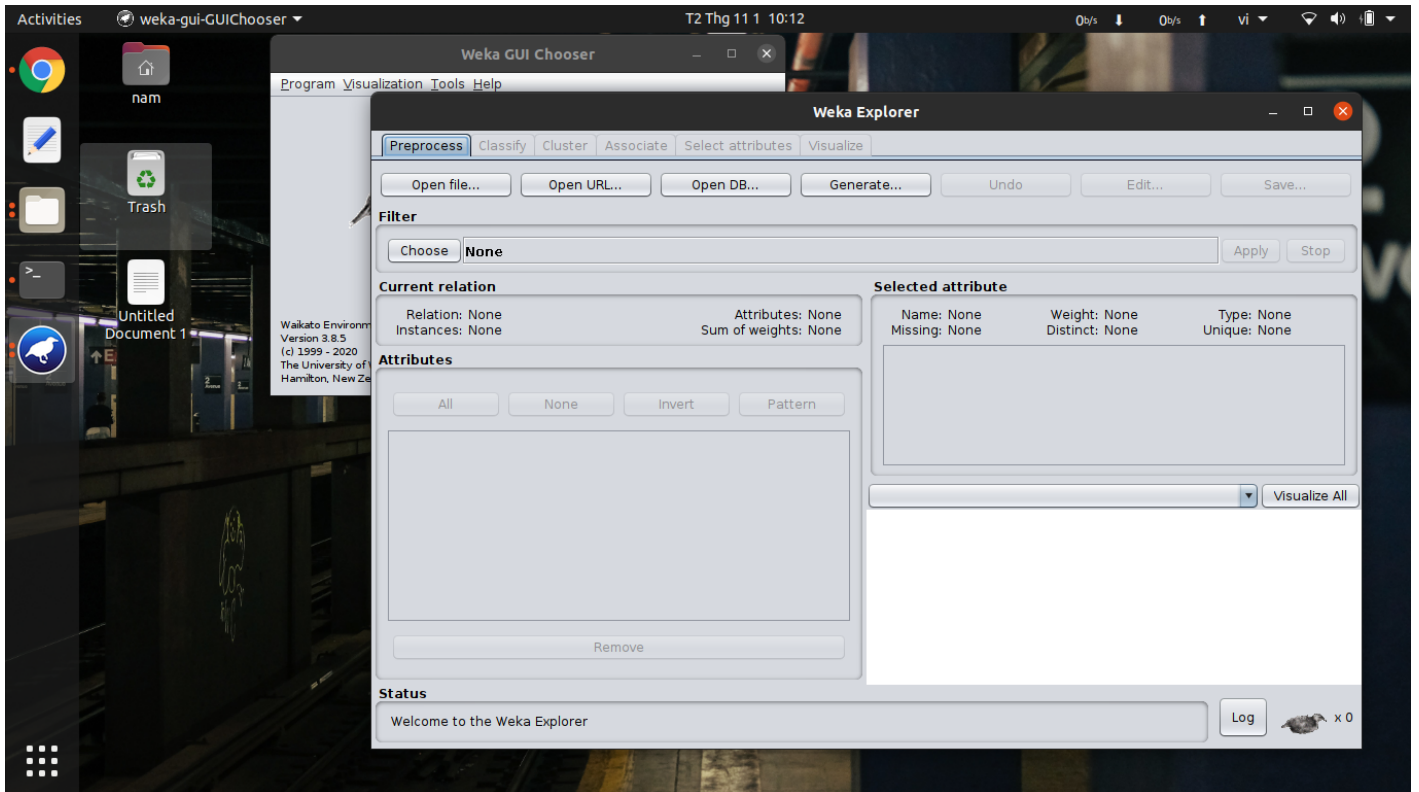


DATA MINING

BÁO CÁO LAB01 Preprocessing Data

Nhóm gồm 1 thành viên là Trần Hoàng Nam, MSSV 18120473, đã thực hiện đầy đủ yêu cầu của đồ án.

1 Cài đặt Weka



Hình 1: Giao diện Weka cùng màn hình desktop

Ý nghĩa các tab:

- **Preprocess:** Tiền xử lý cho dữ liệu, mục đích biến dữ liệu thô sang dữ liệu có định dạng để thuận lợi cho việc phân tích và khai thác dữ liệu.
- **Classify:** Phân lớp dữ liệu, sử dụng mô hình thích hợp để dự đoán nhãn cho dữ liệu mới từ danh sách các nhãn cho trước, dựa vào một bộ tiêu chí nào đó được trích xuất từ dữ liệu.
- **Cluster:** Phân cụm dữ liệu, là thao tác nhóm các điểm dữ liệu có cùng tính chất với nhau, qua đó phân hoạch dữ liệu thành các cụm riêng biệt.
- **Associate:** Tìm ra và khai thác các mối kết hợp hay tương quan trong dữ liệu.
- **Select Attributes:** Trích chọn các đặc trưng quan trọng nhất trong dữ liệu, đó là các đặc trưng mang nhiều thông tin nhất để suy diễn ra các dự đoán.
- **Visualize:** Trực quan hóa dữ liệu, sử dụng khả năng tiếp nhận thông tin bằng hình ảnh tốt của con người để tối đa thông tin mang lại. Giúp con người đánh giá dữ liệu, phát hiện insight, đánh giá mô hình,...

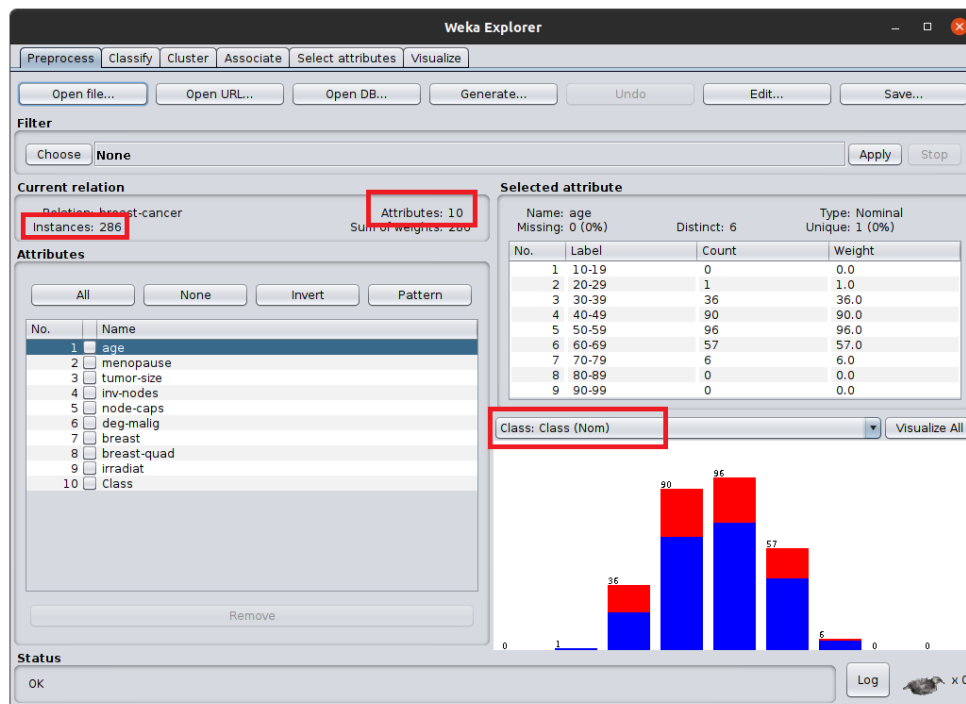
Ý nghĩa các nhóm điều khiển

- **Current relation:** Cung cấp các thông tin relation của file đang được chọn như tên relation, số lượng thuộc tính, số quan sát, tổng trọng số.
- **Attributes:** Hiển thị tất cả các thuộc tính của dữ liệu. Có thể chọn một thuộc tính bất kì để phân tích. Chọn nhiều thuộc tính bằng cách check hoặc dùng nhóm câu lệnh All, None, Invert, Pattern để thực hiện các chức năng khác như lấy thuộc tính ra khỏi dữ liệu hoặc ấn chuột phải để copy range.
- **Selected attribute** Nơi đây sẽ xuất hiện tất cả thông tin của thuộc tính đã chọn ở **Attributes** như tên, kiểu, số quan sát thiếu, số giá trị phân biệt, số giá trị độc nhất. Ngoài ra còn thể hiện các thông tin như label, count, weight nếu là thuộc tính định danh; min, max, trung bình, độ lệch chuẩn nếu là thuộc tính số. Cuối cùng là histogram hiển thị tần suất xuất hiện của các giá trị(chia bin nếu là thuộc tính kiểu số), nó có thể là stacked histogram theo thuộc tính làm lớp(class) được chọn khác.

2 Làm quen với Weka

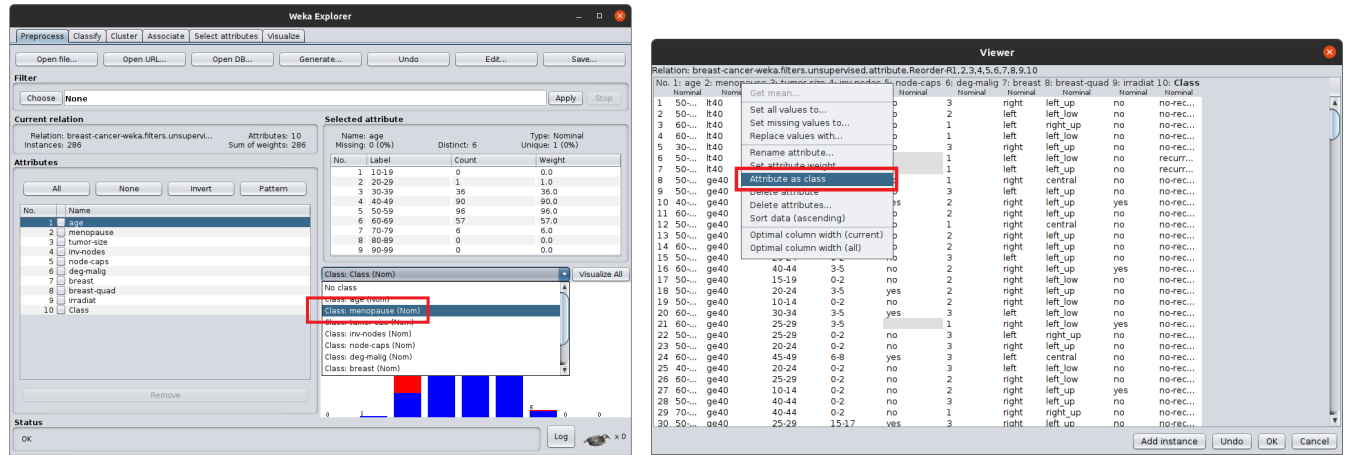
2.1 Đọc dữ liệu vào Weka

Câu hỏi 1,2 Tập dữ liệu có 286 mẫu, 10 thuộc tính.



Hình 2: Các thông tin cơ bản

Câu hỏi 3 Thuộc tính làm lớp là **Class**(xem hình 2). Có thể thay đổi thuộc tính làm lớp, bằng cách chọn ở select menu phía trên histogram hoặc ấn edit → chuột phải vào thuộc tính → Attribute as class → OK.



Hình 3: Ví dụ hai cách thay đổi thuộc tính **menopause** thành thuộc tính phân lớp

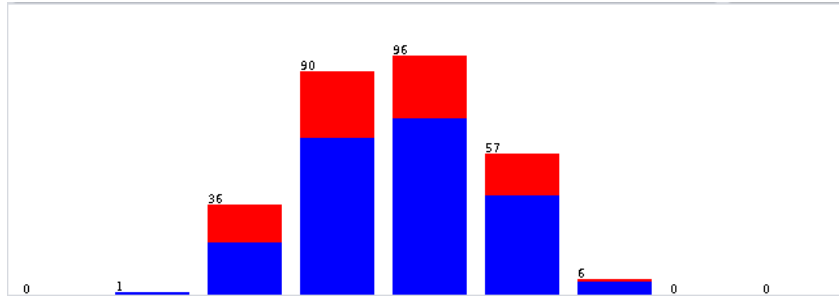
Name: menopause Missing: 0 (0%)	Distinct: 3	Type: Nominal Unique: 0 (0%)
Name: tumor-size Missing: 0 (0%)	Distinct: 11	Type: Nominal Unique: 0 (0%)
Name: inv-nodes Missing: 0 (0%)	Distinct: 7	Type: Nominal Unique: 1 (0%)
Name: node-caps Missing: 8 (3%)	Distinct: 2	Type: Nominal Unique: 0 (0%)
Name: breast Missing: 0 (0%)	Distinct: 2	Type: Nominal Unique: 0 (0%)
Name: breast-quad Missing: 1 (0%)	Distinct: 5	Type: Nominal Unique: 0 (0%)
Name: irradiat Missing: 0 (0%)	Distinct: 2	Type: Nominal Unique: 0 (0%)
Name: Class Missing: 0 (0%)	Distinct: 2	Type: Nominal Unique: 0 (0%)
Name: deg-malig Missing: 0 (0%)	Distinct: 3	Type: Nominal Unique: 0 (0%)
Name: age Missing: 0 (0%)	Distinct: 6	Type: Nominal Unique: 1 (0%)

Hình 4: Kiểm tra missing values

Câu hỏi 4 Như hình 4, có hai thuộc tính bị thiếu dữ liệu là **node-caps** và **breast-quad**. Trong đó nhiều nhất là **node-caps** với số lượng 8, ít nhất là **breast-quad** với số lượng 1. Một số cách để giải quyết vấn đề missing values:

- Xóa những dòng chứa missing value.
- Với cột dữ liệu kiểu số (liên tục hoặc không) thì có thể điền vào bằng mean, median hoặc mode của tất cả các giá trị còn lại
- Với cột dữ liệu kiểu categorical thì thay bằng giá trị có tần suất xuất hiện cao nhất trong cột, nếu số lượng missing values nhiều thì có thể phân tất cả vào một lớp mới.
- Sử dụng các thuật toán ít bị ảnh hưởng bởi việc thiếu dữ liệu như KNN, Naive Bayes

Câu hỏi 5 Lấy đồ thị đại diện khi ta chọn thuộc tính **age** ở **Attributes**, ta được như sau

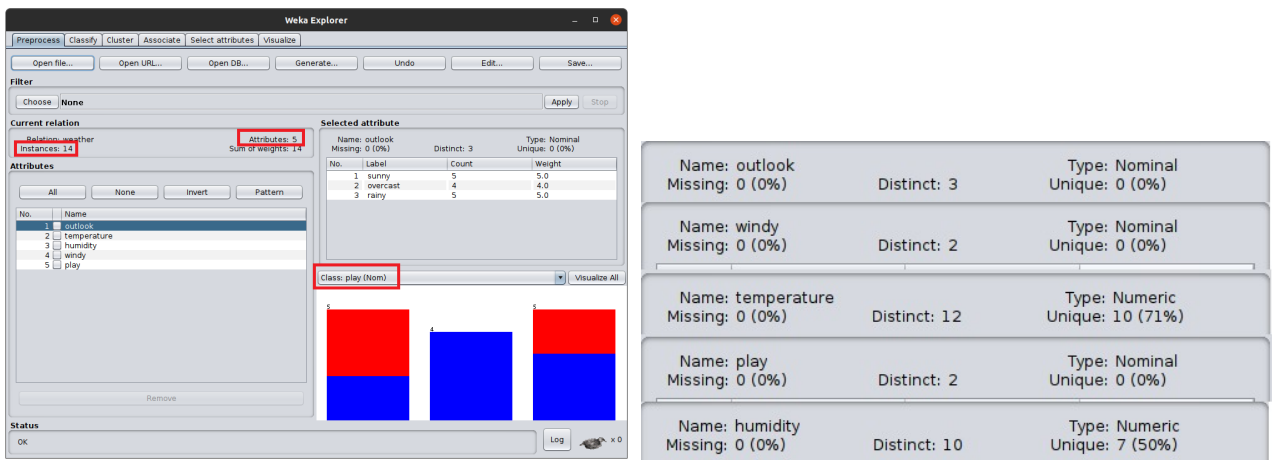


Hình 5: Biểu đồ có được sau khi chọn thuộc tính **age**

Đồ thị này thể hiện số lượng tổng số người bị ung thư vú đã khảo sát trong các độ tuổi khác nhau. Ngoài ra còn thể hiện tổng số người tái phát (được thể hiện bằng màu đỏ) và tổng số người không tái phát (được thể hiện bằng màu xanh) theo các độ tuổi, cũng như mối tương quan giữa các đại lượng này. Sau khi đặt nhãn cho các trục x là các nhóm tuổi, đặt nhãn cho trục y là số lượng, đặt legend cho đồ thị; ta có thể đặt tên là: "Tình trạng tái phát của các bệnh nhân ung thư vú ở các nhóm tuổi".

2.2 Khám phá tập dữ liệu Weather

Câu hỏi 1 Tập dữ liệu có 5 thuộc tính, 14 mẫu. Thuộc tính lớp là **play**. Các thuộc tính categorical là **outlook**, **windy**, **play** và thuộc tính numeric là **temperature**, **humidity**.



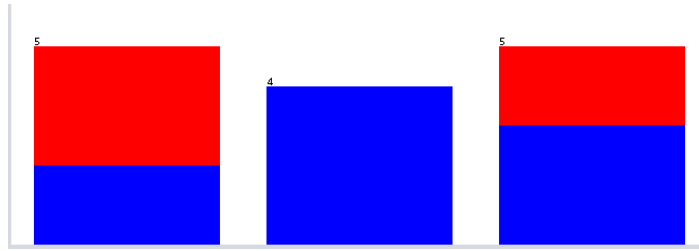
Hình 6: Thông tin tập dữ liệu, thuộc tính

Câu hỏi 2 Five-number summary của các thuộc tính **temperature** và **humidity**, weka chỉ cung cấp giá trị min và max.

	min	25%	50%	75%	max
temperature	64	69.25	72	78.75	85
humidity	65	71, 25	82.5	90	96

Câu hỏi 3 Trong các biểu đồ sau đây với các thuộc tính khác, nhãn của trục y là số lượng, màu xanh ứng với việc quyết định đi chơi, màu đỏ thì không.

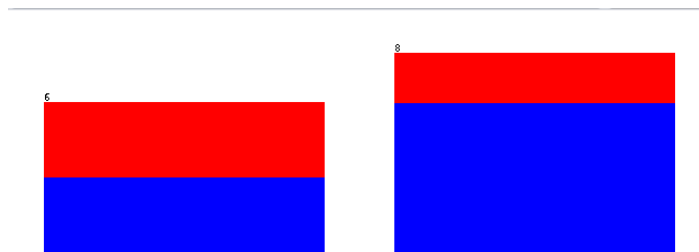
Với thuộc tính **outlook**, ta có biểu đồ hiển thị sau



Hình 7: Biểu đồ có được sau khi chọn thuộc tính **outlook**, nhãn của trục x từ trái qua phải là 'sunny', 'overcast', 'rainy'

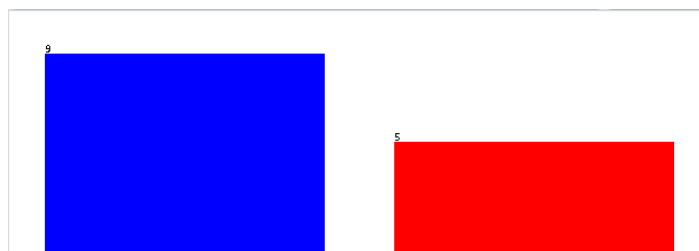
Ta thấy, tất cả mọi người chọn đi chơi nếu trời âm u. Khi trời nắng thì có xu hướng không đi, khi trời mưa thì có xu hướng đi chơi.

Với thuộc tính **windy**, ta có biểu đồ hiển thị sau



Hình 8: Biểu đồ có được sau khi chọn thuộc tính **windy**, nhãn của trục x từ trái qua phải là 'TRUE', 'FALSE'

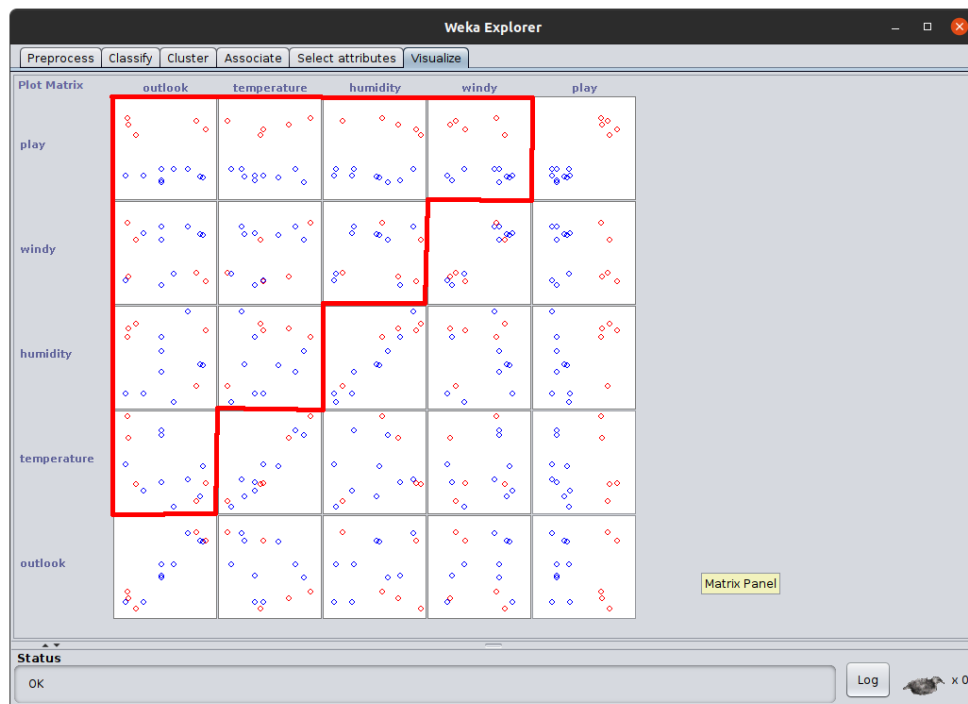
Biểu đồ cho thấy mọi người thích thú đi chơi khi trời lặng gió, khi trời có gió thì một nửa người đi, một nửa không. Với thuộc tính **play**, ta có biểu đồ hiển thị sau



Hình 9: Biểu đồ có được sau khi chọn thuộc tính **play**, nhãn của trục x từ trái qua phải là 'yes', 'no'

Biểu đồ cho thấy số người đi chơi nhiều hơn hẳn số người không đi

. **Câu hỏi 4** Các đồ thị ở đây là scatterplot matrix. Trông không có vẻ có mối tương quan với nhau, không thấy dạng tương quan tuyến tính positive correlation và negative correlation



Hình 10: Xem xét các đồ thị con trong vùng màu đỏ không thấy sự tương quan

2.3 Khám phá tập dữ liệu Tín dụng Đức

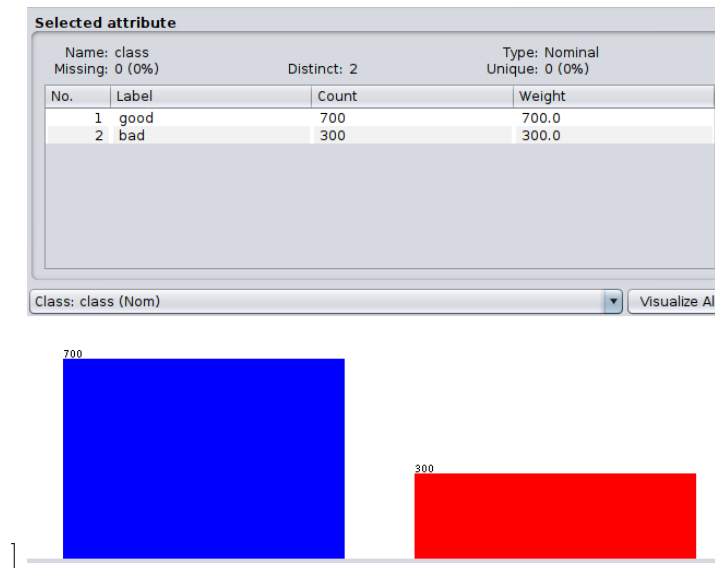
Câu hỏi 1 Nội dung của phần ghi chú mô tả về tập dữ liệu, cụ thể các thông tin quan trọng sau: tên bộ dữ liệu, nguồn dữ liệu, số mẫu, số lượng thuộc tính, mô tả thuộc tính; cùng một số thông tin khác.

Từ mô tả ta thấy tập dữ liệu có 10000 mẫu, 21 thuộc tính. Mô tả 5 thuộc tính như sau

Name: checking_status		Type: Nominal	
Missing: 0 (0%)		Unique: 0 (0%)	
Distinct: 4			
No.	Label	Count	Weight
1	<0	274	274.0
2	0<=X<200	269	269.0
3	>=200	63	63.0
4	no checking	394	394.0
Name: duration		Type: Numeric	
Missing: 0 (0%)		Unique: 5 (1%)	
Distinct: 33			
Statistic		Value	
Minimum		4	
Maximum		72	
Mean		20.903	
StdDev		12.059	
Name: credit_history		Type: Nominal	
Missing: 0 (0%)		Unique: 0 (0%)	
Distinct: 5			
No.	Label	Count	Weight
1	no credits/all paid	40	40.0
2	all paid	49	49.0
3	existing paid	530	530.0
4	delayed previously	88	88.0
5	critical/other existing cre...	293	293.0
Name: credit_amount		Type: Numeric	
Missing: 0 (0%)		Unique: 847 (85%)	
Distinct: 921			
Statistic		Value	
Minimum		250	
Maximum		18424	
Mean		3271.258	
StdDev		2822.737	
Name: savings_status		Type: Nominal	
Missing: 0 (0%)		Unique: 0 (0%)	
Distinct: 5			
No.	Label	Count	Weight
1	<100	603	603.0
2	100<=X<500	103	103.0
3	500<=X<1000	63	63.0
4	>=1000	48	48.0
5	no known savings	183	183.0

Hình 11: Mô tả 5 thuộc tính

Câu hỏi 2 Thuộc tính lớp là **class**, từ số liệu và đồ thị trong hình sau, ta thấy phân bố bị lệch về phía 'good'.



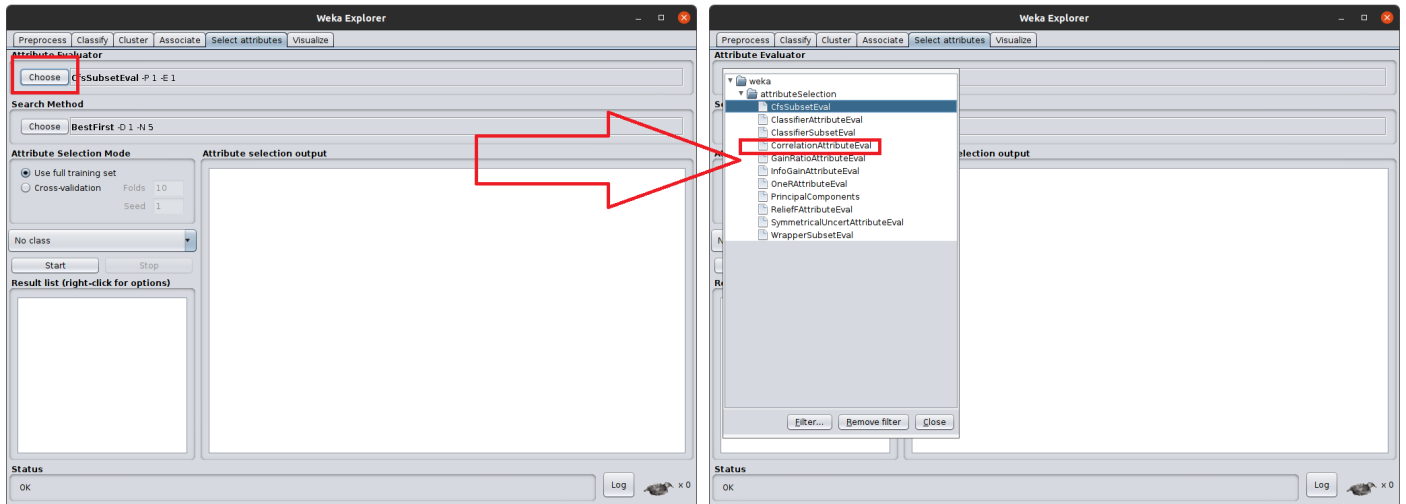
Hình 12: Phân bố thuộc tính phân lớp bị lệch

Câu hỏi 3

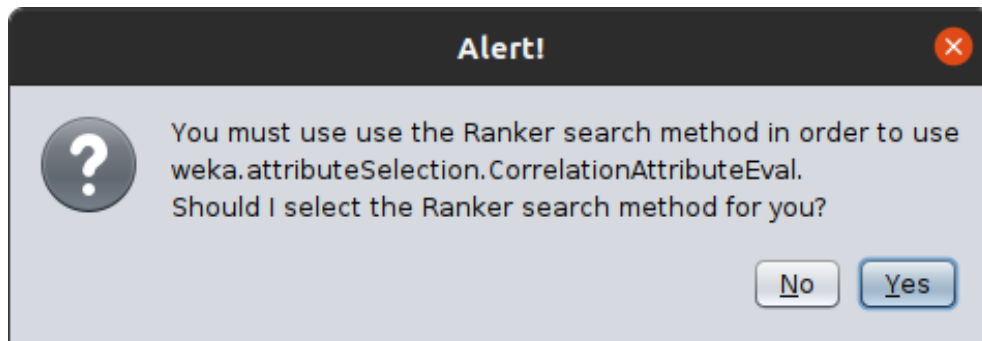
- **CfsSubsetEval**: Thuật toán tìm kiếm một tập con các thuộc tính có hiệu quả tốt (tương quan giữa các thuộc tính thấp, tương quan với thuộc tính lớp cao)
- **ClassifierAttributeEval**: Đánh giá giá trị của một thuộc tính bằng bộ phân loại mà người dùng chọn.
- **ClassifierSubsetEval**: Đánh giá các tập con thuộc tính dựa trên tập train hoặc một tập test riêng biệt.
- **CorrelationAttributeEval**: Đánh giá dựa trên tương quan của tập con thuộc tính với thuộc tính lớp
- **GainRatioAttributeEval**: Đánh giá dựa trên tỉ lệ lợi của thuộc tính trên thuộc tính lớp
- **InfoGainAttributeEval**: Đánh giá dựa trên thông tin mang lại của thuộc tính trên thuộc tính lớp
- **OneRAttributeEval**: Đánh giá giá trị của thuộc tính dựa trên bộ phân lớp OneR
- **PrincipalComponents**: Thực hiện phân tích thành phần chính và biến đổi dữ liệu
- **ReliefAttributeEval**: Đánh giá giá trị của thuộc tính bằng cách lấy mẫu liên tục và xem xét giá trị gần nhất của thuộc tính của các mẫu cùng hay khác lớp. Có thể sử dụng cả những lớp dữ liệu rời rạc hoặc liên tục.
- **SymmetricalUncertAttributeEval**: Đánh giá giá trị của thuộc tính dựa trên độ bất ổn định đối xứng trên thuộc tính lớp.
- **WrapperSubsetEval**: Đánh giá tập thuộc tính dựa trên sơ đồ học.

Câu hỏi 4 Sử dụng bộ lọc **CorrelationAttributeEval**

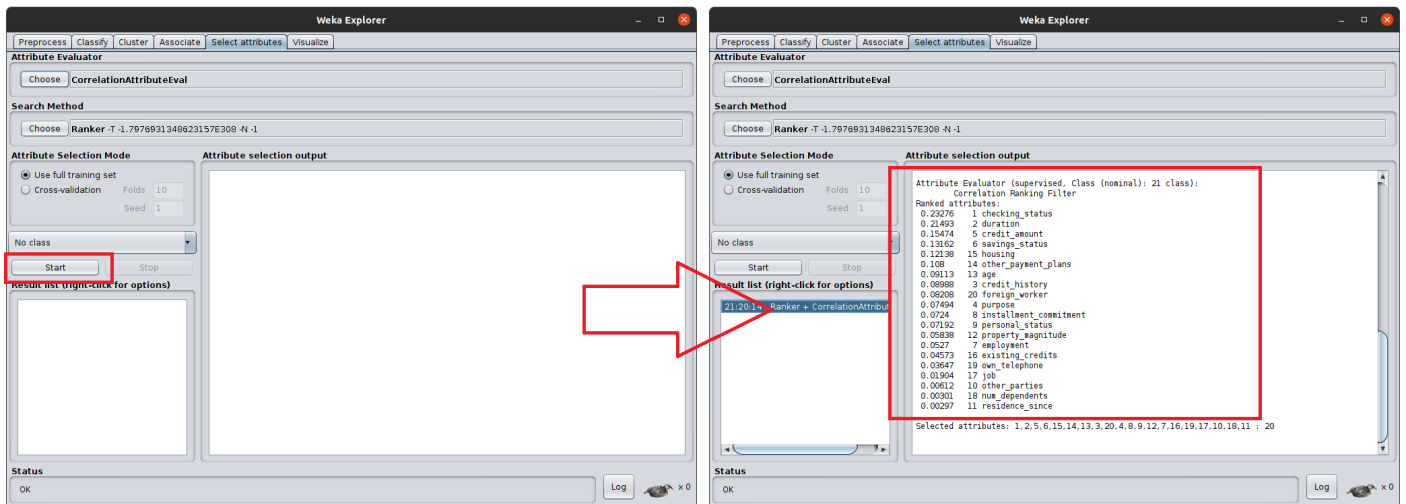
Từ tab **Select attributes**, chọn **Choose** → **CorrelationAttributeEval**



Hệ thống hỏi dùng phương pháp tìm kiếm là Ranker, chọn yes



Ấn **Start**, chờ một lúc và xem kết quả như hình sau



Từ bảng kết quả, ta chọn được 5 thuộc tính có tương quan cao nhất.

3 Cài đặt tiền xử lý dữ liệu

Một số quy ước của chương trình

- Cú pháp tham số dòng lệnh: `python3 main.py <tên chức năng> <input file> <các tùy chọn khác>`

- Tên của 8 chức năng là `list_missing`, `count_missing`, `impute`, `remove_col`, `remove_row`, `remove_duplicate`, `normalize`, `calculate`
- Tất cả các tùy chọn như sau, sẽ đề cập chi tiết hơn cho từng chức năng.

```

nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py -h
usage: main.py [-h] [-m METHOD] [-o OUTFILE] [-c COLUMNS [COLUMNS ...]]
               [--threshold THRESHOLD] [--formula FORMULA]

Process some functions.

optional arguments:
  -h, --help            show this help message and exit
  -m METHOD, --method METHOD
                        Method to be used by some functions
  -o OUTFILE, --outfile OUTFILE
                        Output file name
  -c COLUMNS [COLUMNS ...], --columns COLUMNS [COLUMNS ...]
                        List of input columns
  --threshold THRESHOLD
                        Percent, 50 by default
  --formula FORMULA     Formula to generate new column using existing columns
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$

```

- Để đơn giản, ta quy ước nếu tất cả các phần tử không thiếu của một cột là kiểu số thì cột có kiểu số, trường hợp khác là categorical
- Tất cả kiểu số đều quy về số thực.

3.1 Chức năng 1, liệt kê các cột thiếu phần tử

Chạy lệnh `python3 main.py list_missing house-prices.csv`, chương trình liệt kê các cột thiếu như hình bên trái

```

nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing house-prices.csv
There are missing values in these following columns:
LotFrontage
Alley
MasVnrType
MasVnrArea
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2
FireplaceQu
GarageType
GarageYrBlt
GarageFinish
GarageQual
GarageCond
PoolQC
Fence
MiscFeature
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$

```

```

nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py count_missing house-prices.csv
Number of rows with missing value: 1000
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$

```

3.2 Chức năng 2, đếm số hàng thiếu phần tử

Chạy lệnh `python3 main.py count_missing house-prices.csv`, chương trình in ra như hình bên phải

3.3 Chức năng 3, điền vào chỗ thiếu

Ở chức năng 3 sử dụng 3 option là **column**, **method**, **outfile**. Trong đó:

- **column** có thể nhận nhiều tham số để chỉ các cột cần điền vào. Nếu không được chỉ định thì mặc định sẽ điền vào toàn bộ các cột. Các cột có tên không hợp lệ sẽ bị bỏ qua.
- **method** chỉ phương pháp điền cho thuộc tính numeric, hỗ trợ mean và median, mặc định là mean.
- **outfile** chỉ tên của file cần lưu, mặc định là file hiện thời.

Ngoài ra với các thuộc tính categorical có nhiều mode, để dữ liệu cân bằng, chương trình thay các giá trị bị thiếu bởi lần lượt các mode. Kết quả thực nghiệm trên các lệnh:

- `python3 main.py impute house-prices.csv --column LotFrontage --outfile result.csv --method mean`
 - Tất cả các giá trị bị thiếu của **LotFrontage** được thay bằng trung bình của nó: 69.3035
 - Dùng chức năng 1 trên file result.csv thì không thấy cột LotFrontage(xem hình bên trái dưới đây)
- `python3 main.py impute house-prices.csv -c LotFrontage Alley --outfile result.csv -m median`
 - Tất cả các giá trị bị thiếu của **LotFrontage** được thay bằng trung vị của nó: 68
 - Tất cả các giá trị bị thiếu của **Alley** được thay bằng mode của nó: Grvl
 - Dùng chức năng 1 trên file result.csv thì không thấy cột LotFrontage, Alley(xem hình bên phải dưới đây)

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py impute house-prices.csv --column LotFrontage --outfile result.csv --method mean
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing result.csv
There are missing values in these following columns:
Alley
MasVnrType
MasVnrArea
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2
FireplaceQu
GarageType
GarageYrBlt
GarageFinish
GarageQual
GarageCond
PoolQC
Fence
MiscFeature
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$
```

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py impute house-prices.csv -c LotFrontage Alley --outfile result.csv -m median
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing result.csv
There are missing values in these following columns:
MasVnrType
MasVnrArea
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2
FireplaceQu
GarageType
GarageYrBlt
GarageFinish
GarageQual
GarageCond
PoolQC
Fence
MiscFeature
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$
```

- `python3 main.py impute house-prices.csv`
 - Tất cả giá trị bị thiếu ở bảng đều đã được điền
 - Lưu lại vào chính file house-prices.csv
 - Dùng chức năng 1,2 không phát hiện giá trị bị thiếu
- (các demo phía sau dùng lại file cũ trước khi bị thay đổi)

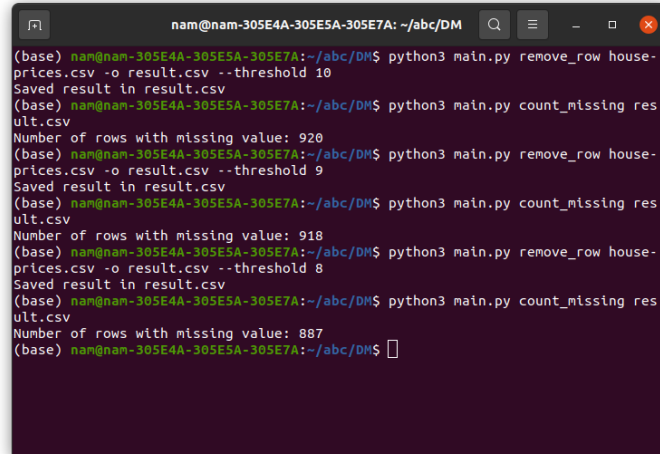
```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py impute house-prices.csv
Saved result in house-prices.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing house-prices.csv
There are missing values in these following columns:
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py count_missing house-prices.csv
Number of rows with missing value: 0
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$
```

3.4 Chức năng 4, xóa dòng với ngưỡng cho trước

Chức năng 4 cần 2 tham số, **threshold**, **outfile**

- **threshold** là ngưỡng xóa, tính theo phần trăm, mặc định là 50
- **outfile** tương tự như chức năng 3

Thử một vài giá trị ngưỡng, ta thu được các dòng thiếu còn lại như sau



```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_row house-
prices.csv -o result.csv --threshold 10
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py count_missing res
ult.csv
Number of rows with missing value: 920
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_row house-
prices.csv -o result.csv --threshold 9
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py count_missing res
ult.csv
Number of rows with missing value: 918
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_row house-
prices.csv -o result.csv --threshold 8
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py count_missing res
ult.csv
Number of rows with missing value: 887
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$
```

3.5 Chức năng 5, xóa cột với ngưỡng cho trước

Chức năng 5 cần 2 tham số, **threshold**, **outfile**, hoạt động giống hệt chức năng 4. Thử một vài giá trị ngưỡng, ta thu được các dòng thiếu còn lại

```
(base) nam@nam-305E4A-305E5A-305E7A: ~/abc/DM$ python3 main.py remove_col house-prices.csv -o result.csv --threshold 5
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing result.csv
There are missing values in these following columns:
LotFrontage
MasVnrArea
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2
GarageType
GarageYrBlt
GarageFinish
GarageQual
GarageCond
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$

(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_col house-prices.csv -o result.csv --threshold 10
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing result.csv
There are missing values in these following columns:
MasVnrArea
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2
GarageType
GarageYrBlt
GarageFinish
GarageQual
GarageCond
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$

(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_col house-prices.csv -o result.csv --threshold 5
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing result.csv
There are missing values in these following columns:
MasVnrArea
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$

(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_col house-prices.csv -o result.csv --threshold 2
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py list_missing result.csv
There are missing values in these following columns:
MasVnrArea
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$
```

Hình 13: Ngưỡng lần lượt 50-10-5-2

3.6 Chức năng 6, xóa lặp

Chức năng 6 cần 1 tham số **outfile**.

Chạy `python3 main.py remove_duplicate house-prices.csv -o result.csv`, file kết quả chỉ còn 716 dòng thiếu

```
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py remove_duplicate house-prices.csv -o result.csv
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py count_missing result.csv
Number of rows with missing value: 716
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$
```

3.7 Chức năng 7, chuẩn hóa

Ở chức năng 7 sử dụng 3 option là **column**, **method**, **outfile**. Trong đó:

- **column** có thể nhận nhiều tham số để chỉ các cột cần chuẩn hóa. Nếu không được chỉ định thì mặc định sẽ chuẩn hóa toàn bộ các cột. Các cột categorical bị bỏ qua, các cột tên không hợp lệ bị bỏ qua
- **method** chỉ phương pháp chuẩn hóa cho thuộc tính numeric, hỗ trợ min-max và Z-score, mặc định là min-max.
- **outfile** chỉ tên của file cần lưu, mặc định là file hiện thời.

Với hai lệnh sau đây, ta đã chuẩn hóa thuộc tính **Id** theo hai cách:

```
python3 main.py normalize house-prices.csv -c Id -o result.csv -m Z-score
```

```
python3 main.py normalize house-prices.csv -c Id -o result.csv -m min-max
```

Kết quả kiểm tra với thư viện pandas, ta thấy kết quả chênh lệch rất nhỏ.

(Ở 4 hình dưới lệnh có hơi khác là có dấu ngoặc đơn ở min-max, Z-score là do em không để ý à, nhưng **kết quả không thay đổi**)

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py normalize house-prices.csv -c Id -o result.csv -m 'Z-score'
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3
Python 3.8.8 (default, Apr 13 2021, 19:58:26)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> df = pd.read_csv('house-prices.csv')
>>> df_result = pd.read_csv('result.csv')
>>> Id = df['Id']
>>> Id_result = df_result['Id']
>>> normalized_Id = (Id - Id.mean())/(Id.std())
>>> print(normalized_Id - Id_result)
0      0.000000e+00
1      2.220446e-16
2      0.000000e+00
3      0.000000e+00
4      0.000000e+00
...
995     0.000000e+00
996     0.000000e+00
997     0.000000e+00
998     0.000000e+00
999     0.000000e+00
Name: Id, Length: 1000, dtype: float64
>>>
```

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py normalize house-prices.csv -c Id -o result.csv -m 'min-max'
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3
Python 3.8.8 (default, Apr 13 2021, 19:58:26)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> df = pd.read_csv('house-prices.csv')
>>> df_result = pd.read_csv('result.csv')
>>> Id = df['Id']
>>> Id_result = df_result['Id']
>>> normalized_Id = (Id - Id.min())/(Id.max() - Id.min())
>>> print(normalized_Id - Id_result)
0      0.000000e+00
1      0.000000e+00
2     -1.110223e-16
3      0.000000e+00
4      0.000000e+00
...
995     0.000000e+00
996     5.551115e-17
997     0.000000e+00
998     0.000000e+00
999     0.000000e+00
Name: Id, Length: 1000, dtype: float64
>>>
```

Thử với 2 lệnh khác:

```
python3 main.py normalize house-prices.csv -c Id MSSubClass MSZoning -o result.csv -m Z-score
```

```
python3 main.py normalize house-prices.csv -c Id MSSubClass MSZoning -o result.csv -m min-max
```

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py normalize house-prices.csv -c Id MSSubClass MSZoning -o result.csv -m 'Z-score'
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3
Python 3.8.8 (default, Apr 13 2021, 19:58:26)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> MSC = pd.read_csv('house-prices.csv')['MSSubClass']
>>> MSC_result = pd.read_csv('result.csv')['MSSubClass']
>>> normalized_MSC = (MSC - MSC.mean())/(MSC.std())
>>> print(normalized_MSC - MSC_result)
0      0.000000e+00
1      0.000000e+00
2     -5.551115e-17
3      0.000000e+00
4      0.000000e+00
...
995     5.551115e-17
996     5.551115e-17
997     5.551115e-17
998     5.551115e-17
999     0.000000e+00
Name: MSSubClass, Length: 1000, dtype: float64
>>>
```

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py normalize house-prices.csv -c Id MSSubClass MSZoning -o result.csv -m 'min-max'
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3
Python 3.8.8 (default, Apr 13 2021, 19:58:26)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> MSC = pd.read_csv('house-prices.csv')['MSSubClass']
>>> MSC_result = pd.read_csv('result.csv')['MSSubClass']
>>> normalized_MSC = (MSC - MSC.min())/(MSC.max() - MSC.min())
>>> print(normalized_MSC - MSC_result)
0      0.000000e+00
1      0.000000e+00
2     2.775558e-17
3     6.938894e-18
4      0.000000e+00
...
995     2.775558e-17
996     2.775558e-17
997     2.775558e-17
998     2.775558e-17
999     0.000000e+00
Name: MSSubClass, Length: 1000, dtype: float64
>>>
```

Ta được kết quả tương tự với cột **MSSubClass**, cột **Id** kết quả như ví dụ trên, cột **MSZoning** là categorical nên chương trình tự bỏ qua.

3.8 Chức năng 8, tính toán

Chức năng 4 cần 2 tham số, **formula**, **outfile**

- **formula** được định dạng là <tên cột mới>=<các phép tính trên cột cũ>. Với các phép toán phức tạp có dấu ngoặc tròn, phải đặt **formula**(hoặc chỉ phần sau dấu bằng) trong cặp dấu ngoặc kép.
- **outfile** tương tự như các chức năng trên.

Nếu có thuộc tính bị thiếu hoặc có phép chia cho 0, biểu thức sẽ bị thiếu. Nếu có các thuộc tính có tên không hợp lệ hoặc là thuộc tính categorical thì chương trình in ra lỗi.

Một số ví dụ:

- `python3 main.py calculate house-prices.csv -o result.csv --formula MyCol=Id`
- `python3 main.py calculate house-prices.csv -o result.csv --formula MyCol=-Id`
- `python3 main.py calculate house-prices.csv -o result.csv --formula MyCol=-Id+MSSubClass`
- `python3 main.py calculate house-prices.csv -o result.csv --formula MyCol=-Id+2*MSSubClass-LotArea/OverallQual`
- `python3 main.py calculate house-prices.csv -o result.csv --formula MyCol="(Id*MSSubClass*LotArea/(OverallQual+OverallCond))-LotArea/Id"`

Kiểm tra trường hợp phức tạp nhất với pandas, và ta thấy kết quả giống nhau.

```
nam@nam-305E4A-305E5A-305E7A: ~/abc/DM
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3 main.py calculate house-prices.csv -o result.csv --formula MyCol="(Id*MSSubClass*LotArea/(OverallQual+OverallCond))-LotArea/Id"
Saved result in result.csv
(base) nam@nam-305E4A-305E5A-305E7A:~/abc/DM$ python3
Python 3.8.8 (default, Apr 13 2021, 19:58:26)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> df = pd.read_csv('house-prices.csv')
>>> result_by_outputfile = pd.read_csv('result.csv')['MyCol']
>>> result_by_pandas = result_by_pandas = (df.Id*df.MSSubClass*df.LotArea/(df.OverallQual+df.OverallCond))-df.LotArea/df.Id
>>> result_by_outputfile - result_by_pandas
0      3.725290e-09
1      1.490116e-08
2      0.000000e+00
3      0.000000e+00
4      0.000000e+00
...
995    7.450581e-09
996    0.000000e+00
997    0.000000e+00
998    0.000000e+00
999    0.000000e+00
Length: 1000, dtype: float64
>>>
```