

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
THÀNH PHỐ HỒ CHÍ MINH



BÀI TẬP 1: TIỀN XỬ LÝ DỮ  
LIỆU VỚI WEKA

**I. THÔNG TIN SINH VIÊN**

Họ và tên: TRẦN NHẬT HUY

Mssv: 1612272

Email: [nhathuy13598@gmail.com](mailto:nhathuy13598@gmail.com)

Sđt: 0354 878 677

**II. BẢNG BÁO CÁO CÔNG VIỆC**

STT	CÁC CÂU HỎI	MỨC ĐỘ HOÀN THÀNH	GHI CHÚ
1	a. Định nghĩa sự hợp nhất dữ liệu	100%	
	b. Có vấn đề nhận hiện thực thể hay không? Giải quyết (nếu có)	100%	
	c. Có vấn đề dữ liệu dư thừa không? Giải quyết (nếu có)	100%	
	d. Có mâu thuẫn dữ liệu không? Giải quyết (nếu có)	100%	
	e. Tích hợp 2 dataset	100%	
	f. Chụp lại màn hình	100%	
2	a. Xem thuộc tính age và trả lời các câu hỏi	100%	
	b. Liệt kê five-number summary của thuộc tính age	100%	
	c. Bao nhiêu thuộc tính số, có thứ tự, rời rạc/danh sách?	100%	
	d. Giải thích đồ thị trong Explorer	100%	
	e. Dán ảnh chụp các đồ thị vào bài làm	100%	
	f. Nhận xét về đồ thị	100%	
	g. Dán đồ thị bạn cho rằng có khả năng đoán bệnh tim tốt nhất	100%	
	h. Những cặp thuộc tính nào tương quan?	100%	
3	a. Có bao nhiêu thuộc tính trong dataset?	100%	
	b. Liệt kê các phương pháp lọc thuộc tính	100%	
	c. So sánh các phương pháp trong textbook và weka	100%	
4	a. Dữ liệu thiếu	80%	Chưa cài đặt một phương pháp trong weka
	b. Dữ liệu nhiễu	100%	
	c. Dữ liệu tạp	100%	
	d. Lưu dataset đã làm sạch	100%	
5	a. Xây dựng thuộc tính	100%	
	b. Chuẩn hóa	100%	
	c. Chọn 1 phương pháp chuẩn hóa	100%	
	d. Lưu dataset đã chuẩn hóa	100%	
6	Lấy mẫu	100%	

### III. CHI TIẾT BÀI LÀM

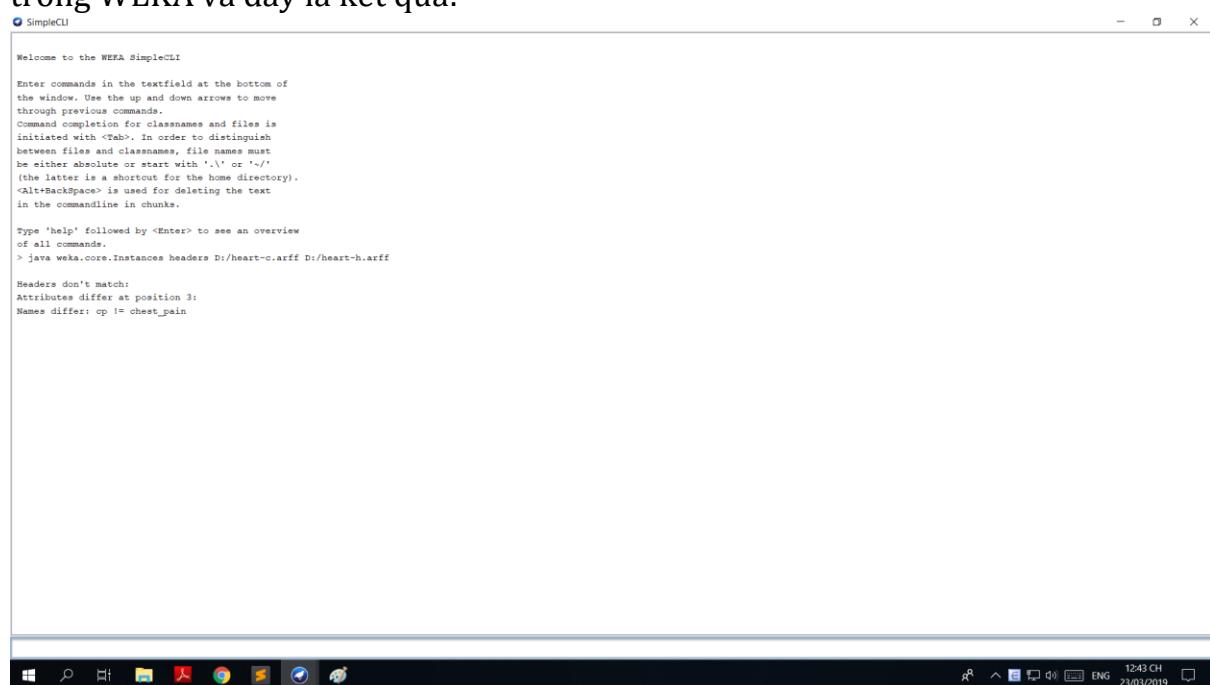
#### 1. CHUẨN BỊ DỮ LIỆU – TÍCH HỢP DỮ LIỆU (INTEGRATION)

##### a) Định nghĩa sự tích hợp dữ liệu

**Sự tích hợp dữ liệu (Data Integration)** là kết hợp các dữ liệu từ nhiều nguồn khác nhau để tạo thành một kho dữ liệu mạch lạc, các nguồn này có thể bao gồm nhiều cơ sở dữ liệu (database), khối dữ liệu (data cube) hoặc là **file phẳng (flat file)**.

##### b) Có vấn đề về nhận diện thực thể (entity identification) trong 2 dataset này hay không? Nếu có, giải quyết như thế nào?

Trong 2 data này có vấn đề về nhận diện thực thể. Ta sẽ kiểm tra bằng lệnh trong WEKA và đây là kết quả:



```

SimpleCLI

Welcome to the WEKA simpleCLI

Enter commands in the textfield at the bottom of
the window. Use the up and down arrows to move
through previous commands.
Command completion for classnames and files is
initiated with <Tab>. In order to distinguish
between files and classnames, file names must
be either absolute or start with '.' or '/'
(the latter is a shortcut for the home directory).
<Alt+BackSpace> is used for deleting the text
in the commandline in chunks.

Type 'help' followed by <Enter> to see an overview
of all commands.
> java weka.core Instances headers D:/heart-c.arff D:/heart-h.arff

Headers don't match:
Attributes differ at position 3:
Names differ: cp != chest_pain

```

Hình 1: Bi lỗi attributes cp != chest\_pain

Ta sẽ tiến hành sửa lại thuộc tính **cp** thành **chest\_pain**

```

D:\heart-c.arff (New folder) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES heart-c.arff
FOLDERS New folder
282 % From: 2 To: flat
283 % From: 3 To: down
284 %
285 %
286 % Relabeled values in attribute 'thal'
287 % From: 6 To: fixed_defect
288 % From: 3 To: normal
289 % From: 7 To: reversible_defect
290 %
291 %
292 % Relabeled values in attribute 'num'
293 % From: '0' To: '<50'
294 % From: '1' To: '>50_1'
295 % From: '2' To: '>50_2'
296 % From: '3' To: '>50_3'
297 % From: '4' To: '>50_4'
298 %
299 @relation cleveland-14-heart-disease
300 @attribute 'age' real
301 @attribute 'sex' { female, male }
302 @attribute 'chest_pain' { typ_angina, asympt, non_anginal, atyp_angina }
303 @attribute 'trestbps' real
304 @attribute 'chol' real
305 @attribute 'fbs' { t, f }
306 @attribute 'restecg' { left_vent_hyper, normal, st_t_wave_abnormality }
307 @attribute 'thalach' real
308 @attribute 'exang' { no, yes }
309 @attribute 'oldpeak' real
310 @attribute 'slope' { up, flat, down }
311 @attribute 'ca' real
312 @attribute 'thal' { fixed_defect, normal, reversible_defect }
313 @attribute 'num' { '<50', '>50_1', '>50_2', '>50_3', '>50_4' }
314 @data
315 63,male,typ_angina,145,233,t,left_vent_hyper,150,no,2,3,down,0,fixed_defect,'<50'
316 67,male,asympt,160,286,f,left_vent_hyper,108,yes,1,5,flat,3,normal,'>50_1'
317 67,male,asympt,120,229,f,left_vent_hyper,129,yes,2,6,flat,2,reversible_defect,'>50_1'
318 37,male,non_anginal,130,250,f,normal,187,no,3,5,down,0,normal,'<50'
319 41,female,atyp_angina,130,204,f,left_vent_hyper,172,no,1,4,up,0,normal,'<50'

```

71 characters selected

Tab Size: 4 Plain Text

1243 CH 23/03/2019

Hình 2: Sửa lại trong file heart-c.arff

### Sửa lại thuộc tính cp trong file heart-c.arff thành chest\_pain

Kiểm tra lại xem còn bị lỗi hay không

```

SimpleCLI

Welcome to the WEKA SimpleCLI

Enter commands in the textfield at the bottom of
the window. Use the up and down arrows to move
through previous commands.
Command completion for classnames and files is
initiated with <Tab>. In order to distinguish
between files and classnames, file names must
be either absolute or start with '.' or '-'
(the latter is a shortcut for the home directory).
<Alt+Backspace> is used for deleting the text
in the commandline in chunks.

Type 'help' followed by <Enter> to see an overview
of all commands.
> java weka.core Instances headers D:/heart-c.arff D:/heart-h.arff

Headers don't match:
Attributes differ at position 3:
Names differ: cp != chest_pain
> java weka.core Instances headers D:/heart-c.arff D:/heart-h.arff

Headers don't match:
Attributes differ at position 11:
Labels differ at position 1: up != down

```

1244 CH 23/03/2019

Hình 3: Lỗi Labels differ at position 1 up != down

Tiến hành sửa lỗi, lần này chúng ta chỉ cần thay đổi thứ tự của up và down cho phù hợp

```

D:\heart-c.arff (New folder) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES heart-c.arff FOLDERS
282 % From: 2 To: flat
283 % From: 3 To: down
284 %
285 %
286 % Relabeled values in attribute 'thal'
287 % From: 6 To: fixed_defect
288 % From: 3 To: normal
289 % From: 7 To: reversible_defect
290 %
291 %
292 % Relabeled values in attribute 'num'
293 % From: '0' To: '<50'
294 % From: '1' To: '>50_1'
295 % From: '2' To: '>50_2'
296 % From: '3' To: '>50_3'
297 % From: '4' To: '>50_4'
298 %
299 @relation cleveland-14-heart-disease
300 @attribute 'age' real
301 @attribute 'sex' { female, male }
302 @attribute 'chest_pain' { typ_angina, asympt, non_anginal, atyp_angina }
303 @attribute 'trestbps' real
304 @attribute 'chol' real
305 @attribute 'fbs' { t, f }
306 @attribute 'restecg' { left_vent_hyper, normal, st_t_wave_abnormality }
307 @attribute 'thalach' real
308 @attribute 'exang' { no, yes }
309 @attribute 'oldpeak' real
310 @attribute 'slope' { down, flat, up }
311 @attribute 'ca' real
312 @attribute 'thal' { fixed_defect, normal, reversible_defect }
313 @attribute 'num' { '<50', '>50_1', '>50_2', '>50_3', '>50_4' }
314 @data
315 63,male,typ_angina,145,233,t,left_vent_hyper,150,no,2,3,down,0,fixed_defect,'<50'
316 67,male,asympt,160,286,f,left_vent_hyper,108,yes,1,5,flat,3,normal,'>50_1'
317 67,male,asympt,120,229,f,left_vent_hyper,129,yes,2,6,flat,2,reversible_defect,'>50_1'
318 37,male,non_anginal,130,250,f,normal,187,no,3,5,down,0,normal,'<50'
319 41,female,atyp_angina,130,204,f,left_vent_hyper,172,no,1,4,up,0,normal,'<50'

36 characters selected
Tab Size: 4 Plain Text
12:44 CH 23/03/2019

```

Hình 4: Sửa lại thứ tự trong file heart-c.arff

### Tiến hành kiểm tra lại

```

SimpleCLI

Welcome to the WEKA SimpleCLI

Enter commands in the textfield at the bottom of
the window. Use the up and down arrows to move
through previous commands.
Command registration for classnames and files is
initiated with <Tab>. In order to distinguish
between files and classnames, file names must
be either absolute or start with '.' or './'
(the latter is a shortcut for the home directory).
<Alt+Delete> is used for deleting the text
in the commandline in chunks.

Type 'help' followed by <Enter> to see an overview
of all commands.
> java weka.core Instances headers D:/heart-c.arff D:/heart-h.arff

Headers don't match:
Attributes differ at position 3:
Names differ: cp != chest_pain
> java weka.core Instances headers D:/heart-c.arff D:/heart-h.arff

Headers don't match:
Attributes differ at position 11:
Labels differ at position 1: up != down
> java weka.core Instances headers D:/heart-c.arff D:/heart-h.arff

Headers match

1_d.png - Paint

```

Hình 5: Lần này không có lỗi nên có thông báo Header match

c) Có vấn đề dữ liệu dư thừa (redundancy) trong 2 dataset này hay không? Nếu có, giải quyết như thế nào?

**Dư thừa dữ liệu (Redundancy)** là một vấn đề quan trọng trong hợp nhất dữ liệu. Một thuộc tính có thể được suy ra từ một thuộc tính khác hoặc 1 tập các thuộc

tính. Sự không nhất quán trong cách đặt tên thuộc tính hoặc chiều cũng gây nên dư thừa dữ liệu.

Một số loại dư thừa dữ liệu có thể được phát hiện bằng cách **phân tích quan hệ (correlation analysis)**. Với thuộc tính số, ta xác định quan hệ giữa 2 thuộc tính A và B bằng cách tính **hệ số quan hệ (correlation coefficient)**. Với thuộc tính phân loại hoặc rời rạc, ta xác định bằng cách sử dụng **kiểm tra Chi-square (Chi-square test)**.

Bên cạnh việc phát hiện dư thừa dữ liệu giữa các thuộc tính, **sự trùng lặp dữ liệu (duplication)** cũng cần phải thực hiện.

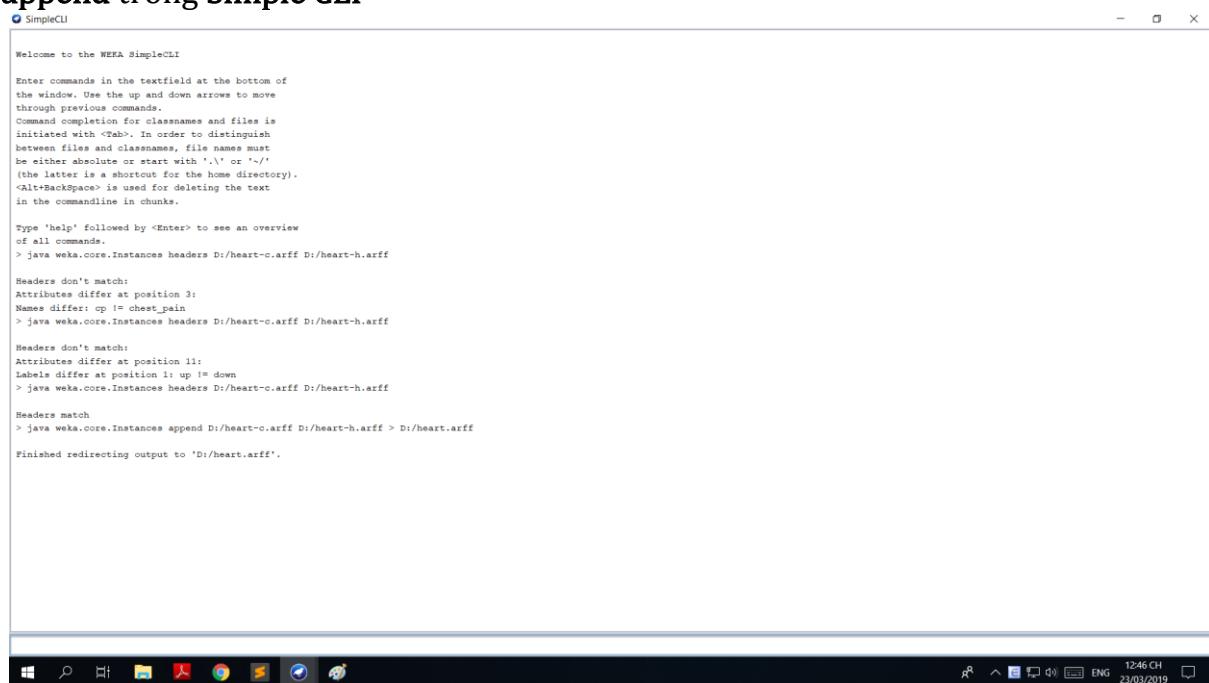
Trong 2 data này có sự dư thừa dữ liệu, cụ thể là sự trùng lặp dữ liệu. Chúng ta sẽ giải quyết vấn đề này bằng cách sử dụng bộ lọc mà Weka cung cấp sẵn sau khi đã hợp nhất 2 dataset lại với nhau

*d) Có sự mâu thuẫn dữ liệu (data value conflicts) trong 2 dataset này hay không? Nếu có, giải quyết như thế nào?*

Trong 2 dataset này có **mâu thuẫn dữ liệu (data value conflicts)**, cụ thể là giá trị của attribute **slope** mà chúng ta đã chỉnh sửa ở trên

*e) Tích hợp 2 dataset này lại thành 1 dataset để chuẩn bị cho các câu hỏi tiếp theo. Nạp dataset sau khi tích hợp vào Explorer. Bạn có bao nhiêu mẫu? Bao nhiêu thuộc tính?*

Để tích hợp 2 dữ liệu lại với nhau ta sẽ sử dụng lệnh **java weka.core Instances append** trong Simple CLI



```

SimpleCLI

Welcome to the WEKA SimpleCLI

Enter commands in the textfield at the bottom of
the window. Use the up and down arrows to move
through previous commands.
Command completion for classnames and files is
initiated with <Tab>. In order to distinguish
between files and classnames, file names must
be either absolute or start with '\.' or './'
(the latter is a shortcut for the home directory).
<Alt+BackSpace> is used for deleting the text
in the commandline in chunks.

Type 'help' followed by <Enter> to see an overview
of all commands.
> java weka.core.Instances headers D:/heart-c.arff D:/heart-h.arff

Headers don't match:
Attributes differ at position 3:
Names differ: cp != chest_pain
> java weka.core.Instances headers D:/heart-c.arff D:/heart-h.arff

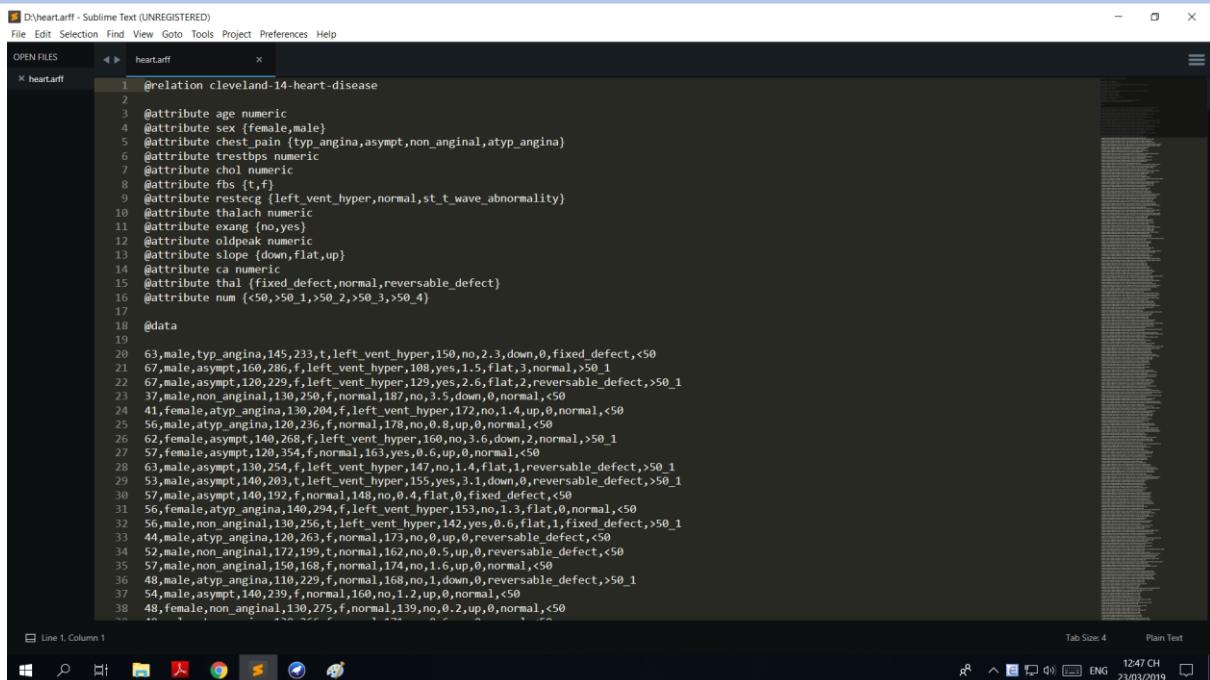
Headers don't match:
Attributes differ at position 11:
Labels differ at position 1: up != down
> java weka.core.Instances headers D:/heart-c.arff D:/heart-h.arff

Headers match
> java weka.core.Instances append D:/heart-c.arff D:/heart-h.arff > D:/heart.arff

Finished redirecting output to 'D:/heart.arff'.

```

Hình 6: Tạo file heart.arff bằng lệnh append



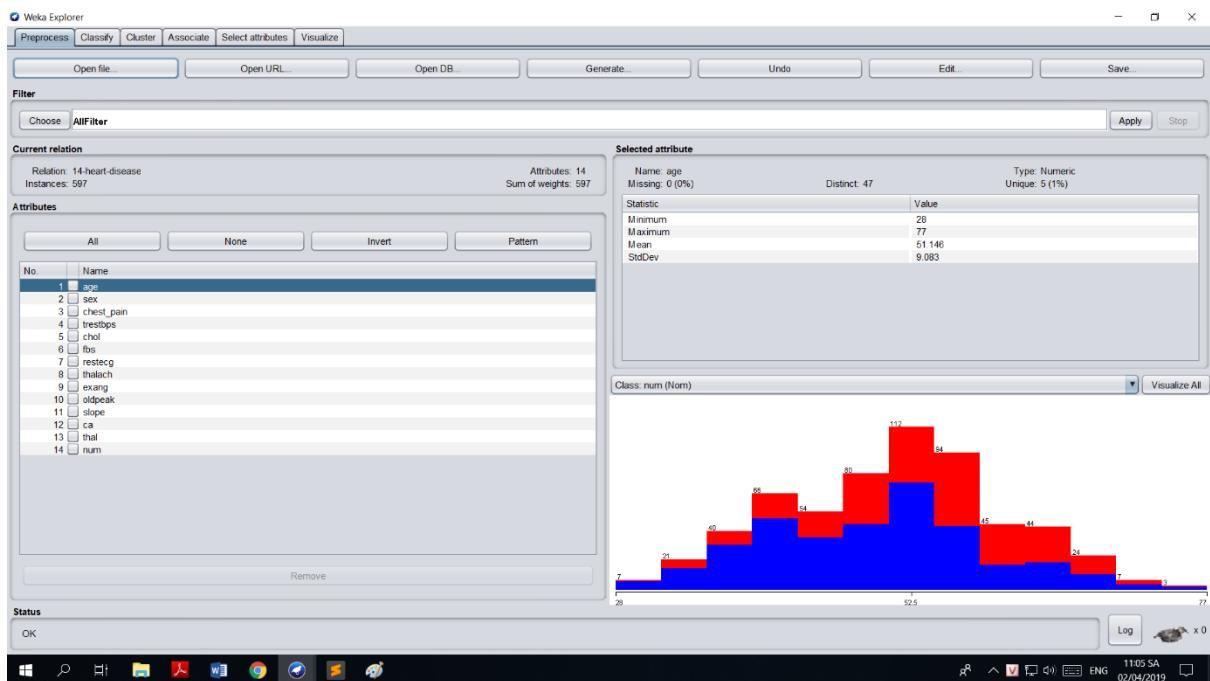
```

@relation cleveland-14-heart-disease
@attribute age numeric
@attribute sex {female, male}
@attribute chest_pain {typ_angina, asympt, non_anginal, atyp_angina}
@attribute trestbps numeric
@attribute chol numeric
@attribute fbs {t,f}
@attribute restecg {left_vent_hyper, normal, st_t_wave_abnormality}
@attribute thalach numeric
@attribute exang {no,yes}
@attribute oldpeak numeric
@attribute slope {down,flat,up}
@attribute ca numeric
@attribute thal {fixed_defect,normal,reversible_defect}
@attribute num (<50,>50_1,>50_2,>50_3,>50_4)
@data
20 63,male,typ_angina,145,233,t,left_vent_hyper,150,no,2.3,down,0,fixed_defect,<50
21 67,male,asympt,160,286,f,lelf_vent_hyper,108,yes,1.5,flat,3,normal,>50_1
22 67,male,asympt,120,229,f,lelf_vent_hyper,129,yes,2.6,flat,2,reversible_defect,>50_1
23 37,male,non_anginal,130,250,f,normal,187,no,3.5,down,0,normal,<50
24 41,female,atyp_angina,130,284,f,lelf_vent_hyper,172,no,1.4,up,0,normal,<50
25 56,male,atyp_angina,120,236,f,normal,178,no,0.8,up,0,normal,<50
26 62,female,asympt,140,268,f,lelf_vent_hyper,160,no,3.6,down,2,normal,>50_1
27 57,female,asympt,120,354,f,normal,163,yes,0.6,up,0,normal,<50
28 63,male,asympt,130,254,f,lelf_vent_hyper,147,no,1.4,flat,1,reversible_defect,>50_1
29 53,male,asympt,140,203,t,lelf_vent_hyper,155,yes,3.1,down,0,reversible_defect,>50_1
30 57,male,asympt,140,192,f,normal,148,no,0.4,flat,0,fixed_defect,<50
31 56,female,atyp_angina,140,294,f,lelf_vent_hyper,153,no,1.3,flat,0,normal,<50
32 56,male,non_anginal,130,250,t,lelf_vent_hyper,142,yes,0.6,flat,1,fixed_defect,>50_1
33 44,male,atyp_angina,120,263,f,normal,173,no,0,up,0,reversible_defect,<50
34 52,male,non_anginal,172,199,t,normal,162,no,0.5,up,0,reversible_defect,<50
35 57,male,non_anginal,150,168,f,normal,174,no,1.6,up,0,normal,<50
36 48,male,atyp_angina,110,229,f,normal,168,no,1,down,0,reversible_defect,>50_1
37 54,male,asympt,140,239,f,normal,160,no,1.2,up,0,normal,<50
38 48,female,non_anginal,130,275,f,normal,139,no,0.2,up,0,normal,<50

```

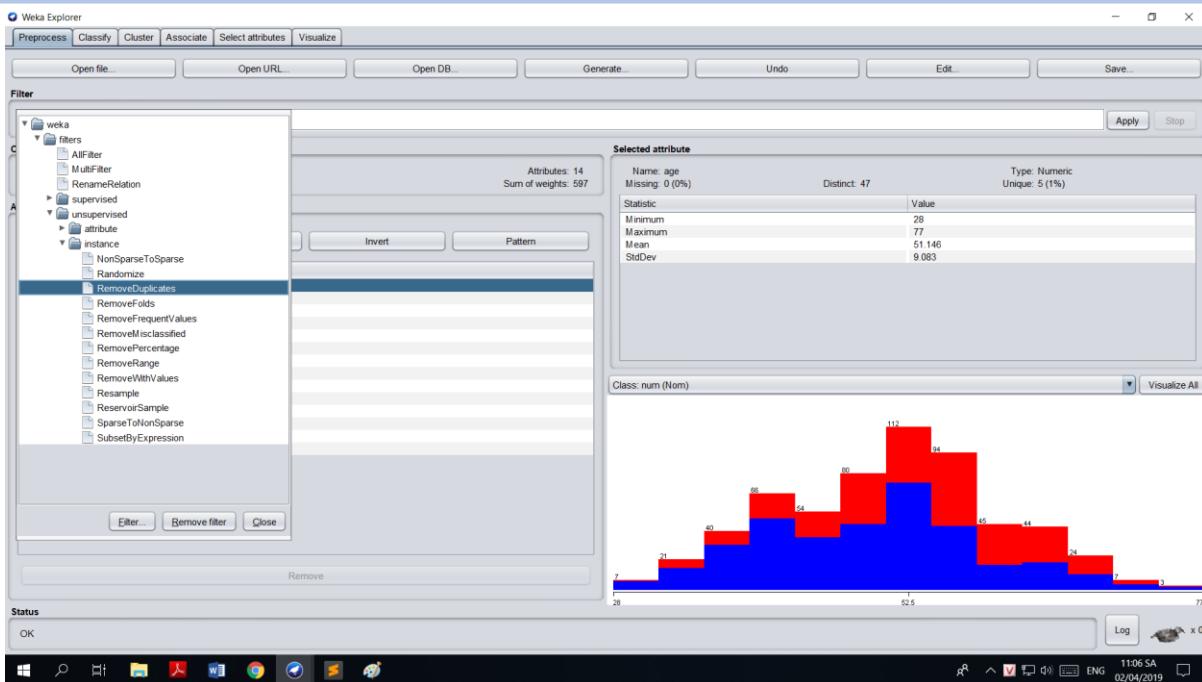
Hình 7: File heart.arff sau khi sử dụng lệnh append

Ta sửa lại tên quan hệ là **14-heart-disease**  
Load dataset vào Explorer của Weka



Hình 8: Dataset trong Weka

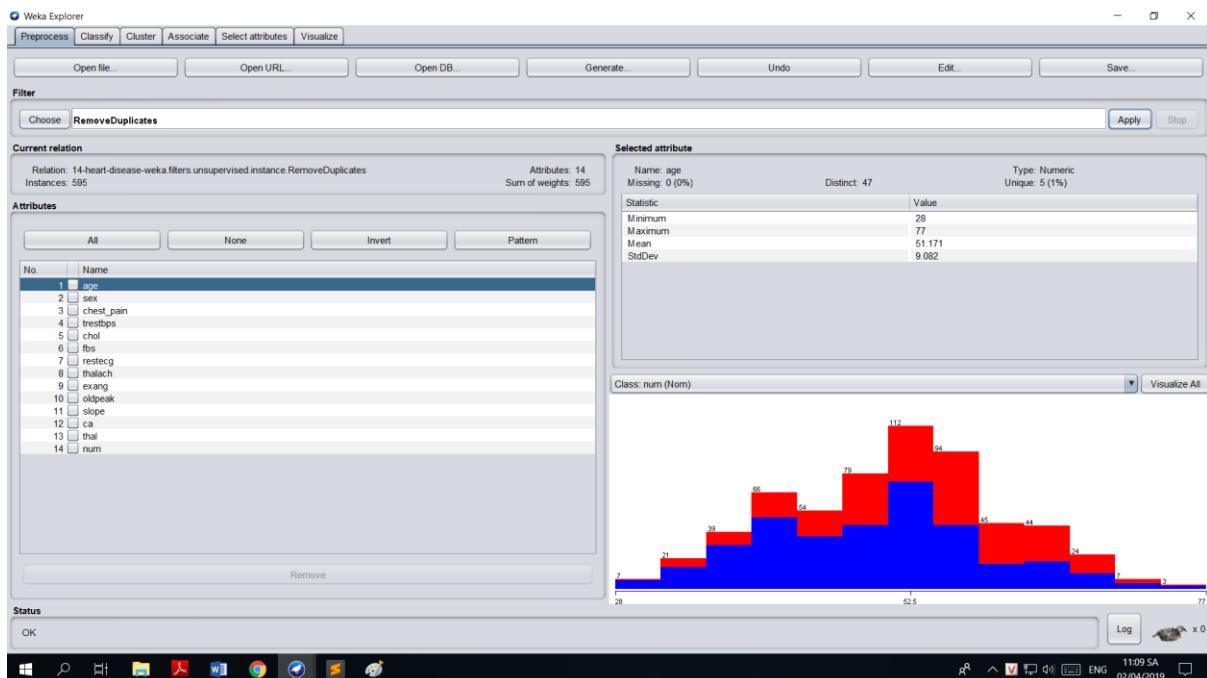
Ta tiến hành loại bỏ **Duplicate Instances** bằng bộ lọc như trong hình



Hình 9: Dùng bộ lọc RemoveDuplicate

f) Chụp lại màn hình của cửa sổ Explorer của bạn.

Ta chọn **Apply** và đây là kết quả



Ban đầu có **597 instances**, sau khi lọc chúng ta còn **595**

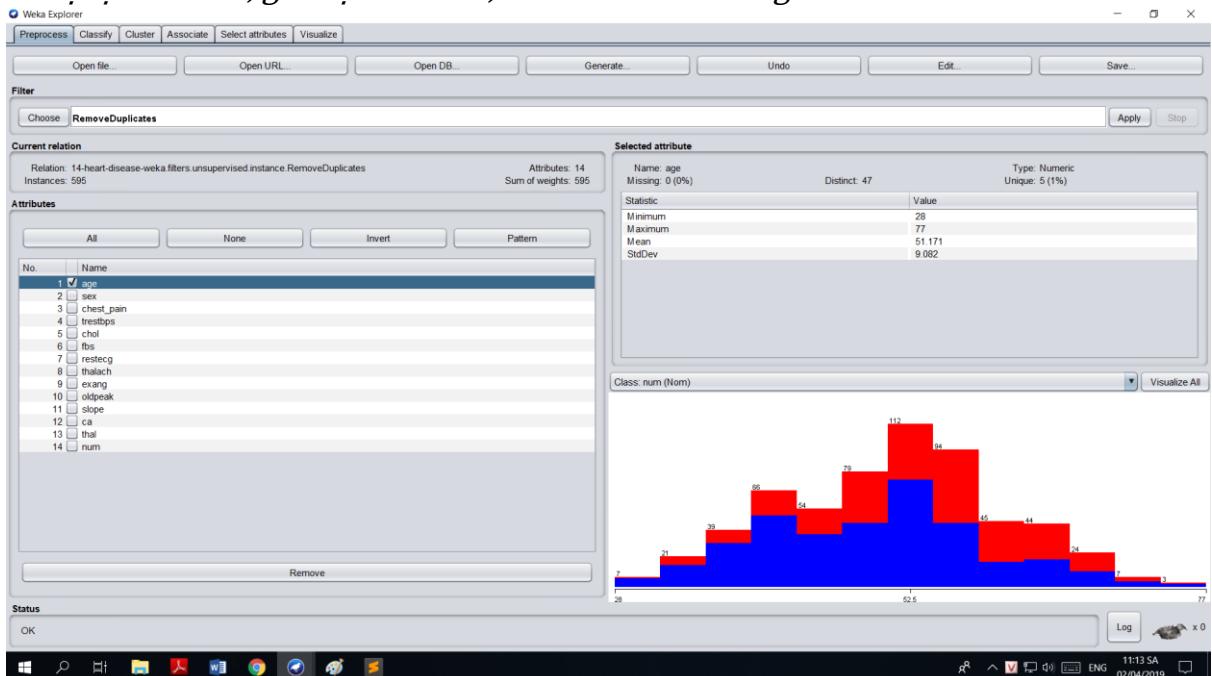
Nhấn **save** để lưu lại dataset này cho các bước sau

File **heart-unfilter.arff** là file chưa sử dụng bộ lọc RemoveDuplicate

File **heart.arff** là file đã sử dụng bộ lọc RemoveDuplicate

## 2. TÓM TẮT MÔ TẢ DỮ LIỆU – DESCRIPTIVE DATA SUMMARIZATION

a) Trong tab **Preprocess**, xem xét thuộc tính **age** và trả lời câu hỏi: trung bình, độ lệch chuẩn, giá trị nhỏ nhất, lớn nhất của nó là gì?



Hình 10: Thuộc tính Age

**Trung bình (Mean)** là: 51.171

**Độ lệch chuẩn (StdDev)** là: 9.082

**Giá trị nhỏ nhất (Minimum)** là: 28

**Giá trị lớn nhất (Maximum)** là: 77

b) Liệt kê five-number summary của thuộc tính này. Weka có cung cấp những con số này hay không?

Min: 28

Q<sub>1</sub>: 44

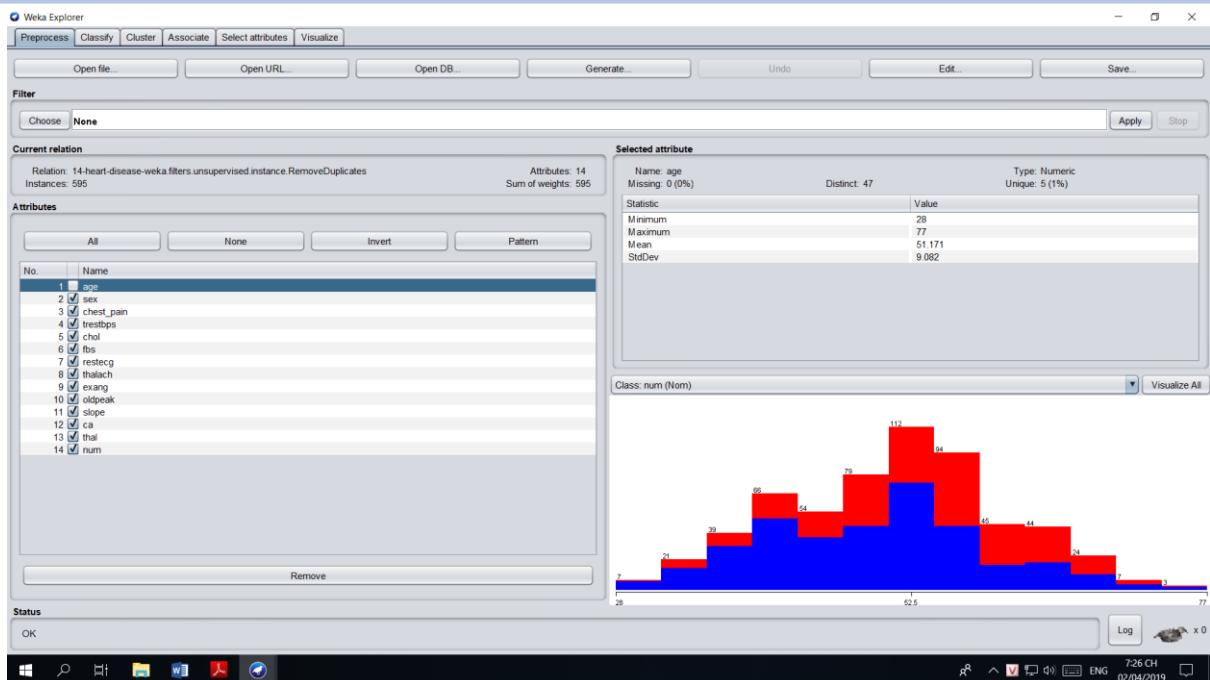
Median: 52

Q<sub>2</sub>: 88

Max: 77

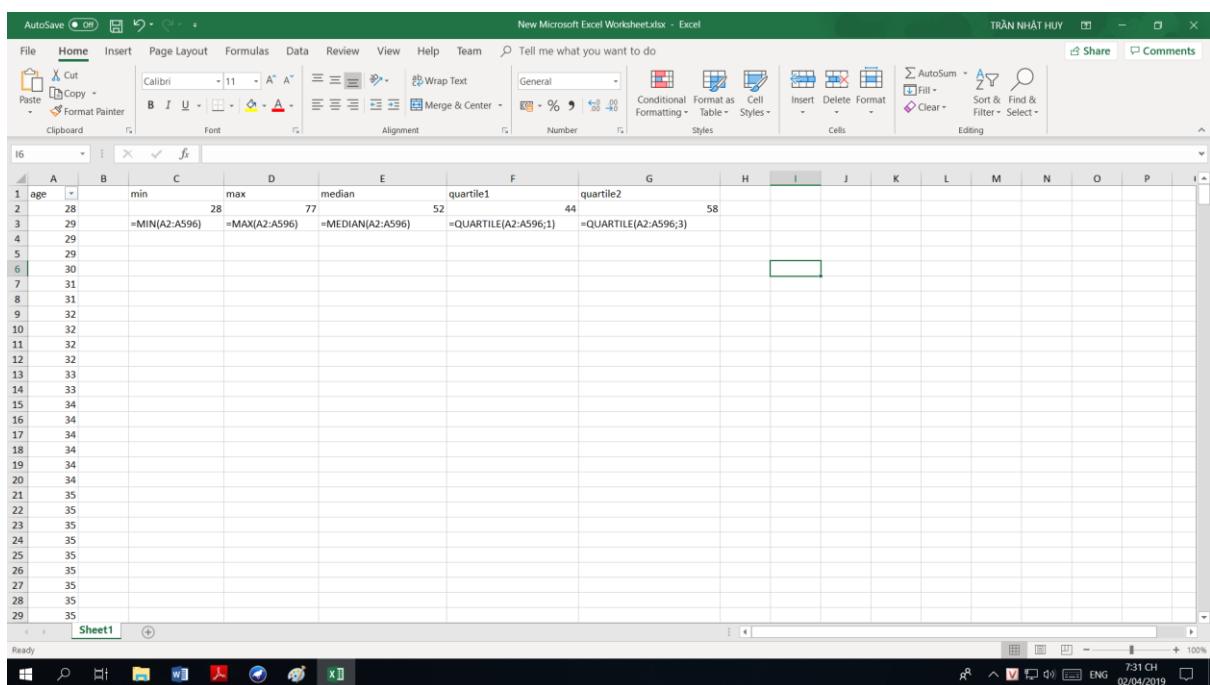
Weka chỉ cung cấp 1 phần các con số này cụ thể là min, max còn Q1, Median, Q2 thì chúng ta phải tự tìm. Ở đây, em sử dụng Excel để tìm.

Ta chọn thuộc tính **age** để giữ lại, xóa attribute khác và lưu thành một file **heart\_age.arff**



Hình 11: Nhấn Remove để loại bỏ và Save lại

Ta sao chép data trong file **heart\_age.arff** vào excel để thao tác như trong hình



Hình 12: Thao tác làm trong Excel

- c) Cho biết thuộc tính nào là số (*numeric*), thuộc tính nào là có thứ tự (*ordinal*) và thuộc tính nào là rời rạc/danh sách (*categorical/nominal*).

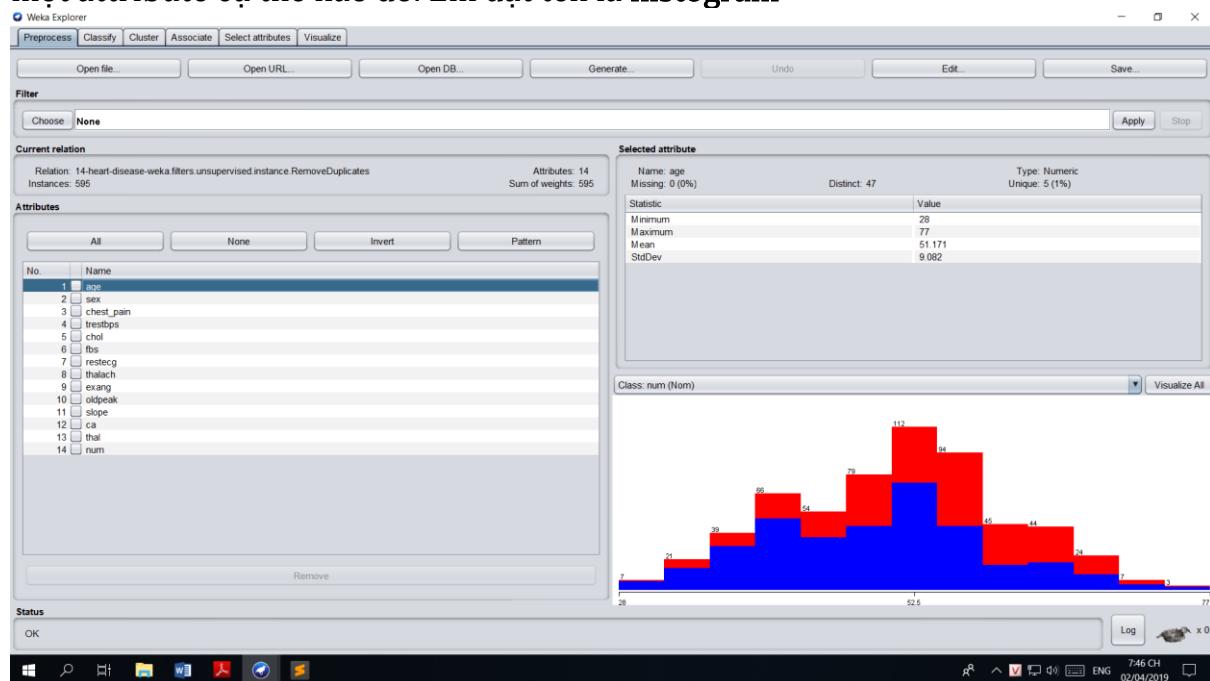
Thuộc tính số (*numeric*): age, trestbps, chol, thalach, oldpeak, ca.

Thuộc tính có thứ tự (ordinal): Không có

Thuộc tính rời rạc/danh sách (categorical/nominal): sex, chest\_pain, fbs, restecg, exang, slope, thal, num

- d) Giải thích ý nghĩa của đồ thị trong cửa sổ Explorer. Bạn đặt tên cho đồ thị này là gì? Màu xanh và màu đỏ có nghĩa gì (chú ý các pop-up hiện lên khi di chuyển chuột trên đồ thị). Đồ thị này biểu diễn cho cái gì?

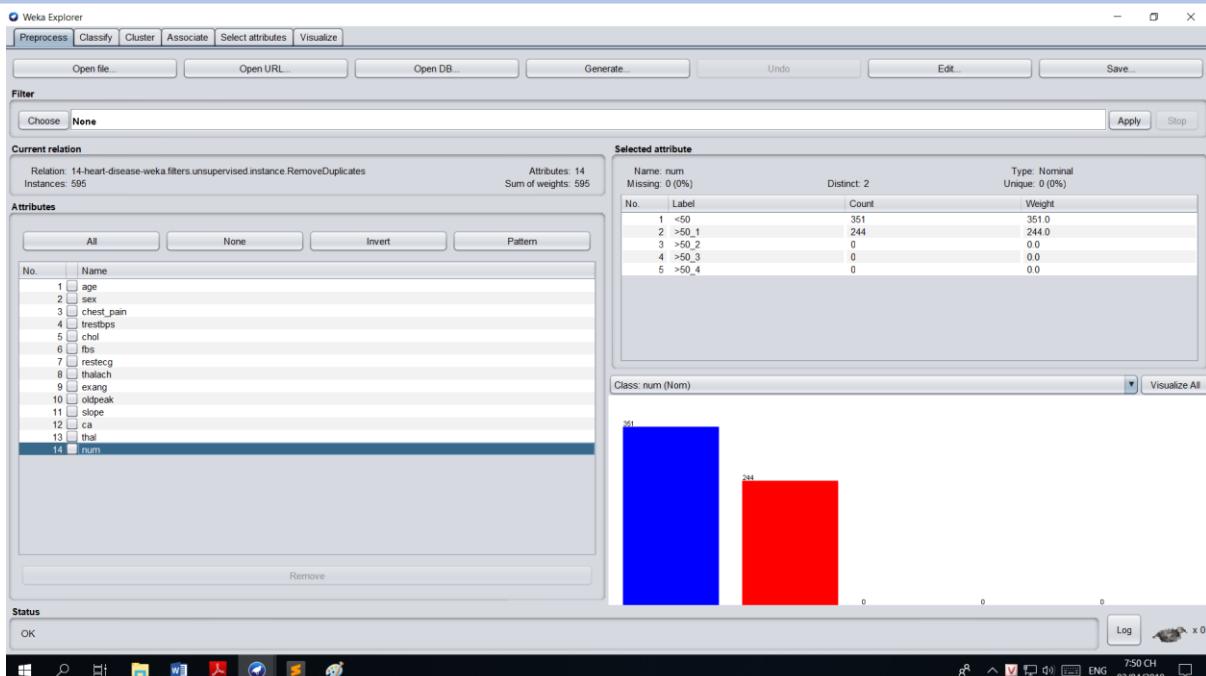
Đồ thị trong cửa sổ Explorer thể hiện phân bố của kết quả cần dự đoán theo một attribute cụ thể nào đó. Em đặt tên là **histogram**



Hình 13: Phân bố của kết quả dựa theo thuộc tính age

Màu xanh thể hiện class num có value là < 50

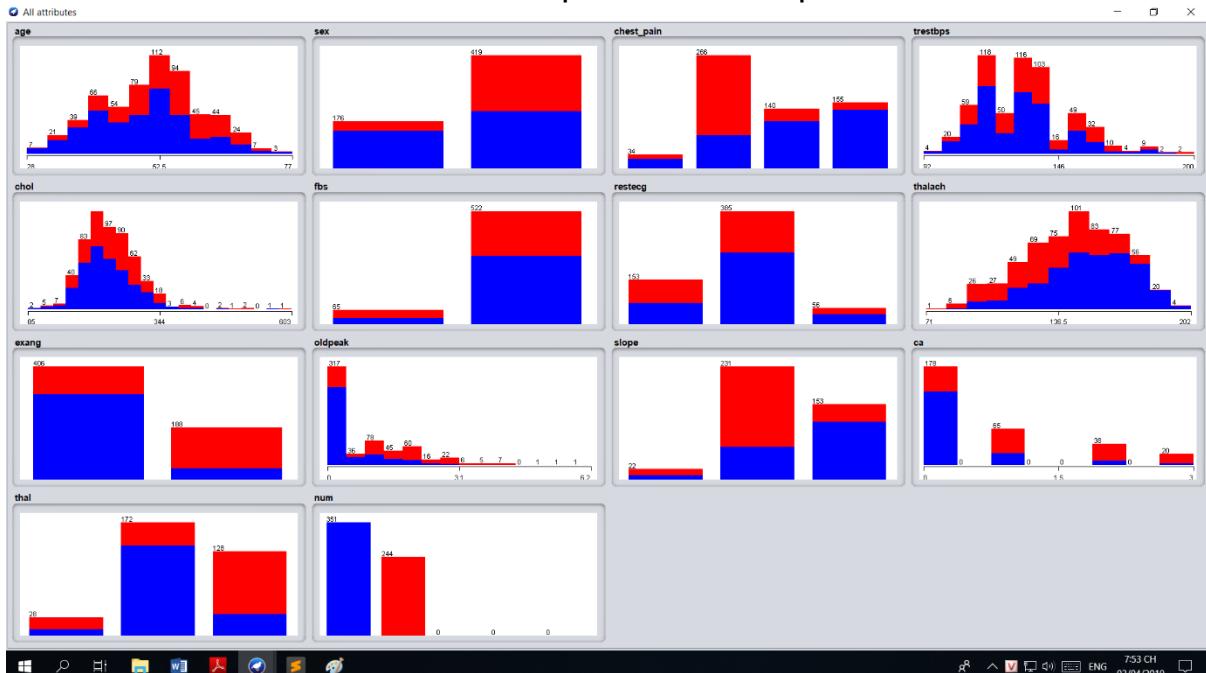
Màu đỏ thể hiện class num có value là > 50\_1



Hình 14: Chọn thuộc tính num để xác định ý nghĩa

e) Lần lượt xem xét các thuộc tính khác của dataset dưới dạng đồ thị. Dán các ảnh chụp màn hình vào bài làm.

Ta sẽ nhấn nút **Visualize all** để hiển thị tất cả các đồ thị



Hình 15: Đồ thị cho tất cả các attribute

f) Nhận xét của bạn từ những đồ thị đó?

Thuộc tính **num** sẽ có phân bố gần giống với các attribute được chọn

- g) *Chuyển sang tab Visualize. Thuật ngữ sử dụng trong textbook để đặt tên cho các đồ thị là gì? Chọn jitter tối đa, chú ý cột num (cột cuối cùng), theo bạn các thuộc tính nào có vẻ như dẫn đến bệnh tim nhiều nhất? Dán vào bài làm hình ảnh đồ thị của thuộc tính mà bạn cho rằng có khả năng dự đoán bệnh tim tốt nhất (Y) như là một hàm của num(X).*

Thuật ngữ sử dụng trong textbook là **Scatter plot matrix**

Theo em thì thuộc tính **oldpeak** có vẻ dẫn đến bệnh tim nhiều nhất



Hình 16: Đồ thị của num(X) và Oldpeak(Y)

- h) *Có những cặp thuộc tính khác nhau nào có vẻ như tương quan với nhau không?*

Có các thuộc tính tuy khác nhau nhưng có vẻ tương quan với nhau như:

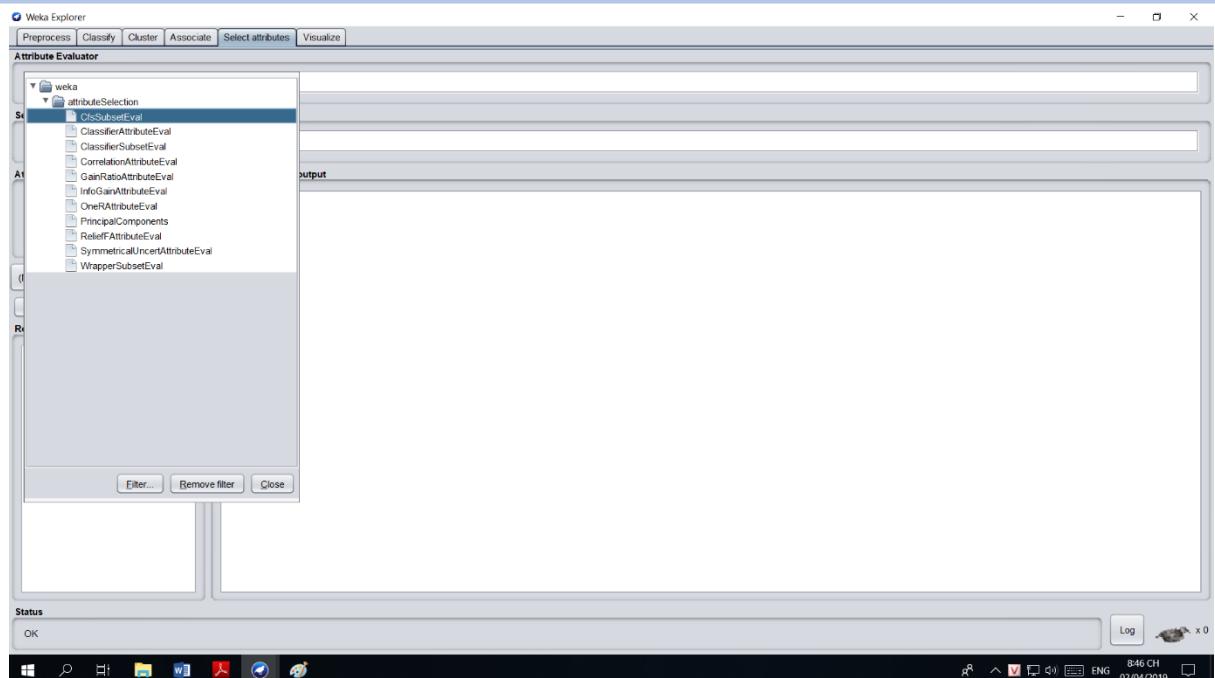
- thal, cal, slope, restecg
- oldpeak, thalach, chol, tresbps, chest\_pain, age
- sex, exang

### 3. CHUẨN BỊ DỮ LIỆU – CHỌN LỌC DỮ LIỆU

- a) *Bạn hãy cho biết có bao nhiêu thuộc tính trong những dataset trước khi xử lý?*

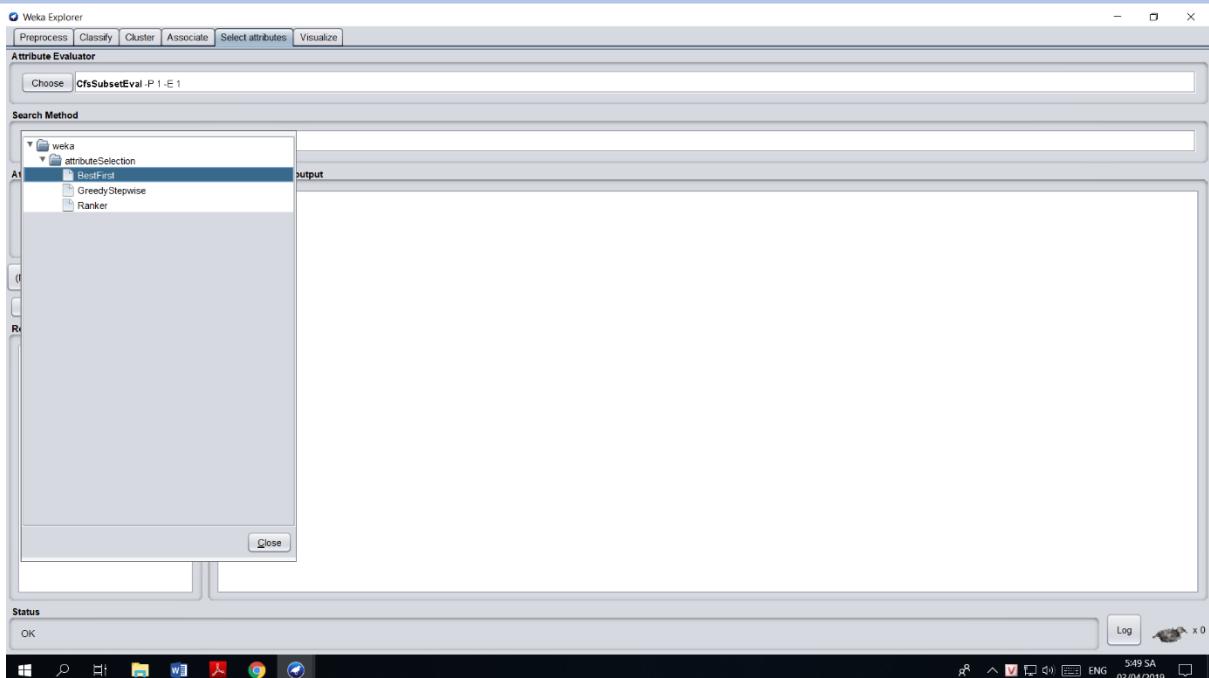
Có tất cả 14 thuộc tính trong dataset với thuộc tính **num** là cần dự đoán

- b) *Sử dụng tab Select attributes. Liệt kê những lựa chọn khác nhau của Weka để chọn lọc thuộc tính, giải thích ngắn gọn từng phương pháp.*



Hình 17: Các phương pháp Attribute Evaluator

STT	TÊN PHƯƠNG PHÁP	MỤC ĐÍCH
1	CfsSubsetEval	Đánh giá khả năng dự đoán của từng attribute riêng lẻ và mức độ dư thừa (redundancy) giữa chúng, ưu tiên tập các attribute mà có quan hệ cao với lớp (class) nhưng có sự tương quan lẫn nhau thấp. Các giá trị thiếu (Missing values) sẽ được xem như là giá trị rác hoặc giá trị của nó sẽ phụ thuộc vào phân bố của các giá trị
2	GainRatioAttributeEval	Đánh giá những thuộc tính bằng cách đo tỉ lệ tăng (gain ratio) của thuộc tính đó cho class.
3	InfoGainAttributeEval	Đánh giá những thuộc tính bằng cách đo độ tăng thông tin (Information Gain) cho class. Phương pháp này sẽ rời rạc hóa các thuộc tính numeric bằng cách dùng MDL-base discretization method
4	OneRAttributeEval	Sử dụng một phương pháp đơn giản được kết thừa từ OneR classifier. Nó có thể dùng dữ liệu huấn luyện để đánh giá hoặc nó có thể áp dụng internal cross-validation
5	PrincipalComponents	Biến đổi một tập các attribute . Các attribute mới được xếp hạng dựa vào thứ tự của eigenvalues. Tập con (Subset) được chọn bằng cách chọn số lượng vừa đủ các vector trị riêng (eigenvector) để thỏa mãn một phương sai cho trước.
6	ReliefFAttributeEval	Đây là phương pháp dựa vào instance. Phương pháp này lấy mẫu một các ngẫu nhiên và kiểm tra các instance gần đó (neighboring insances) mà có cùng hoặc khác class. Phương pháp này hoạt động cả khi class mang giá trị rác hoặc liên tục.
7	SymmetricalUncertAttributeEval	Đánh giá một thuộc tính bằng cách đo symmetrical uncertainty của nó cho class.
8	WrapperSubsetEval	Sử dụng classifier để đánh giá tập attribute và áp dụng cross-validation để dự đoán độ chính xác của lược đồ học (learning scheme) cho từng tập.



Hình 18: Các phương pháp Search

STT	TÊN PHƯƠNG PHÁP	MỤC ĐÍCH
1	BestFirst	Thực hiện leo đồi tham lam (greedy hill climbing) với quay lui (backtracking). Nó có thể tìm kiếm tiến (forward) từ một tập attribute rỗng, lui (backward) từ tập chứa toàn bộ attribute hoặc có thể bắt đầu từ một trạng thái cụ thể nào đó và tìm kiếm theo 2 hướng
2	GreedyStepwise	Tìm kiếm tham lam trong không gian các tập attribute. Nó cũng có thể tìm kiếm tới và lui. Tuy nhiên, nó không sử dụng quay lui mà dừng lại ngay khi thêm hoặc xóa đi thuộc tính tốt nhất còn lại mà làm giảm số liệu đánh giá
3	Ranker	Nó sắp xếp các thuộc tính bằng các sự đánh giá độc lập và phải được sử dụng kết hợp với phương pháp đánh giá single-attribute (single-attribute evaluator). Phương pháp này không chỉ xếp hạng các thuộc tính (attributes) mà còn thực hiện chọn các thuộc tính bằng các loại bỏ những thuộc tính xếp hạng thấp

c) So sánh với các phương pháp chọn lọc dữ liệu trong textbook, có phương pháp nào không có trong Weka hay phương pháp nào trong Weka không có trong textbook?

Phương pháp không có trong Weka là ChiSquaredAttributeEval

Phương pháp trong Weka mà không có trong textbook là ClassifierAttributeEval, ClassifierSubsetEval, CorrelationAttributeEval

#### 4. CHUẨN BỊ DỮ LIỆU – LÀM SẠCH DỮ LIỆU

a) Các giá trị thiếu (Missing values): Liệt kê các phương pháp đã học để xử lý dữ liệu thiếu. Weka đã cài đặt những phương pháp nào? Bạn hãy chọn 1 phương pháp để xử lý giá trị thiếu trong dataset, giải thích tại sao bạn chọn

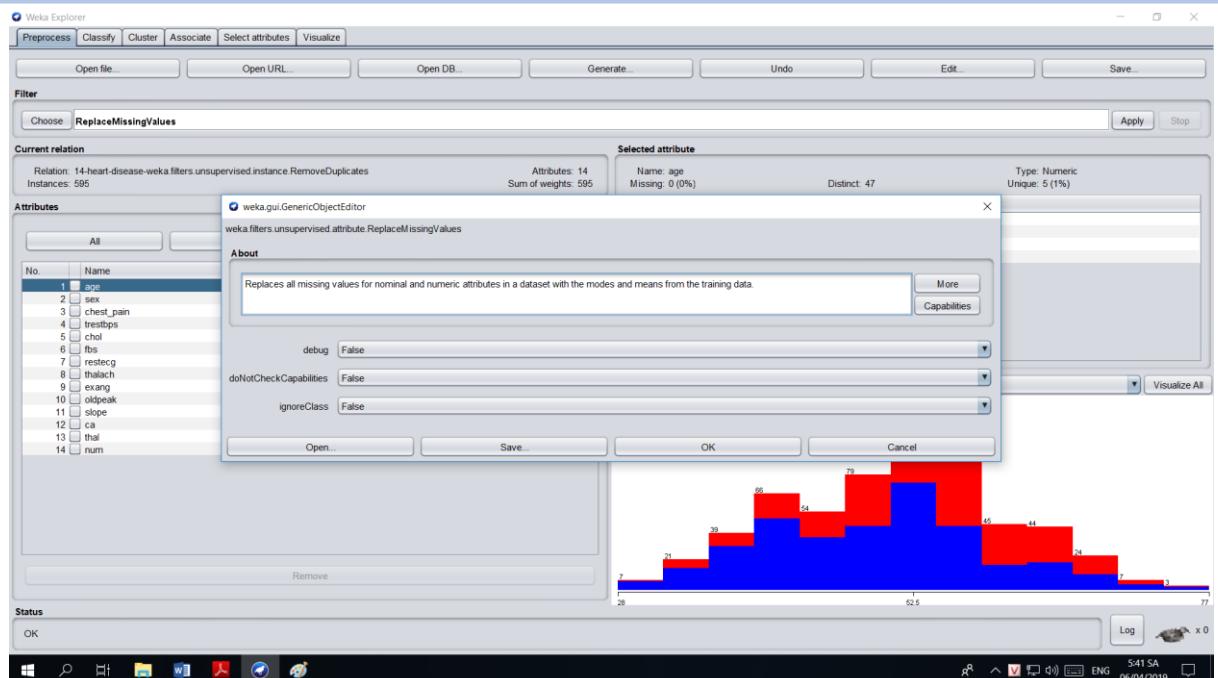
*phương pháp đó. Cài đặt 1 phương pháp khác mà bạn thích nếu nó không có trong Weka*

Các phương pháp để xử lý dữ liệu thiếu là:

STT	TÊN PHƯƠNG PHÁP	MỤC ĐÍCH
1	Loại bỏ các dòng (Ignoring the tuple)	Phương pháp này được sử dụng khi dòng đó không có (giả thiết rằng việc khai thác đòi hỏi phân lớp). Phương pháp này không hiệu quả trừ khi dòng đó có vài thuộc tính thiếu giá trị và nó sẽ hoạt động kém nếu số lượng giá trị bị mất của mỗi thuộc tính biến động lớn
2	Thêm bằng cách thủ công (Fill in the missing value manually)	Phương pháp này nhìn chung thì tốn thời gian và có thể không khả thi nếu dataset lớn với nhiều giá trị thiếu
3	Sử dụng một hằng số để thay thế giá trị thiếu (Use a global constant to fill in the missing value)	Thay thế tất cả các giá trị thiếu của thuộc tính bằng một hằng số giống nhau như “Unknown” hoặc $-\infty$ . Nếu giá trị thiếu được thay thế bằng “Unknown” thì chương trình khai thác sẽ nhầm lẫn, tưởng rằng nó đang làm việc với một khái niệm đặc biệt. Do đó, phương pháp này dễ nhưng không hoàn hảo.
4	Use the attribute mean to fill in the missing value	Sử dụng trung bình của attribute để thay thế dữ liệu thiếu
5	Use the attribute mean for all samples belonging to the same class as given tuple	Sử dụng trung bình của attribute cho tất cả các mẫu cùng thuộc về một lớp giống như tuple được cho
6	Sử dụng giá trị có khả năng nhất để thay thế giá trị thiếu (Use the most probable value to fill in the missing value)	Có thể được xác định bằng hồi quy, các phương pháp suy luận dựa trên công thức Bayes hoặc cây quyết định quy nạp

Weka đã cài đặt phương pháp: (3), (4)

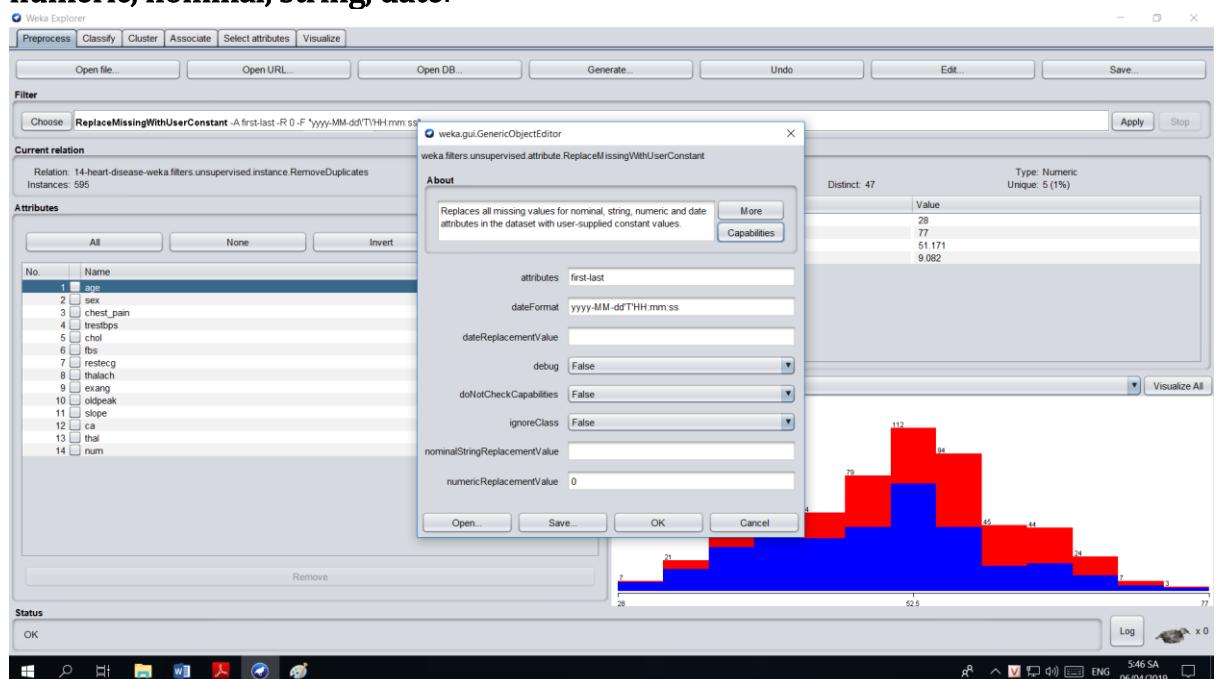
Bộ lọc **ReplaceMissingValues** thay thế các dữ liệu thiếu của tất cả thuộc tính số hoặc nominal bằng mode và trung bình của dữ liệu huấn luyện



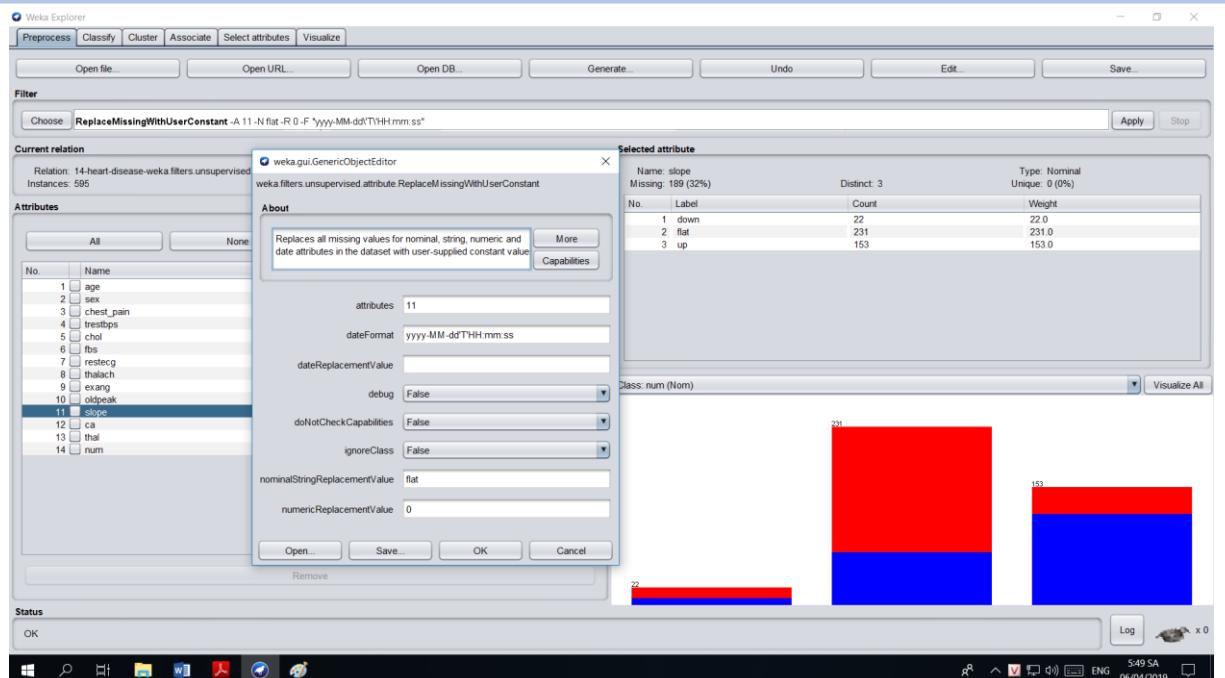
Hình 19: Bộ lọc ReplaceMissingValues

Ta sẽ chạy thử bộ lọc này ở câu kế tiếp nên chúng ta chuyển sang các bộ lọc khác.

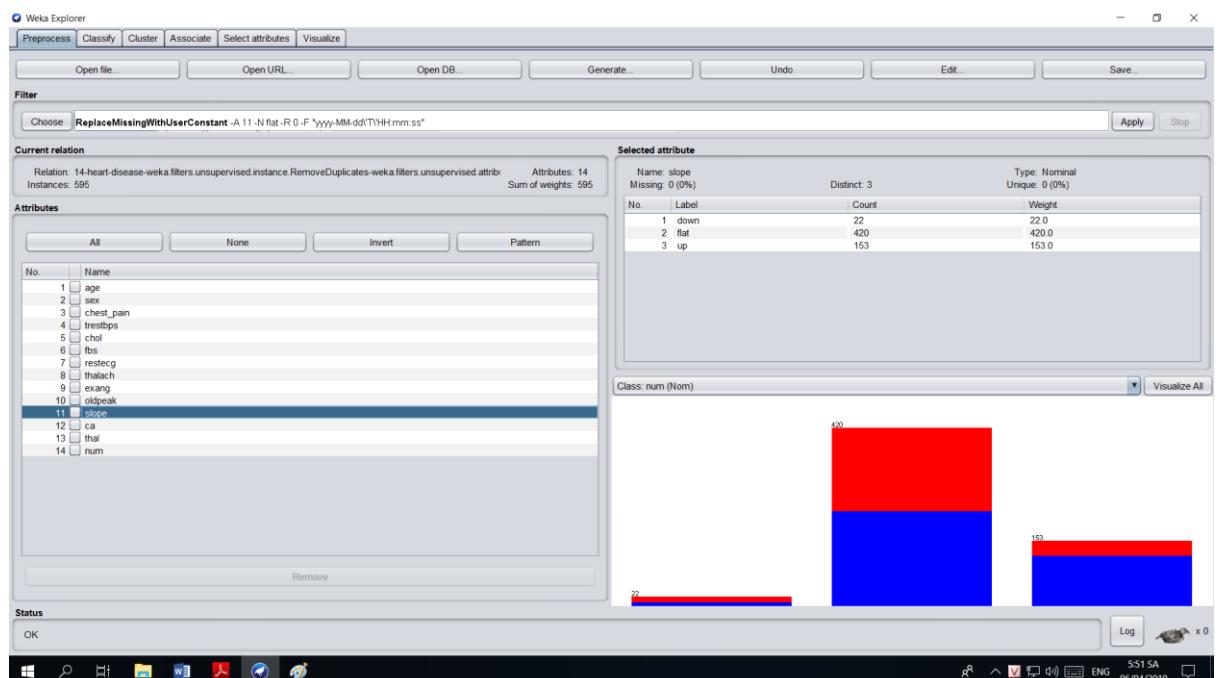
Bộ lọc **ReplaceMissingWithUserConstant** thay thế các dữ liệu thiếu với hằng số mà người dùng cung cấp. Bộ lọc này hoạt động trên các thuộc tính có kiểu numeric, nominal, string, date.



Hình 20: Bộ lọc ReplaceMissingWithUserConstant



Hình 21: Sử dụng bộ lọc này cho thuộc tính slope, ta mặc định là flat cho missing values



Hình 22: Sau khi chạy thì slope không còn missing values

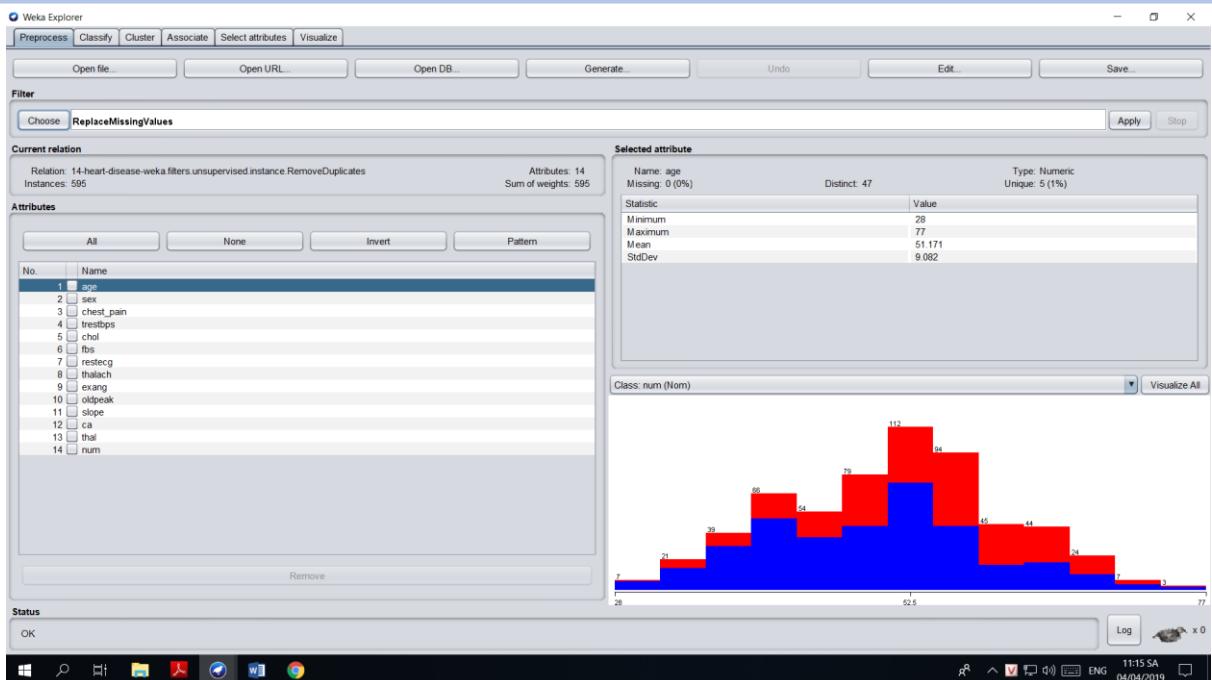
b) **Dữ liệu nhiễu (Noisy data):** Liệt kê các phương pháp đã học để loại bỏ các dữ liệu nhiễu, Weka đã cài đặt những phương pháp nào?

STT	TÊN PHƯƠNG PHÁP	MỤC ĐÍCH
1	Binning	Phương pháp này làm trộn dữ liệu đã được sắp xếp bằng cách dựa vào các giá trị xung quanh. Do đó, phương pháp này là làm trộn cục bộ
2	Regression	Data có thể được làm trộn bằng cách xây dựng một hàm hồi quy cho nó. Hồi quy tuyến tính tìm đường thẳng tốt nhất để biểu diễn 2 attribute, do đó, một thuộc tính sẽ được suy ra từ thuộc tính còn lại. <b>Multiple linear regression</b> là phiên bản mở rộng của <b>linear regression</b> , có nhiều hơn 2 attribute tham gia và dữ liệu được biểu diễn bằng một mặt phẳng đa chiều
3	Clustering	Điểm <b>outlier</b> có thể được phát hiện bằng phương pháp phân cụm. Những giá trị nào mà nằm ngoài tất cả các cụm thì được xem là <b>outlier</b>

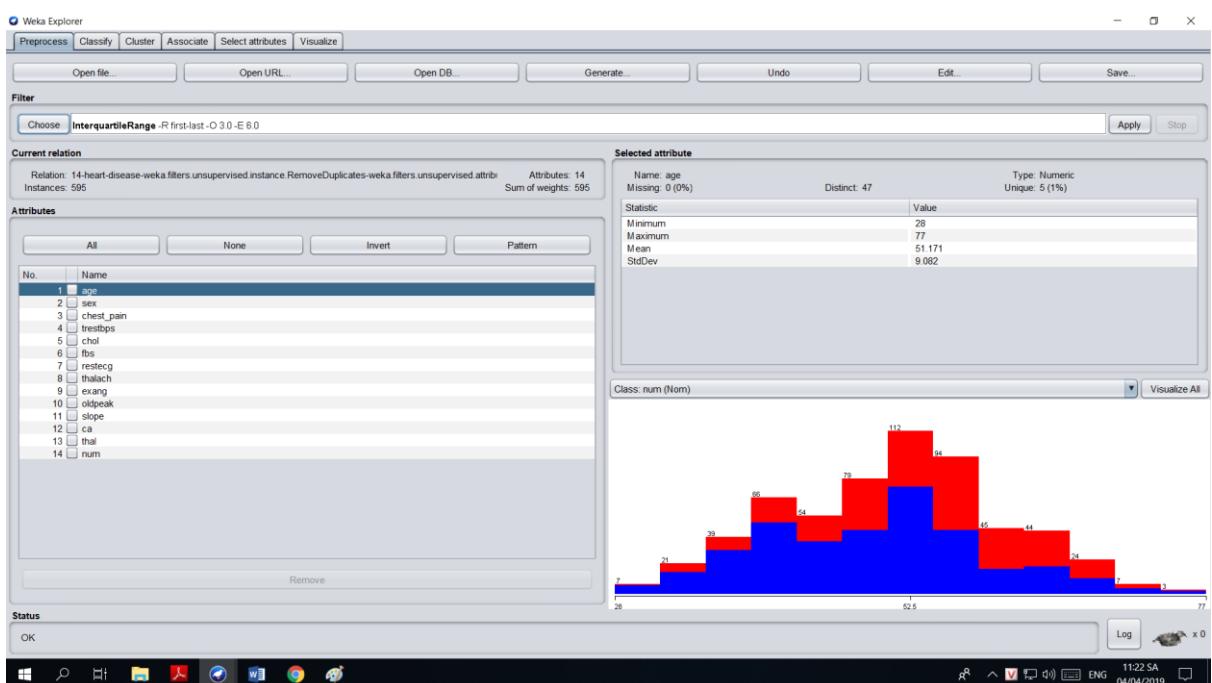
c) **Dò tìm dữ liệu tạp (Outlier detection):** Liệt kê các phương pháp đã học để dò tìm dữ liệu tạp. Bạn dò tìm dữ liệu tạp bằng Weka như thế nào? Có dữ liệu tạp trong dataset đã cho hay không? Nếu có, liệt kê một số dữ liệu tạp.

Các phương pháp dò tìm dữ liệu tạp là **phát hiện dựa vào phân bố xác suất** (Statistical Distribution-Based Outlier Detection), **phát hiện dựa vào khoảng cách** (Distance-Based Outlier Detection), **phát hiện outlier cục bộ dựa vào mật độ** (Density-Based Local Outlier Detection), **dựa vào độ sai lệch** (Deviation-Based Outlier Detection)

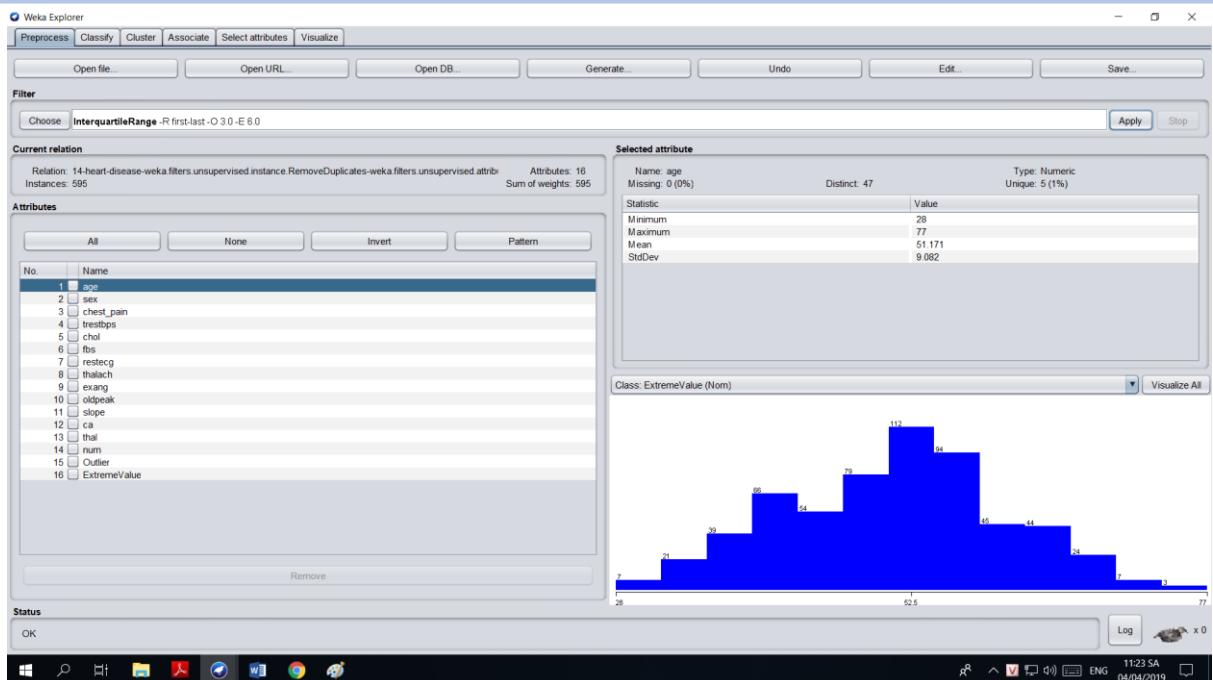
Em dò tìm dữ liệu bằng cách **ReplaceMissingValues** (thay thế các missing values) sau đó sử dụng **InterquartileRange** để tìm ra **Outliers** và **ExtremeValues**. Sau đó, em dùng **RemoveWithValues** để loại bỏ các dòng dữ liệu dựa vào thuộc tính của dòng đó.



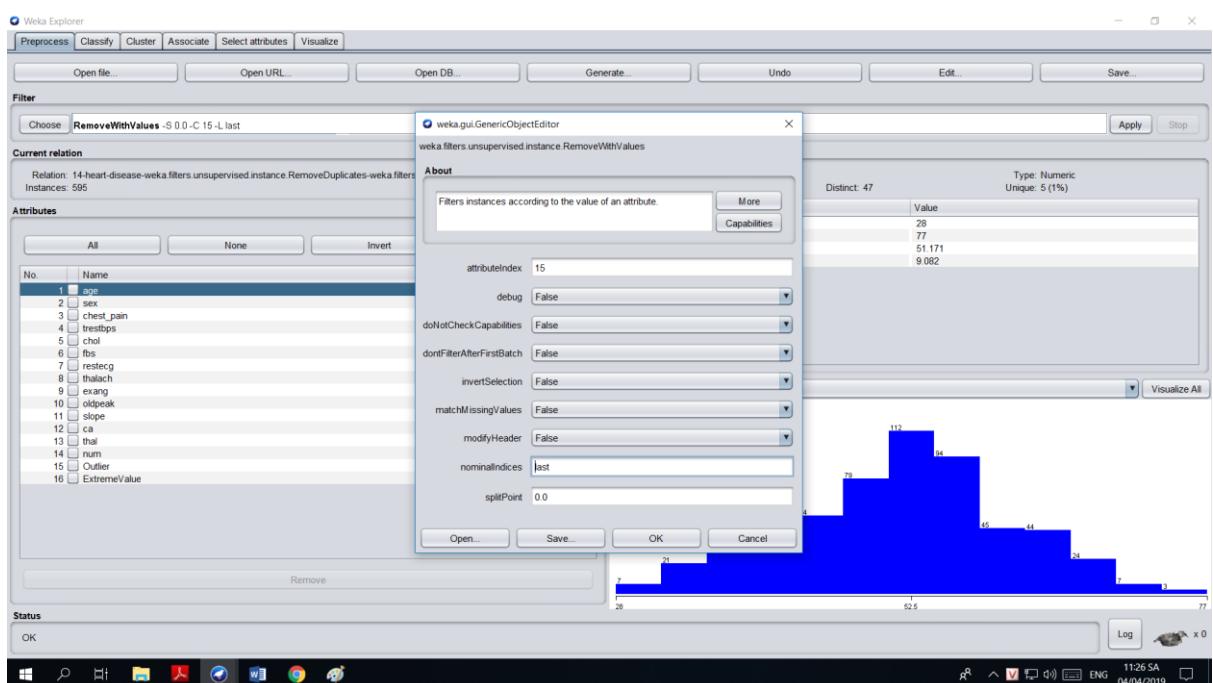
Hình 23: Sử dụng filter ReplaceMissingValues



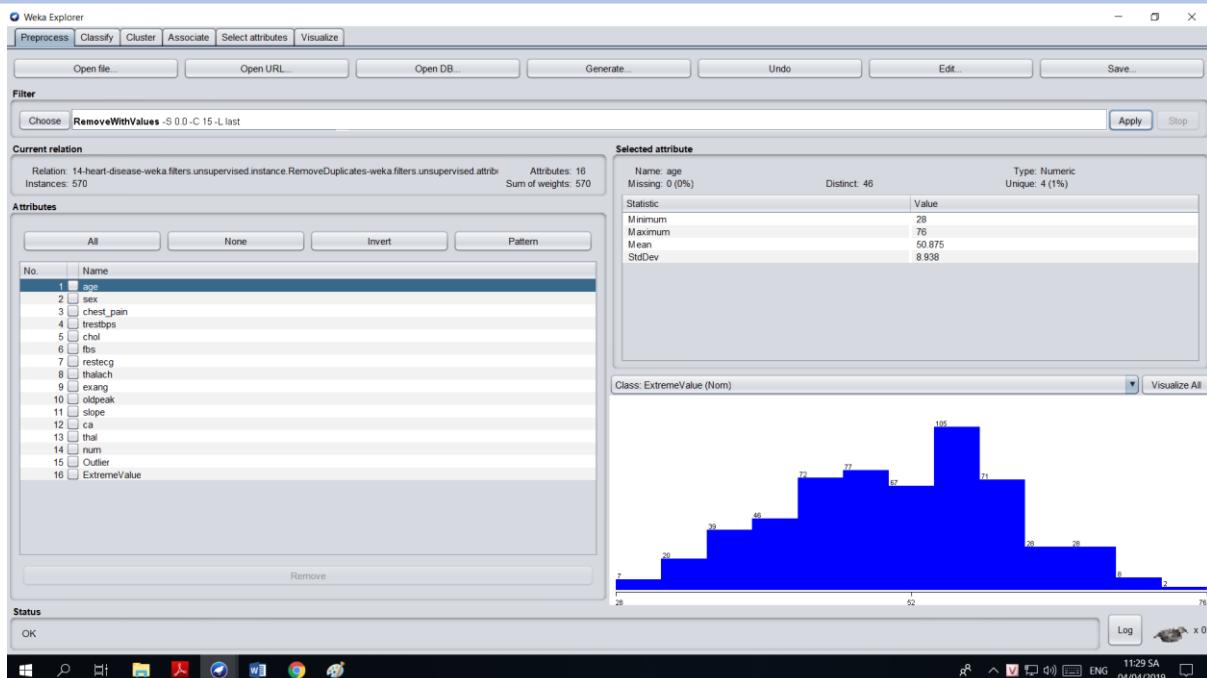
Hình 24: Sử dụng filter InterquartileRange



Hình 25: Tạo ra 2 attribute là Outlier và ExtremeValue

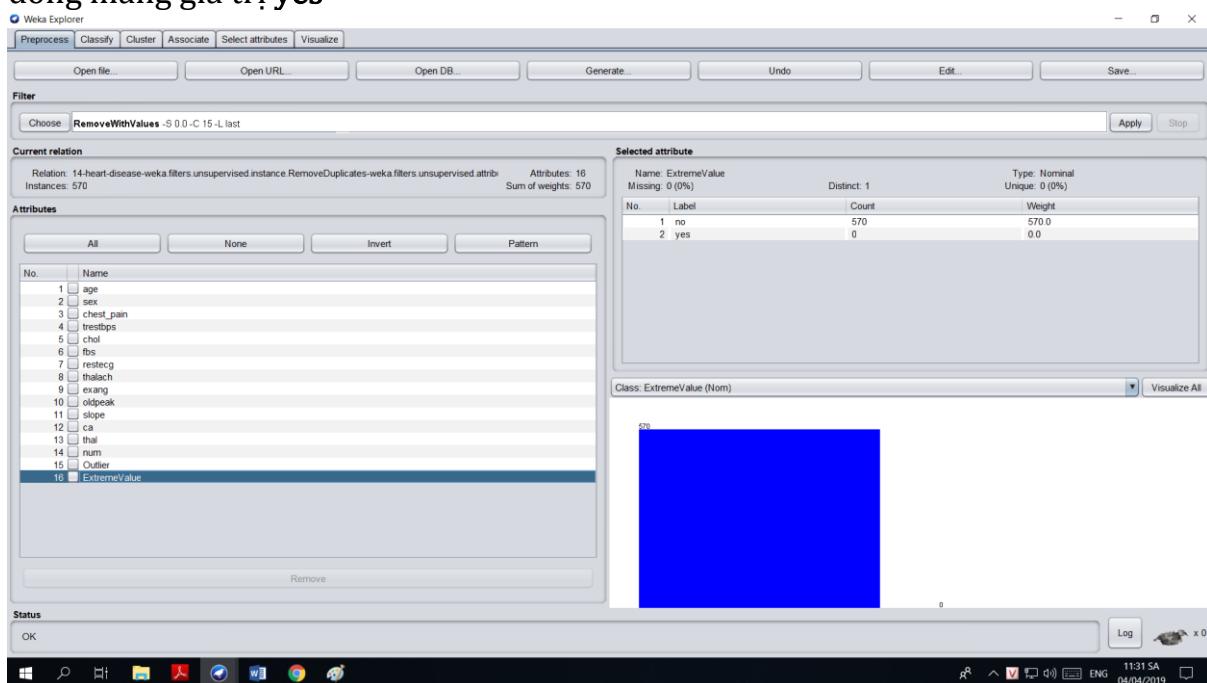


Hình 26: Sử dụng filter RemoveWithValues với tham số là 15 và last



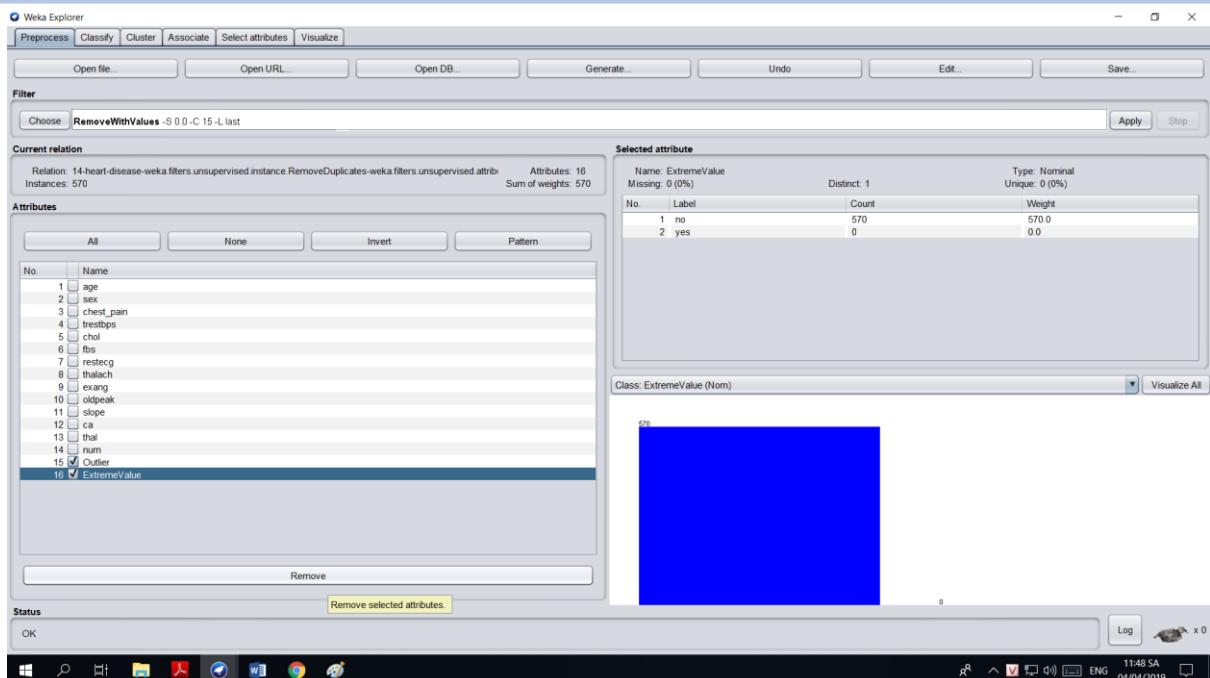
Hình 27: Từ 595 instances chúng ta còn 570

Vì **ExtremeValue** không có instances mang giá trị yes nên ta không cần xóa các dòng mang giá trị yes



Hình 28: Thuộc tính ExtremeValue

Ta sẽ xóa 2 thuộc tính **Outlier** và **ExtremeValue** khỏi dataset và lưu lại thành file **heart-cleaned.arff**



Hình 29: Nhấn Remove để loại bỏ 2 thuộc tính này và Save

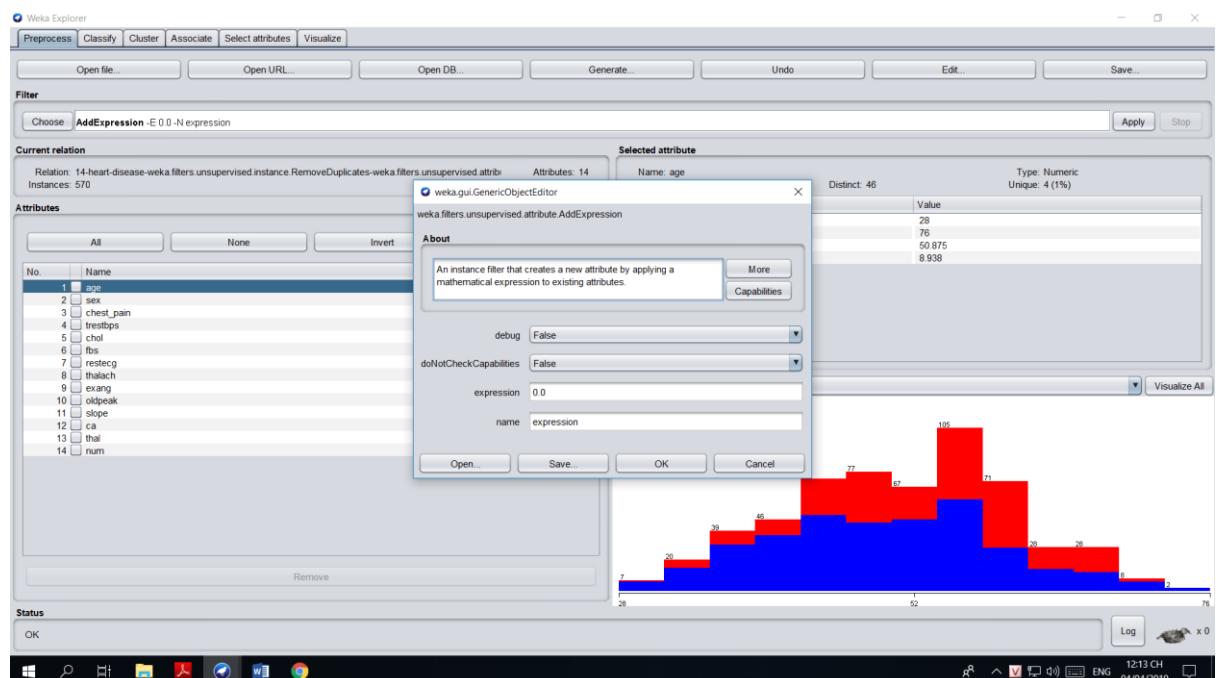
d) Lưu dataset đã làm sạch vào file **heart-cleaned.arff** và dán vào bài làm 1 ảnh chụp cho thấy ít nhất 10 dòng của dữ liệu với tất cả các cột.

Hình 30: Dữ liệu sau khi làm sạch

## 5. CHUẨN BI DỮ LIỆU – CHUYỂN ĐỔI DỮ LIỆU

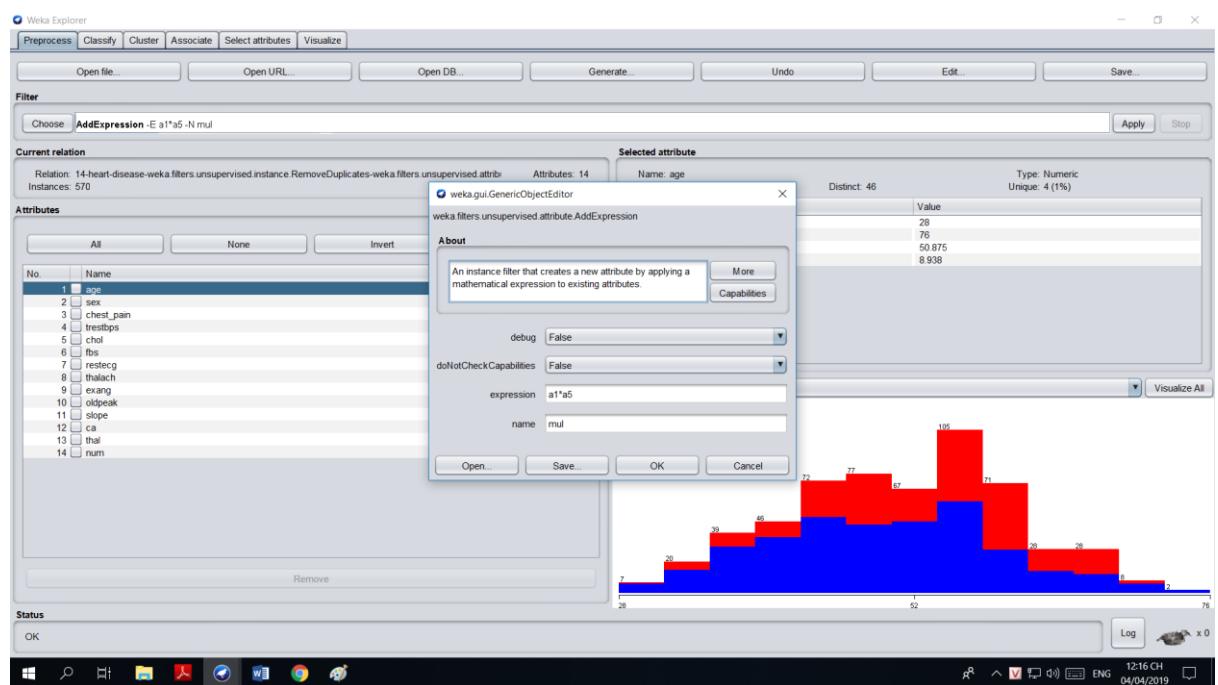
a) **Xây dựng thuộc tính – Attribute construction:** ví dụ, thêm một thuộc tính là tổng của 2 thuộc tính khác. Bô lọc nào của Weka cho phép làm điều này?

Trong Weka, chúng ta sẽ sử dụng bộ lọc **AddExpression** để xây dựng thuộc tính (**Attribute Constructor**)

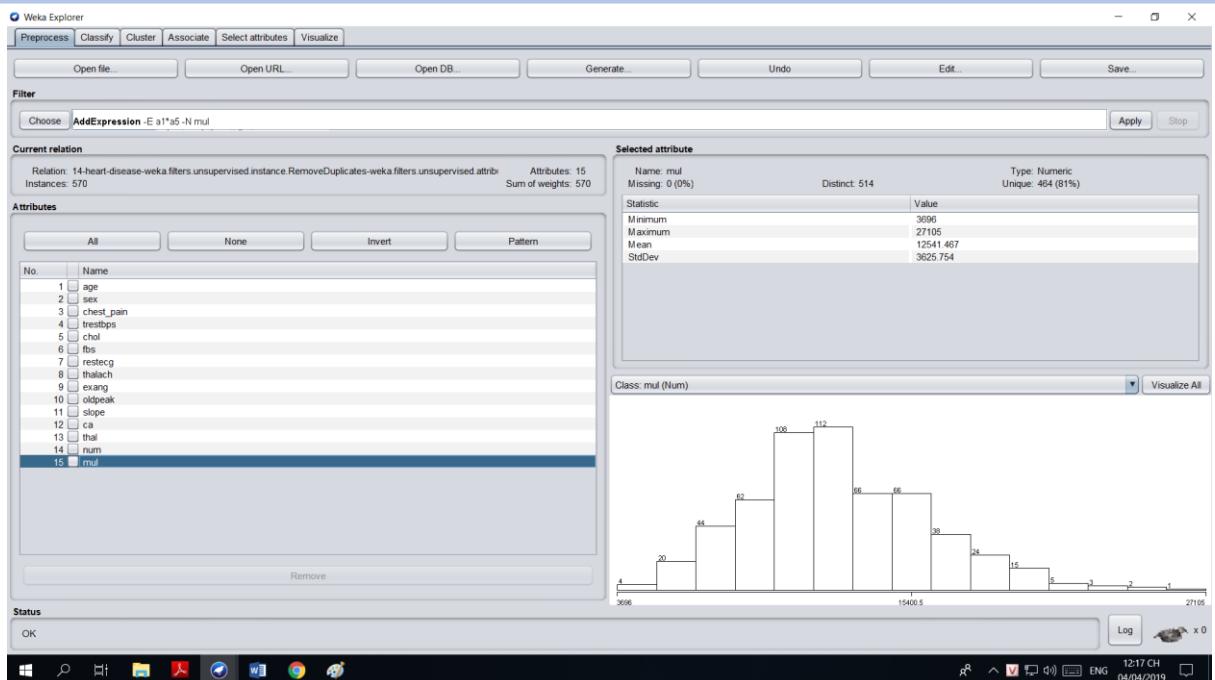


Hình 31: Bộ lọc AddExpression

Ta sẽ gõ công thức vào mục **expression** với attribute có tiền tố là “a” và chỉ số của nó trong dataset



Hình 32: Tạo thuộc tính mul bằng cách nhân a1 và a5



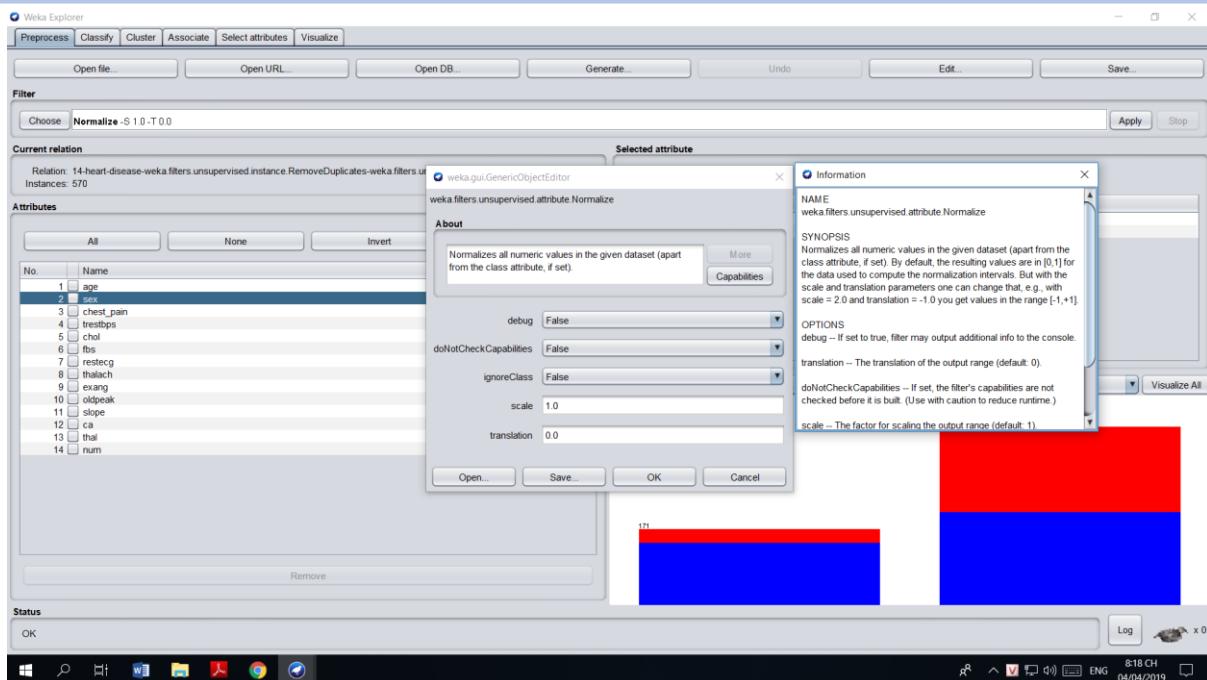
Hình 33: Attribute mới được tạo ra có tên là mul

- b) Chuẩn hóa – Normalize một thuộc tính. Bộ lọc nào của Weka cho phép làm điều này? Bộ lọc đó có thể **chuẩn hóa Min-max không**, **chuẩn hóa Z-score** hay **chuẩn hóa thập phân** hay không? Cho biết cụ thể cách thức thực hiện những chuẩn hóa này trong Weka.

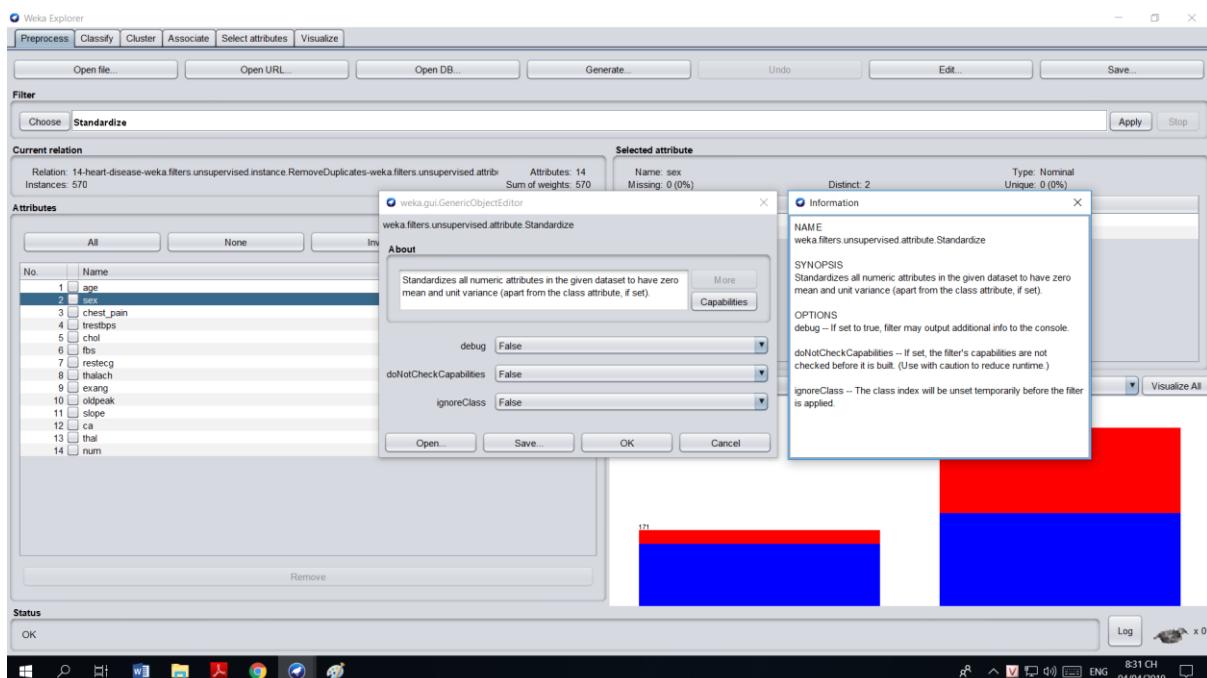
**Chuẩn hóa (Normalize)** một thuộc tính là chúng ta biến đổi bằng cách **scailing** dữ liệu của chúng sao cho giá trị nằm trong một khoảng nhỏ cụ thể như là [0.0 ; 1.0]. Có rất nhiều cách chuẩn hóa nhưng ta quan tâm nhất 3 loại sau:

STT	TÊN PHƯƠNG PHÁP	MỤC ĐÍCH
1	Min-max normalization	<p>Sử dụng phép biến đổi tuyến tính dữ liệu gốc. Giả sử, chúng ta có thuộc tính <b>A</b> với giá trị nằm trong khoảng <math>[min_A; max_A]</math>. Ta muốn chuẩn hóa một giá trị <b>v</b> của <b>A</b> thành <b>v'</b> thuộc đoạn <math>[new\_min_A; new\_max_A]</math> bằng cách sau:</p> $v' = \frac{v - min_A}{max_A - min_A} (new\_max_A - new\_min_A) + new\_min_A$ <p>Phương pháp này bảo toàn mối quan hệ giữa các giá trị trong dữ liệu gốc. Tuy nhiên, phương pháp này gặp lỗi cho các dữ liệu trong tương lai mà nằm ngoài khoảng giá trị của <b>A</b></p>
2	z-score normalization	<p>Giá trị của thuộc tính <b>A</b> được chuẩn hóa dựa vào trung bình (mean) và độ lệch chuẩn (standard deviation) của <b>A</b>. Giá trị <b>v</b> của <b>A</b> được chuẩn hóa thành <b>v'</b> bằng cách sau: <math>v' = \frac{v - \bar{A}}{\sigma_A}</math>, với <math>\bar{A}</math> và <math>\sigma_A</math> lần lượt là trung bình và độ lệch chuẩn của <b>A</b>. Phương pháp này tốt khi mà giá trị <b>min</b> và <b>max</b> của <b>A</b> là không xác định được hoặc dữ liệu nhiễu gây sai cho phương pháp min-max normalization</p>
3	Chuẩn hóa thập phân (Normalization by decimal scaling)	<p>Chuẩn hóa bằng cách di chuyển chấm thập phân của các giá trị thuộc tính <b>A</b>. Số chấm thập phân di chuyển sẽ phụ thuộc vào giá trị tuyệt đối lớn nhất của <b>A</b>. Giá trị <b>v</b> của <b>A</b> được chuẩn hóa thành <b>v'</b> bằng cách sau: <math>v' = \frac{v}{10^j}</math>, với <b>j</b> là số nguyên nhỏ nhất để <math>\text{Max}( v' ) &lt; 1</math></p>

Trong Weka, chúng ta được cung cấp 2 bộ lọc để chuẩn hóa là **min-max normalization** và **z-score normalization**



Hình 34: Min-max normalization



Hình 35: z-score normalization

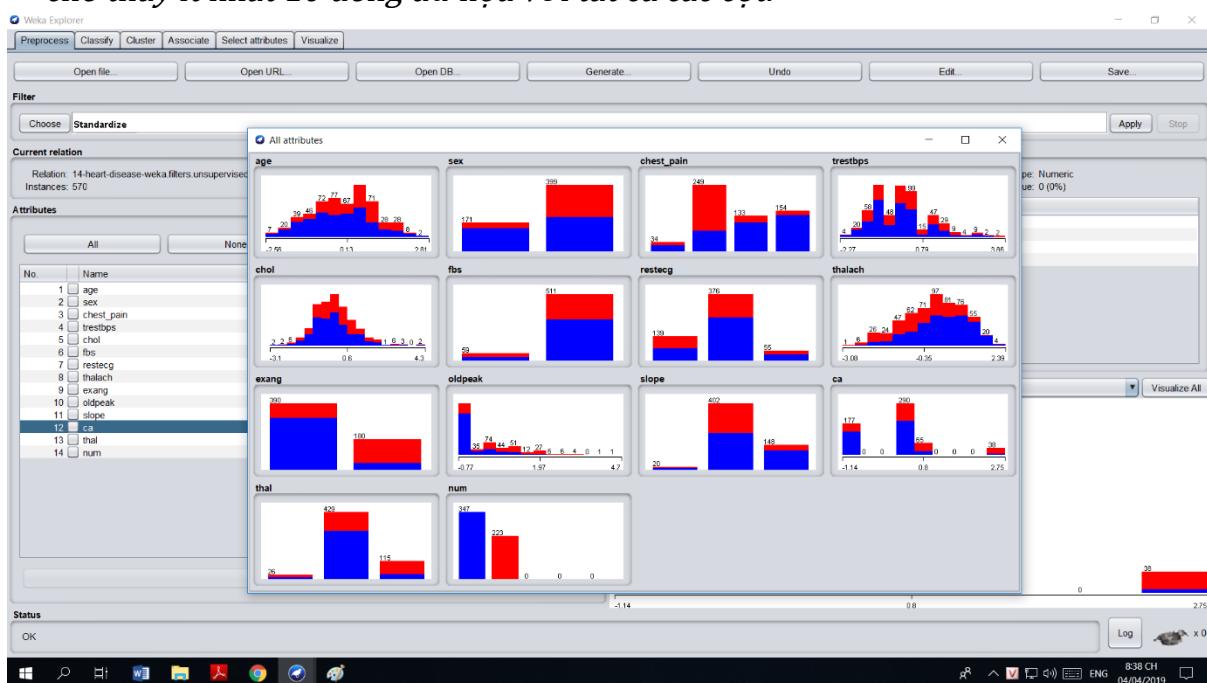
c) Chọn 1 phương pháp và tiến hành chuẩn hóa tất cả các thuộc tính là số thực, giải thích sự lựa chọn của bạn.

Ta chọn phương pháp z-score normalization để chuẩn hóa vì các thuộc tính là số thực trong dataset đa phần có phân phối giống với Gauss trừ cao nên sử dụng phương pháp này sẽ phù hợp hơn



Hình 36: Phân phối của thuộc tính số có phân phối giống Gauss

d) Lưu dataset đã chuẩn hóa vào file **heart-normal.arff** và chụp ảnh màn hình cho thấy ít nhất 10 dòng dữ liệu với tất cả các cột.



Hình 37: Kết quả sau khi dùng chuẩn hóa z-score

```

C:\Users\nhat_huy\Desktop\1612272\baitap\heart-normal.arff - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES
  heart-normal.arff
FOLDERS
  baitap
  heart-normal.arff
1 @relation heart-normal
2
3 @attribute age numeric
4 @attribute sex {female,male}
5 @attribute chest_pain {typ_angina,asympt,non_anginal,atyp_angina}
6 @attribute trestbps numeric
7 @attribute chol numeric
8 @attribute fbs {1,0}
9 @attribute restecg {left_vent_hyper,normal,st_t_wave_abnormality}
10 @attribute thalach numeric
11 @attribute exang {no,yes}
12 @attribute oldpeak numeric
13 @attribute slope {down,flat,up}
14 @attribute ca numeric
15 @attribute thal {fixed_defect,normal,reversible_defect}
16 @attribute num {<50,>50_1,>50_2,>50_3,>50_4}
17
18 @data
19 1.356578, male, typ_angina, 0.738078, -0.239508, t, left_vent_hyper, 0.216111, no, 1.478228, down, -1.14411, fixed_defect, <50
20 1.3804126, male, asympt, -0.679795, -0.316777, f, left_vent_hyper, -0.660313, yes, 1.770929, flat, 2.753299, reversible_defect, >50_1
21 -1.552479, male, non_anginal, -0.112646, 0.088885, , normal, 1.766288, no, 2.649633, down, -1.14411, normal, <50
22 -1.104991, female, atyp_angina, -0.112646, -0.799707, f, left_vent_hyper, 1.134271, no, 0.680123, up, 1.14411, normal, <50
23 0.573371, male, atyp_angina, -0.679795, -0.181556, f, normal, 1.384678, no, 0.14721, up, -1.14411, normal, <50
24 1.244692, female, asympt, 0.454503, 0.436594, f, left_vent_hyper, 0.633457, no, 2.7466, down, 2.753299, normal, >50_1
25 0.688528, female, asympt, -0.679795, 2.097873, f, normal, 0.75866, yes, -0.180414, up, -1.14411, normal, <50
26 1.356578, male, asympt, -0.112646, 0.166153, f, left_vent_hyper, 0.090908, no, 0.600123, flat, 0.804595, reversible_defect, >50_1
27 0.23771, male, asympt, 0.454503, -0.819024, t, left_vent_hyper, 0.424784, yes, 2.258765, down, -1.14411, reversible_defect, >50_1
28 0.688528, male, asympt, 0.454503, -1.031513, f, normal, 0.132642, no, 0.375548, flat, 1.14411, fixed_defect, <50
29 0.573371, female, atyp_angina, 0.454503, 0.938841, f, left_vent_hyper, 0.341315, no, 0.502556, flat, -1.14411, normal, <50
30 0.573371, male, non_anginal, -0.112646, 0.204788, t, left_vent_hyper, -0.117765, yes, -0.180414, flat, 0.804595, fixed_defect, >50_1
31 -0.769271, male, atyp_angina, -0.679795, 0.340008, f, normal, 1.176005, no, -0.765816, up, -1.14411, reversible_defect, <50
32 0.125824, male, non_anginal, 2.26938, -0.896293, t, normal, 0.716926, no, -0.277981, up, -1.14411, reversible_defect, <50
33 0.688528, male, non_anginal, 1.021652, 1.495126, f, normal, 1.21774, no, 0.795258, up, -1.14411, normal, <50
34 -0.321724, male, atyp_angina, -1.246944, 0.316777, f, normal, 0.967333, no, 0.269855, down, -1.14411, reversible_defect, >50_1
35 0.349597, male, asympt, 0.454503, -0.123605, f, normal, 0.633457, no, 0.404989, up, -1.14411, normal, <50
36 -0.321724, female, non_anginal, -0.112646, 0.571815, f, normal, -0.242968, no, -0.570682, up, -1.14411, normal, <50
37 -0.209837, male, atyp_angina, 0.112646, 0.39796, f, normal, 1.092536, no, -0.180414, up, -1.14411, normal, <50
38 1.468465, male, typ_angina, -1.246944, -0.664498, f, left_vent_hyper, -0.034296, yes, 0.990392, flat, 1.14411, normal, <50

```

Hình 38: heart-normal.arff

## 6. CHUẨN BỊ DỮ LIỆU – RÚT GỌN DỮ LIỆU

Các cơ sở dữ liệu thường rất lớn, không thể thao tác trực tiếp được. Các kỹ thuật rút gọn dữ liệu được áp dụng để tiền xử lý dữ liệu. Trong tab **Preprocess**, bên cạnh việc chọn lọc thuộc tính, một phương pháp để rút gọn dữ liệu là **chọc lọc** các dòng trong một dataset, hay còn gọi là **lấy mẫu (sampling)**. Làm cách nào để lấy mẫu với các bộ lọc của Weka? Nó có thể thực hiện 2 phương pháp chính là: **Simple Random Sample Without Replacement**, và **Simple Random Sample With Replacement** hay không?

**Rút gọn dữ liệu (Data reduction)** là kỹ thuật được áp dụng để tạo ra một dataset với cách trình bày giản lược và nhỏ hơn rất nhiều so với dữ liệu gốc nhưng vẫn giữ được sự toàn vẹn của dữ liệu gốc. Khi đó, làm việc trên dữ liệu giản lược sẽ đơn giản hơn rất nhiều. Các chiến lược cho rút gọn dữ liệu gồm:

- Data cube aggregation
- Chọn lọc thuộc tính (Attribute subset selection)
  - ✓ Stepwise forward selection
  - ✓ Stepwise backward elimination
  - ✓ Combination of forward selection and backward elimination
  - ✓ Cây quyết định quy nạp (Decision tree induction)
- Giảm số chiều (Dimensionality reduction)
  - ✓ Biến đổi Wavelet (Wavelet transforms)
  - ✓ Principal Components Analysis
- Giảm số lượng (Numerosity reduction)
  - ✓ Regression and log linear models

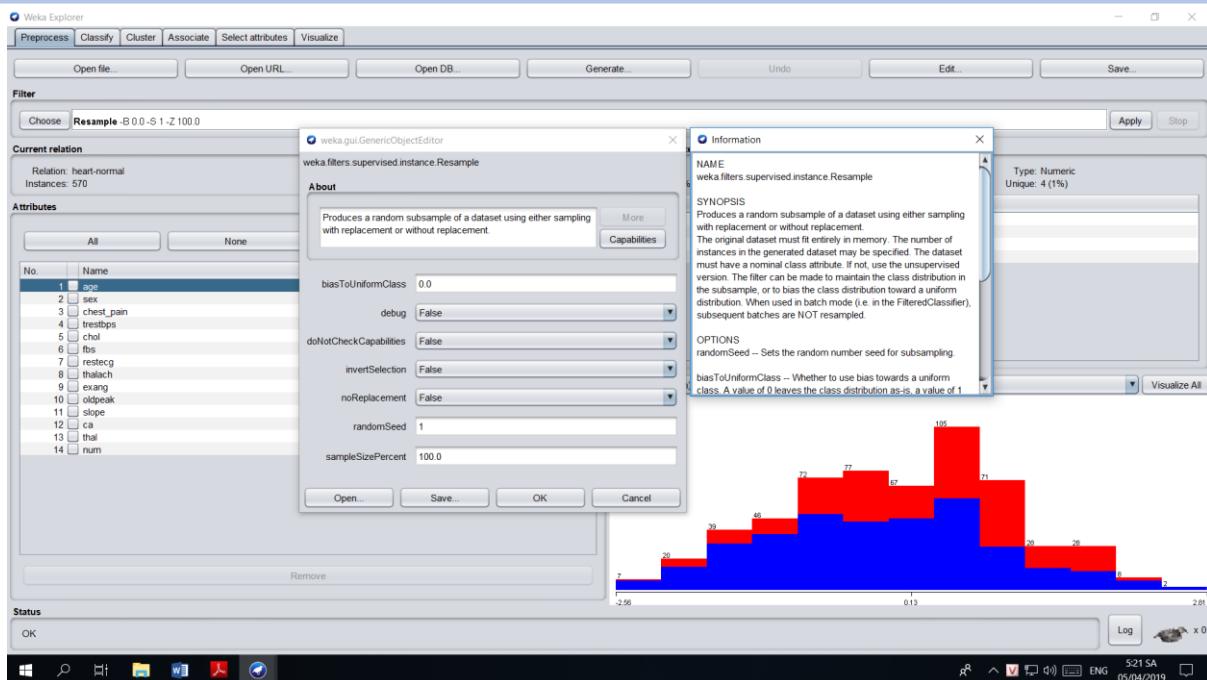
- ✓ Histograms
- ✓ Clustering
- ✓ Sampling
- Discretization and concept hierarchy generation

**Lấy mẫu (Sampling)** là một phương pháp thuộc nhóm **Numerosity reduction**.

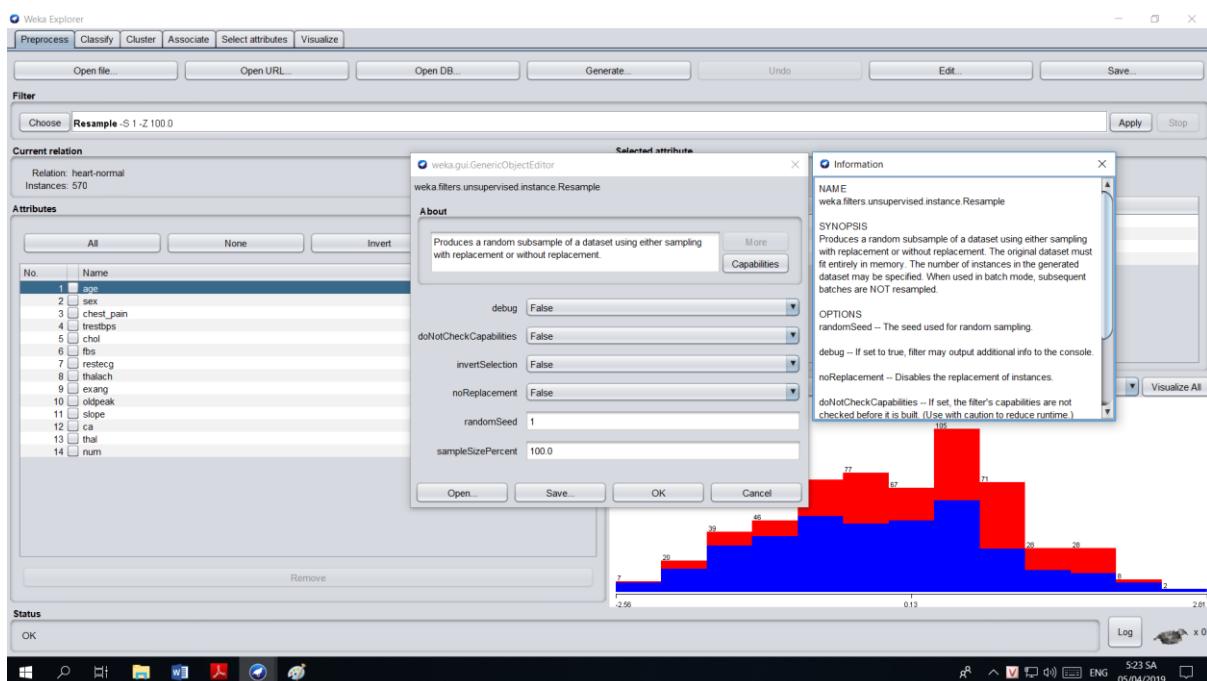
Lấy mẫu được sử dụng vì nó cho phép một dataset lớn được biểu diễn bởi một mẫu ngẫu nhiên nhỏ hơn dataset gốc.

STT	TÊN PHƯƠNG PHÁP	MỤC ĐÍCH
1	Lấy mẫu ngẫu nhiên không hoàn lại (Simple random sample without replacement)	Lấy mẫu bằng cách lấy ngẫu nhiên $s$ dòng dữ liệu trong dataset, với xác suất lấy mỗi dòng là như nhau và bằng $\frac{1}{N}$ với $N$ là số dòng dữ liệu trong dataset
2	Lấy mẫu ngẫu nhiên có hoàn lại (Simple random sample with replacement)	Lấy ngẫu nhiên tuy nhiên sau mỗi lần lấy 1 dòng dữ liệu thì dòng đó không bị loại bỏ khỏi dataset vào lần lấy sau. Do đó, chúng ta sẽ có thể lấy trùng dòng dữ liệu này một cách ngẫu nhiên vào các lần lấy kế tiếp
3	Cluster sample	Nếu các dữ liệu trong dataset được nhóm thành $M$ cluster riêng lẻ nhau thì chúng ta có thể thực hiện phương pháp lấy ngẫu nhiên $s$ cluster với $s < M$
4	Mẫu phân tầng (Stratified sample)	Nếu dataset được chia làm các phần riêng lẻ nhau gọi là <b>tầng (strata)</b> thì chúng ta có thể tạo ra một mẫu phân tầng bằng cách lấy mẫu ngẫu nhiên tại mỗi tầng

Chúng ta có thể lấy mẫu bằng cách sử dụng **bộ lọc Resample**. Có 2 bộ lọc là **bộ lọc Resample** trong nhóm **Unsupervised** và **Supervised**



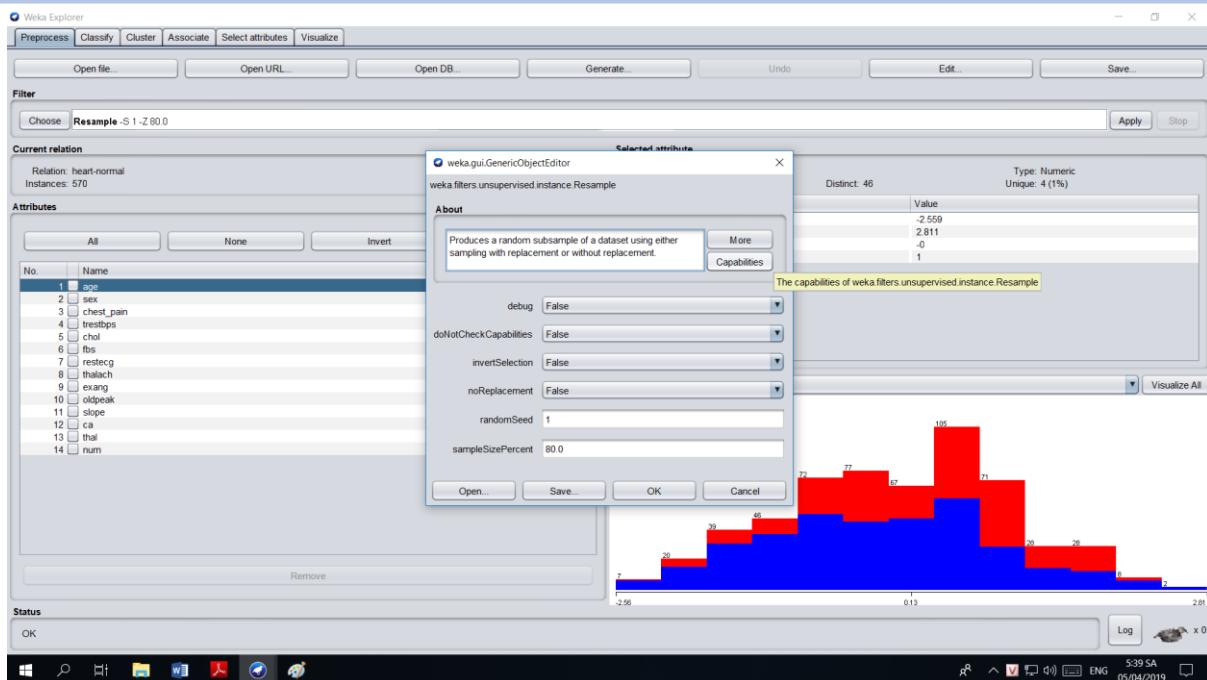
Hình 39: Bộ lọc Resample trong nhóm Supervised



Hình 40: Bộ lọc Resample trong nhóm Unsupervised

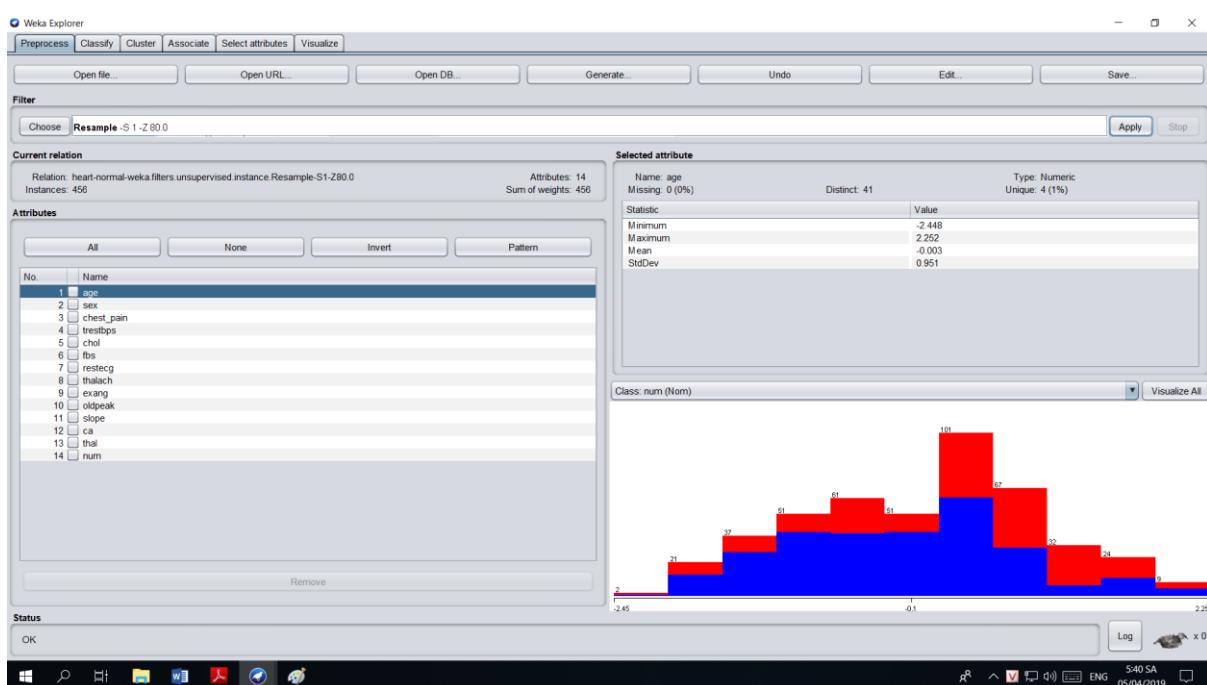
**Bộ lọc Resample** trong **Supervised** chỉ làm việc với **nominal class** hoặc **binary class** và cả 2 bộ lọc này đều cho tùy chọn **Simple random sample with replacement** và **Simple random sample without replacement**

Ta sẽ làm thử với bộ lọc Resample trong **Supervised**



Hình 41: Ta thiết lập lấy 80% instances trong sampleSizePercent

Từ 570 instances chúng ta còn 456 instances



Hình 42: Kết quả sau khi lấy mẫu

Lưu lại thành file **heart-sample.arff**

```

C:\Users\nhat huy\Desktop\161227\baitap\heart-sample.arff (baitap) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES
  heart-sample.arff
FOLDERS
  baitap
@relation heart-sample
1 @attribute age numeric
2 @attribute sex {female, male}
3 @attribute chest_pain {typ_angina, asympt, non_anginal, atyp_angina}
4 @attribute trestbps numeric
5 @attribute chol numeric
6 @attribute fbs {<=1}
7 @attribute restecg {left_vent_hyper, normal, st_t_wave_abnormality}
8 @attribute thalach numeric
9 @attribute exang {no, yes}
10 @attribute oldpeak numeric
11 @attribute slope {down, flat, up}
12 @attribute ca numeric
13 @attribute thal {fixed_defect, normal, reversible_defect}
14 @attribute num {<=50, >50_1, >50_2, >50_3, >50_4}
15
16
17
18 @data
19 0.125824, female, non_anginal, -0.396221, 0.513863, f, normal, -0.242968, no, -0.765816, flat, 0.157184, normal, <50
20 0.349597, male, asympt, -1.246944, -0.761072, f, left_vent_hyper, -1.536738, yes, -0.765816, flat, 0.804595, normal, >50_1
21 0.013937, female, non_anginal, -1.246944, -1.070147, f, normal, 1.035924, no, -0.765816, flat, 0.157184, normal, <50
22 -0.545497, male, asympt, -1.246944, -0.142922, f, st_t_wave_abnormality, -0.291234, yes, 0.299855, flat, 0.157184, normal, <50
23 0.013937, female, asympt, 1.588801, 1.112696, f, normal, 0.216111, yes, 0.299855, flat, 0.157184, normal, >50_1
24 0.349597, male, asympt, 0.566365, 0.784304, f, left_vent_hyper, -1.202862, yes, 2.356332, flat, 2.753299, normal, >50_1
25 -1.328705, male, atyp_angina, -0.679795, 0.799707, f, normal, 0.007439, no, -0.765816, flat, 0.157184, normal, <50
26 -0.321724, male, asympt, 0.452936, 0.552497, f, left_vent_hyper, 0.883864, no, -0.277981, flat, -1.14411, reversible_defect, >50_1
27 -1.216818, male, asympt, -1.246944, -1.514443, f, left_vent_hyper, -1.286331, yes, 1.185526, flat, -1.14411, reversible_defect, >50_1
28 0.349597, male, asympt, -1.246944, -0.761072, f, left_vent_hyper, -1.536738, yes, -0.765816, flat, 0.804595, normal, >50_1
29 -0.769271, male, atyp_angina, -0.679795, 1.186051, f, normal, -0.117765, no, 0.299855, flat, 0.157184, normal, <50
30 0.349597, male, asympt, 3.857398, -0.91561, f, normal, -0.117765, yes, 1.185526, flat, 0.157184, normal, >50_1
31 1.916013, male, asympt, 0.681363, -1.012196, t, normal, -0.159499, no, 2.551466, flat, 2.753299, reversible_defect, >50_1
32 0.349597, female, atyp_angina, -0.112646, 0.146836, f, st_t_wave_abnormality, 0.424784, no, -0.765816, flat, 0.157184, normal, <50
33 1.580352, female, non_anginal, 1.588801, 2.213777, f, left_vent_hyper, 0.257846, no, 0.014721, up, -1.14411, normal, <50
34 -0.43361, male, non_anginal, 0.454503, -1.012196, f, normal, 0.007439, yes, 0.299855, flat, 0.157184, normal, >50_1
35 0.349597, male, non_anginal, 0.679795, 0.243422, f, left_vent_hyper, 0.090908, no, -0.375548, flat, -1.14411, reversible_defect, <50
36 0.349597, male, atyp_angina, 1.588801, 1.151331, f, normal, 1.259474, no, -0.765816, flat, 0.157184, normal, <50
37 -0.881158, female, non_anginal, 0.566365, 0.625852, f, normal, 0.842129, no, -0.570682, flat, -1.14411, normal, <50
38 -0.545497, female, non_anginal, 0.567933, -1.321271, f, left_vent_hyper, 0.633457, yes, 0.600123, down, -1.14411, normal, <50

```

Hình 43: Dữ liệu sau khi lấy mẫu