BỘ GIÁO DỤC VÀ ĐÀO TẠO TPHCM

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Khoa CNTT CLC

BÁO CÁO

Bộ môn: KHAI THÁC DỮ LIỆU VÀ ỨNG DỤNG

Đồ án 1: Tiền xử lý dữ liệu

| GIẢNG VIÊN HƯỚNG DẪN |

Nguyễn Khánh Toàn

Thành viên nhóm:

Nguyễn Ngọc Phước

THÀNH PHỐ HỒ CHÍ MINH - THÁNG 11 NĂM 2021

Em cam đoan đồ án này em tự xây dựng và nghiên cứu không sao chụp bất kỳ nhóm nào.

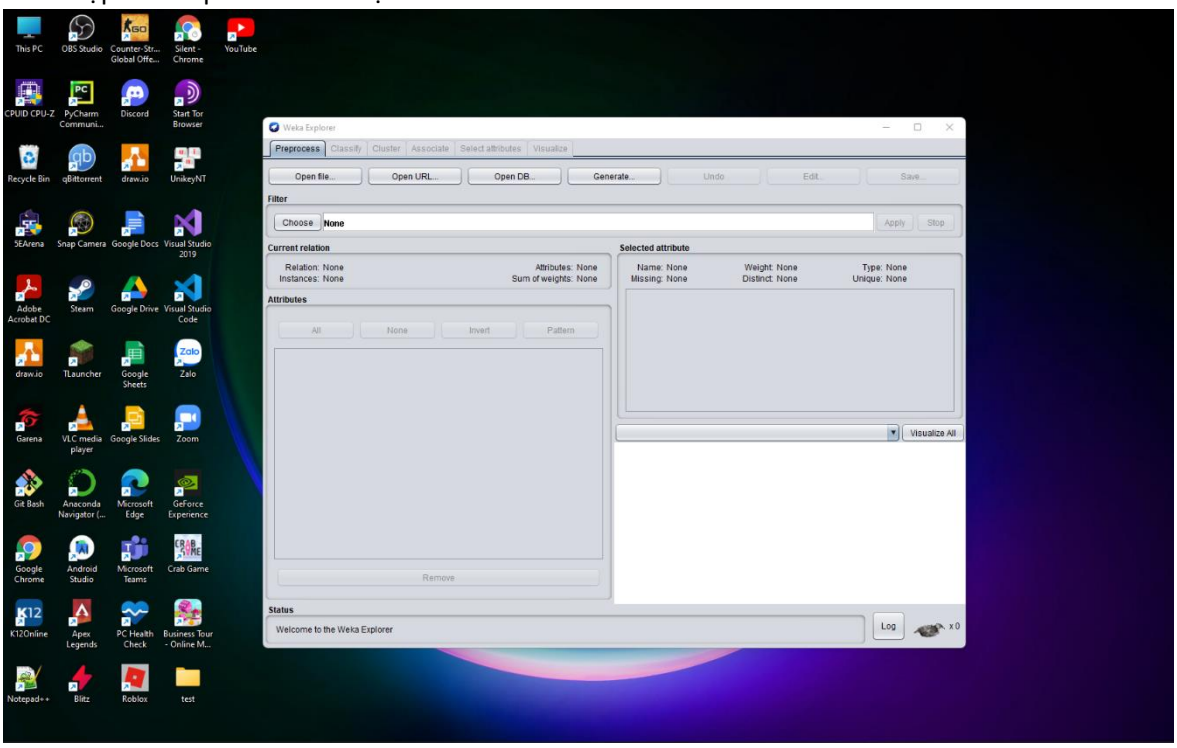
Mục lục

Table of Contents

Phần 1: Yêu cầu 1, cài đặt Weka

1.1 Ảnh chụp màn hình Weka sau khi cài đặt

- Ảnh chụp desktop sau khi cài đặt:



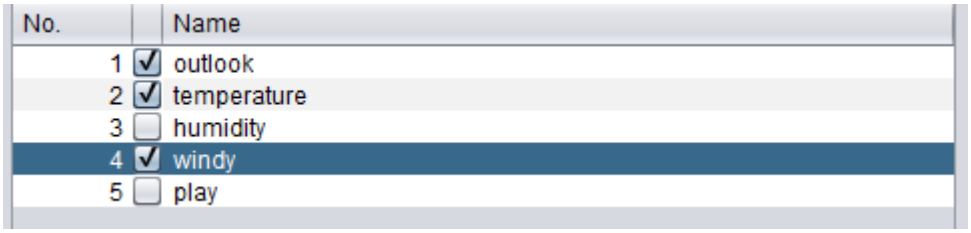
1.2 Ý nghĩa các nhóm điều khiển

1.2.1 Current relation

- Thể hiện thông tin chung về dataset hiện tại, gồm 4 field:
  - Relation: tên của quan hệ dữ liệu hiện tại (tên chung của tập dataset)
  - Instances: số mẫu của dataset hiện tại
  - Atributes: số thuộc tính của dataset hiện tại

1.2.2 Attributes

- Chọn thuộc tính để xem xét, ảnh hưởng đến nhóm hiển thị: selected atribute và visualize trong cùng nhóm này.
- Người dùng có thể chọn thuộc tính bằng cách tích 1 hoặc nhiều thuộc tính trong danh sách các thuộc tính:

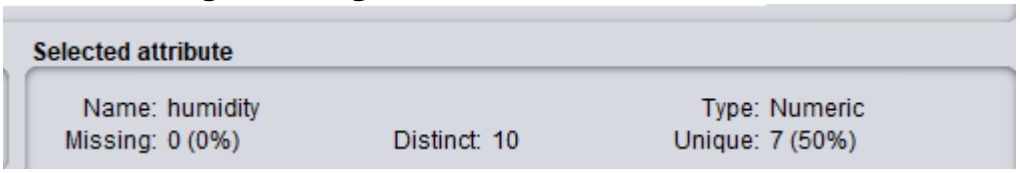


- Người dùng có thể sử dụng các nút điều khiển tổng quát để điều khiển việc chọn thuộc tính:

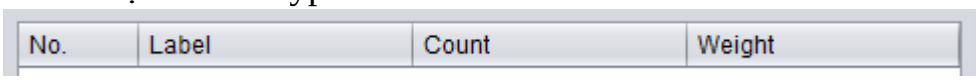
- All: chọn tất cả thuộc tính
- None: bỏ chọn tất cả thuộc tính
- Invert: bỏ chọn những thuộc tính đã chọn, chọn những thuộc tính chưa chọn và ngược lại
- Pattern: sử dụng cú pháp regex để tìm và chọn thuộc tinh

1.2.3 Selected attribute

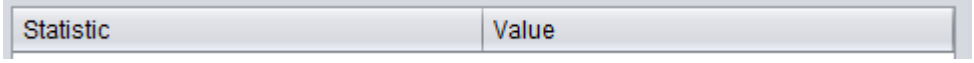
- Hiển thị thông tin chung về thuộc tính đã chọn:



- Gồm 4 field quan trọng:
  - Name: tên thuộc tính
  - Missing: số dữ liệu bị mất, thiếu
  - Distinct: số số giá trị riêng biệt của thuộc tính
  - Type: kiểu dữ liệu của thuộc tính
  - Unique: những giá trị độc nhất
  - Sum of weight: tổng trọng số của thuộc tính
- Bên cạnh đó còn có các dạng bảng biểu diễn:
  - Cho thuộc tính có type là norminal:



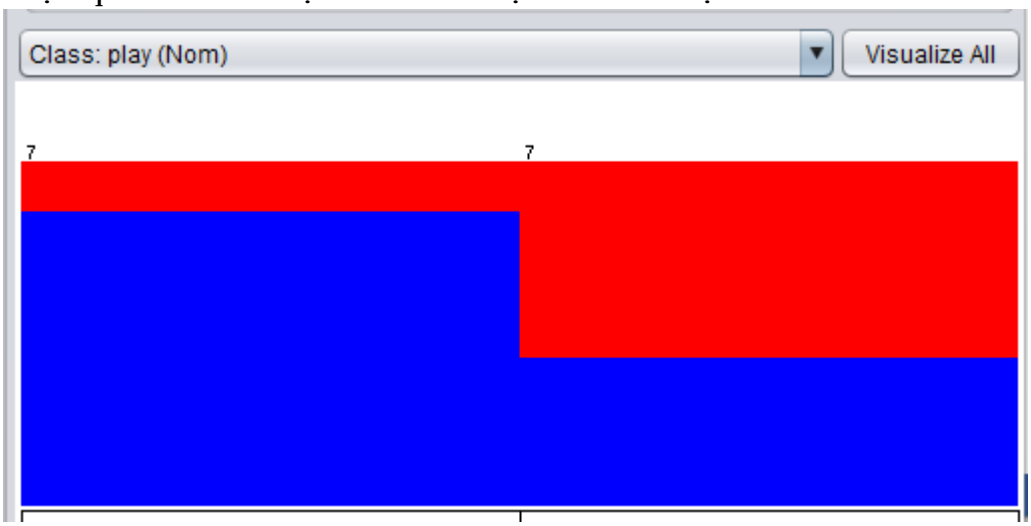
    - No: số thứ tự giá trị
    - Label: tên giá trị
    - Count: số giá trị trong thuộc tính
    - Weight: trọng số của giá trị
  - Cho thuộc tính có type là numeric



  - Statistic: các thông số thống kê như mean, max, min, standard deviation

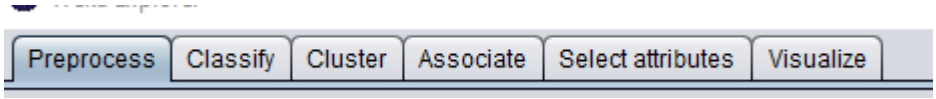
- Value: giá trị của các thông số kể trên

- Trực quan hóa dữ liệu đối với thuộc tính đã chọn:



- Người dùng có thể chọn dữ liệu được trực quan theo class, mỗi lựa chọn sẽ tương ứng với một thuộc tính, bên cạnh đó còn thêm class thuộc tính chính và class none (để thể hiện không chọn class nào)
- Phổ màu của từng class trên sẽ dựa trên thuộc tính được chọn làm class chính, đây cũng chính là thuộc tính phân loại
- Nút visualize all sẽ trực quan hóa tất cả các thuộc tính trên theo class

1.3 Giải thích sơ lược các tab



Preprocess: Chọn dataset và chỉnh sửa nó theo nhiều cách khác nhau

Classify: Huấn luyện các thuật toán học mang chức năng phân loại hoặc hồi quy và đánh giá chúng

Cluster: Học gom nhóm dữ liệu dựa trên dataset

Associate: học những luật kết hợp của data và đánh giá chúng

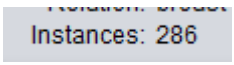
Select atributes: chọn những khía cạnh liên quan nhất của dataset

Visualize: Xem những thể hiện 2d khác nhau của data và tương tác với chúng

Phần 2: Làm quen với weka

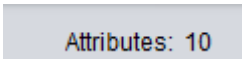
2.1 Đọc dữ liệu vào Weka

2.1.1 Tập dữ liệu có bao nhiêu mẫu (instances)?



Tập dữ liệu có 286 mẫu

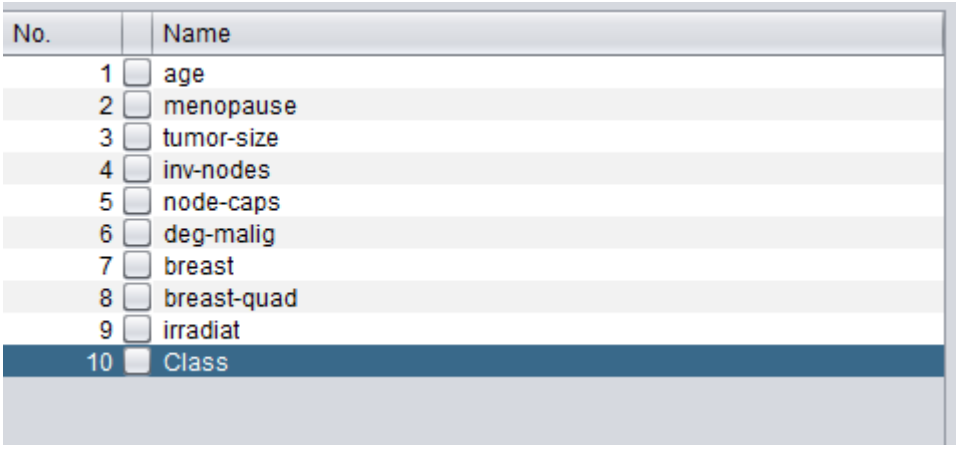
2.1.2 Tập dữ liệu có bao nhiêu thuộc tính (attributes)?



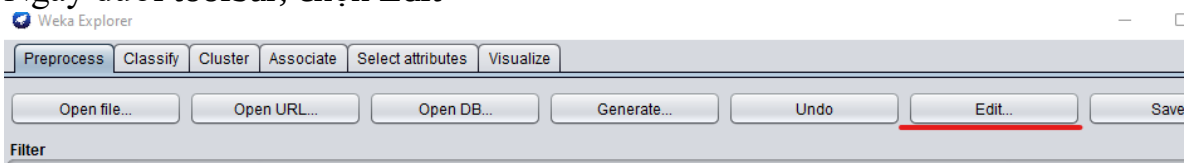
Tập dữ liệu có 10 thuộc tính

2.1.3 Thuộc tính nào được dùng làm lớp (class)? Có thể thay dổi thuộc tính dùng làm lớp hay không? Nếu có thì bằng cách nào?

- Thuộc tính được dùng làm lớp luôn là thuộc tính **cuối cùng** trong bảng attributes, cụ thể ở đây chính là thuộc tính "Class", thuộc tính số 10

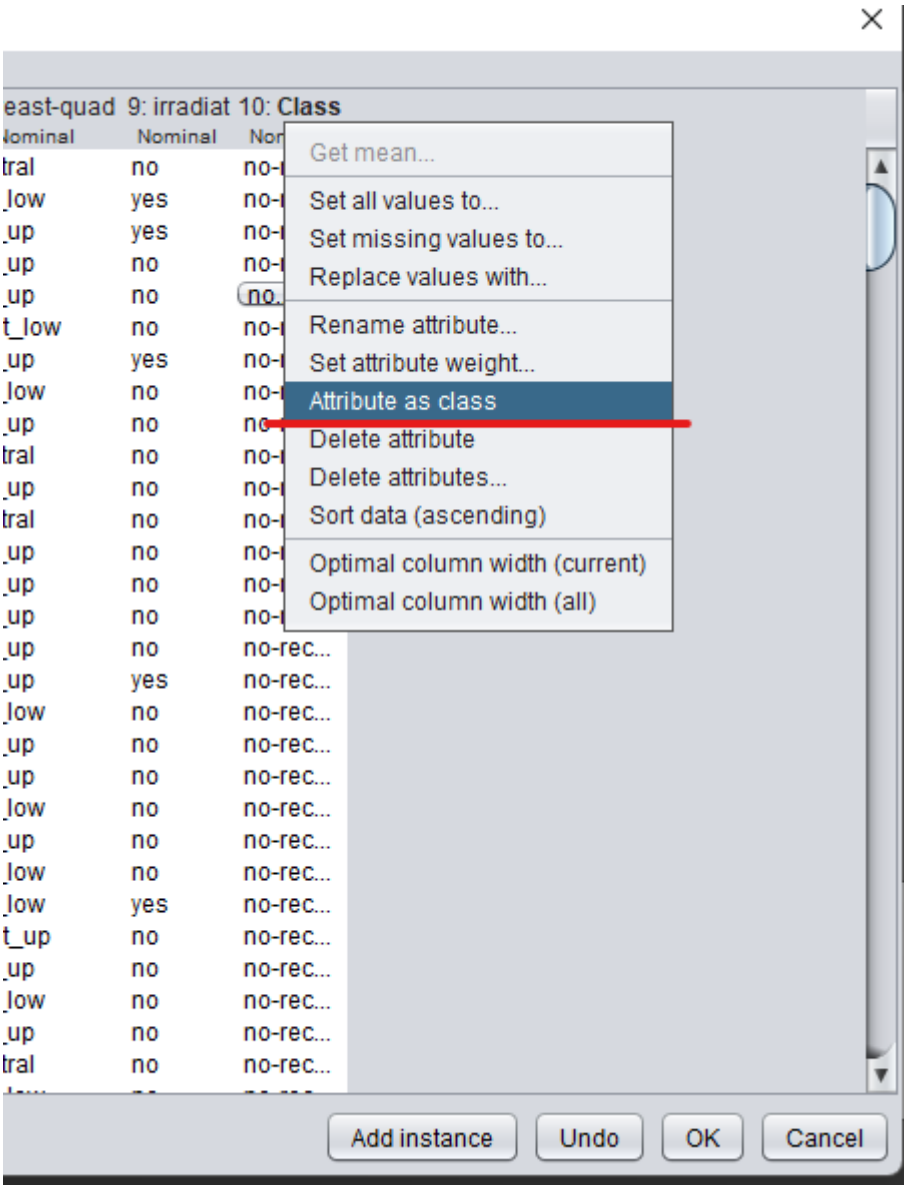


- Thuộc tính phân lớp này có thể được thay dổi tùy ý
- Cách thức thay đổi như sau:
  - Ngay dưới toolbar, chọn Edit



  - Lúc này màn hình sẽ hiển thị danh sách tất cả các thuộc tính, tuy ở đây chỉ có 10 thuộc tính, tương ứng 10 cột, nhưng trong trường hợp data có nhiều thuộc tính hơn, ta có thể kéo để chọn thuộc tính thích hợp
  - Xác định thuộc tính muốn chọn làm class

- Chuột phải vào hang tên gọi của cột thuộc tính đó, và chọn "Attribute as class"

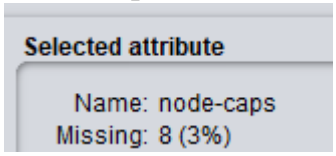


- Sau đó chọn "OK" để xác nhận chọn thuộc tính đã chọn làm thuộc tính class

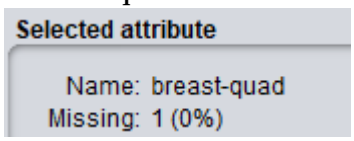
2.1.4 Tìm hiểu chi tiết từng thuộc tính trong khung attribute và cho biết: có bao nhiêu thuộc tính bị thiếu dữ liệu(missing value)? Thuộc tính nào thiếu dữ liệu ít nhất / nhiều nhất? Trình bày tổng quát các cách để giải quyết vấn đề missing values

- Có 2 thuộc tính bị thiết dữ liệu, đó là:

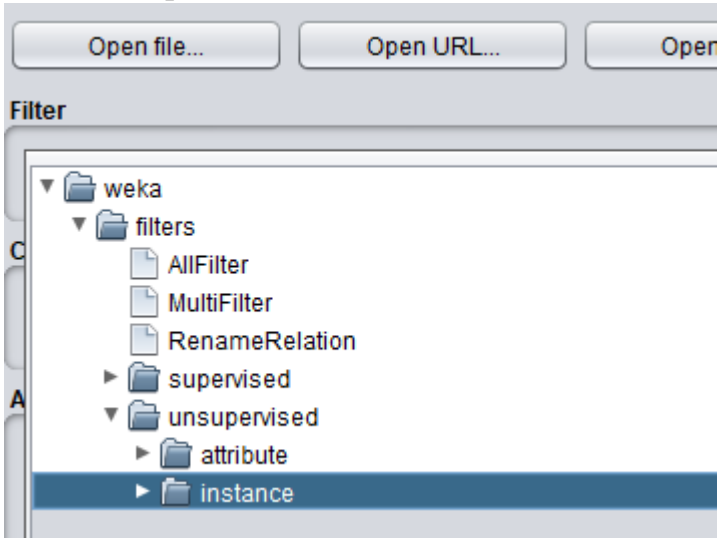
- Node-caps: thiếu 8 mẫu



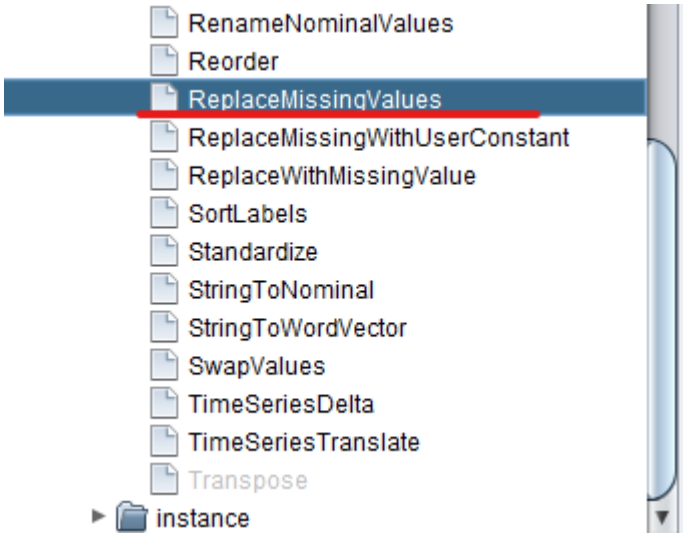
- Breast-quad: thiết 1 mẫu



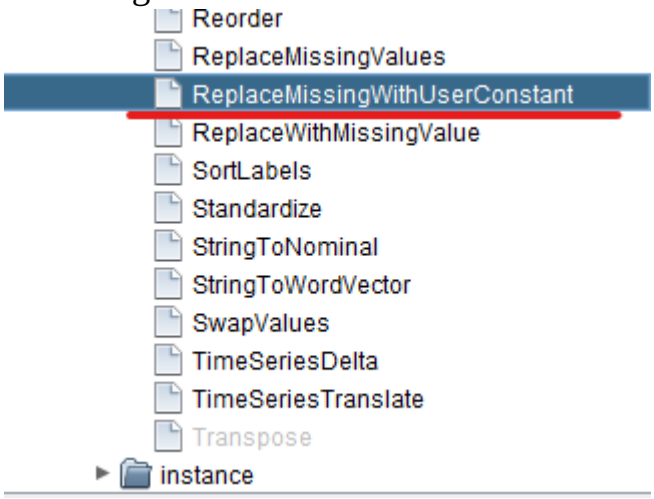
- Dựa vào số liện trên:
  - Node-caps là thuộc tinh thiếu dữ liệu nhiều nhất
  - Breast-quad là thuộc tính thiếu dữ liệu ít nhất
- Cách giải quyết vấn đề missing value in weka
  - Dưới toolbar chọn filter
  - Chọn unsupervised



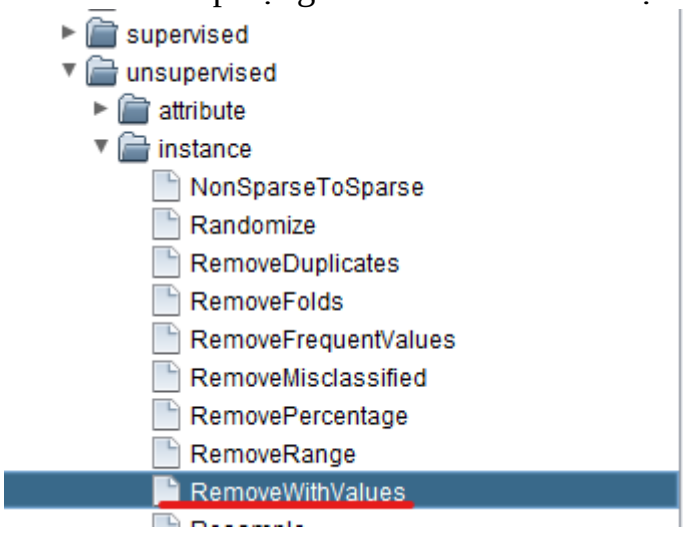
  - Ta sẽ chủ yếu giải quyết vấn đề missing value trong tab unsuppervised này, cụ thể có 3 cách chinh:
    - Replace missing value: sử dụng mean/median impute để điền missing value, đây là phương pháp thích hợp cho thuộc tính numeric



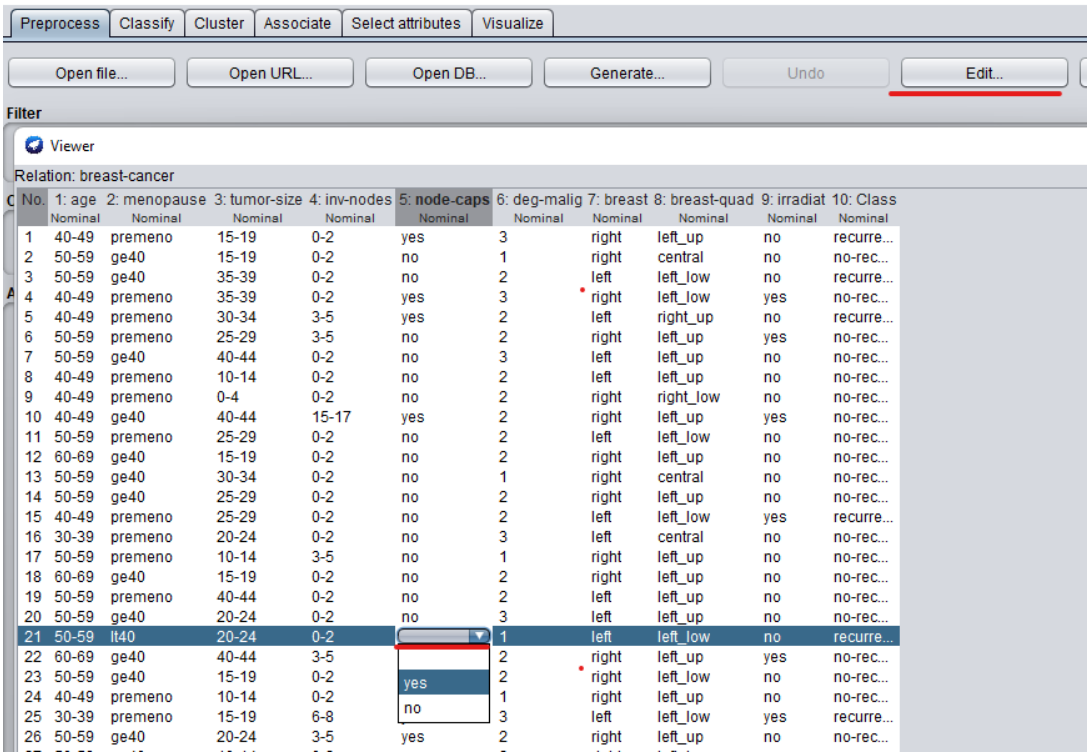
- Replace missing value with user constant: sử dụng mode để điền missing value, đât là phương pháp thích hợp cho thuộc tinh categorical



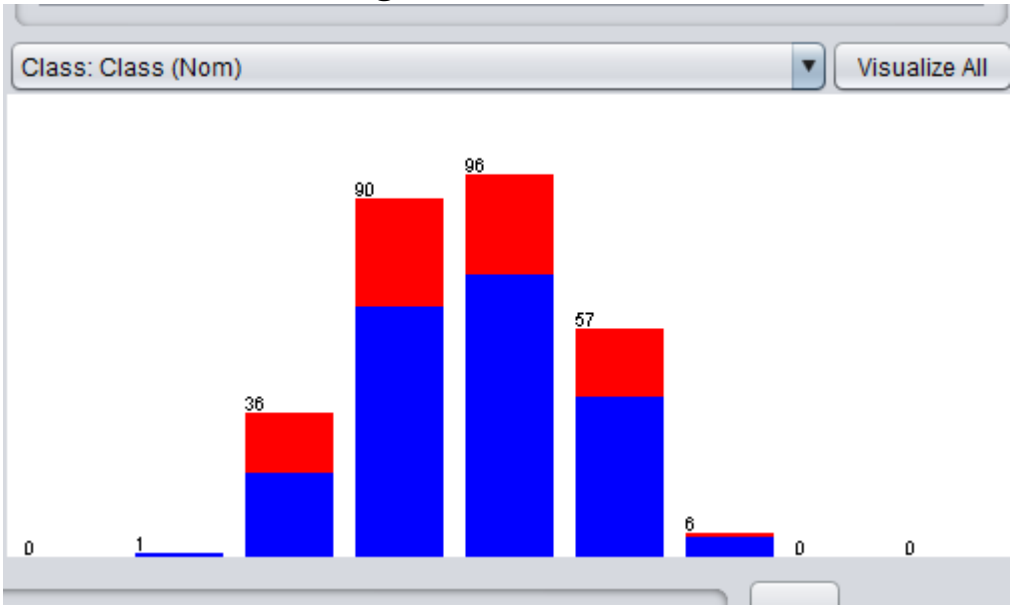
- Remove with value : xóa mẫu có value bị thiếu, cách này chỉ nên xem xét áp dụng khi 2 cách trên thất bại



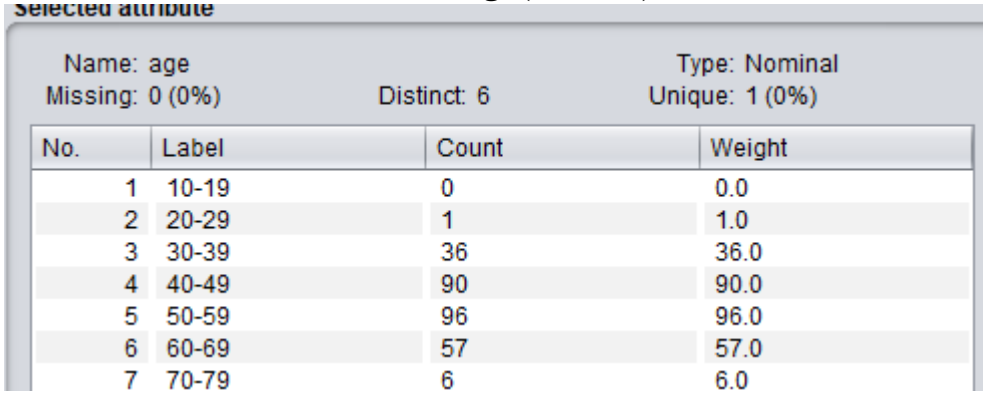
- Còn 1 cách thứ 4 đó là tự điền data thủ công, ta chọn edit và tìm ô bị thiếu để điền vào, tuy nhiên cách này không có tính ứng dụng cao



2.1.5 Giải thích ý nghĩa đồ thị trong cửa sổ Explorer. Bạn đặt tên cho đồ thị này là gì? Màu xanh đỏ có nghĩa là gì? Đồ thị này biểu diễn cho cái gì?



- Tên: biểu đồ biểu diễn tương quan giữa thuộc tinh và phân lớp kết quả, cụ thể trên hình là biểu đồ tương quan giữa thuộc tính age (tuổi tác) và class (class thể hiện ung thư vú hay không)
- Màu xanh đỏ có ý nghĩa: với từng label của thuộc tinh đang chọn, cụ thể với danh sách label cho nhôm age(tuổi tác):


Selected attribute

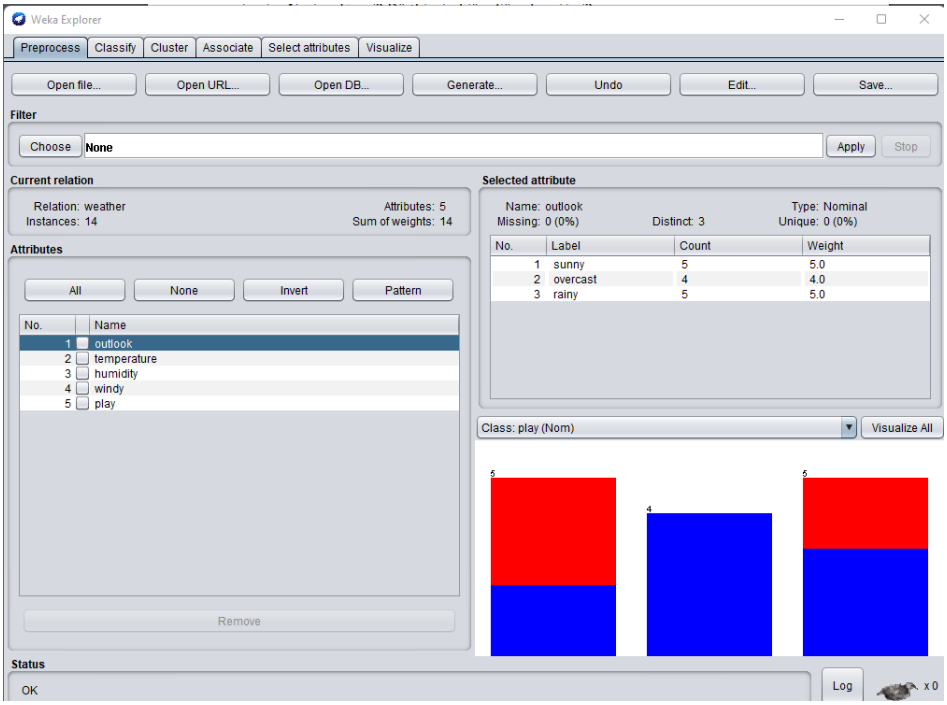
Name: age Type: Nominal
Missing: 0 (0%) Distinct: 6 Unique: 1 (0%)

No.	Label	Count	Weight
1	10-19	0	0.0
2	20-29	1	1.0
3	30-39	36	36.0
4	40-49	90	90.0
5	50-59	96	96.0
6	60-69	57	57.0
7	70-79	6	6.0

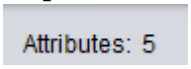
  - Màu xanh là tương ứng giữa giá trị của label đó thì có bao nhiêu người ung thư vú (hoặc không ung thư vú)
  - Màu đỏ là tương ứng giữa giá trị của label đó thì có bao nhiêu người không ung thư vú (hoặc ung thư vú)
- Đồ thị biểu diễn sự tương quan giữa thuộc tinh (đang chọn trong nhôm attribute) và class (được chọn ở hình trên, thường là chọn thuộc tinh class)

2.2 Khám phá tập dữ liệu Weather

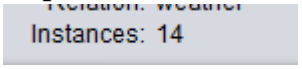
2.2.1 Tập dữ liệu có bao nhiêu thuộc tính? Bao nhiêu mẫu? Phân loại các thuộc tính theo kiểu dữ liệu. Thuộc tính nào là lớp?



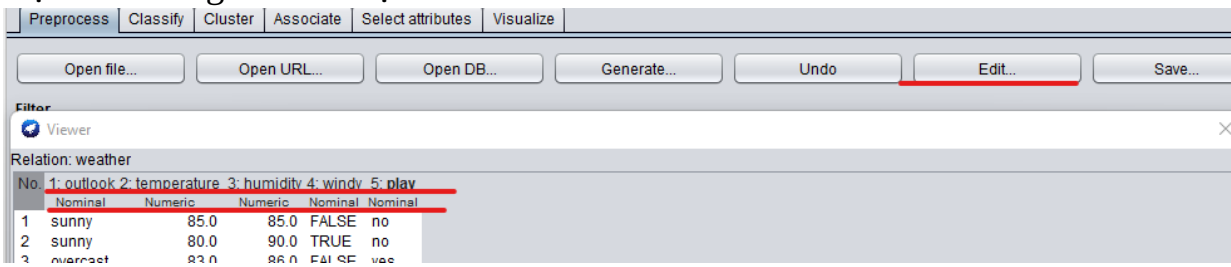
- Tập dữ liệu có 5 thuộc tính



- Tập dữ liệu có 14 mẫu



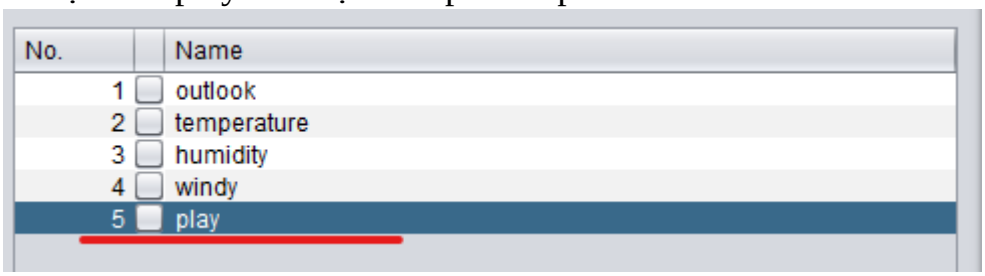
- Dựa vào thông tin khi chọn Edit



Các thuộc tính numeric: temperature, humidity

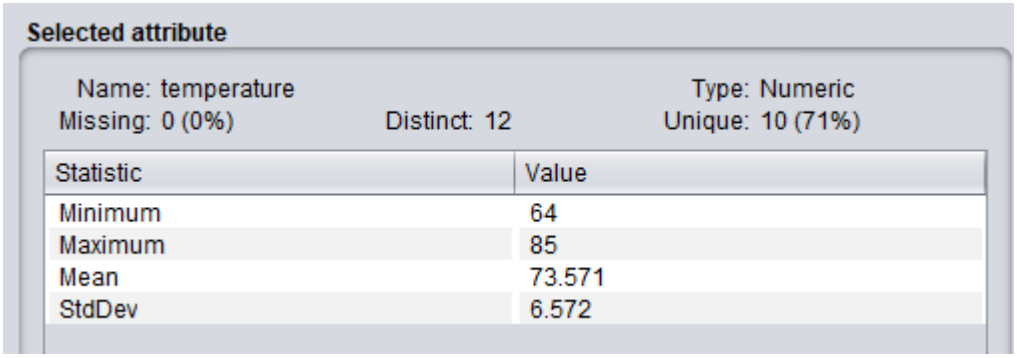
Các thuộc tính theo dạng categorical: outlook, windy, play

- Thuộc tính play là thuộc tính phân lớp

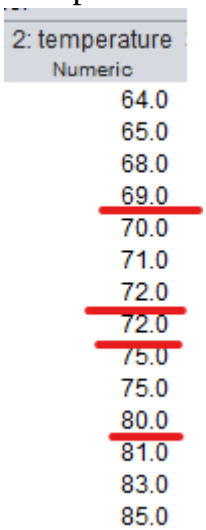


2.2.2 Liệt kê file-number summary của temperature và humidity. Weka có cung cấp những giá trị này không?

- Đối với thuộc tính numeric, weka chỉ cung cấp các giá trị sau theo mặc định



- Ta có thể thấy rằng với five-number summary, còn thiếu 3 dữ kiện là q1, q3 và median, bởi five-number summary bao gồm:
  - Min
  - Q1: median của nửa dưới dữ liệu
  - Median
  - Q3: median của nữa trên giữa liệu
  - Max
- Vậy đối với từng thuộc tính numeric có nhu cầu, ta cần tính them Q1, Q3 và median. Weka có hỗ trợ tính toán thông qua filter-> unsupervised -> attribute -> MathExpression / AddExpression, tuy nhiên ở đây ta có thể tính tay do số mẫu ít
  - Temperature



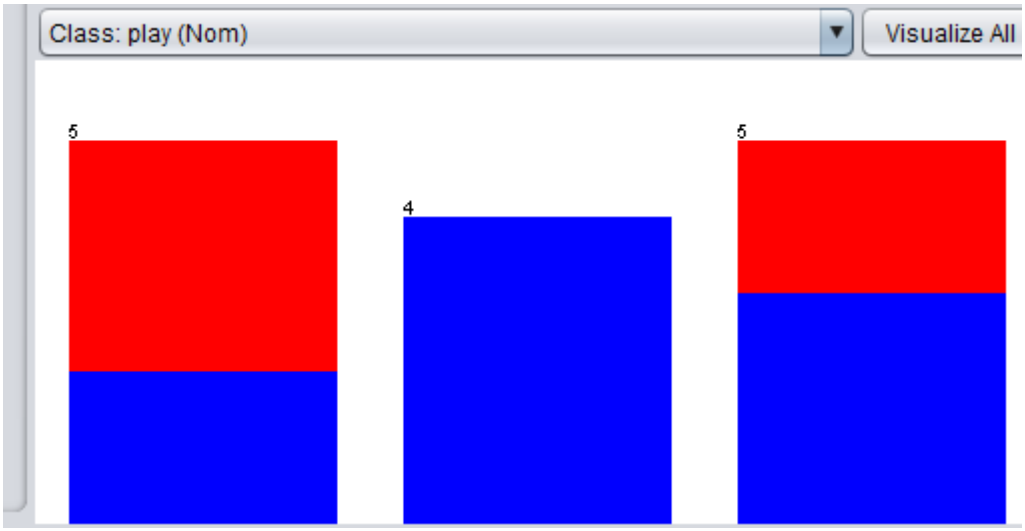
    - Do có 14 mẫu, Q2, hay median sẽ được tính theo (72 + 72) / 2 = 72
    - Q1 = 69
    - Q3 = 80
  - Humidity

3: humidity
Numeric
65.0
70.0
70.0
70.0
75.0
80.0
80.0
85.0
86.0
90.0
90.0
91.0
95.0
96.0

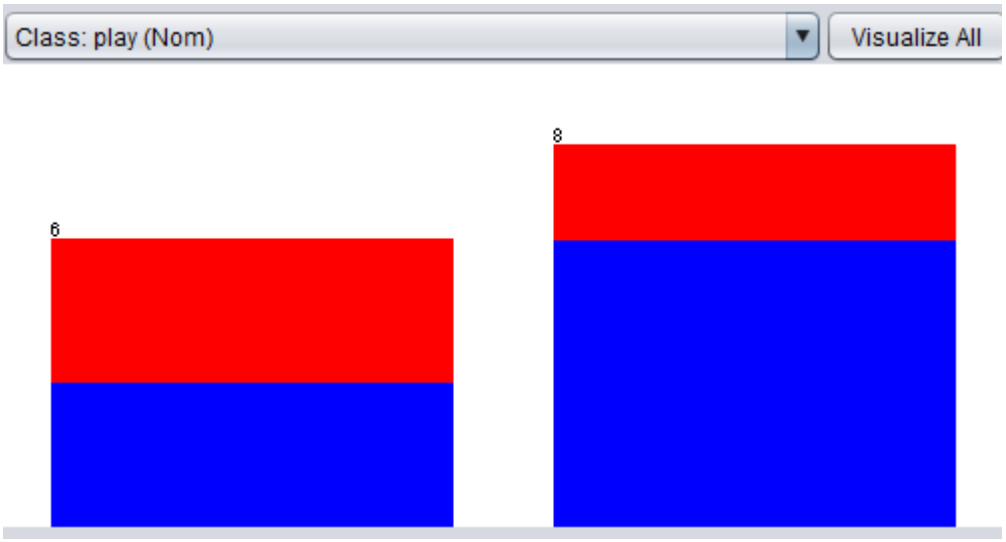
- Do có 14 mẫu, Q2 hay median sẽ được tính theo (80+85)/2 = 82.5
- Q1 = 70
- Q3 = 90

2.2.3 Lần lượt xem xét các thuộc tính khác của data set dưới dạng đồ thị

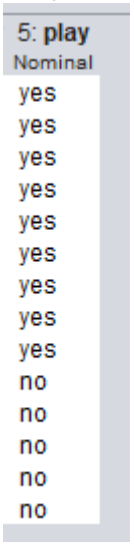
- Outlook



  - Là thuộc tính có 3 label
    - Sunny: 5 mẫu (2 yes - 3 no)
    - Overcast: 4 mẫu (4 yes)
    - Rainy: 5 mẫu (3 yes – 2 no)
- Windy:

- 

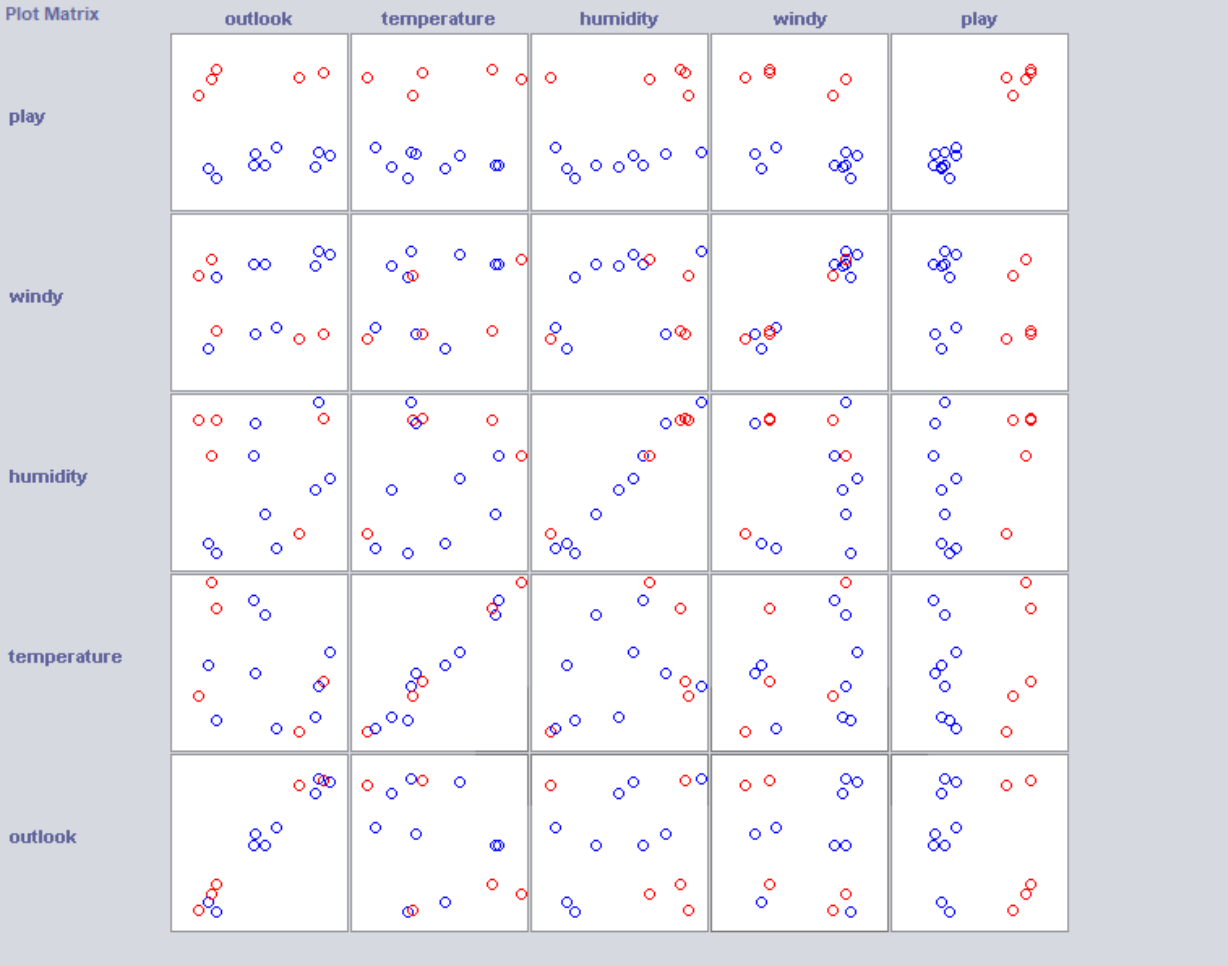
    - Là thuộc tính có 2 label
        - True: 6 mẫu (3 yes – 3 no)
        - False: 8 mẫu (6 yes – 2 no)
- Play:



    - Là thuộc tính phân lớp, gồm 2 label:
        - Yes: 9 mẫu
        - No 5 mẫu

2.2.4 Chuyển sang tab visualize. Thuật ngữ textbook cho đồ thị này là gì? Những cặp thuộc tính khác nhau nào có vẻ tương quan?

- Với từng đồ thị thành phần là **scatter plot**, đồ thị lớn ở đây, hay ma trận tổng quát chứa các thành phần đó gọi là **scatter plot matrix**



- Một số cặp thuộc tính "trông" có vẻ tương quan
  - Outlook – temperature
  - Temperature – windy
  - Humidity – play

2.3 Khám phá tập dữ liệu tín dụng Đức

2.3.1 Nội dung của phần ghi chú nói về điều gì? Tập dữ liệu có bao nhiêu mẫu? Bao nhiêu thuộc tính? Mô tả 5 thuộc tính

- Nội dung ghi chú:
  - Title: German Credit là title của bảng dữ liệu này
  - Sources information: là nguồn của dữ liệu, tác giả thu thập
  - Number of instances: số mẫu
  - Attribute description: mô tả về thuộc tính, miền giá trị của thuộc tính
  - Cost matrix: ma trận chi phí mà dataset sử dụng
- Tập dữ liệu có 1000 mẫu

```
%
% 3. Number of Instances:  1000
%
```

- Tập dữ liệu có 21 thuộc tính

```
Number of Attributes german: 20 (7 numerical, 13 categorical)
```

Ở đây tuy để 20, nhưng ta cần phải xem xét thêm 1 thuộc tính phân lớp, nên 20+1 = 21

- Mô tả 5 thuộc tính bất kì
  - Duration: là thuộc tính thể hiện khoảng thời gian (theo tháng), thuộc thuộc tính NUMERIC

```
Attribute 2:  (numerical)
            Duration in month
```

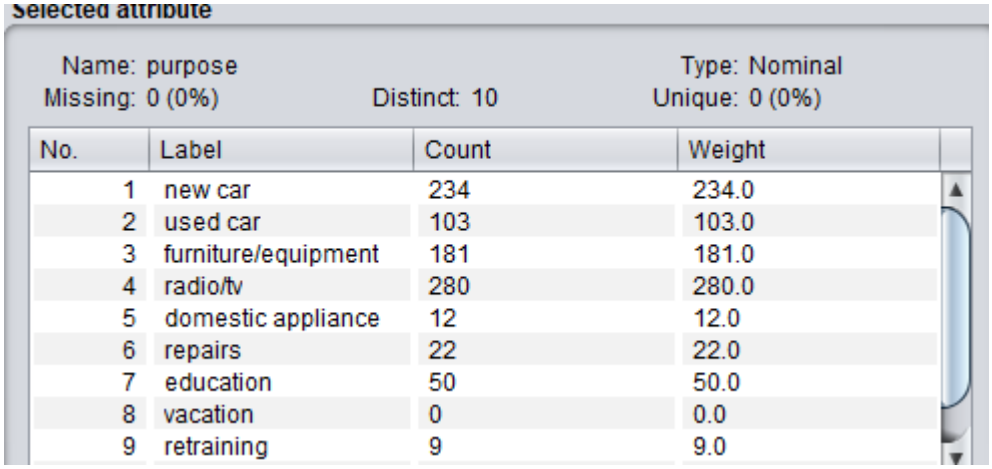
Selected attribute

Name: duration Type: Numeric
Missing: 0 (0%) Distinct: 33 Unique: 5 (1%)

Statistic	Value
Minimum	4
Maximum	72
Mean	20.903
StdDev	12.059

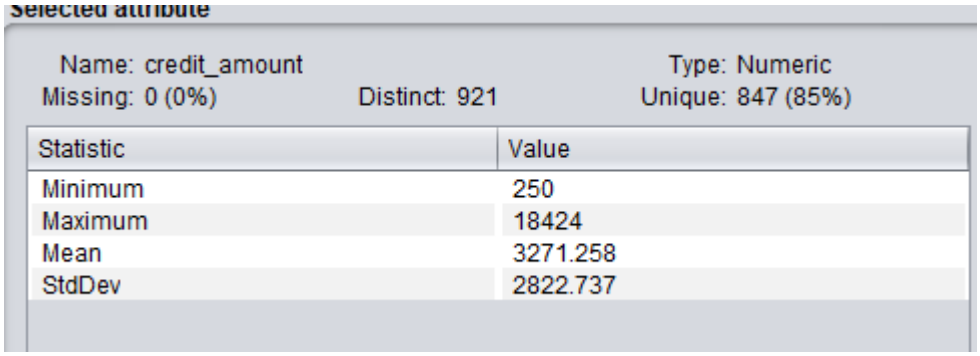
  - Purpose: là thuộc tính CATEGORICAL, thể hiện mục đích của chi tiêu

```
Attribute 4:  (qualitative)
            Purpose
            A40 : car (new)
            A41 : car (used)
            A42 : furniture/equipment
            A43 : radio/television
            A44 : domestic appliances
            A45 : repairs
            A46 : education
            A47 : (vacation - does not exist?)
            A48 : retraining
            A49 : business
            A410 : others
```



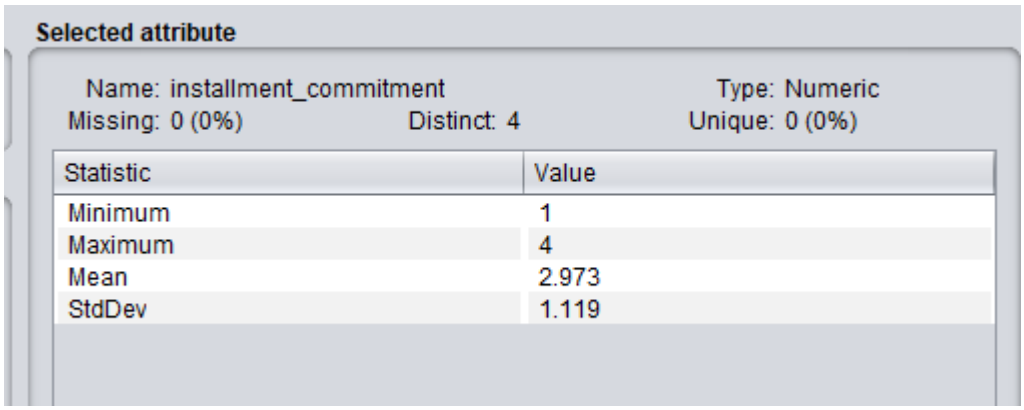
- Credit amount: là thuộc tính NUMERIC, biểu diễn số lượng của credit

```
Attribute 5:  (numerical)
            Credit amount
```



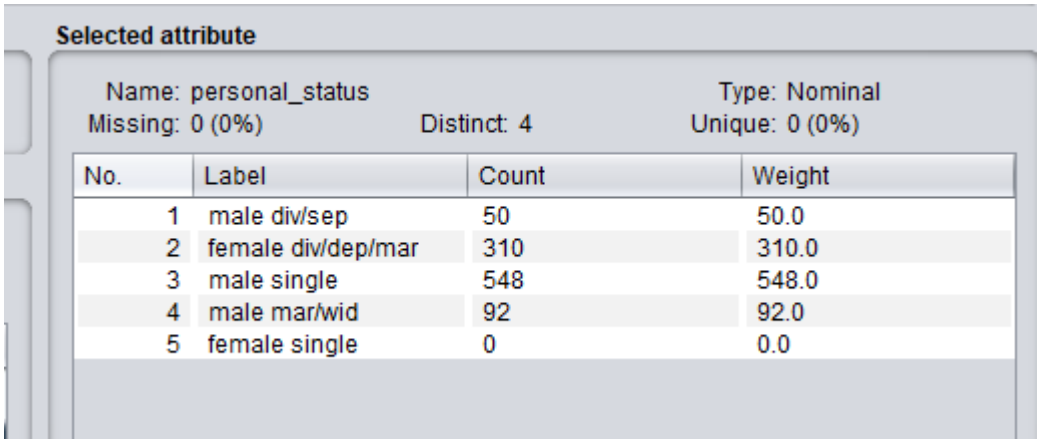
- Installment commitment: là thuộc tính CATEGORICAL, biểu hiện rate của installment theo phần tram thu nhập thừa

```
Attribute 8:  (numerical)
            Installment rate in percentage of disposable income
```



- Personal status: là thuộc tính CATEGORICAL, thể hiện giới tính và tình trạng hôn nhân

```
%
% Attribute 9:  (qualitative)
%              Personal status and sex
%              A91 : male   : divorced/separated
%              A92 : female : divorced/separated/married
%                A93 : male   : single
%              A94 : male   : married/widowed
%              A95 : female : single
%
```

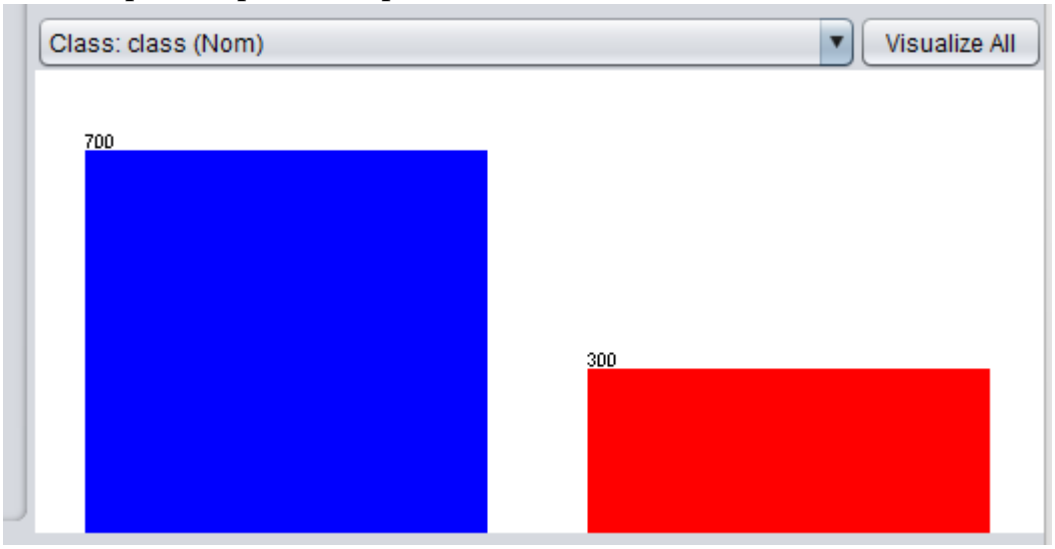


2.3.2 Tên của thuộc tính lớp là gì? Cân bằng hay lệch

- Thuộc tính phân lớp là phân loại khách hang đó good hay bad

```
(1 = Good,  2 = Bad)
```

- Dữ liệu phân lớp lệch về phía Good



2.3.3 Liệt kê lựa chọn khác nhau của weka để lựa chọn thuộc tính và giải thích ngắn gọn

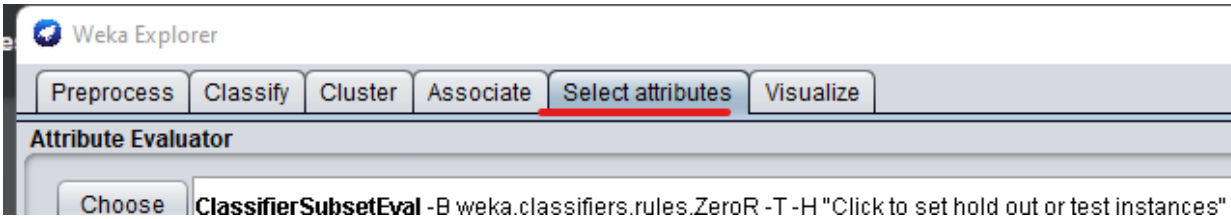
- Attribute evaluator:

- CfsSubsetEval: Đánh giá giá trị của một tập hợp con các thuộc tính bằng cách xem xét khả năng dự đoán của từng đối tượng cùng với mức độ dư thừa giữa chúng.
- ClassifierAttributeEval : Đánh giá giá trị của một thuộc tính bằng cách sử dụng bộ phân loại do người dùng chỉ định.
- ClassifierSubsetEval: Đánh giá các tập hợp con thuộc tính trên dữ liệu đào tạo hoặc một tập hợp thử nghiệm riêng biệt.
- CorrelationAttributeEval: Đánh giá giá trị của một thuộc tính bằng cách đo lường mối tương quan giữa nó và lớp.
- GainRatioAttributeEval: Đánh giá giá trị của một thuộc tính bằng cách đo hệ số khuếch đại liên quan đến lớp.
- InfoGainAttributeEval: Đánh giá giá trị của một thuộc tính bằng cách đo lường thông tin thu được liên quan đến lớp.
- OneRAttributeEval: Đánh giá giá trị của một thuộc tính bằng cách sử dụng bộ phân loại OneR classifier.
- PrincipalComponents: Thực hiện phân tích và chuyển đổi các thành phần chính của dữ liệu.
- ReliefFAttributeEval: Đánh giá giá trị của một thuộc tính bằng cách liên tục lấy mẫu một cá thể và xem xét giá trị của thuộc tính đã cho của cùng một lớp và khác lớp.
- SymmetricalUncertAttributeEval: Đánh giá giá trị của một thuộc tính bằng cách đo độ đo đối xứng đối với lớp.
- WrapperSubsetEval: Đánh giá các tập thuộc tính bằng cách sử dụng một chiến lược học (máy)

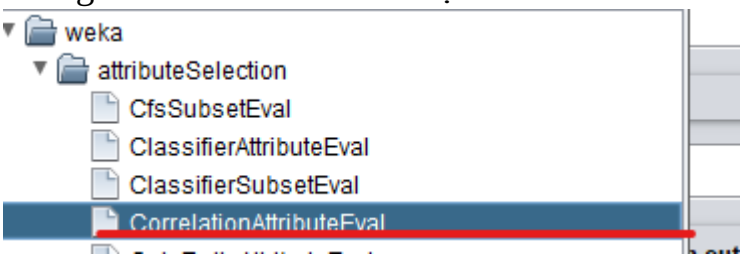
- Search method:
  - BestFirst: sử dụng thuật toán vét cạn hillclimbing
  - Ranker: xếp hạng các thuộc tính từ phù hợp nhất đến ít phù hợp nhất
  - Greedy step wise: tìm vét cạn tiến hoặc lùi trong không gian các subset thuộc tính

2.3.4 Chọn 5 thuộc tính có tương quan cao nhất với thuộc tính lớp

- Để chọn các thuộc tính có độ tương quan cao với thuộc tính lớp, ta có thể sử dụng CorrelationSubsetEval và Ranker search method để tiến hành đánh giá, các bước thực hiện như sau:
- Trên thanh tabar chọn Select attributes

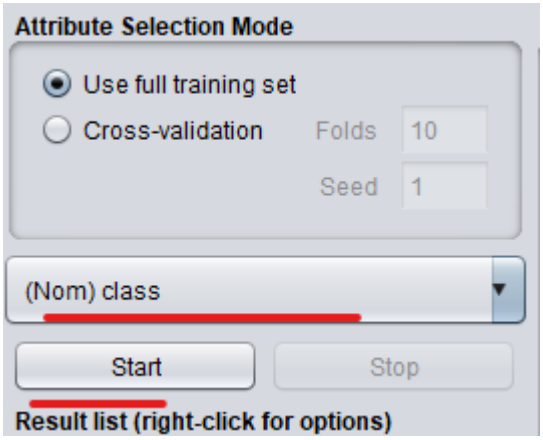


- Trong Attribute Evaluator chọn CorrelationSubsetEval



- Ở search method chọn ranker

- Chọn thuộc tính class và bấm start



- Trong kết quả có được, ta chọn 5 thuộc tính đầu tiên

```
ttribute Evaluator (supervised, Class (nom
        Correlation Ranking Filter
anked attributes:
0.23276    1 checking_status
0.21493    2 duration
0.15474    5 credit_amount
0.13162    6 savings_status
0.12138   15 housing
0.108     14 other_payment_plans
0.09113   13 age
0.08988    3 credit_history
0.08208   20 foreign_worker
0.07494    4 purpose
```

- Vậy ta có các thuộc tính sau có sự tương quan cao nhất với lớp
  - Checking_status
  - Duration

- Credit_amount
- Savings_status
- Housing

Phần 3: Cài đặt tiền xử lý dữ liệu

3.1 Sử dụng phần mềm

- Cài đặt thư viện: phần mềm yêu cầu cài đặt 1 thư viện, đó là https://pypi.org/project/tabulate/
- Hướng dẫn sử dụng phần mềm trên Console đã được **tích hợp sẵn** vào phầm mềm, ta có thể gõ lệnh sau để có thêm chi tiết
  - **python3 preprocess.py -h**
- Tuy nhiên, hướng dẫn này cũng được mô tả tương đối chi tiết (với các chức năng cần thiết phục vụ đồ án), phần hướng dẫn này sẽ được đinh kèm dưới dạng file pdf nộp cùng với đồ án: **usage.pdf,** việc đọc sơ file này trước khi sử dụng phần mềm được khuyến khích.

3.2 Một số tiêu điểm

- Để dễ dàng hơn cho việc viết document, document của từng hàm riêng biệt đã được viết thẳng vào source code (một phương pháp thường thấy đối với các thư viện python), thông tin chi tiết document này có thể được truy cập dễ dàng khi mở các file **.py** bằng bất kì text editor nào
- Sử dụng thư viện **typing** để hỗ trợ quá trinh viết document được suôn sẻ hơn, vì thư viện này cho phép định nghĩa kiểu dữ liệu đầu vào cũng như trả về, chi tiết xem tại: https://docs.python.org/3/library/typing.html (đây là thư viện có sẵn)
- Quá trình thao tác trên file đều là tự cài đặt, thư viện **csv** chỉ có vai trò đọc file, các bước thao tác trên dữ liệu có thể bao gồm 3 bước chinh
  - Đọc dữ liệu (sử dụng thư viện **csv**)
  - Xử lý dữ liệu (tự cài đặt)
  - Lưu dữ liệu (vào file khác)
- Thư viện cần cài đặt thêm là **tabulate,** mục đích cho thư viện này là vẽ bảng, trực quan hóa kết quả ở một số yêu cầu
- Sử dụng cấu trúc dữ liệu Stack (tự cài đặt) và thuật toán chuyển infix sang post-fix (tự cài đặt) để tiến hanh tinh toán trên các thuộc tinh

3.3 Báo cáo kết quả sử dụng phần mềm với từng chức năng

3.3.1 Liệt kê các cột bị thiếu dữ liệu

- Dòng lệnh sau sẽ liệu kê các thuộc tính bị thiếu dữ liệu

```
$ python preprocess.py -f ../data/house-prices.csv list -mc
Attribute with missing values
```

attribute
Alley
FireplaceQu
PoolQC
Fence
MiscFeature
MasVnrType
BsmtQual

- Ngoài ra, để có thông tin chi tiết hơn, ta còn có thể sử dụng lệnh sau để thể hiện từng thuộc tinh thì thiếu bao nhiêu dữ liệu

```
$ python preprocess.py -f ../data/house-prices.csv list -m
Generals missing data:
```

attribute	missing instance
Alley	941
FireplaceQu	501
PoolQC	1000
Fence	815
MiscFeature	963
MasVnrType	593
BsmtQual	27
BsmtCond	27

3.3.2 Đếm số dông bị thiếu dữ liệu

```
$ python preprocess.py -f ../data/house-prices.csv list -mr
Number of rows with missing value: 1000
```

- Ta có thể thấy ở đây có 1000 dòng bị thiếu dữ liệu

3.3.3 Điền giá trị thiếu

- Điền giá trị bị thiếu bằng phương pháp mean và lưu vào file mean_fill.csv

```
$ python preprocess.py -f ../data/house-prices.csv fill -ft mean -o mean_fill.csv
mean
filling N/A value with MEAN...
Saved to mean_fill.csv
done!
```

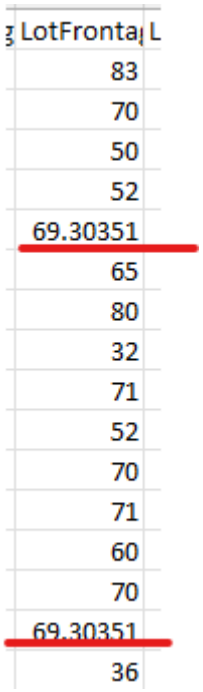
- Sau khi chạy xong, file mới sẽ được sinh ra với tất cả các thuộc tinh được fill như sau
  - PoolQC là thuộc tính trống hoàn toàn, không xác định là NUMERIC hay CATEGORICAL nên phần mềm sẽ điền giá trị fallback là 0

PoolQC
0
0
0
0
0
0
0
0
0

  - Fence là thuộc tính CATEGORICAL, được điền theo chế độ MODE, đây là chế độ mặc định

Fence
MnPrv
MnPrv
MnPrv
MnPrv
GdWo
MnPrv
MnPrv
MnPrv
MnPrv
MnPrv
MnPrv
MnPrv

  - LotFrontage là thuộc tính NUMERIC, được điền giá trị MEAN



LotFronta	L
83	
70	
50	
52	
69.30351	
65	
80	
32	
71	
52	
70	
71	
60	
70	
69.30351	
36	

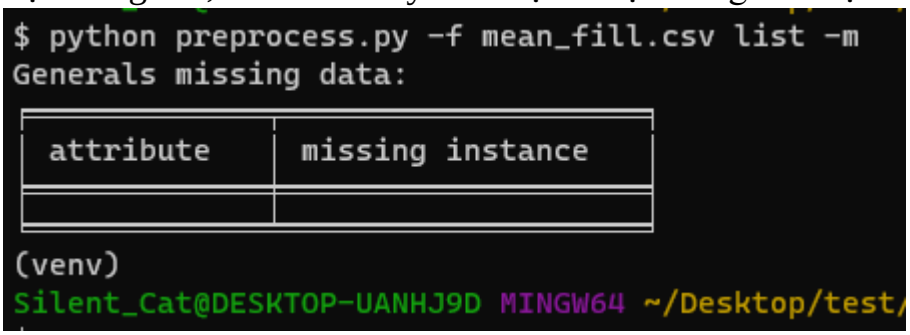
- Ngược lại, nếu kiểu điền thuộc tính là MEDIAN, ta có thể chạy lệnh sau, lúc này thì kết quả sẽ được lưu vào file median_fill.csv

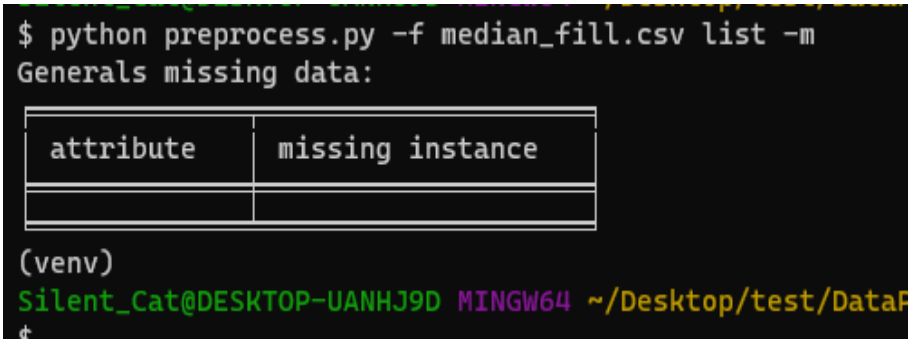
```
$ python preprocess.py -f ../data/house-prices.csv fill -ft median -o median_fill.csv
filling N/A value with MEDIAN...
Saved to median_fill.csv
done!
```

- Ví dụ thuộc tính LotFrontage sau khi được điền với MEDIAN thì kết quả không còn là số như điền kiểu MEAN nữa

g LotFronta
83
70
50
52
63
65
80
32
71
52
70
71
60
70
63
36
34
35
51
44
108
71

- Ta có thể sử dụng lệnh ở chức năng a, b để kiểm tra độ thiếu của dữ liệu trong file, ta có thể thấy toàn bộ dữ liệu trống đã được điền





3.3.4 Xóa các dòng dữ liệu bị thiết với ngưỡng cho trước

- Xóa các dòng dữ liệu chỉ cụ thể số thuộc tính, ở đây là xóa những dông bị thiếu từ 4 thuộc tinh trở lên và lưu vào file del_row_int.csv

```
$ python preprocess.py -f ../data/house-prices.csv delthres -t row -ti 4 -o del_row_int.csv
deleting missing rows with a given threshold...
Saved to del_row_int.csv
done!
```

- Dữ liệu sau khi xóa chỉ còn 34 dòng dữ liệu (không tinh header)

	Id	MSSubCla	MSZoning	LotFronta	LotArea	Street	Alley	LotShape	LandCont	Utilities	LotConfig	LandSlope	Neighborh	Condition	Condition	Bl
2	1053	60	RL	100	9500	Pave		Reg	Lvl	AllPub	Corner	Gtl	mes	Artery	Norm	1F
3	954	60	RL		11075	Pave		IR1	Lvl	AllPub	Inside	Mod	Mitchel	Norm	Norm	1F
4	1056	20	RL	104	11361	Pave		Reg	Lvl	AllPub	Inside	Gtl	NWAmes	Norm	Norm	1F
5	298	60	FV	66	7399	Pave	Pave	IR1	Lvl	AllPub	Inside	Gtl	Somerst	Norm	Norm	1F
6	1436	20	RL	80	8400	Pave		Reg	Lvl	AllPub	Inside	Gtl	mes	Norm	Norm	1F
7	891	50	RL	60	8064	Pave		Reg	Lvl	AllPub	Corner	Gtl	mes	Artery	Norm	1F
8	767	60	RL	80	10421	Pave		Reg	Lvl	AllPub	Inside	Gtl	NWAmes	Norm	Norm	1F
9	734	20	RL	80	10000	Pave		Reg	Lvl	AllPub	Corner	Gtl	Sawyer	Feedr	Norm	1F
10	993	60	RL	80	9760	Pave		Reg	Lvl	AllPub	Inside	Mod	mes	Norm	Norm	1F
11	1084	20	RL	80	8800	Pave		Reg	Lvl	AllPub	Inside	Gtl	mes	Norm	Norm	1F
12	1436	20	RL	80	8400	Pave		Reg	Lvl	AllPub	Inside	Gtl	mes	Norm	Norm	1F
13	298	60	FV	66	7399	Pave	Pave	IR1	Lvl	AllPub	Inside	Gtl	Somerst	Norm	Norm	1F
14	993	60	RL	80	9760	Pave		Reg	Lvl	AllPub	Inside	Mod	mes	Norm	Norm	1F
15	1329	50	RM	60	10440	Pave	Grvl	Reg	Lvl	AllPub	Corner	Gtl	OldTown	Norm	Norm	1F
16	796	60	RL	70	8400	Pave		Reg	Lvl	AllPub	Inside	Gtl	SawyerW	Norm	Norm	1F
17	323	60	RL	86	10380	Pave		IR1	Lvl	AllPub	Inside	Gtl	SawyerW	Norm	Norm	1F
18	52	50	RM	52	6240	Pave		Reg	Lvl	AllPub	Inside	Gtl	BrkSide	Norm	Norm	1F
19	1458	70	RL	66	9042	Pave		Reg	Lvl	AllPub	Inside	Gtl	Crawfor	Norm	Norm	1F
20	317	60	RL	94	13005	Pave		IR1	Lvl	AllPub	Corner	Gtl	NWAmes	Norm	Norm	1F
21	891	50	RL	60	8064	Pave		Reg	Lvl	AllPub	Corner	Gtl	mes	Artery	Norm	1F
22	1274	80	RL	124	11512	Pave		IR1	Lvl	AllPub	Corner	Gtl	Edwards	Norm	Norm	1F
23	540	20	RL		11423	Pave		Reg	Lvl	AllPub	Inside	Gtl	CollgCr	Norm	Norm	1F
24	1457	20	RL	85	13175	Pave		Reg	Lvl	AllPub	Inside	Gtl	NWAmes	Norm	Norm	1F
25	1436	20	RL	80	8400	Pave		Reg	Lvl	AllPub	Inside	Gtl	mes	Norm	Norm	1F
26	891	50	RL	60	8064	Pave		Reg	Lvl	AllPub	Corner	Gtl	mes	Artery	Norm	1F
27	1077	50	RL	60	10800	Pave	Grvl	Reg	Lvl	AllPub	Inside	Gtl	OldTown	Norm	Norm	1F
28	41	20	RL	84	8658	Pave		Reg	Lvl	AllPub	Inside	Gtl	mes	Norm	Norm	1F
29	870	60	RL	80	9938	Pave		Reg	Lvl	AllPub	Inside	Gtl	SawyerW	Norm	Norm	1F
30	540	20	RL		11423	Pave		Reg	Lvl	AllPub	Inside	Gtl	CollgCr	Norm	Norm	1F
31	993	60	RL	80	9760	Pave		Reg	Lvl	AllPub	Inside	Mod	mes	Norm	Norm	1F
32	1329	50	RM	60	10440	Pave	Grvl	Reg	Lvl	AllPub	Corner	Gtl	OldTown	Norm	Norm	1F
33	993	60	RL	80	9760	Pave		Reg	Lvl	AllPub	Inside	Mod	mes	Norm	Norm	1F
34	41	20	RL	84	8658	Pave		Reg	Lvl	AllPub	Inside	Gtl	mes	Norm	Norm	1F
35	1053	60	RL	100	9500	Pave		Reg	Lvl	AllPub	Corner	Gtl	mes	Artery	Norm	1F
36																

- Thay vì đưa ra số thuộc tinh cụ thể bị thiếu, ta cũng có thể sử dụng phần trăm số thuộc tính để đặt ngưỡng, phần tram này sẽ khoảng từ 0.0-1.0. Cụ thể, lệnh sau đã xóa những dòng bị thiếu 5% số thuộc tính và lưu kết quả vào file del_row_pct.csv

```
$ python preprocess.py -f ../data/house-prices.csv delthres -t row -tp 0.05 -o del_row_pct.csv
deleting missing rows with a given threshold...
Saved to del_row_pct.csv
done!
(venv)
```

- Ta có thể thấy số dòng sau khi xóa còn lại từng đây, 34 dòng dữ liệu (không tính header)

	A	B	C	D	E	F	G	H
1	Id	MSSubCla	MSZoning	LotFronta	LotArea	Street	Alley	LotShape
2	1053	60	RL	100	9500	Pave		Reg
3	954	60	RL		11075	Pave		IR1
4	1056	20	RL	104	11361	Pave		Reg
5	298	60	FV	66	7399	Pave	Pave	IR1
6	1436	20	RL	80	8400	Pave		Reg
7	891	50	RL	60	8064	Pave		Reg
8	767	60	RL	80	10421	Pave		Reg
9	734	20	RL	80	10000	Pave		Reg
10	993	60	RL	80	9760	Pave		Reg
11	1084	20	RL	80	8800	Pave		Reg
12	1436	20	RL	80	8400	Pave		Reg
13	298	60	FV	66	7399	Pave	Pave	IR1
14	993	60	RL	80	9760	Pave		Reg
15	1329	50	RM	60	10440	Pave	Grvl	Reg
16	796	60	RL	70	8400	Pave		Reg
17	323	60	RL	86	10380	Pave		IR1
18	52	50	RM	52	6240	Pave		Reg
19	1458	70	RL	66	9042	Pave		Reg
20	317	60	RL	94	13005	Pave		IR1
21	891	50	RL	60	8064	Pave		Reg
22	1274	80	RL	124	11512	Pave		IR1
23	540	20	RL		11423	Pave		Reg
24	1457	20	RL	85	13175	Pave		Reg
25	1436	20	RL	80	8400	Pave		Reg
26	891	50	RL	60	8064	Pave		Reg
27	1077	50	RL	60	10800	Pave	Grvl	Reg
28	41	20	RL	84	8658	Pave		Reg
29	870	60	RL	80	9938	Pave		Reg
30	540	20	RL		11423	Pave		Reg
31	993	60	RL	80	9760	Pave		Reg
32	1329	50	RM	60	10440	Pave	Grvl	Reg
33	993	60	RL	80	9760	Pave		Reg
34	41	20	RL	84	8658	Pave		Reg
35	1053	60	RL	100	9500	Pave		Reg
36								
37								

3.3.5 Xóa các cột bị thiếu với ngưỡng cho trước

- Xóa các thuộc tính có từ 500 giá trị bị thiếu vào lưu vào del_col_int.csv

```
Silent_Cat@DESKTOP-UANHJ9D MINGW64 ~/Desktop/test/DataPreprocessor/src (main)
$ python preprocess.py -f ../data/house-prices.csv delthres -t col -ti 500 -o del_col_int.csv
deleting missing attributes with a given threshold...
Saved to del_col_int.csv
done!
(venv)
Silent_Cat@DESKTOP-UANHJ9D MINGW64 ~/Desktop/test/DataPreprocessor/src (main)
```

- Ở đây ta có thể thấy thuộc tính PoolQC vốn không có giá trị đã bị xóa, cùng với một số thuộc tính khác

	BK	BL	BM	BN	BO	BP	BQ	BR	BS	BT	BU	BV	BW
eCo	PavedDriv	WoodDec	OpenPorc	EnclosedP	3SsnPorch	ScreenPor	PoolArea	MiscVal	MoSold	YrSold	SaleType	SaleCondi	SalePrice
	Y	0	56	0	0	0	0	0	6	2007	New	Partial	248328

- Tương tự như xóa dòng, xóa cột cũng hỗ trợ xóa theo phần trăm, tuy nhiên ở đây là phần trăm giá trị, cụ thể ở đây ta có thể xóa những thuộc tính nào thiếu hoàn toàn giá trị và lưu vào del_col_pct.csv

```
Silent_Cat@DESKTOP-UANHJ9D MINGW64 ~/Desktop/test/DataPreprocessor/src (main)
$ python preprocess.py -f ../data/house-prices.csv delthres -t col -tp 1 -o del_col_pct.csv
deleting missing attributes with a given threshold...
Saved to del_col_pct.csv
done!
(venv)
```

- Ở đây ta có thể thấy chỉ thuộc tính PoolQc không có giá trị bị xóa, còn các thuộc tính khác, miễn là có giá trị thì sẽ không bị xóa

	BP	BQ	BR	BS	BT	BU	BV	BW	BX	BY	BZ	CA	CB
dDec	OpenPorc	EnclosedP	3SsnPorch	ScreenPor	PoolArea	Fence	MiscFeatu	MiscVal	MoSold	YrSold	SaleType	SaleCondi	SalePrice
0	56	0	0	0	0			0	6	2007	New	Partial	248328

3.3.6 Xóa các mẫu trùng lặp

- Xóa các dòng dữ liệu bị trùng và lưu vào file deldup.csv

```
$ python preprocess.py -f ../data/house-prices.csv deldup -t row -o deldup.csv
deleting duplicate row...
Saved to deldup.csv
done!
(venv)
```

- Xóa xong thì dữ liệu còn 716 dòng (không tính header)

712	188	50	RL	60	10410	Pave
713	192	60	RL		7472	Pave
714	903	60	RL	63	7875	Pave
715	237	20	RL	65	8773	Pave
716	389	20	RL	93	9382	Pave
717	1186	50	RL	60	9738	Pave

deldup

3.3.7 Chuẩn hóa thuộc tính

- Chuẩn hóa thuộc tính LotArea theo min-max và lưu vào file min_max_norm.csv

```
Silent_Cat@DESKTOP-UANHJ9D MINGW64 ~/Desktop/test/DataPreprocessor/src (main)
$ python preprocess.py -f ../data/house-prices.csv norm -t min-max -a LotArea -o min_max_norm.csv
performing min-max normalization...
Saved to min_max_norm.csv
done!
(venv)
```

a	LotArea	S
3	0.039164	F
0	0.039131	F
0	0.021158	F
2	0.022524	F
	0.051533	F
5	0.03493	F
0	0.034332	F
2	0.014141	F
1	0.050204	F
2	0.022281	F

- Chuẩn hóa thuộc tính LotFrontage dưới dạng z-score và lưu vào file z_score_norm.csv

```
Silent_Cat@DESKTOP-UANHJ9D MINGW64 ~/Desktop/test/DataPreprocessor/src (main)
$ python preprocess.py -f ../data/house-prices.csv norm -t z-score -a LotFrontage -o z_score_norm.cs
performing z-score normalization...
Saved to z_score_norm.csv
done!
(venv)
```

z	LotFronta	I
	0.643854	
	0.032741	
	-0.90743	
	-0.81341	
	-0.2023	
	0.502828	

3.3.8 Tính giá trị biểu thức thuộc tính

- Tạo một thuộc tinh mới tên là newAttribute, chứa kết quả của phép tinh giữa các thuộc tính: ‘(MSSubClass + LotFrontage) * OverallQual’

```
$ python preprocess.py -f ../data/house-prices.csv acalc -c '(MSSubClass + LotFrontage) * OverallQual' -a newAttribute -o attr
ibute_calc.csv
Saved to attribute_calc.csv
done!
```

newAttribute
721
640
600
492
775
500
912
786
328
450
455
320
360

Tham khảo

https://docs.python.org/3/library/csv.html

https://docs.python.org/3/library/typing.html

https://docs.python.org/3/library/argparse.html