit, bộ dữ liệu của chúng ta sẽ được biểu diễn trực quan bằng bảng biểu (cửa sổ viewer) với hàng đầu tiền là danh sách các thuộc tính.



No.	1: age	2: sex	3: region	4: income	5: married	6: children	7: car	8: save_act	9: current_act	10: mortgage	11: pep
	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal
1	35_51	FEM...	INNE...	0_24386	NO	1	NO	NO	NO	NO	YES
2	35_51	MALE	TOWN	24387...	YES	3	YES	NO	YES	YES	NO
3	52_...	FEM...	INNE...	0_24386	YES	0	YES	YES	YES	NO	NO
4	0_34	FEM...	TOWN	0_24386	YES	3	NO	NO	YES	NO	NO
5	52_...	FEM...	RURAL	43759...	YES	0	NO	YES	NO	NO	NO
6	52_...	FEM...	TOWN	24387...	YES	2	NO	YES	YES	NO	YES
7	0_34	MALE	RURAL	0_24386	NO	0	NO	NO	YES	NO	YES
8	52_...	MALE	TOWN	24387...	YES	0	YES	YES	YES	NO	NO
9	35_51	FEM...	SUBU...	24387...	YES	2	YES	NO	NO	NO	NO
10	52_...	MALE	TOWN	0_24386	YES	2	YES	YES	YES	NO	NO
11	52_...	FEM...	TOWN	43759...	YES	0	NO	YES	YES	NO	NO
12	52_...	FEM...	INNE...	24387...	NO	0	YES	YES	YES	YES	NO
13	35_51	FEM...	TOWN	0_24386	YES	1	NO	YES	YES	YES	YES
14	52_...	FEM...	TOWN	43759...	YES	1	YES	YES	YES	YES	YES
15	35_51	MALE	RURAL	0_24386	YES	0	NO	YES	YES	YES	NO
16	35_51	FEM...	INNE...	0_24386	YES	0	YES	YES	YES	YES	NO
17	35_51	FEM...	TOWN	0_24386	YES	2	NO	NO	NO	YES	NO
18	35_51	FEM...	SUBU...	24387...	YES	0	NO	YES	NO	YES	NO
19	52_...	FEM...	INNE...	24387...	YES	0	NO	YES	NO	NO	YES
20	0_34	MALE	TOWN	0_24386	YES	0	YES	YES	YES	NO	NO
21	52_...	MALE	INNE...	43759...	YES	2	NO	YES	NO	NO	YES
22	35_51	MALE	TOWN	0_24386	YES	2	NO	YES	YES	NO	NO
23	52_...	MALE	INNE...	24387...	YES	0	NO	YES	YES	NO	NO
24	0_34	FEM...	TOWN	0_24386	NO	0	YES	YES	YES	YES	NO
25	0_34	MALE	INNE...	0_24386	NO	2	YES	YES	YES	NO	NO
26	52_...	MALE	INNE...	24387...	YES	0	YES	YES	YES	YES	NO
27	35_51	MALE	INNE...	0_24386	YES	0	NO	YES	YES	YES	NO
28	35_51	FEM...	TOWN	0_24386	YES	1	NO	NO	YES	NO	YES
29	35_51	FEM...	INNE...	24387...	NO	3	YES	NO	YES	YES	NO

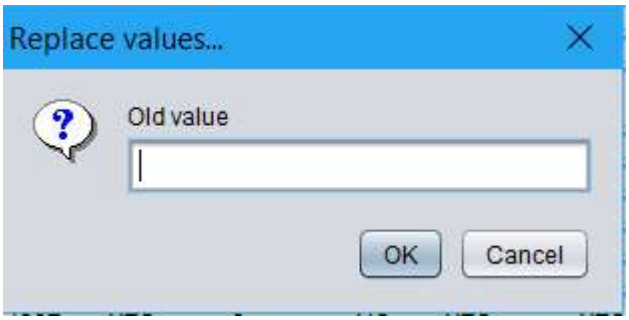
Nháy chuột phải vào tên một thuộc tính, một dropdown menu để chỉnh sửa cho cột dữ liệu thuộc tính được chọn.

Set all value to: Đặt giá trị cho tất cả các dòng tại thuộc tính được chọn. Giá trị này do người dùng thiết lập.



Set missing values to: Ở các mẫu, tại thuộc tính được chọn, ta đặt giá trị cho nó nếu nó rỗng.

Replace values with: Thay thế giá trị cũ thành một giá trị mới.



Get mean: Lấy giá trị trung bình của thuộc tính

Rename Attribute, Delete Attribute: Đổi tên và Xoá thuộc tính.

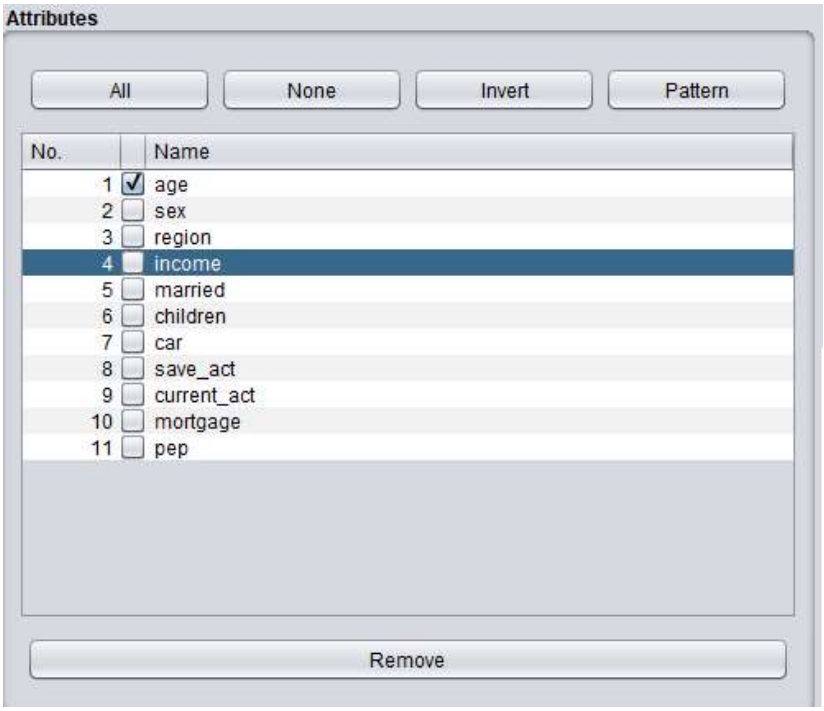
Sort data: Sắp xếp tăng dần theo thuộc tính được chọn.

Lọc dữ liệu từ những bộ lọc có sẵn: (rời rạc hoá dữ liệu,….)

3.3. Biểu diễn thông tin về tập dữ liệu

Ở phần Current Relation sẽ hiện thị cho ta vài thông tin về bộ dữ liệu: tên quan hệ: “bank”, Số thuộc tính là 11, số mẫu là 600.

Ở phần Attributes sẽ hiện thị các thuộc tính theo đúng thứ tự được khai báo trong file input (ở đây là *.arff).



Các button:

All: đánh tick cho tất cả các thuộc tính.

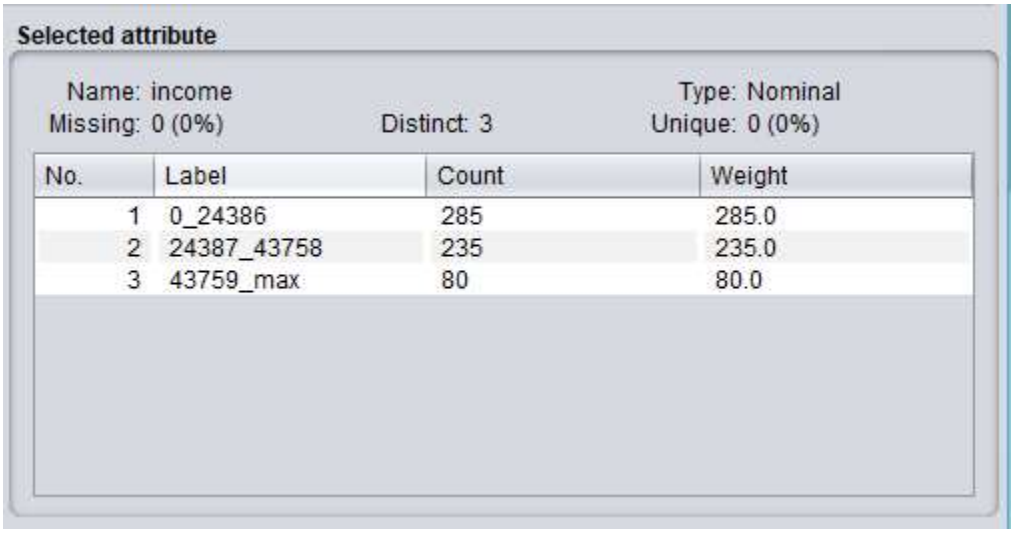
None: bỏ chọn tick tất cả các thuộc tính.

Invert: Đảo chọn tick.

Remove: xoá thuộc tính ra khỏi quan hệ.

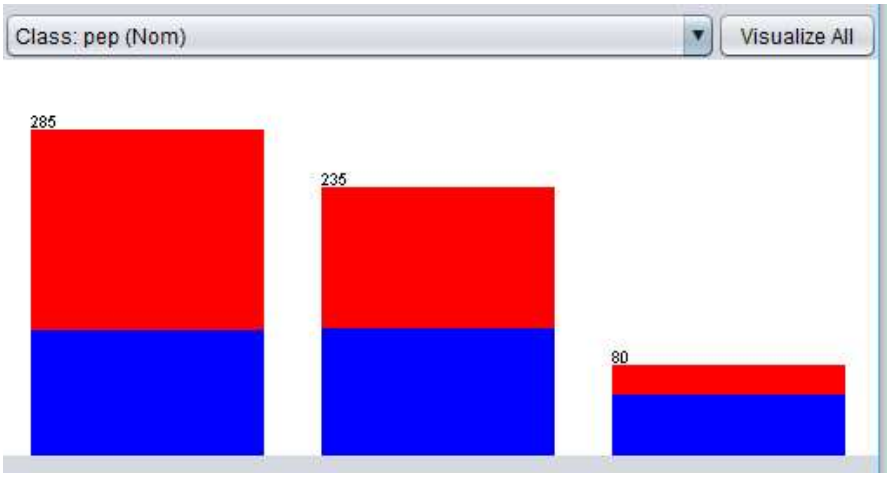
Ở mỗi thuộc tính được chọn<bôi đen>, khung bên

(Seleted attribute) phải sẽ hiện thị các giá trị mà thuộc tính đang nhận và số lượng mẫu cho từng giá trị.



Ví dụ: tên thuộc tính là income, thuộc tính nhận 3 giá trị là 0_24386 (có 285 mẫu), 24387_43758(có 235 mẫu) và 43759_max(có 80 mẫu). Và không có mẫu nào bị thiếu giá trị. Kiểu là Nominal.

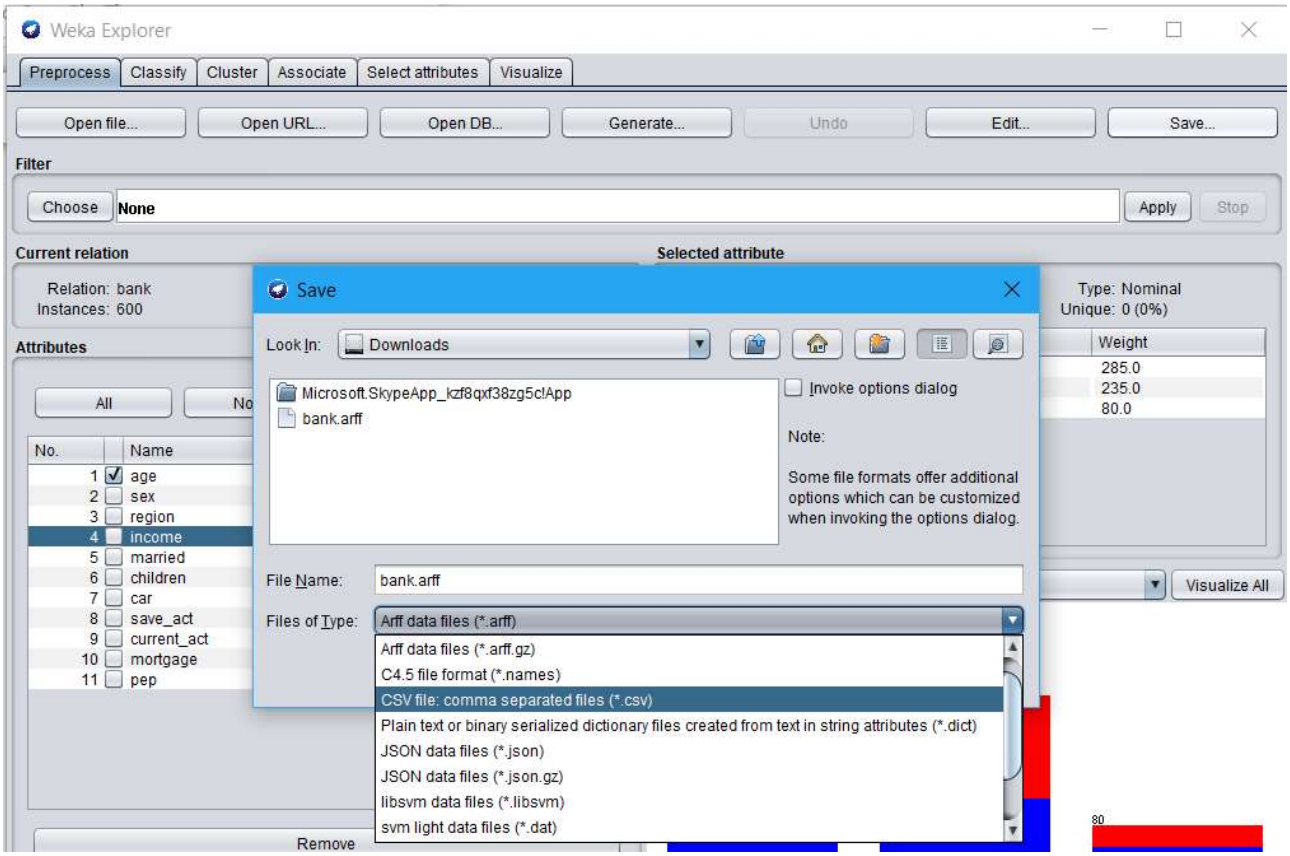
Ở phần visualize, thể hiện trực quan về thuộc tính được chọn.



Class: là biểu diễn theo class đó. Ví dụ, thuộc tính đang chọn là income và ta biểu diễn theo thuộc tính pep. Vì pep có hai giá trị nên hai màu đỏ xanh biểu diễn hai giá trị đó. Ta thấy rằng ở cột đầu tiên, với 285 mẫu nhận giá trị 0-24386 thì tỉ lệ số mẫu mang giá trị của thuộc tính pep là không đều, khoảng (60:40).

3.4. Lưu trữ dữ liệu

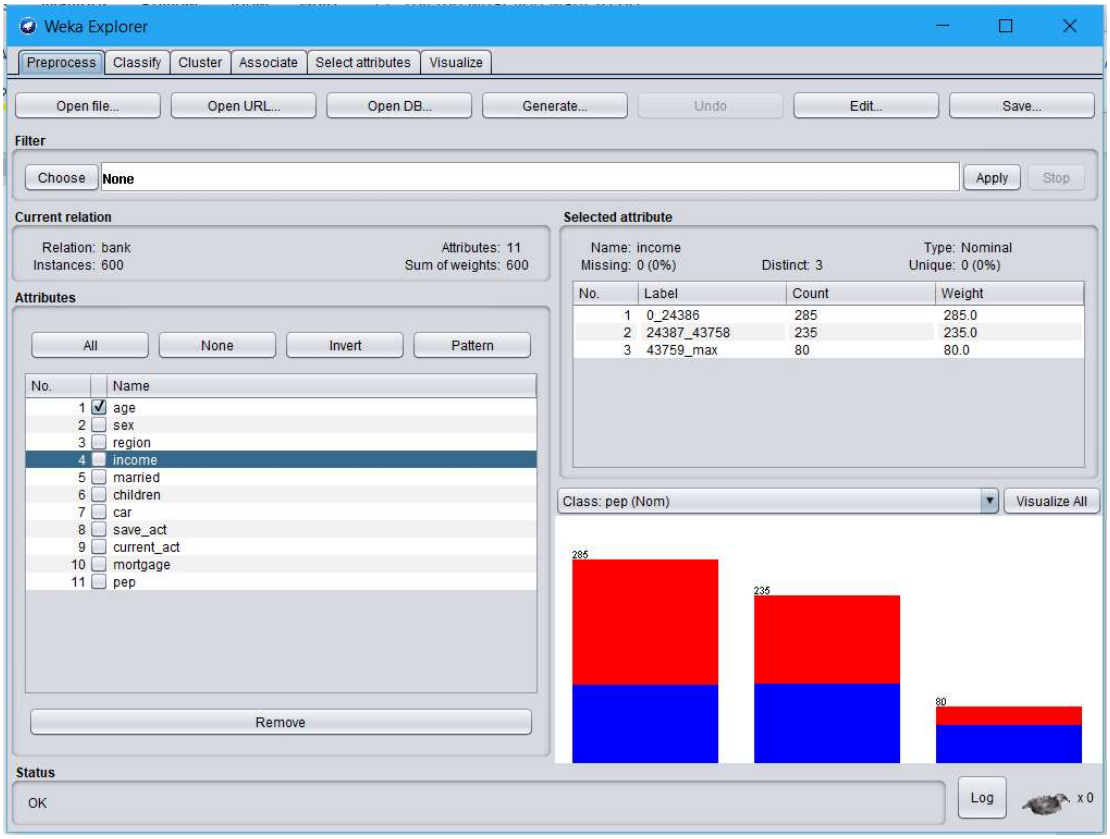
Sau khi hiệu chỉnh dữ liệu, ta có thể lưu lại <nhập nút save ở trên cùng bên phải> dữ liệu dưới dạng các định dạng file cho phép như trong hình…


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Stop
Current relation
Relation: bank
Instances: 600
Selected attribute
Type: Nominal
Unique: 0 (0%)
Attributes
All
No.
Name
1 age
2 sex
3 region
4 income
5 married
6 children
7 car
8 save_act
9 current_act
10 mortgage
11 pep
Weight
285.0
235.0
80.0
Visualize All
Remove
Save
Look In:
Downloads
Microsoft.SkypeApp_kzf8qxf38zg5c!App
bank.arff
Invoke options dialog
Note:
Some file formats offer additional options which can be customized when invoking the options dialog.
File Name:
bank.arff
Files of Type:
Arff data files (*.arff)
Arff data files (*.arff.gz)
C4.5 file format (*.names)
CSV file: comma separated files (*.csv)
Plain text or binary serialized dictionary files created from text in string attributes (*.dict)
JSON data files (*.json)
JSON data files (*.json.gz)
libsvm data files (*.libsvm)
svm light data files (*.dat)
80

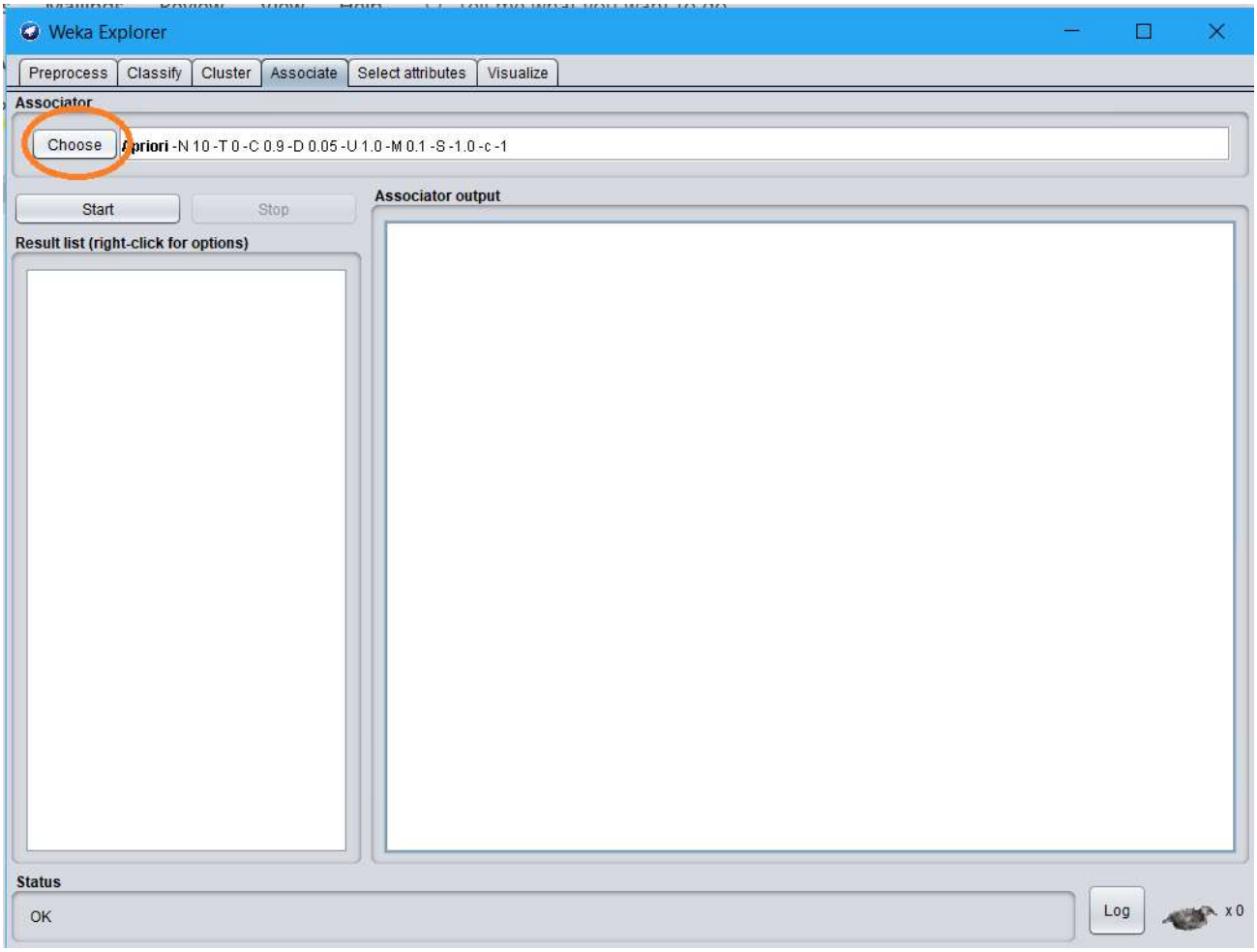
4. Associate: Khai thác luật kết hợp.

4.1. Bước 1: Chọn tập dữ liệu

Chọn dữ liệu và hiểu chỉnh dữ liệu ở tab preprocessing như ở phần 1.

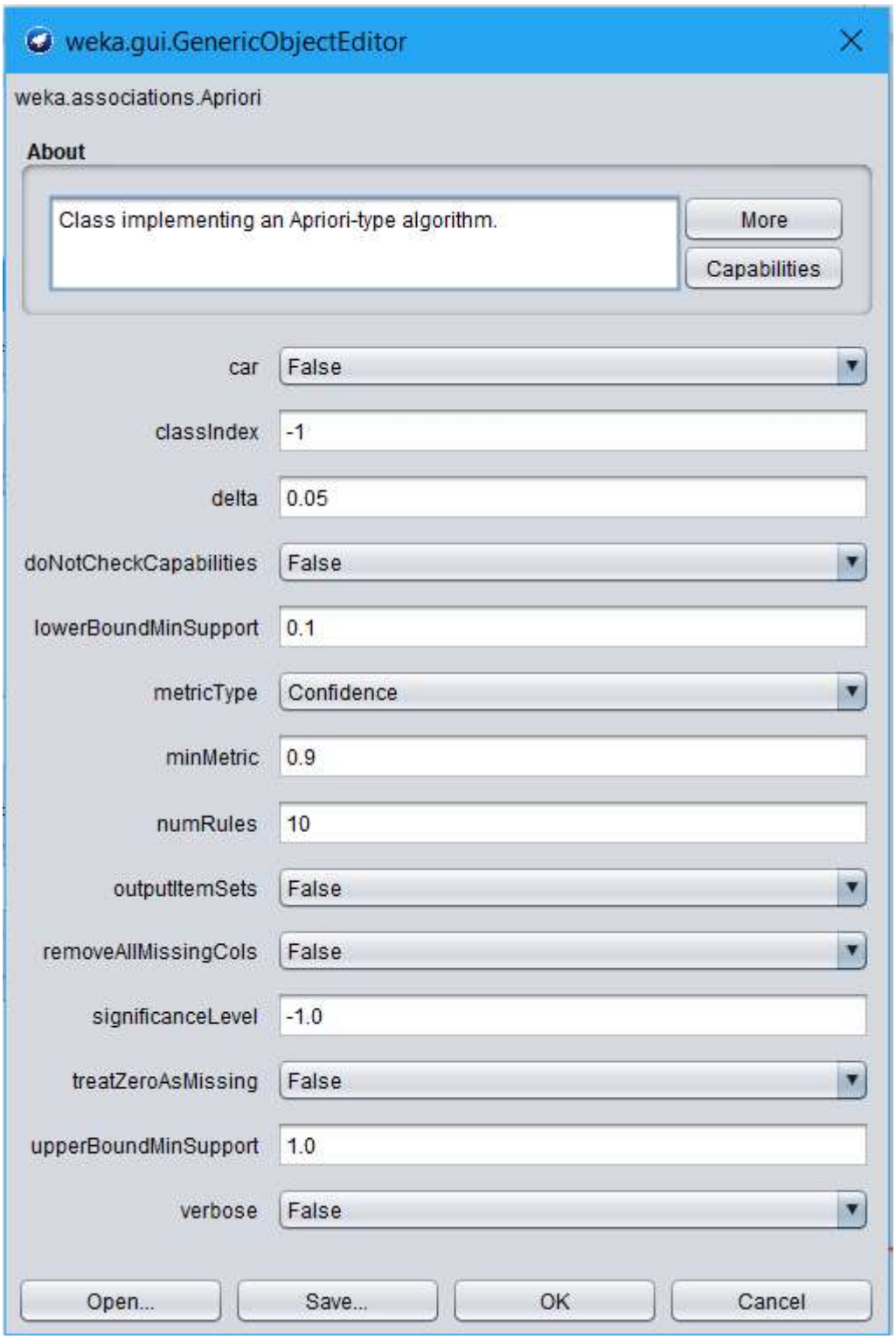


4.2. Bước2: Chọn thuật toán khai thác luật kết hợp



ở phần Associator, Ta nhấn nút Choose để chọn thuật toán khai thác. Ta thường dùng thuật toán Apriori

Nhấn vào ô textbox kế bên để chỉnh sửa các thuật tính của thuật toán.



Các tham số chính của thuật toán Apriori:

lowerBoundMinSupport(-M): chặn dưới minSupport

upperBoundMinSupport(-U): chặn trên minSupport

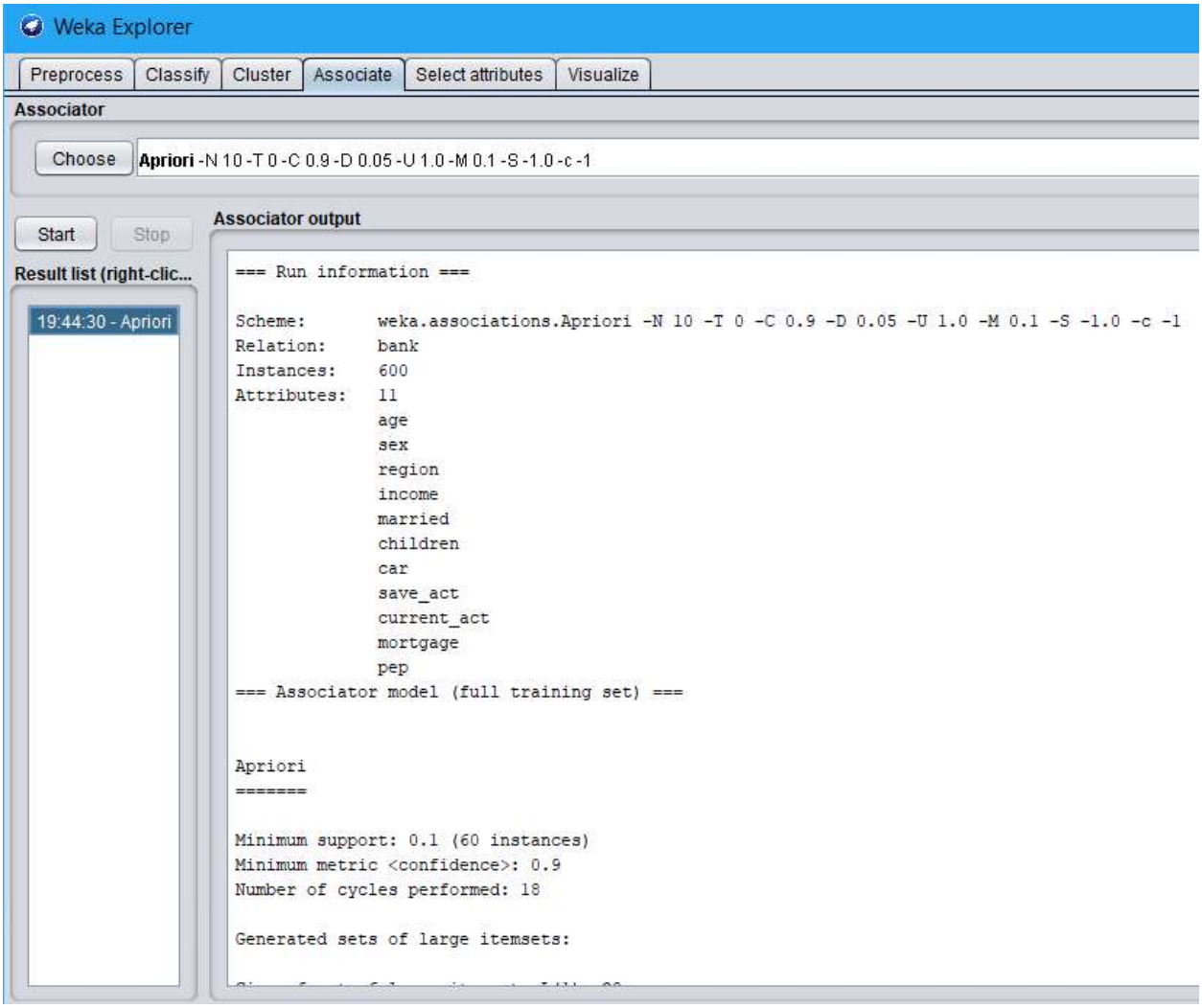
delta(-D): hệ số giảm support khi lặp, giảm support cho đến khi đạt minsupport hay đã phát sinh đủ luật.

metricType(-T): độ đo tính quan trọng /lý thú của luật bao gồm : Confidence, Lift, Leverage, Conviction.

minMetric(-C): độ đo tin cậy nhỏ nhất. Chỉ xét những luật có điểm lớn hơn giá trị này.

NumRules(-N) : số luật cần tìm.

4.3. Bước 3: Tiến hàng khai thác



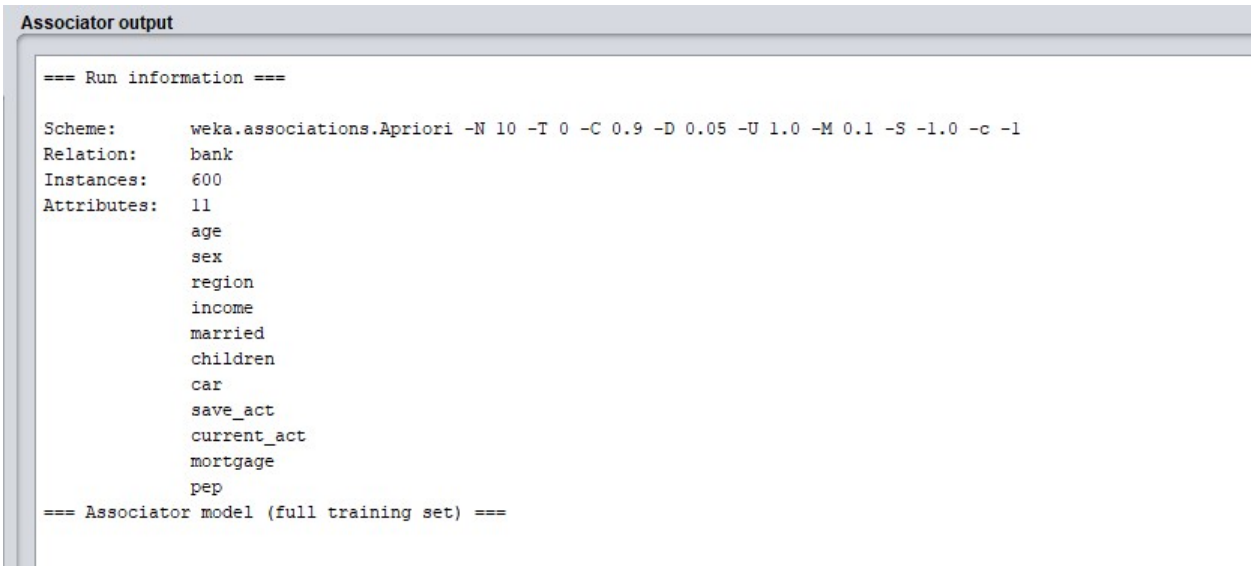
Sau khi thiết lập các tham số của thuật toán , t nhấn nút Start để bắt đầu.

Ở Result list, thể hiện danh sách các kết quả đã thực hiện

Ở Associator output, thể hiện kết quả sau khi chạy thuật toán.

4.4. Bước 4: Đọc kết quả

Ở phần đầu của output, thể hiện thuật toán cùng với thông tin của bộ dữ liệu được dùng (như ở phần preprocessing)



```
Associator output

=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     bank
Instances:    600
Attributes:   11
              age
              sex
              region
              income
              married
              children
              car
              save_act
              current_act
              mortgage
              pep
=== Associator model (full training set) ===
```

Phần tiếp theo của output là một số thông tin kết quả như sau: giá trị minsupport, giá trị min confidence, số vòng lặp và kích thước của các tập phổ biến

```
Apriori
=======

Minimum support: 0.1 (60 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 28

Size of set of large itemsets L(2): 232

Size of set of large itemsets L(3): 524

Size of set of large itemsets L(4): 277

Size of set of large itemsets L(5): 33
```

Phần quan trọng nhất của output là các luật tìm được

```
Best rules found:

 1. income=43759_max 80 ==> save_act=YES 80    <conf:(1)> lift:(1.45) lev:(0.04) [24] conv:(24.8)
 2. age=52_max income=43759_max 76 ==> save_act=YES 76    <conf:(1)> lift:(1.45) lev:(0.04) [23] conv:(23.56)
 3. income=43759_max current_act=YES 63 ==> save_act=YES 63    <conf:(1)> lift:(1.45) lev:(0.03) [19] conv:(19.53)
 4. age=52_max income=43759_max current_act=YES 61 ==> save_act=YES 61    <conf:(1)> lift:(1.45) lev:(0.03) [18] conv:(18.91)
 5. children=0 save_act=YES mortgage=NO pep=NO 74 ==> married=YES 73    <conf:(0.99)> lift:(1.49) lev:(0.04) [24] conv:(12.58)
 6. sex=FEMALE children=0 mortgage=NO pep=NO 64 ==> married=YES 63    <conf:(0.98)> lift:(1.49) lev:(0.03) [20] conv:(10.88)
 7. children=0 current_act=YES mortgage=NO pep=NO 82 ==> married=YES 80    <conf:(0.98)> lift:(1.48) lev:(0.04) [25] conv:(9.29)
 8. children=0 mortgage=NO pep=NO 107 ==> married=YES 104    <conf:(0.97)> lift:(1.47) lev:(0.06) [33] conv:(9.1)
 9. income=43759_max current_act=YES 63 ==> age=52_max 61    <conf:(0.97)> lift:(3.04) lev:(0.07) [40] conv:(14.31)
10. income=43759_max save_act=YES current_act=YES 63 ==> age=52_max 61    <conf:(0.97)> lift:(3.04) lev:(0.07) [40] conv:(14.31)
```

Đọc kết quả:

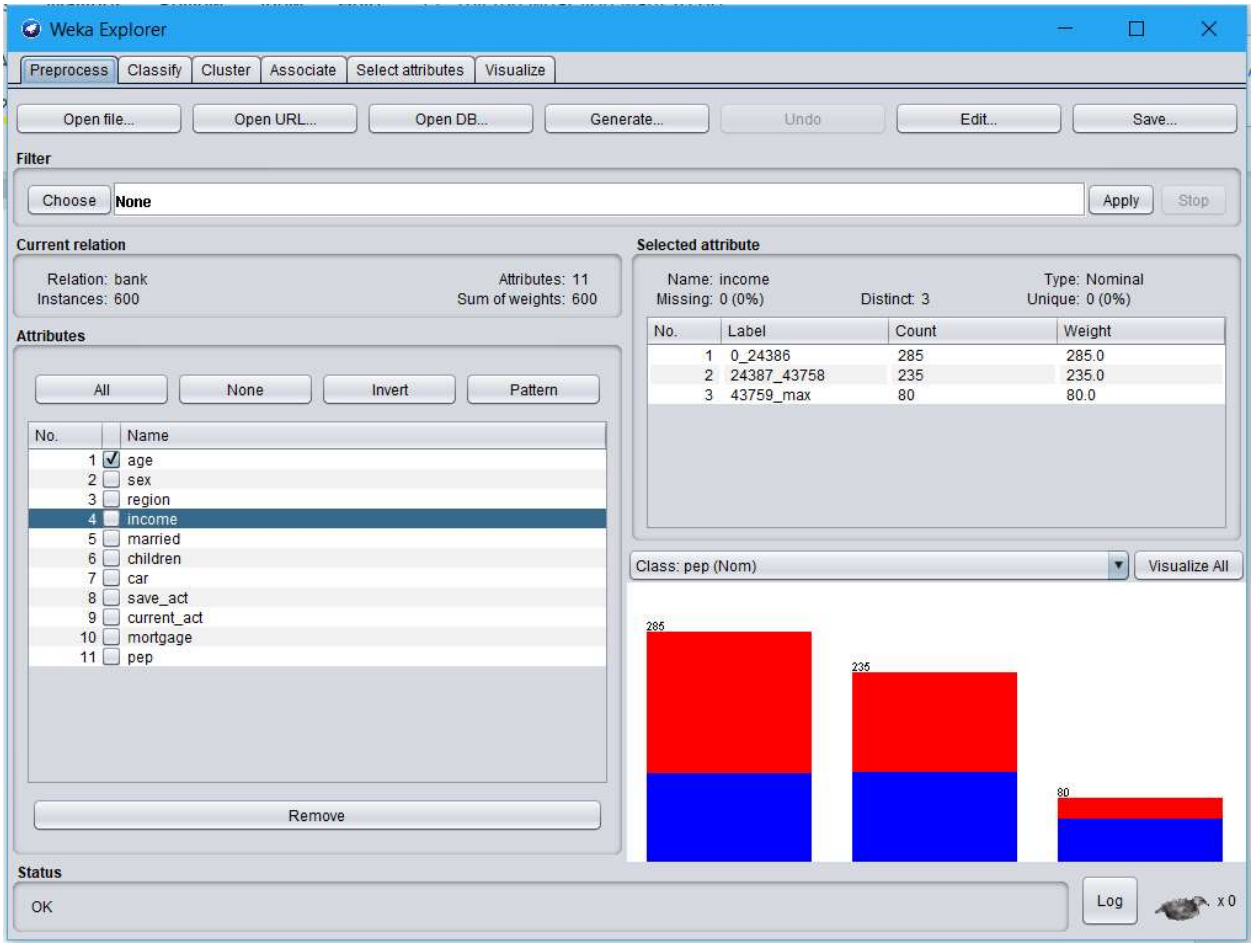
Luật 1: Nếu income là 43759_max (ở 80 mẫu) thì save_act là Yes (ở 80 mẫu) với độ tin cậy là 1.

Luật 2: Nếu age=52_max và income-43759_max (ở 76 mẫu) thì save_act là Yes với độ tin cậy là 1

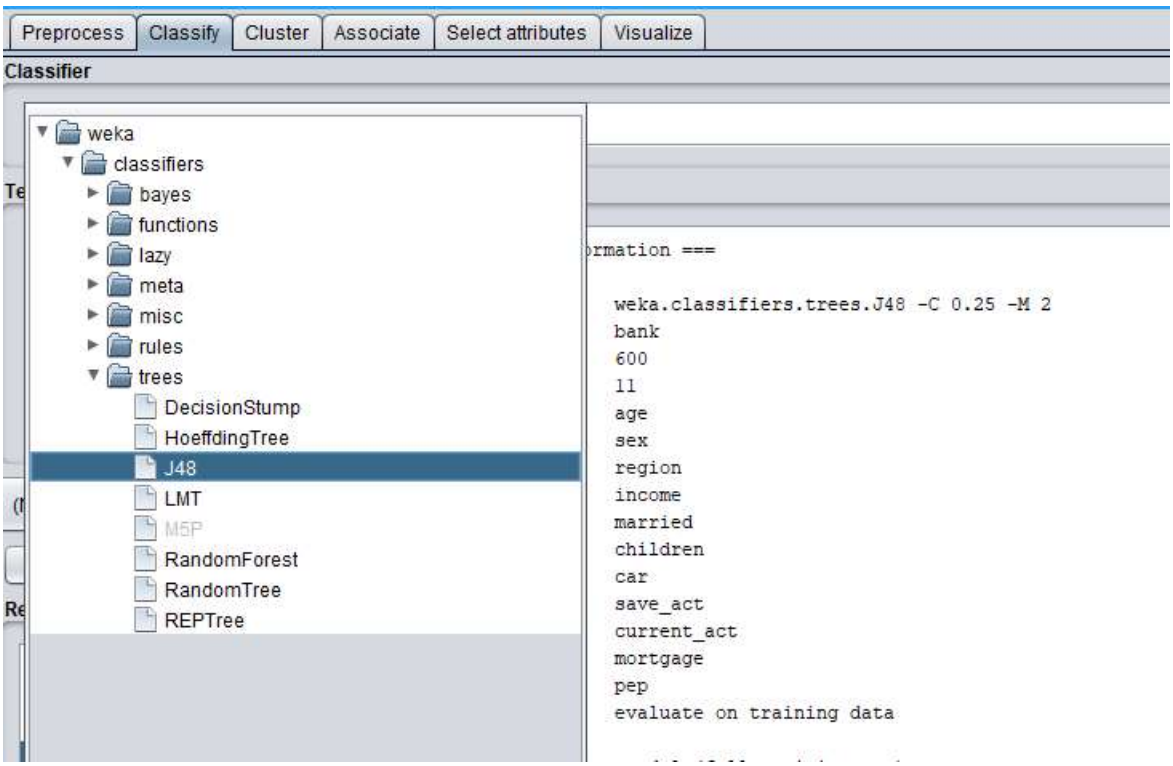
5. Classtify

5.1. Bước 1 Chọn tập dữ liệu và tiền xử lý

Chọn dữ liệu và hiểu chỉnh dữ liệu và tiền xử lý ở tab preprocessing như mục 1.

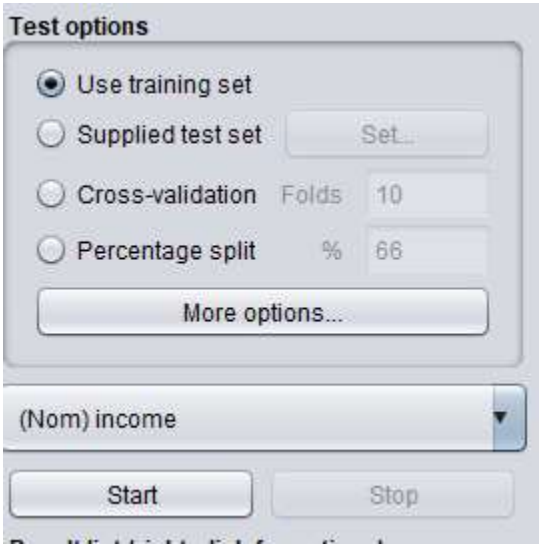


5.2. Bước 2 Chọn thuật toán và xác định các tham số



Ta nhấn nút choose để chọn thuật toán mà bạn muốn. Nhấp ở textbox ở kế bên điều chỉnh các tham số, nếu có. Có nhiều thuật toán là không có tham số. Ở đây mình chọn thuật toán J48 xây dựng cây.

5.3. Bước 3 Chọn kiểu test và tập dữ liệu test nếu cần



Ta chọn các loại test. Ta có các loại test như sau:

Use training set: Sử dụng chính bộ dữ liệu training để test.

Supplied test set: Cung cấp một bộ test set khác.

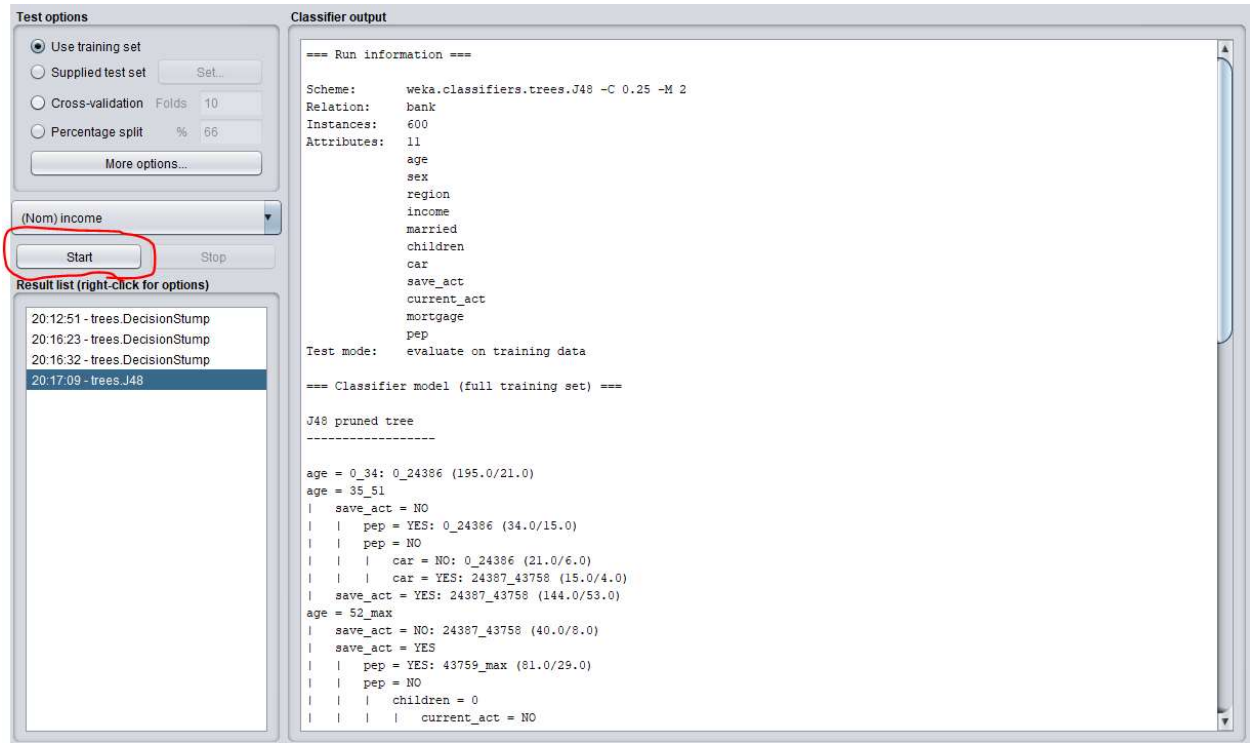
Cross-validation: ~

Percentage split: Chia bộ dữ liệu theo phần trăm. Bao nhiêu nhiêu phần trăm dùng để train , còn lại dùng để test.

ComboBox ở dưới cho ta chọn thuộc tính để ta xây dựng cây. Chẳng hạn ta sẽ xây dựng cây thể hiện income(các lá là các giá trị income).

5.4. Bước 4 Tiến hành phân lớp dữ liệu

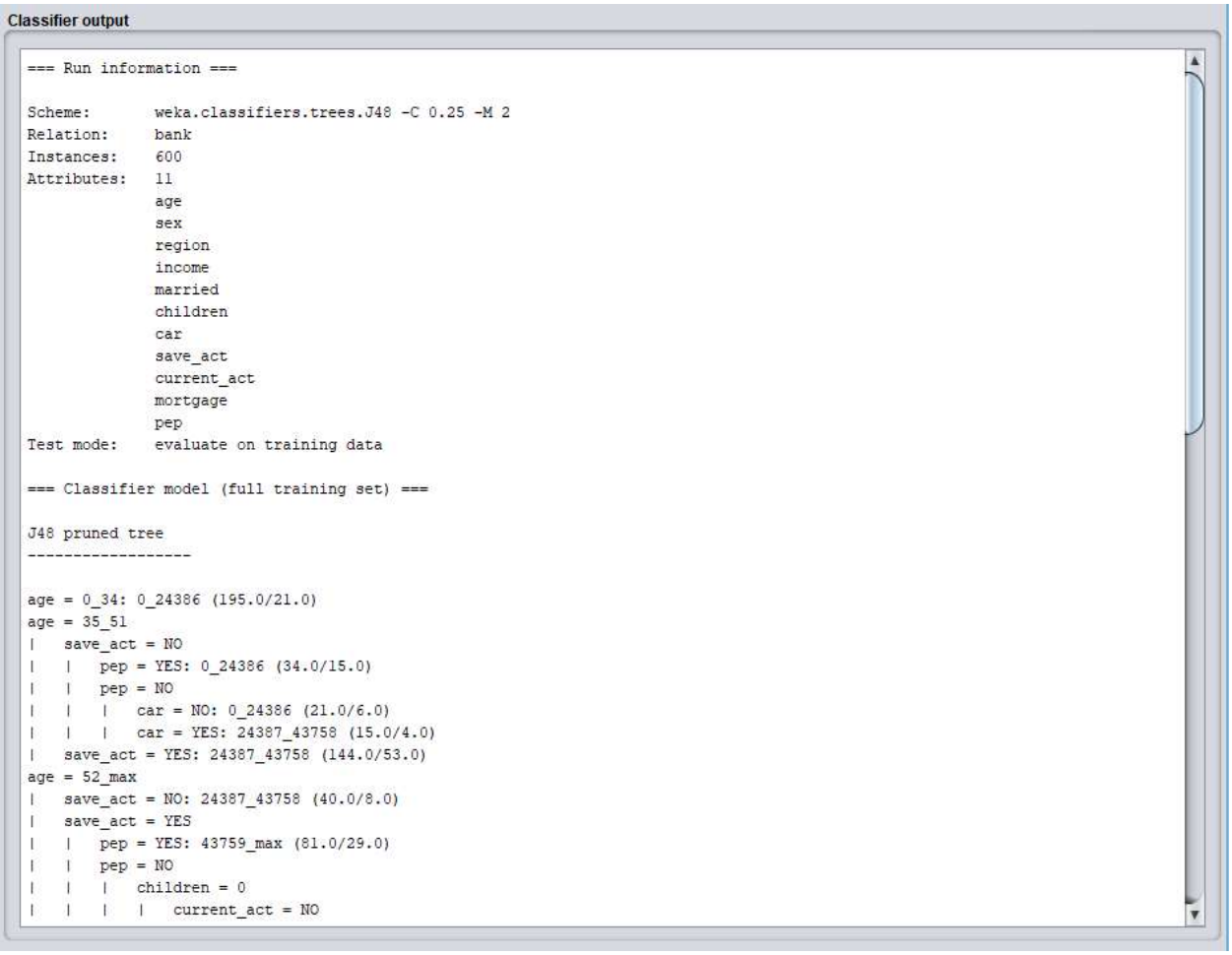
Sau khi thiết lập xong, ta nhấn Start để bắt đầu chạy thuật toán.



Ở phần Result list, thể hiện các kết quả đã thực hiện, sử dụng để so sánh các kết quả thực hiện.

Phần output của thuật toán được hiển thị ở khung lớn bên phải (Classifier Output)

5.5. Bước 5 Đọc kết quả



Phần đầu của output là thông tin chung của thuật toán cũng như bộ dữ liệu train.

Testmode: chế độ test, ở đây ta dùng bộ train làm bộ test luôn.

Classifier model : luôn là dùng toàn bộ bộ training.

Cây tìm được:

```
J48 pruned tree
------------------

age = 0_34: 0_24386 (195.0/21.0)
age = 35_51
|   save_act = NO
|   |   pep = YES: 0_24386 (34.0/15.0)
|   |   pep = NO
|   |   |   car = NO: 0_24386 (21.0/6.0)
|   |   |   car = YES: 24387_43758 (15.0/4.0)
|   save_act = YES: 24387_43758 (144.0/53.0)
age = 52_max
|   save_act = NO: 24387_43758 (40.0/8.0)
|   save_act = YES
|   |   pep = YES: 43759_max (81.0/29.0)
|   |   pep = NO
|   |   |   children = 0
|   |   |   |   current_act = NO
|   |   |   |   |   car = NO
|   |   |   |   |   |   mortgage = NO: 43759_max (2.0)
|   |   |   |   |   |   mortgage = YES: 24387_43758 (2.0)
|   |   |   |   |   car = YES: 24387_43758 (4.0/1.0)
|   |   |   |   current_act = YES: 43759_max (36.0/15.0)
|   |   |   children = 1: 24387_43758 (3.0)
|   |   |   children = 2
|   |   |   |   current_act = NO: 0_24386 (4.0)
|   |   |   |   current_act = YES
|   |   |   |   |   mortgage = NO: 0_24386 (5.0/1.0)
|   |   |   |   |   mortgage = YES: 24387_43758 (3.0/1.0)
|   |   |   children = 3: 24387_43758 (11.0/1.0)

Number of Leaves  :     16

Size of the tree :      28
```

Thuật toán tìm cây là: J48 pruned tree

Số lá là 16 và số nút là 28

```
Time taken to build model: 0.04 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.02 seconds
```

Thời gian để training là 0.04 giây và thời gian để test là 0.02 giây

```
=== Summary ===

Correctly Classified Instances         445               74.1667 %
Incorrectly Classified Instances       155               25.8333 %
Kappa statistic                          0.5857
Mean absolute error                      0.2399
Root mean squared error                  0.3464
Relative absolute error                 59.6377 %
Root relative squared error             77.2456 %
Total Number of Instances              600
```

Phần kết luận:

Có 445 mẫu test đúng, chiếm 74.1667% và 155 mẫu test sai chiếm 25.8333%

Còn lại là các thông số lỗi khác.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.758    0.137    0.834      0.758    0.794      0.626    0.877     0.829     0_24386
               0.655    0.186    0.694      0.655    0.674      0.474    0.802     0.686     24387_43758
               0.938    0.085    0.630      0.938    0.754      0.727    0.951     0.636     43759_max
Weighted Avg.  0.742    0.149    0.752      0.742    0.742      0.580    0.858     0.747
```

Đọ chính xác cho từng phân lớp: ~

```
=== Confusion Matrix ===

   a   b   c   <-- classified as
 216  63   6 |   a = 0_24386
  43 154  38 |   b = 24387_43758
   0   5  75 |   c = 43759_max
```

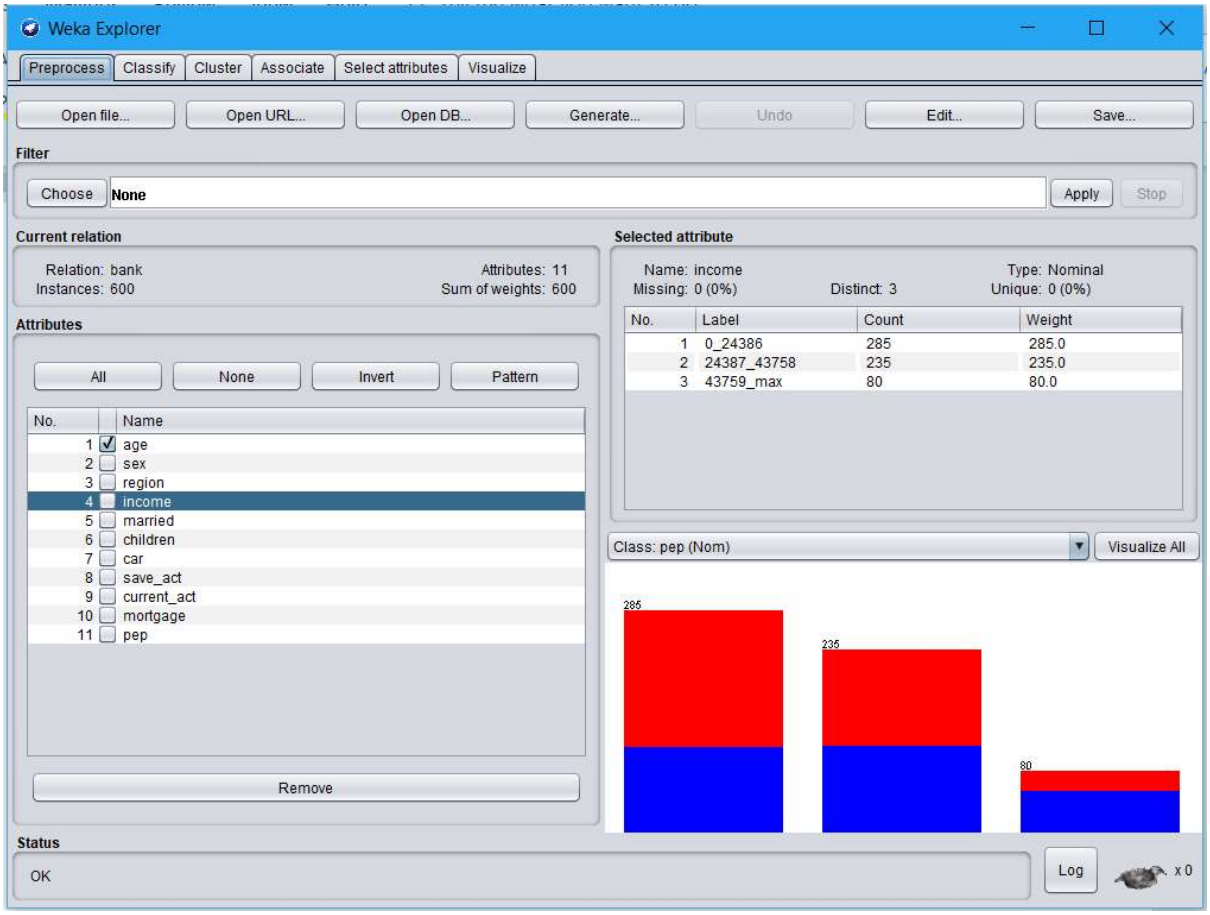
Cho biết bao nhiêu mẫu được gán vào từng phân lớp. Các phần tử của ma trận thể hiện số mẫu test có lớp thật sự là dòng và lớp dự đoán là cột.

Ví dụ ở đâu: thực sự có 285 mẫu ở lớp a, nhưng mô hình chỉ phân được 216 mẫu, 63 mẫu vào lớp b và 6 mẫu vào lớp c. Tổng trên đường chéo chính là số mẫu phân loại đúng, tổng số còn lại là số mẫu sai.

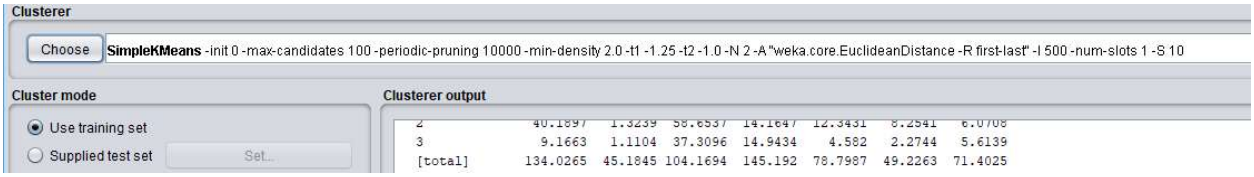
6. Cluster

6.1. Bước 1: Chọn tập dữ liệu và thực hiện tiền xử lý.

Chọn dữ liệu và hiểu chỉnh dữ liệu và tiền xử lý ở tab preprocessing như mục 1.



6.2. Bước 2: Chọn Thuật toán gom nhóm và điều chỉnh tham số.

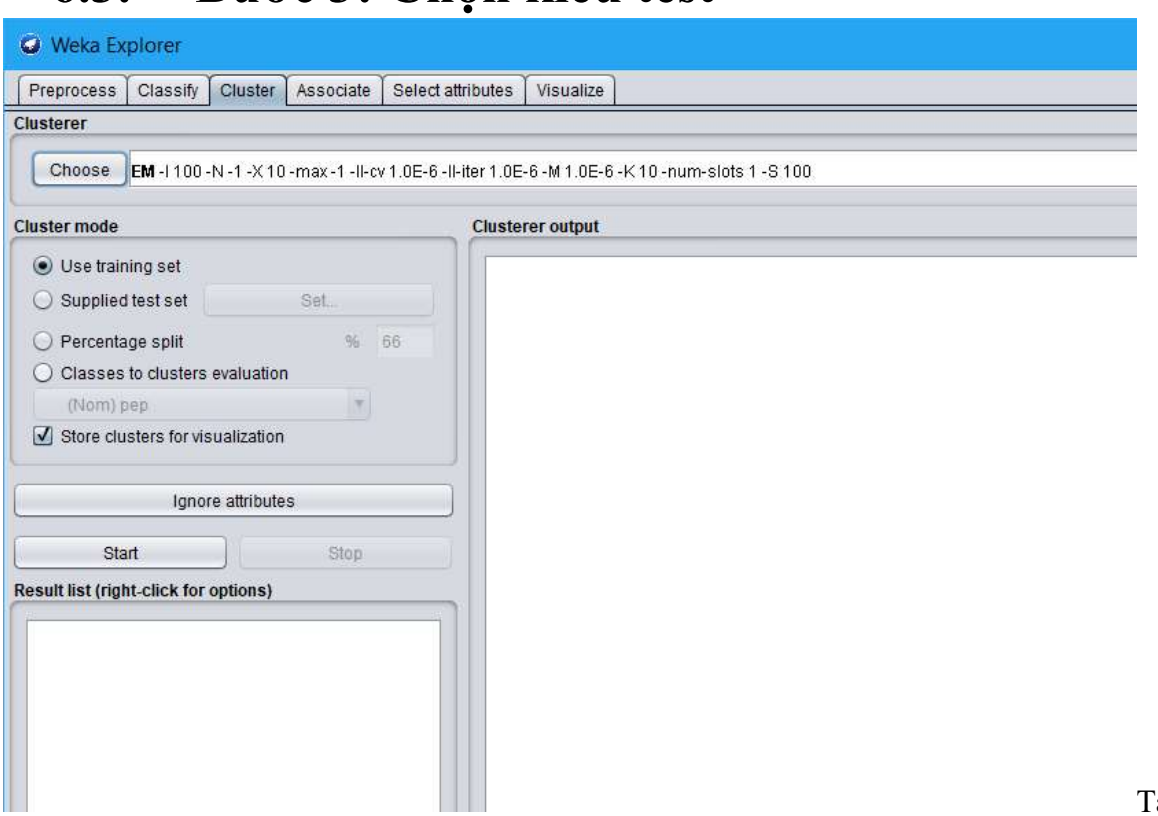


Tương tự các chức năng khác. Ta nhấn choose để chọn thuật toán. Ở đây mình chọn SimpleKmean. Ta nhấn vào textBox kế bên để điều chỉnh tham số nếu cần.

numCluster(-N): số nhóm

seed(-S): giá trị ngẫu nhiên cần gieo.

6.3. Bước 3: Chọn kiểu test



Ta chọn các loại test. Ta có các loại test như sau:

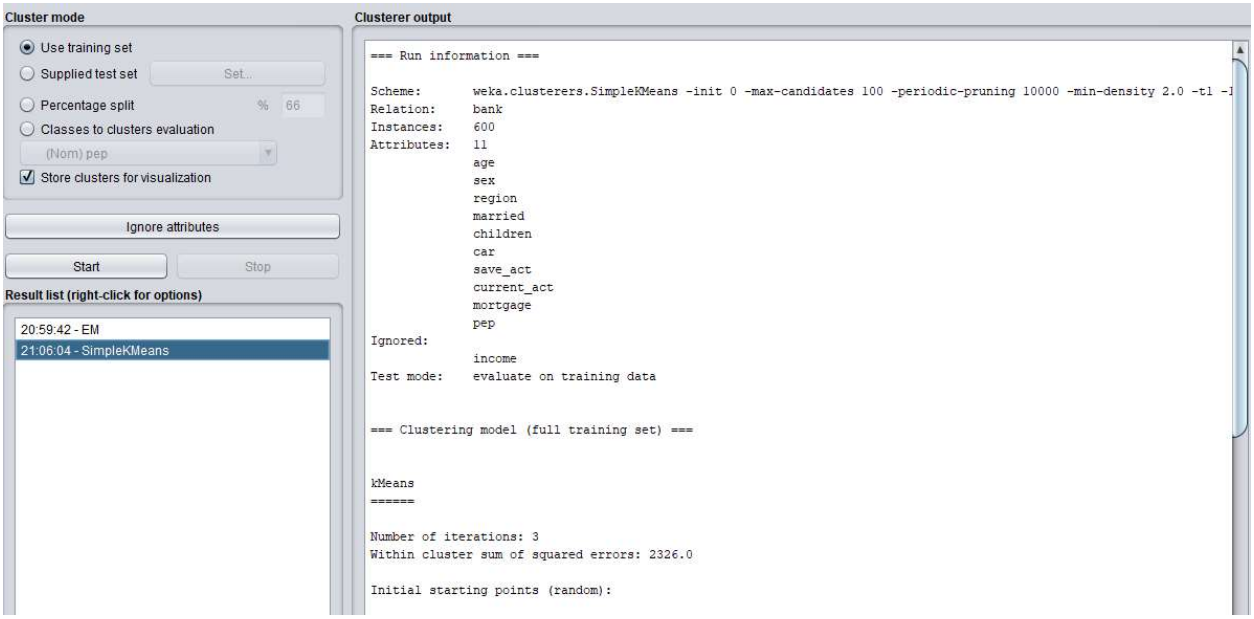
Use training set: Sử dụng chính bộ dữ liệu training để test.

Supplied test set: Cung cấp một bộ test set khác.

Percentage split: Chia bộ dữ liệu theo phần trăm. Bao nhiêu nhiêu phần trăm dùng để train , còn lại dùng để test.

Nút ignore attributes: chọn thuộc tính bị lờ đi trong lúc gom nhóm.

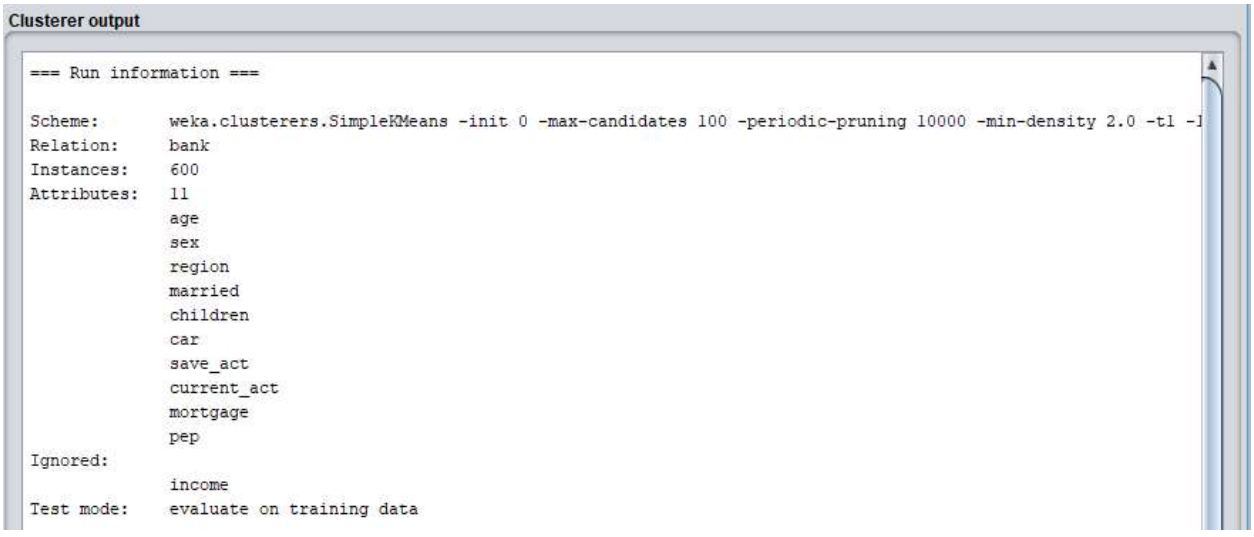
6.4. Bước 4: Tiến hành gom nhóm



Ta nhấn Start để bắt đầu chạy thuật toán

Khung result list lưu lại các kết quả đã thực hiện. Kết quả thuật toán sẽ thể hiện ở khung Clusterer ouput.

6.5. Bước 5: Ghi nhận kết quả.



Thông tin thuật toán cũng như bộ dữ liệu. Thuộc tính bị lờ đi là income và kiểu test là dùng bộ training.

```
kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 2326.0

Initial starting points (random):

Cluster 0: 0_34,FEMALE,RURAL,NO,3,NO,YES,YES,NO,NO
Cluster 1: 52_max,FEMALE,RURAL,YES,2,NO,YES,YES,NO,NO

Missing values globally replaced with mean/mode

Final cluster centroids:
                           Cluster#
Attribute      Full Data          0          1
                 (600.0)    (330.0)    (270.0)
=============================================
age                35_51       0_34     52_max
sex               FEMALE       MALE     FEMALE
region        INNER_CITY INNER_CITY INNER_CITY
married              YES         NO        YES
children               0          0          0
car                   NO         NO        YES
save_act             YES        YES        YES
current_act          YES        YES        YES
mortgage              NO         NO         NO
pep                   NO         NO         NO



Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      330 ( 55%)
1      270 ( 45%)
```

Ta phân thành 2 lớp:

Lớp 0: 0_34,FEMALE,RURAL,NO,3,NO,YES,YES,NO,NO

Lớp 1: 52_max,FEMALE,RURAL,YES,2,NO,YES,YES,NO,NO

Với độ lệch chuẩn là 2326

Có 330 được phần vào lớp 0 và 270 mẫu được phân vào lớp 1.