UNIVERSITY OF SCIENCE - VNUHCM

Department of Computer Science

Teaching Assistant: Nguyen Ngoc Duc

Instructor: Le Hoai Bac

Nguyen Bao Long

Lab 01: Data Preprocessing and Data exploration

Group 1

Cao Hoai Yen Vy, Leader

Au Duong Khang

Due Date: October 22, 2023

Submitted: November 1, 2023

Mục lục

1 Purpose

- Hands-on exploring data through the application of support tools provided by the open-source software WEKA.
- Hands-on preprocessing data in Python.

2 Overview

2.1 Group Information

ID	Name	Email
21127205	**Cao Hoai Yen Vy**	**chyvy21@clc.fitus.edu.vn**
21127621	Au Duong Khang	adkhang21@clc.fitus.edu.vn

****Name of leader is **bold**.*

2.2 Contribution of group member

Objectives	Task	Member
3.1 Install WEKA	Requirement 1	Khang
	Requirement 2	Vy
3.2 Getting Acquainted With WEKA	3.2.1 Exploring Breast Cancer data set	Vy
	3.2.2 Exploring Weather data set	Vy
	3.2.3 Exploring Credit in Germany data se	Khang
3.3 Preprocessing Data in Python	1. Extract columns with missing values	Vy
	2. Count the number of lines with missing data.	Khang
	3. Fill in the missing value using mean, median (for numeric properties) and mode (for the categorical attribute)	Vy
	4. Deleting rows containing more than a particular number of missing values	Khang
	5. Deleting columns containing more than a particular number of missing values	Vy
	6. Delete duplicate samples	Khang
	7. Normalize a numeric attribute using min-max and Z-score methods.	Vy
	8. Performing addition, subtraction, multiplication, and division between two numerical attributes.	Khang
	9. Write command line, read file csv and export file csv	Khang
Present a report in English		Vy
		Khang

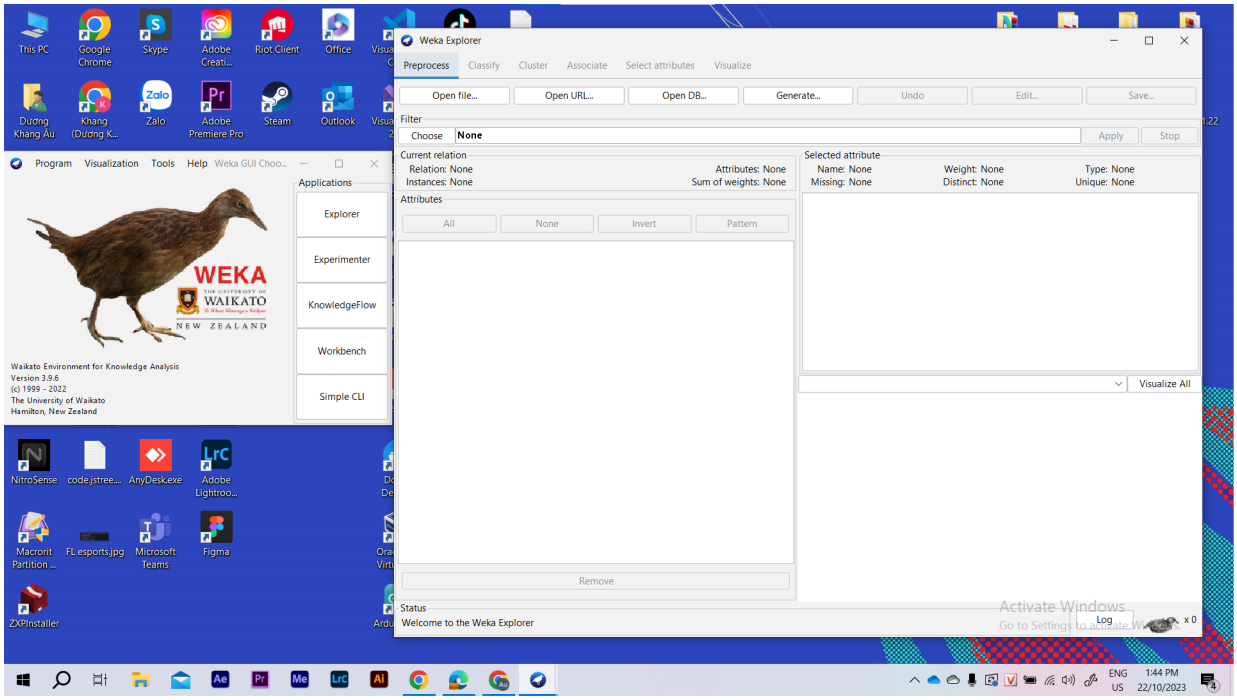
Objectives	Task	Completion
3.1 Install WEKA	Requirement 1	100%
	Requirement 2	100%
3.2 Getting Acquainted With WEKA	3.2.1 Exploring Breast Cancer data set	100%
	3.2.2 Exploring Weather data set	100%
	3.2.3 Exploring Credit in Germany data se	100%
3.3 Preprocessing Data in Python	1. Extract columns with missing values	100%
	2. Count the number of lines with missing data.	100%
	3. Fill in the missing value using mean, median (for numeric properties) and mode (for the categorical attribute)	100%
	4. Deleting rows containing more than a particular number of missing values	100%
	5. Deleting columns containing more than a particular number of missing values	100%
	6. Delete duplicate samples	100%
	7. Normalize a numeric attribute using min-max and Z-score methods.	100%
	8. Performing addition, subtraction, multiplication, and division between two numerical attributes.	100%
	9. Write command line, read file csv and export file csv	100%
	10. Test with house-prices data set with various cases	100%
Summary		100%

3 Requirement

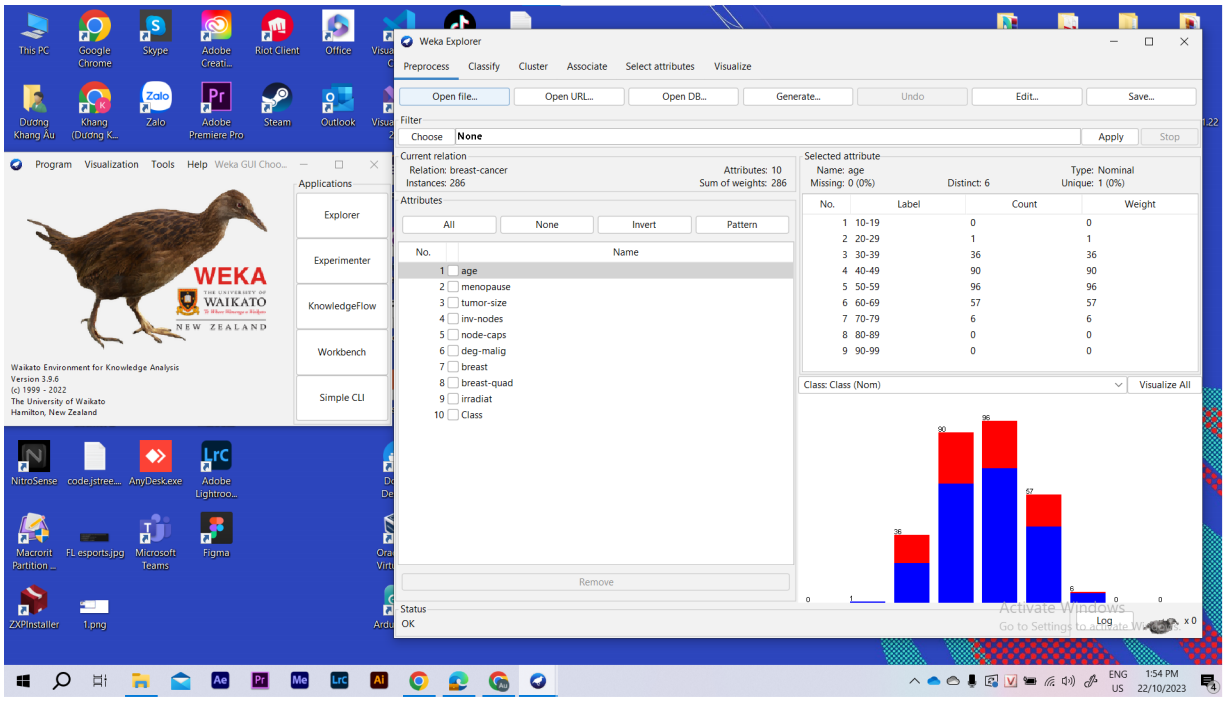
3.1 Install WEKA

3.1.1 Requirement 1

Window of WEKA Explorer



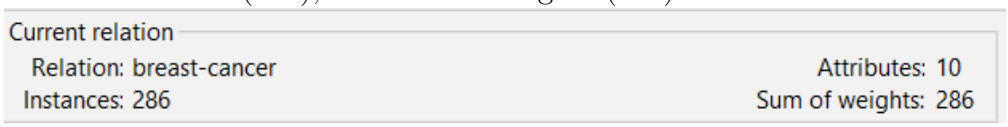
Window when open file .arf (breast-cancer.arf)



3.1.2 Requiredment 2: Explain meaning of some tag in Preprocess tag

We use file namely breast-cancer.arf.

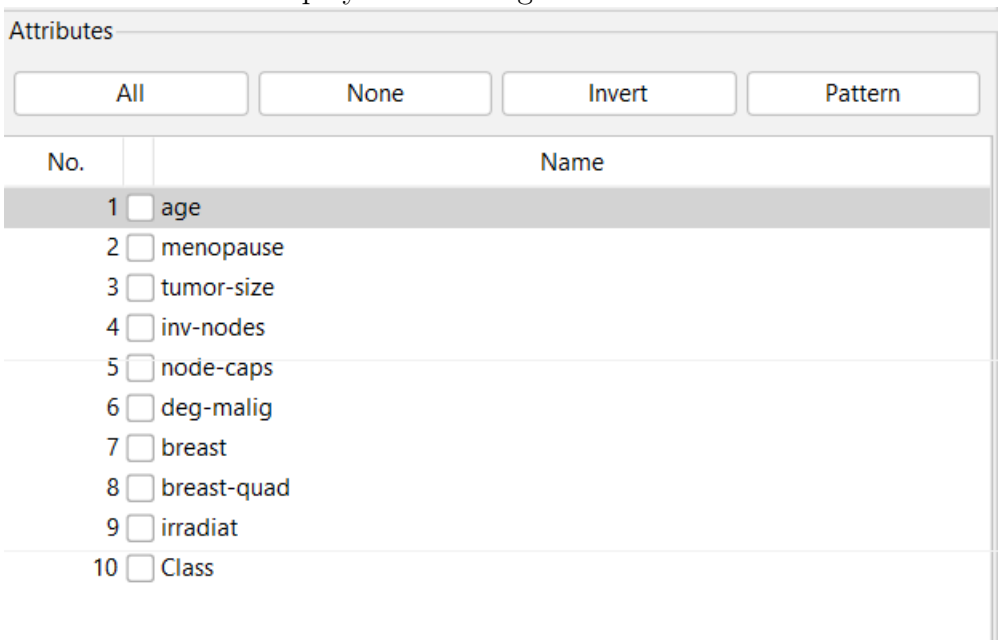
- **Current Relation**: It shows name of the database that is currently loaded (breast-cancer), total attributes (the fields) (10), and number of rows in the table, which named Instances (286), and sum of weights (286).



- **Attributes**: Displays the various fields in the database.

The breast cancer database contain 10 fields - age, menopause, tumor-size, inv-nodes, node-caps, deg-malig, breast, breast-quad, irradiat, class.

When you select an attribute from this list by clicking on it, further details on the attribute itself are displayed on the right hand side.



- **Selected Attribute**: We can observe the displayed attribute name (Name), the type of that attribute (Type), the number of missing values (missing), the number of distinct values (Distinct), and the number of unique values (Unique).

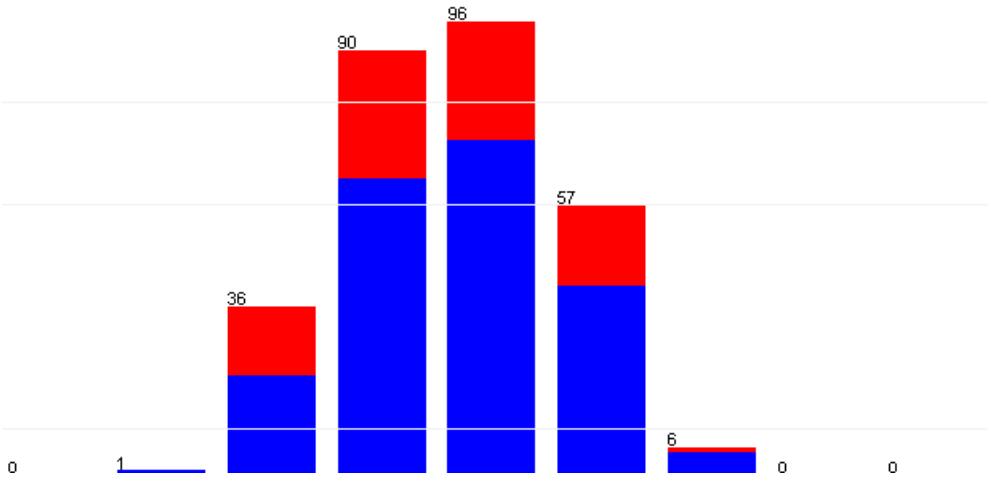
Selected attribute

Name: age Type: Nominal

Missing: 0 (0%) Distinct: 6 Unique: 1 (0%)

No.	Label	Count	Weight
1	10-19	0	0
2	20-29	1	1
3	30-39	36	36
4	40-49	90	90
5	50-59	96	96
6	60-69	57	57
7	70-79	6	6
8	80-89	0	0
9	90-99	0	0

Class: Class (Nom) Visualize All



  - The table underneath this information shows the nominal values of the attribute you have selected.
  - It also shows the count and weight in terms of a percentage for each nominal value.

Other tag in WEKA Explorer:

- **Preprocess**: This section allows you to select a data file and perform data preprocessing tasks to make it suitable for various machine learning algorithms. Preprocessing

tasks can include handling missing values, transforming data, and selecting relevant features.

- **Classify**: In this section, you can access a variety of supervised machine learning algorithms for data classification. These algorithms include Support Vector Machines (SVM), logistic regression, random forest, and more. You can build and evaluate models to classify data into different classes.

- **Cluster**: This section offers several unsupervised machine learning clustering algorithms. These algorithms, such as SimpleKMeans, FilteredClusterer, and HierarchicalClusterer, allow you to group similar data points into clusters based on their characteristics.

- **Associate**: In this section, you can use association rule mining algorithms to discover frequent itemsets or patterns in your data. Algorithms like Apriori, FilteredAssociator, and FPGrowth are available to find commonly occurring patterns in your dataset.

- **Select Attribute**: This section provides tools for attribute selection and feature engineering. You can choose relevant attributes based on various algorithms like ClassifierSubsetEval and PrincipalComponents. This helps in identifying attributes that are highly correlated or important for your analysis.

- **Visualize**: This section allows you to create visual representations of your data after preprocessing. You can explore the relationships between attributes by generating various plots and graphs, helping you gain insights into the data's distribution and patterns. These visualizations are often presented in pairs of attributes for easy analysis.

3.2 Getting Acquainted With WEKA

3.2.1 Exploring Breast Cancer data set

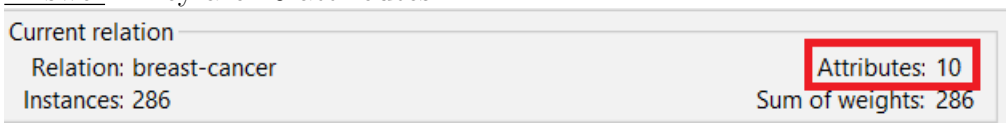
After loading data file namely **breast-cancer.arf** into WEKA explorer successfully, we will answer some questions or perform requirements in the followings:

- How many instances does this data set have?

  Answer: This data set has 286 instances

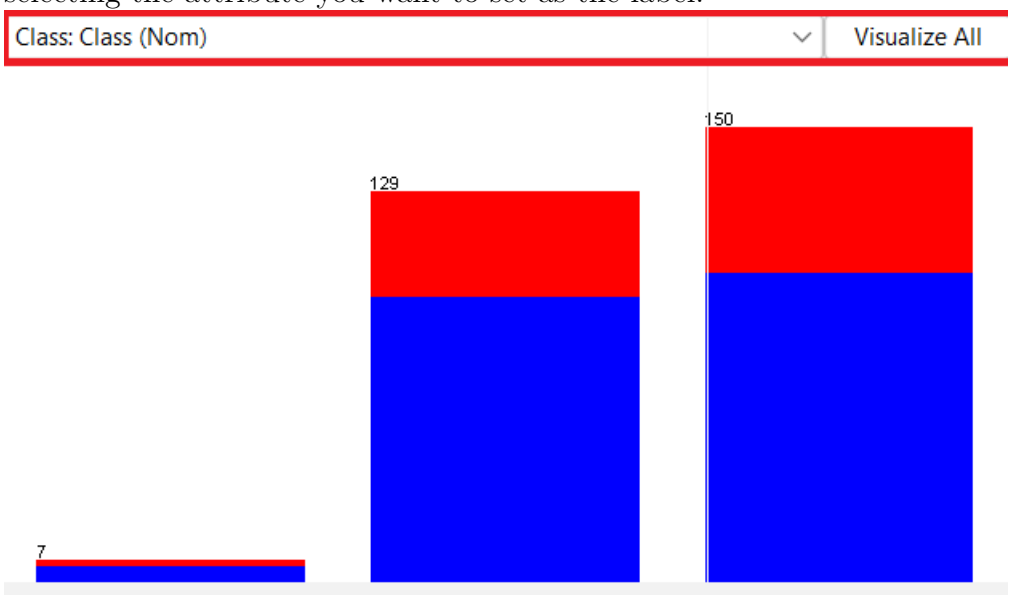
- How many attributes does this data set have?

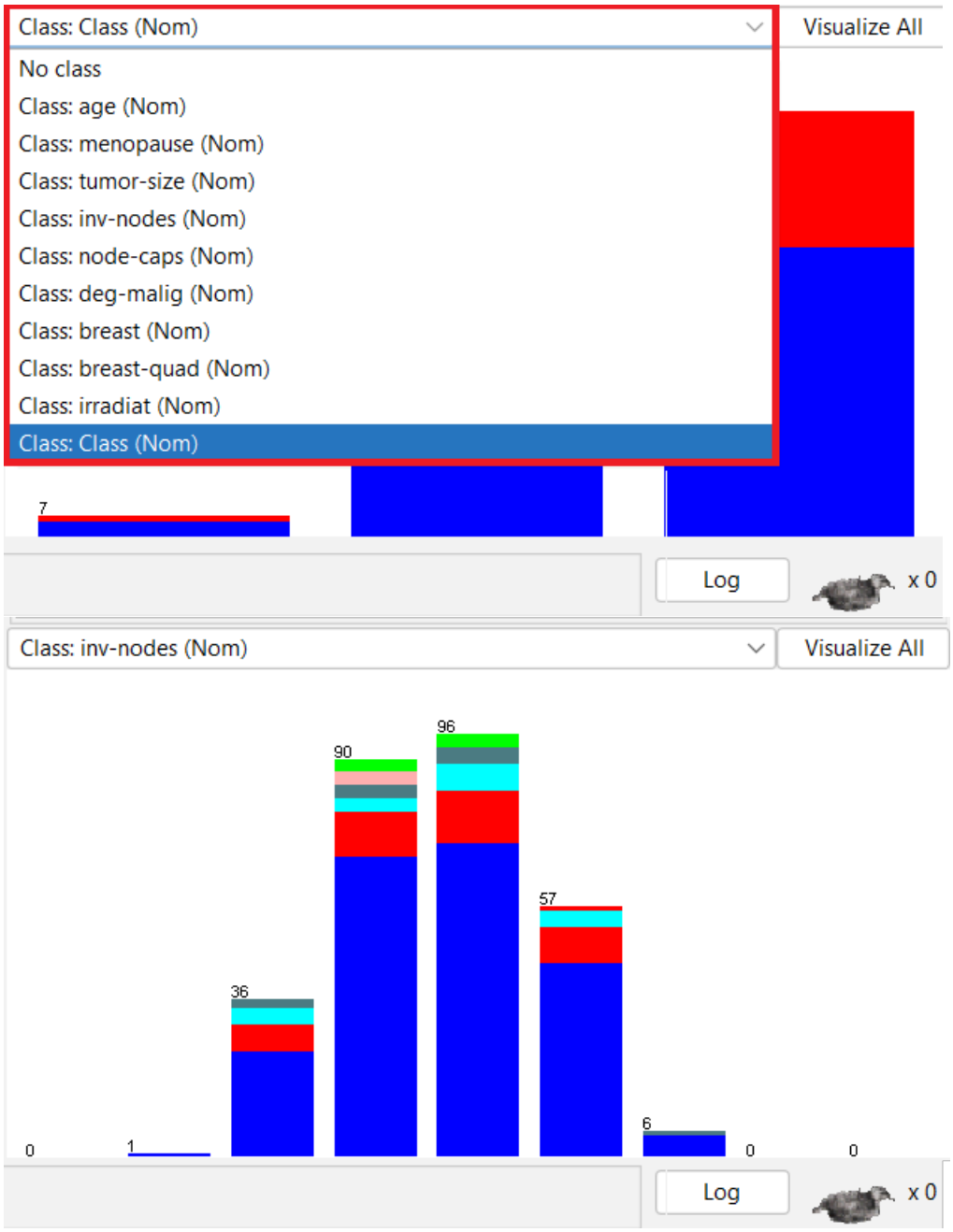
  Answer: They are 10 attributes.



- Which attribute is used for the label? Can it be changed? How?

  Answer: In the breast cancer dataset, this would typically be the "Class" attribute. It can be changed by clicking on the "Class" drop down below Selected attribute and selecting the attribute you want to set as the label.





- What is the meaning of each attribute?

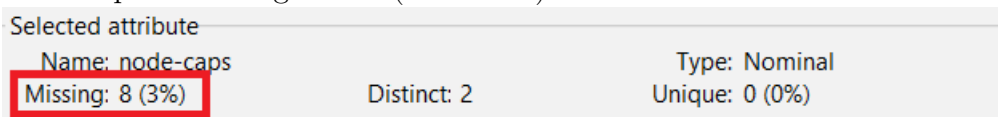
  Answer:

STT	Attribute	Meaning
1	age	This is the age of the patient at the time of diagnosis. This attribute indicates the patient's age.
2	menopause	This attribute indicates whether the patient is in the pre- or post-menopausal stage at the time of diagnosis. This can have an impact on the development of breast cancer.
3	tumor-size	This is the largest diameter (measured in mm) of the tumor removed. This attribute specifies the size of the tumor.
4	inv-nodes	This attribute indicates the number of axillary lymph nodes containing cancer in a range from 0 to 39 that can be seen during histological examination.
5	node-caps	This attribute indicates whether the tumor has spread to other axillary lymph nodes or not.
6	deg-malig	This is the histological grade (ranging from 1 to 3) of the tumor. It reflects the degree of abnormality of cancer cells within the tumor.
7	breast	This attribute specifies the location of breast cancer, whether it's in the left or right breast.
8	breastquad	This attribute describes the quadrant of the breast by dividing it into four quadrants using the nipple as the center point.
9	irradiat	This attribute indicates whether the patient has undergone radiation therapy. Radiation therapy is a treatment method using high-energy X-rays to destroy cancer cells.
10	class	This attribute classifies whether the patient has a relapse of the disease after treatment or not. Typically, it can be "relapse" or "no relapse."

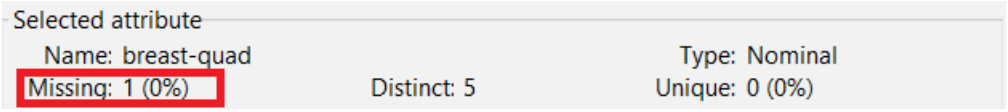
- Lets investigate the missing value status in each attribute and describe in general ways to solve the problem of missing values.

  Answer: There are missing value status in 2 attributes: node-caps and breast-squad

  – Node-caps: 8 missing values (about 3%)

  – Breast-quad: 1 missing value (about 0%)

  General ways to solve the problem of missing values: Handling missing values falls generally into two categories. The two categories are as follows:

  – Deletion

  – Imputation

  1. Deletion:

  – **List-wise Deletion (Row Deletion)**: Remove entire rows with missing values. Suitable when missing values are few and random but can result in significant data loss.

- **Pairwise Deletion (Column Deletion)**: Omit only the attribute with missing values, preserving data for other attributes in the same row. Useful for analyzing different data subsets but can introduce bias.

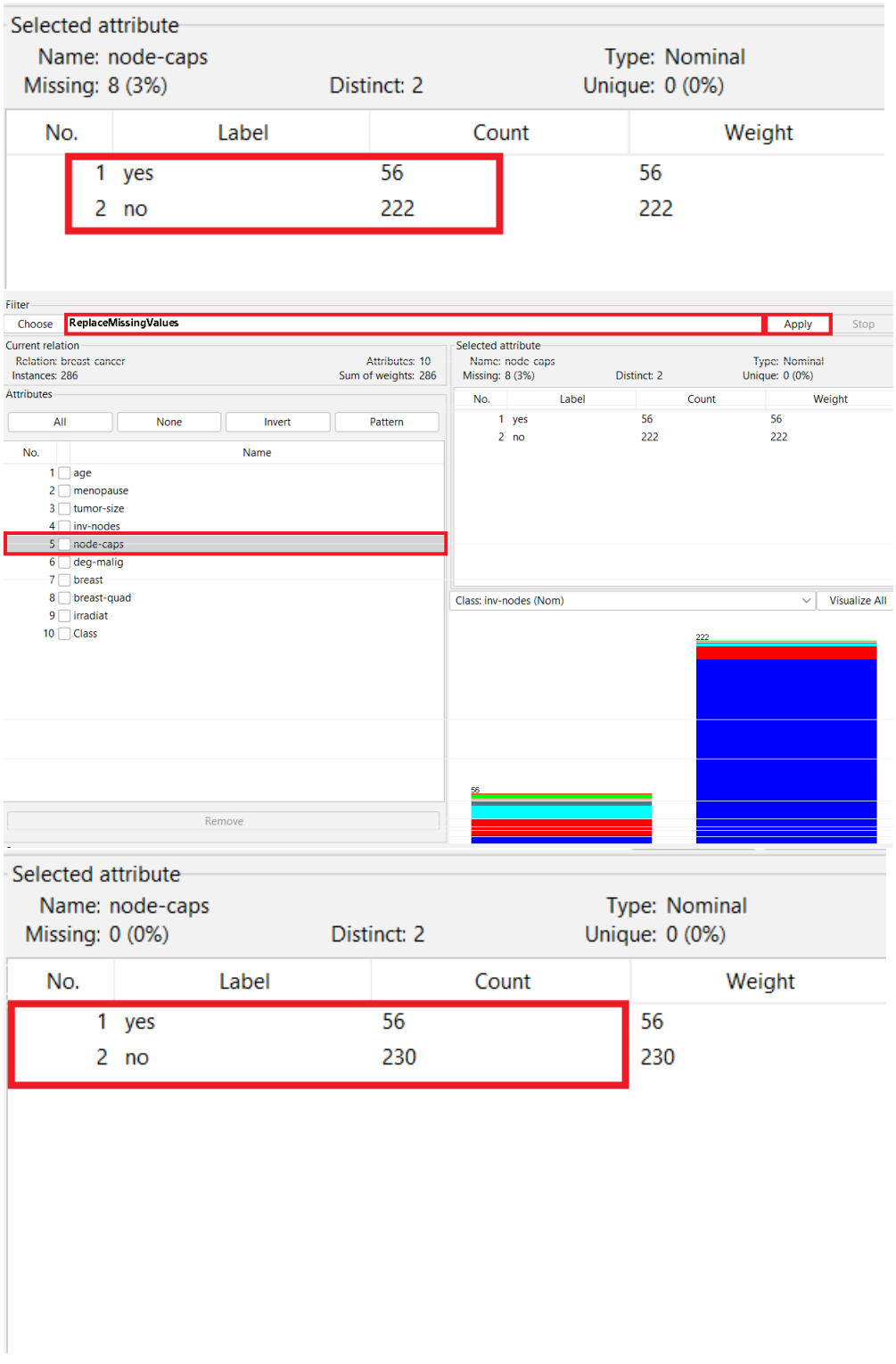
2. Imputation:

- **Mean, Median, or Mode Imputation**: Replace missing numeric values with the attribute's mean, median, or mode. Simple but may not work well for non-normally distributed data.
- **Regression Imputation**: Use regression models to predict missing values based on attribute relationships, suitable for data with strong associations.
- **K-Nearest Neighbors (K-NN) Imputation**: Fill missing values with the average of similar data points. Effective for both numeric and categorical data.
- **Multiple Imputation**: Generate multiple imputed datasets, accounting for imputation uncertainty and introducing data variability.
- **Domain-Specific Imputation**: Experts may suggest specialized methods based on their knowledge of the data and problem context.

- Lets propose solutions to the problem of missing values in the specific attribute.

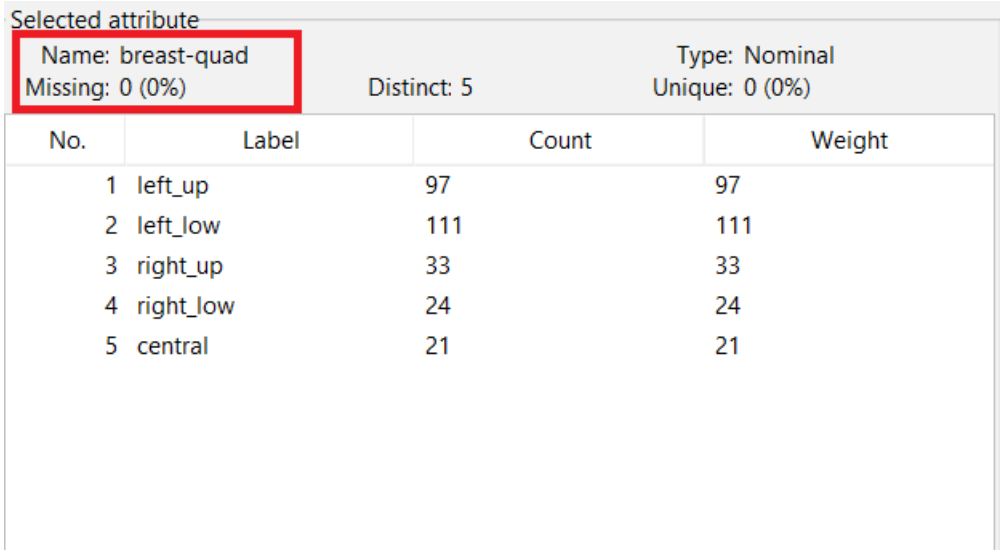
  Answer:

  - "node-caps" attribute: node-caps is classify attribute, the count of value "no" four times as many the count of value "yes". Because of that we choose mode value is "no" for missing value. In WEKA, select Choose $\longrightarrow$ filters$\longrightarrow$ unsupervised$\longrightarrow$ attribute$\longrightarrow$ ReplaceMissingValues$\longrightarrow$ Apply.



– "breast-quad" attribute: "breast-quad" attribute is attribute identifier, so we

choose label which has the largest count. It is "left-low". The way handle missing value like the same the "node-caps" attribute. So we has the result:



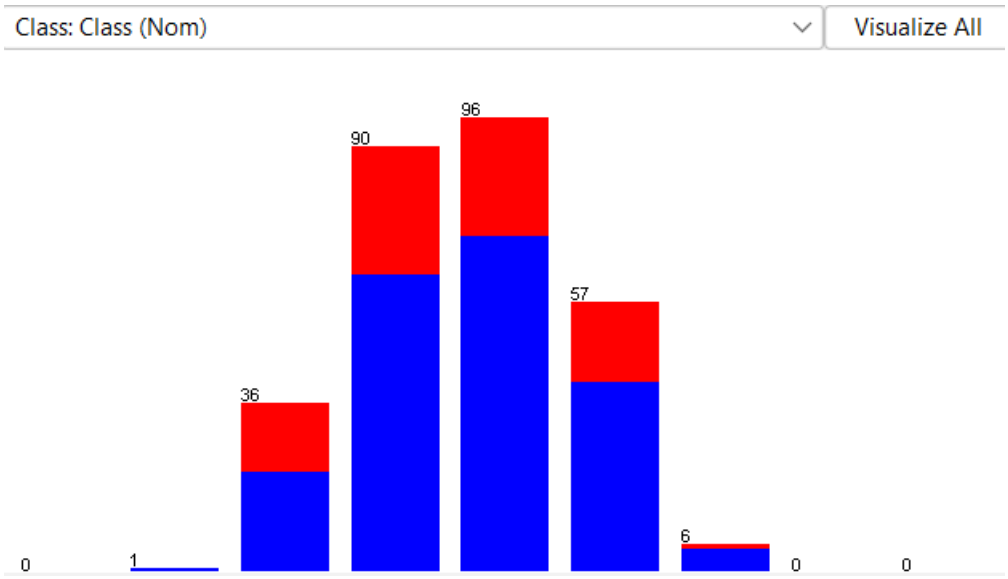
No.	Label	Count	Weight
1	left_up	97	97
2	left_low	111	111
3	right_up	33	33
4	right_low	24	24
5	central	21	21

- Lets explain the meaning of the chart in the WEKA Explorer. Setting the title for it and describing its legend.

  Answer:

  1. Meaning of the chart:

  - Chart type: WEKA Explorer includes scatter plots, line charts, histogram...
  - X-axis and Y-axis: Determine which attributes or variables are represented on the x-axis and y-axis. This indicates the relationships being visualized, such as the correlation between two attributes.
  - For example, if we choose attribute "Age" with label "Class". We will have a chart like this:



– The chart represents the number of breast cancer patients by age, categorized as either non-recurrence-events (in blue) or recurrence-events (in red).

2. Setting the title:

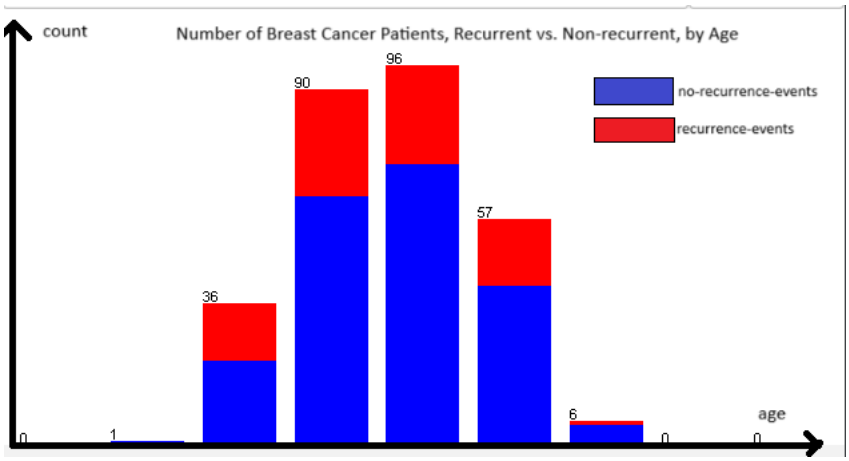
– Title: Number of Breast Cancer Patients, Recurrent vs. Non-recurrent, by Age.

– Legend:

∗ **Blue**: Non-recurrent Breast Cancer Patients

∗ **Red**: Recurrent Breast Cancer Patients

3. Image:

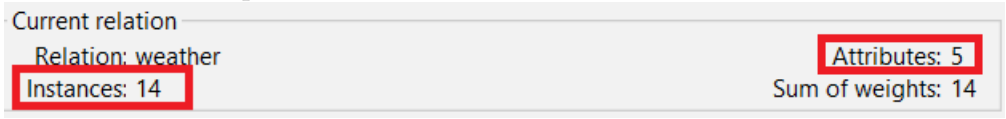


3.2.2 Exploring Weather data set

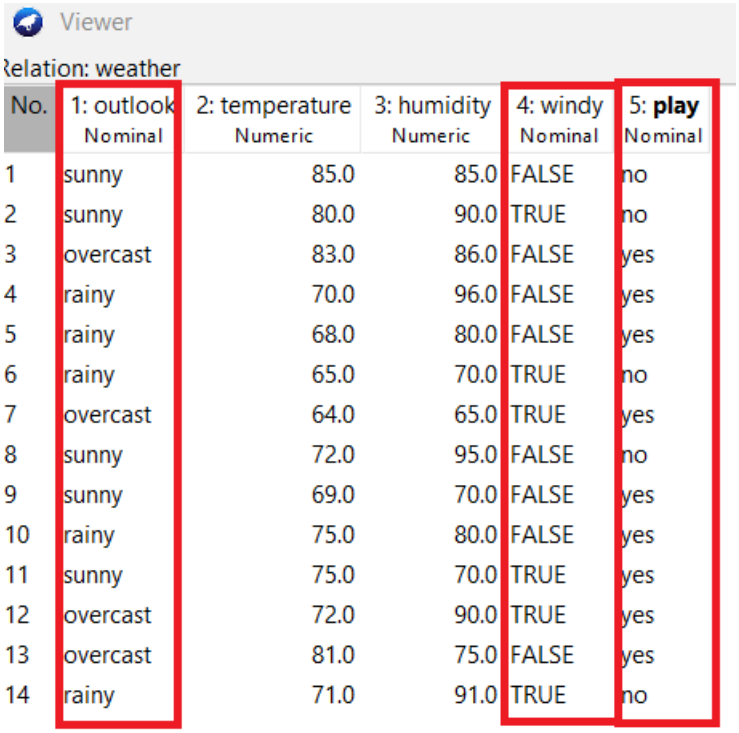
After loading data file namely **weather.numeric.arf** into WEKA explorer successfully, we will answer some questions or perform requirements in the followings:

- How many attributes does this data set have? How many samples? Which attributes have data type categorical? Which attributes have a data type that is numerical? Which attribute is used for the label?

  Answer: This dataset has 5 attributes: outlook, temperature, humidity, windy, play. There are 14 samples in this dataset.



  – Attributes have data type categorical: outlook, windy, play.



No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: **play** Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no

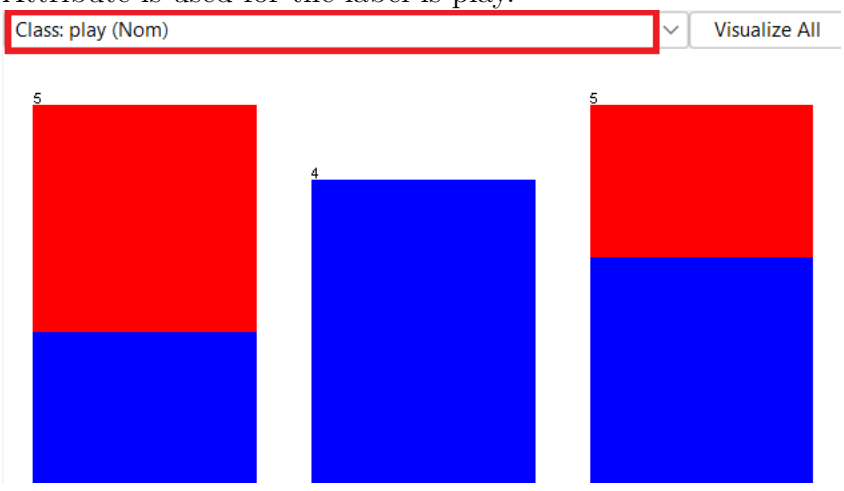
– Attributes have data type numeric: temperature, humidity.

Viewer

Relation: weather

No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: **play** Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no

Attribute is used for the label is play.



- Lets list ***five-number summary*** of two attributes **temperature** and **humidity**. Does WEKA provide these values?

Answer: Five-number summary is minimum, maximum, median, Q1, Q3 of attributes have data type numeric. Five-number summary of temperature and humidity:

Five-number summary	**Temperature**	**Humidity**
Minimum	64	65
Lower Quartile	69.25	71.25
Median	72	82.5
Upper Quartile	78.75	90
Maximum	85	96

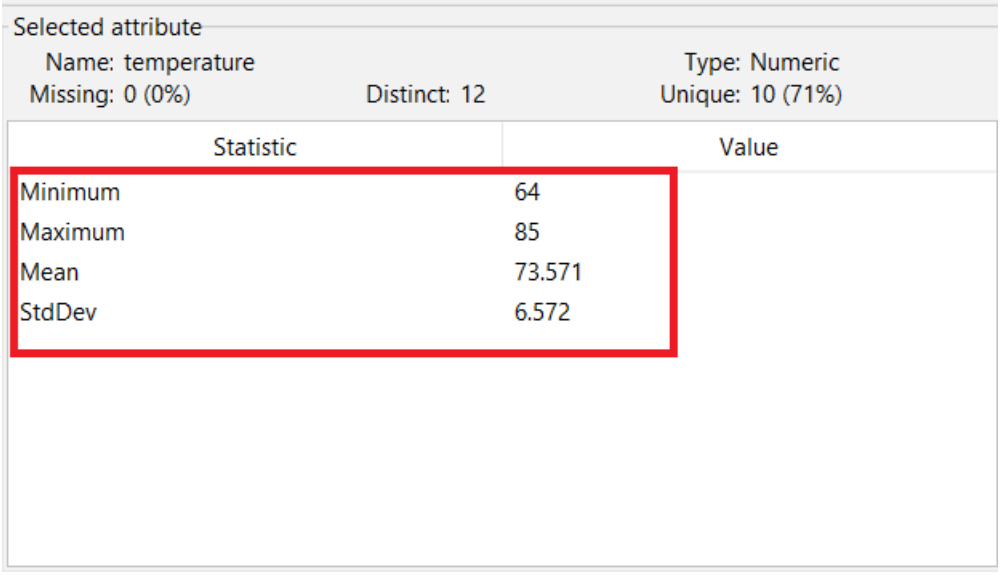
In Weka, for numeric attributes, the "Number Summary" provides four key statistics:

Minimum (minimum): The smallest value in the dataset.

Maximum (maximum): The largest value in the dataset.

Mean (mean): The average value of the numeric attribute.

Standard Deviation (StdDev): A measure of how much the values in the dataset vary from the mean.



Selected attribute

Name: humidity Type: Numeric

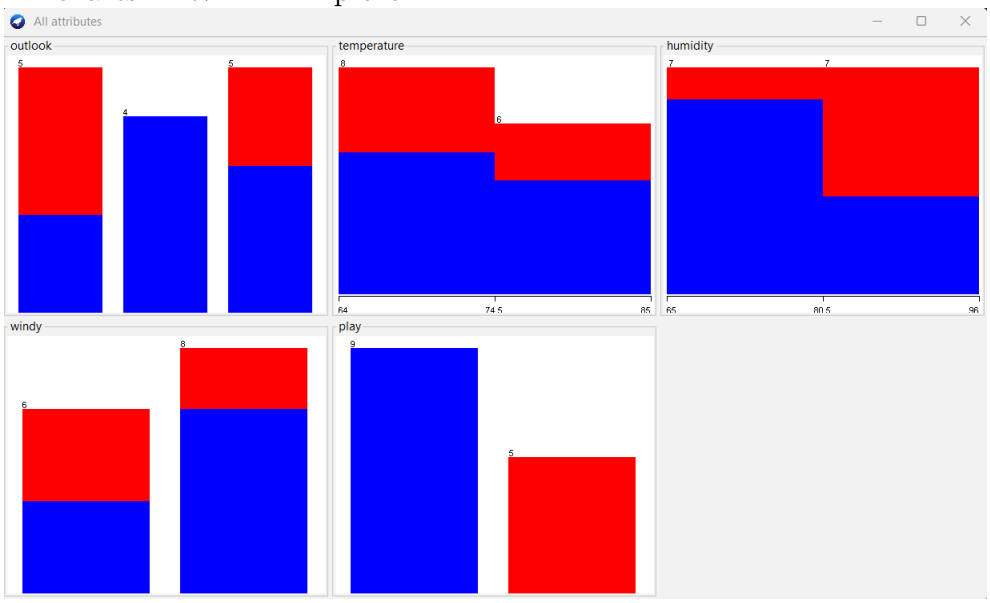
Missing: 0 (0%) Distinct: 10 Unique: 7 (50%)

Statistic	Value
Minimum	65
Maximum	96
Mean	81.643
StdDev	10.285

- Lets explain the meaning of all charts in the WEKA Explorer. Setting the title for it and describing its legend.

  Answer :

  – All charts in WEKA Explorer:



  – Meaning of all charts:

* There are 5 chart of 5 attributes: outlook, temperature, humidity, windy, play. All of chart display the number of values in the attribute, which is selected, according to classify attribute.
* The color represents the proportion of values of the "play" attribute being classified as "yes" in blue and "no" in red.
* For example, if we choose "outlook" attribute, we'll have this chart:

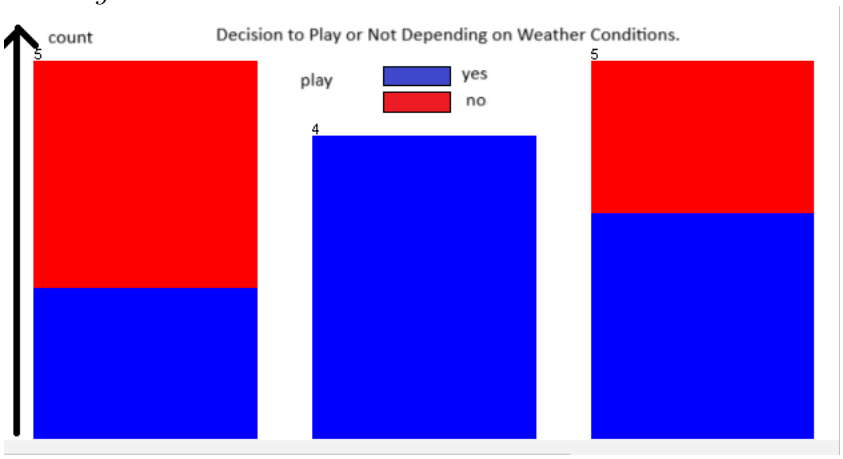
* This is a nominal attribute with three values: sunny, overcast, and rainy. From left to right, each column represents sunny, overcast, and rainy, respectively. Based on the height, we can observe that the first column has 5 "sunny" values, the second has 4 "overcast" values, and the third column has 5 "rainy" values.
* Regarding the colors, for instance, in the first column, there are 2 values with the 'play' as 'yes,' which are colored in blue, while the red color represents 3 values with 'play' as 'no'.

– Setting the title: (we will use attribute Outlook)

* Title: Decision to Play or Not Depending on Weather Conditions.

* Legend:
    - **Blue**: Decide to play (yes)
    - **Red**: Decide not to play (no)
* Analyze similarly with the remaining charts.

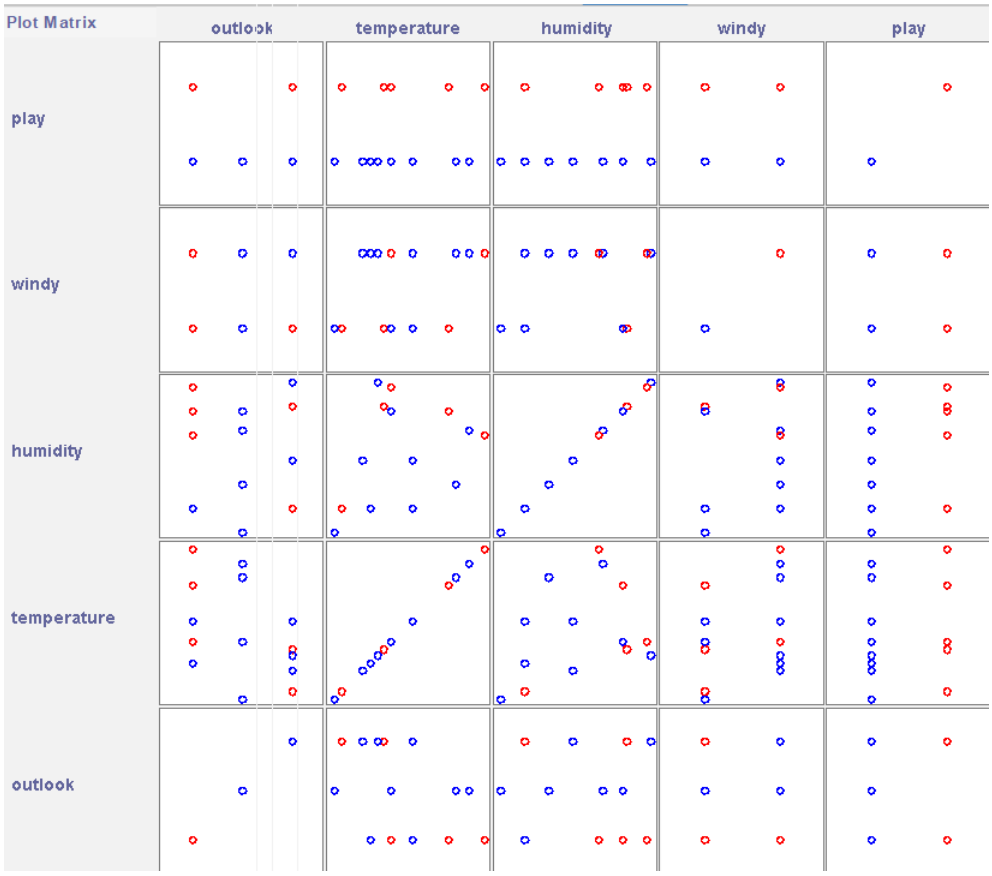
3. Image:



- Lets move to the Visualize tag. Whats the name of this chart? Do you think there are any pairs of different attributes that have correlated

  Answer:

  – Chart in Visualize tag:



- Name of this chart is scatter plot
- From the scatter plot matrix, it appears that there are no attributes that have correlated. Because the points are scattered all over without any discernible pattern.

3.2.3 Exploring Credit in Germany data set

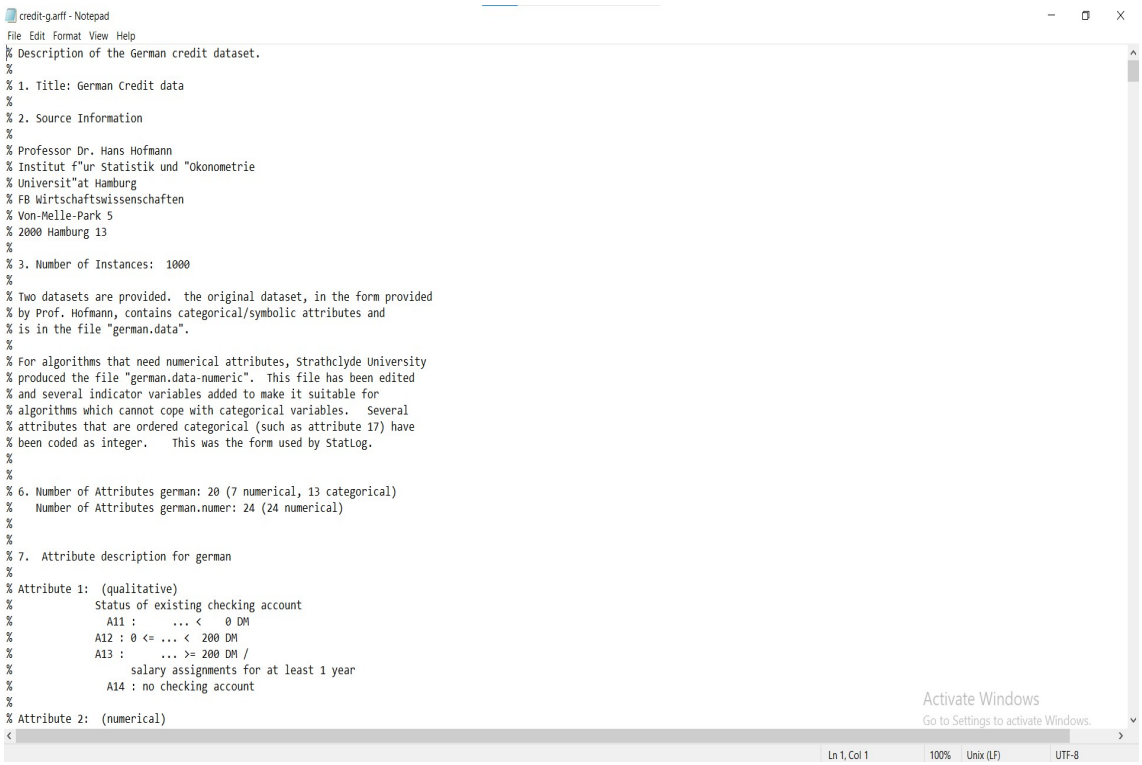
After loading data file namely **credit-g.arf** into WEKA explorer successfully, we will answer some questions or perform requirements in the followings:

- What is the content of the comments section in **credit-g.arf** (when opened with any text editor) about? How many samples does the data set have? How many attributes?

Describe any five attributes (must have both discrete and continuous attributes).

Answer:

– Open file with Notepad:



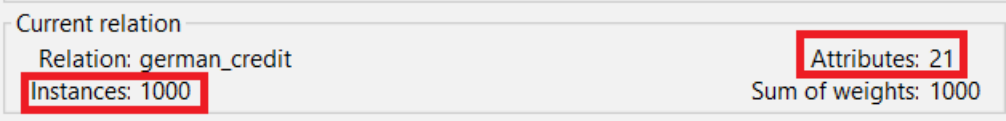
– This is summary of dataset, include: Title, Source Information, Number of Instances, Number of attributes, Attribute description. Basic information about this dataset:

  * Two datasets are provided. the original dataset, in the form provided by Prof. Hofmann, contains categorical/symbolic attributes and is in the file "german.data".

  * For algorithms that need numerical attributes, Strathclyde University produced the file "german.data-numeric". This file has been edited and several indicator variables added to make it suitable for algorithms which cannot cope with categorical variables. Several attributes that are ordered categor-

ical (such as attribute 17) have been coded as integer. This was the form used by StatLog.

– There are 1000 samples and 21 attributes in this dataset.

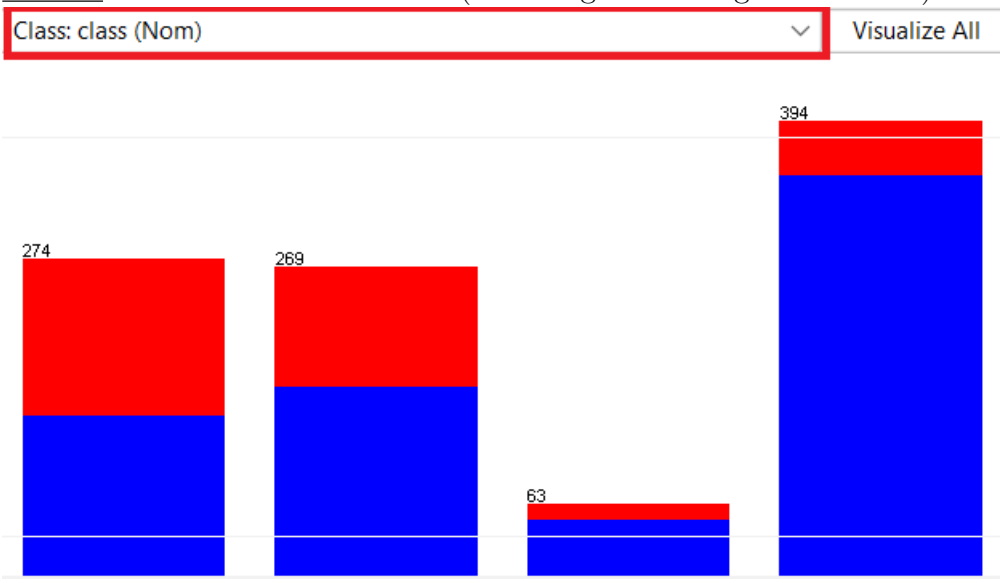


– Describe 5 attributes:

Name	Attribute type	Data type	Meaning
age	Continous	numeric	Age of customer
job	Discrete	nominal	Job of customer
purpose	Discrete	nominal	Purpose of credit loan
credit_amount	Continous	numeric	Account balance in credit card
personal_status	Discrete	nominal	Sex and statement

- Which attribute is used for the label?

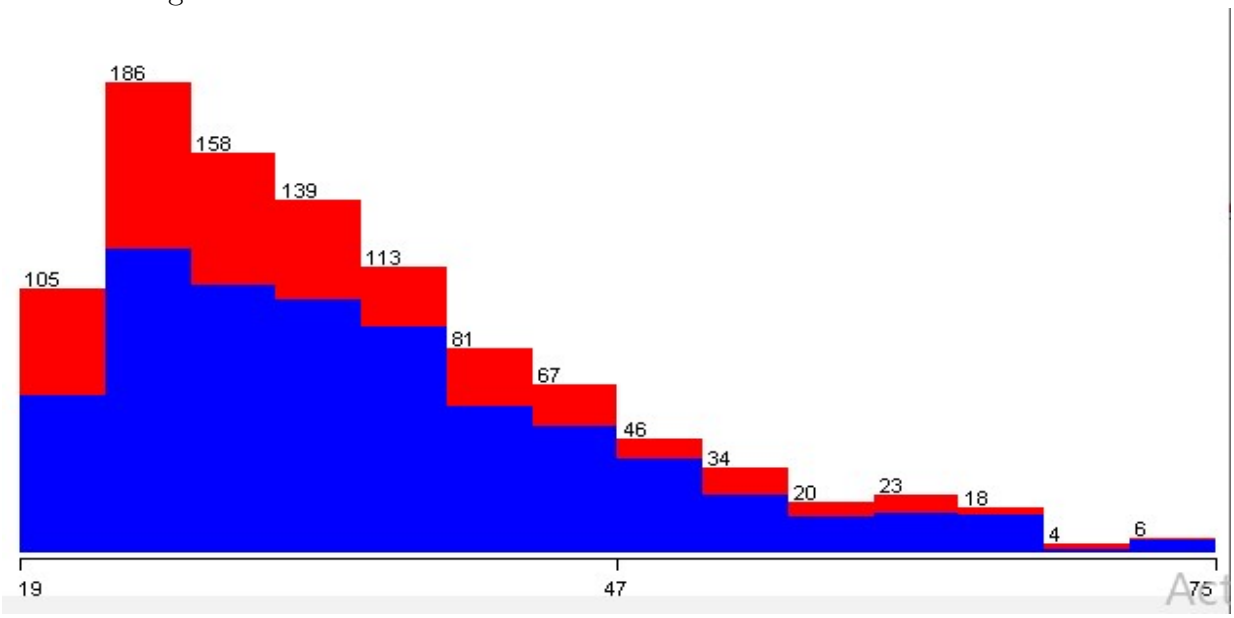
  Answer: That is attribute "class" (including 2 values: good and bad).



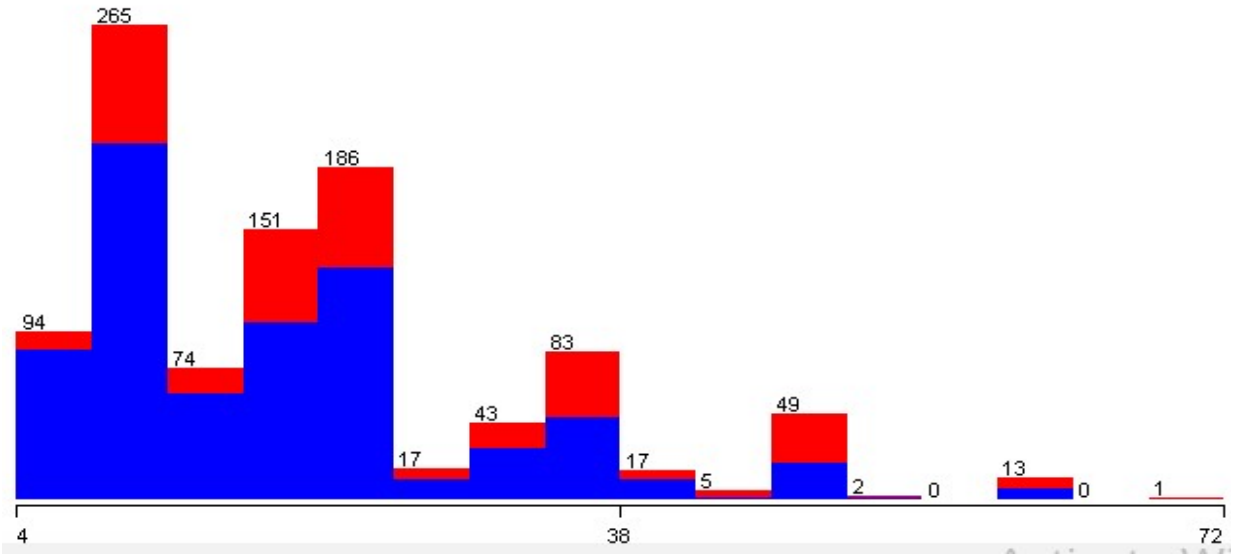
- Lets describe the distribution of continuous attributes? (Left skewed or right skewed ?)

  Answer: There are three attributes have data type continous: age, credit_amount and duration

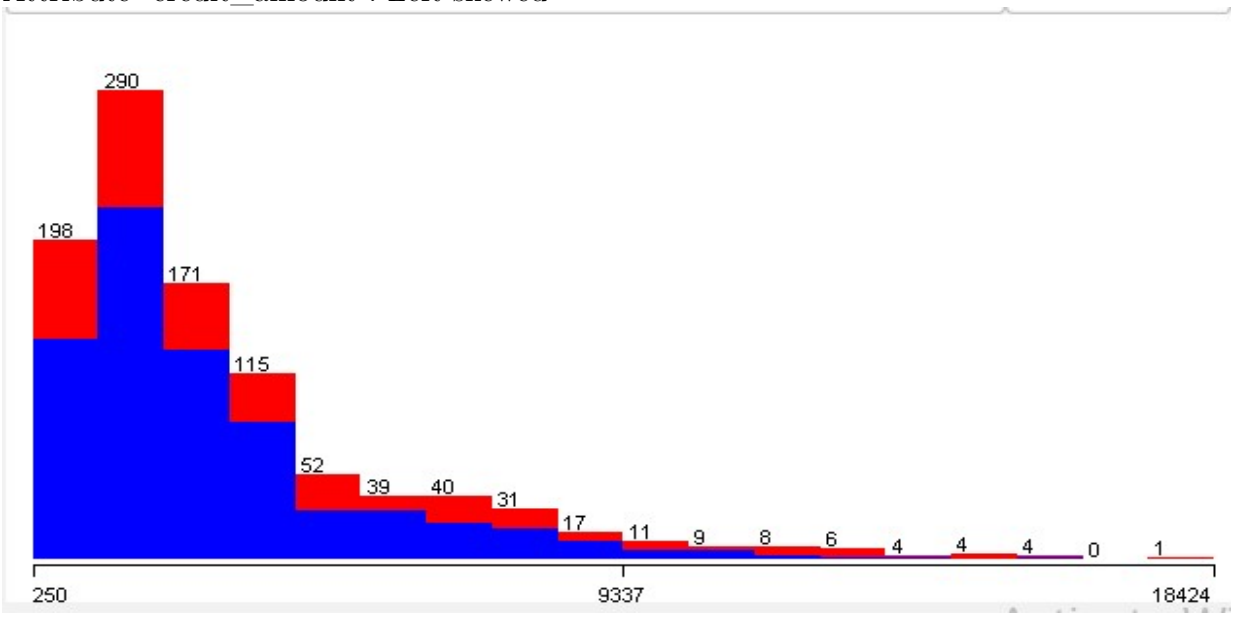
  - Attribute "age": Left skewed



  - Attribute "duration": Left skewed



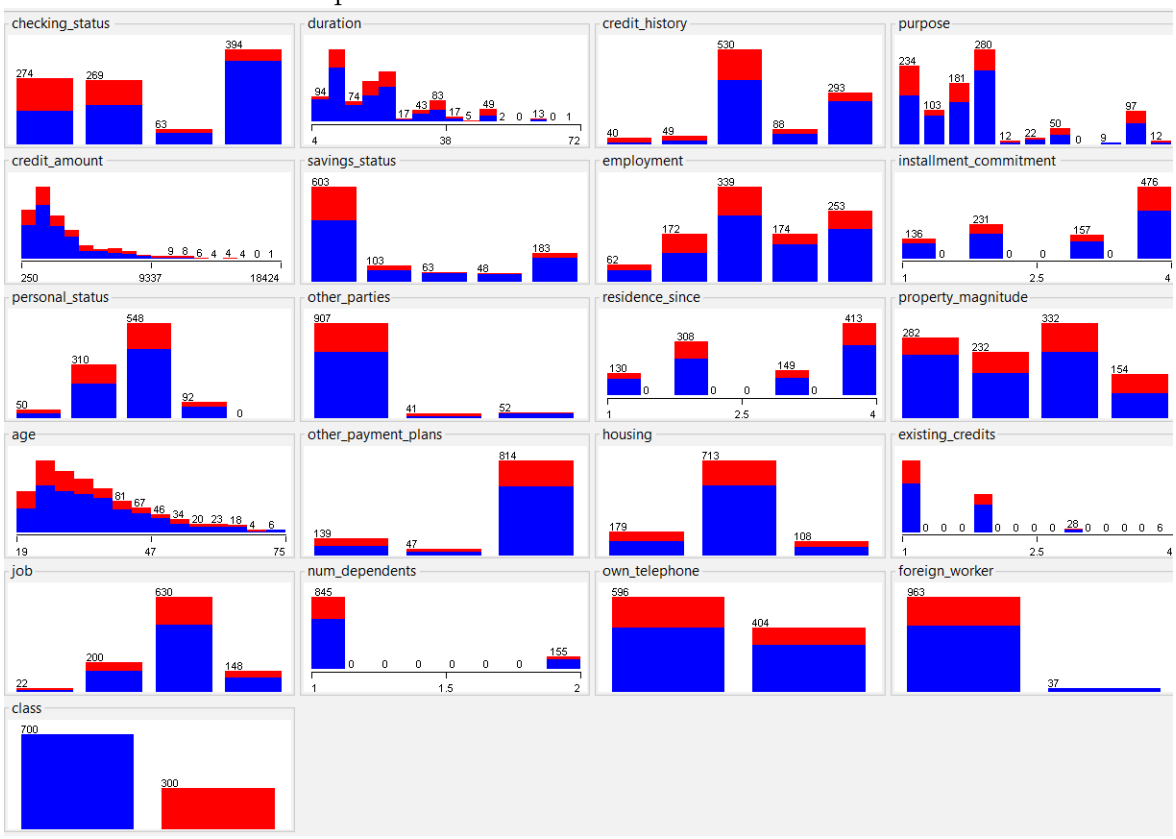
– Attribute "credit_amount": Left skewed



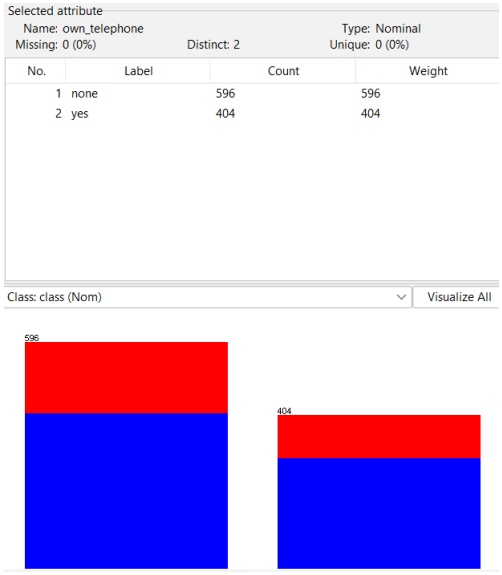
- Lets explain the meaning of all charts in the WEKA Explorer. Setting the title for it and describing its legend.

  Answer:

- All charts in WEKA Explorer:



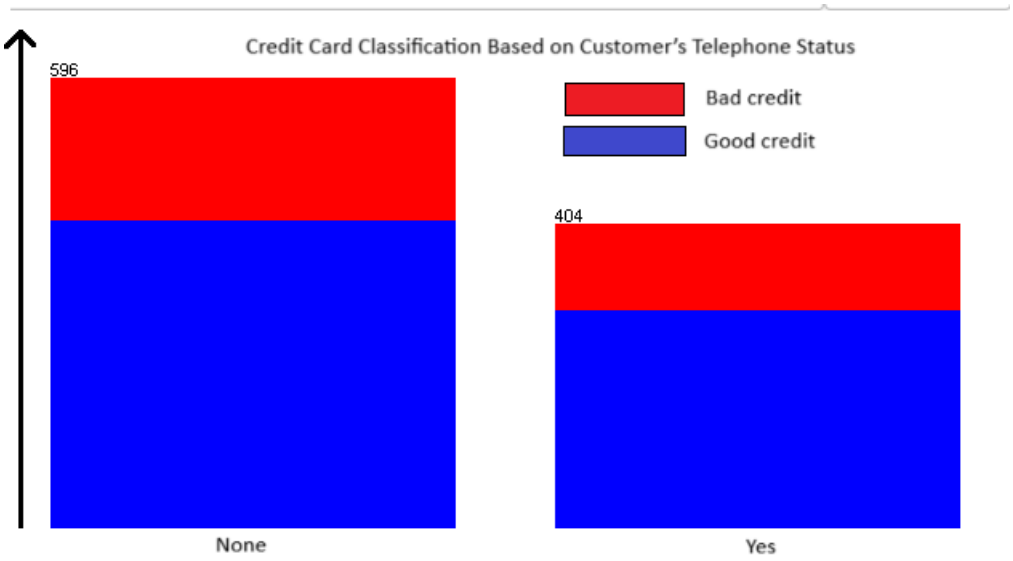
- Meaning of all charts:
  - There are 21 chart of 21 attributes of dataset. All of chart display the number of values in the attribute, which is selected, according to classify attribute.
  - The color represents the proportion of values of the "class" attribute being classified as "good" in blue and "bad" in red.
  - For example, if we choose "own-telephone" attribute, we'll have this chart:



  - This is a nominal attribute with two values: none and yes. From left to right, each column represents yes and none. Based on the height, we can observe that the first column has 596 "none" values, the second has 404 "yes" values.
  - Regarding the colors, for instance, in the first column, The color blue represents values classified as "good," while the color red represents values with a class of "bad."

- Setting the title: (we will use attribute own phone)
  - Title: Credit Card Classification Based on Customer's Telephone Status.
  - Legend:
    - **Blue**: Good credit (good)
    - **Red**: Bad credit (bad)
  - Analyze similarly with the remaining charts.

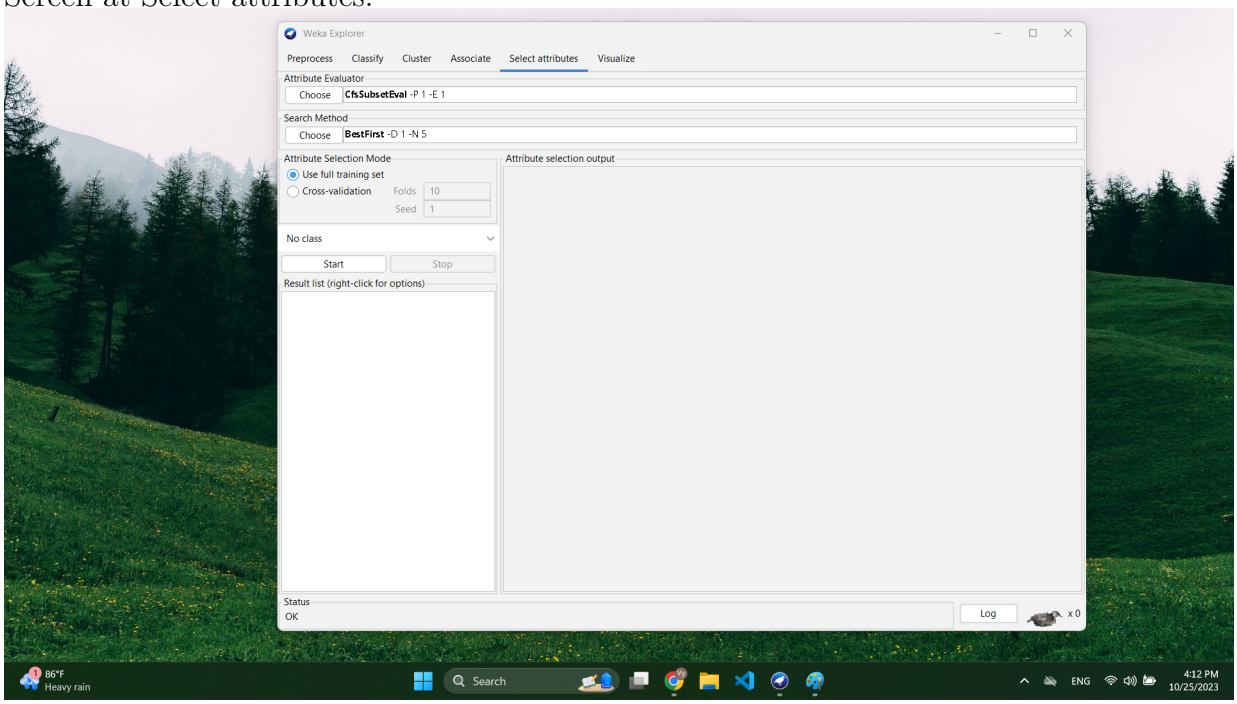
3. Image:



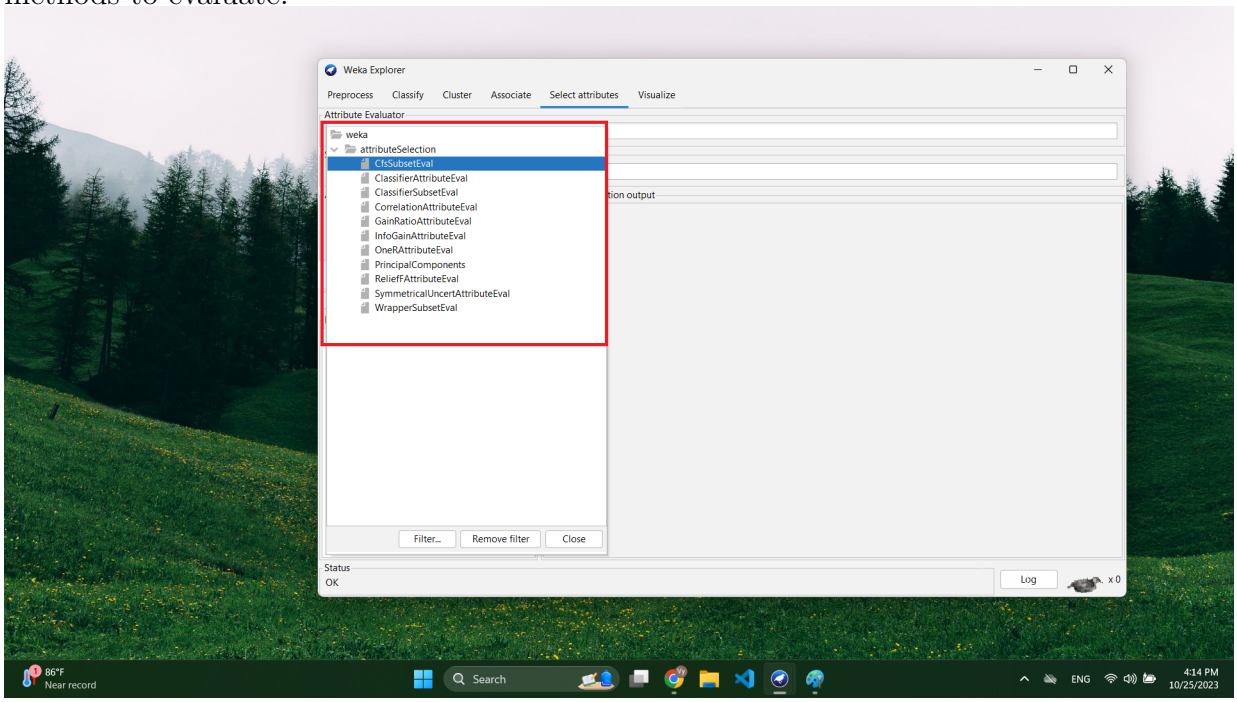
- Lets move to the Select attributes tag. Describe all of the options for attribute selection.

  Answer: Options for attribute selection

  Screen at Select attributes:

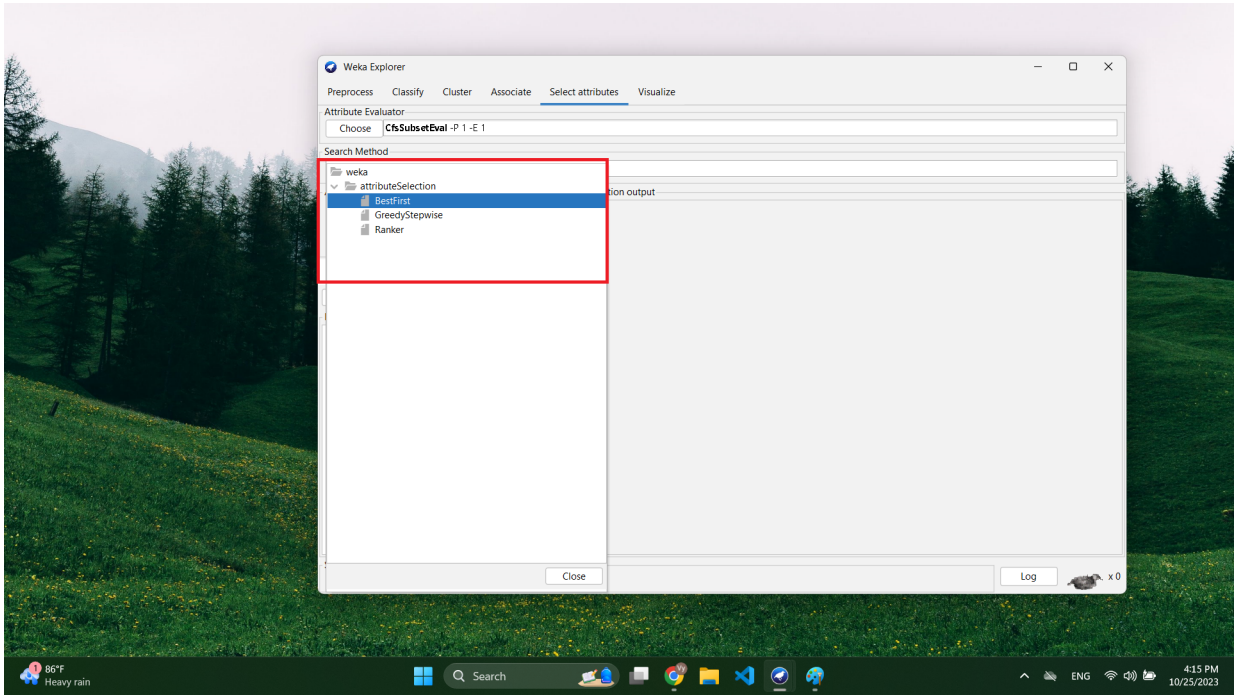


– Attribute Evaluator: use for evaluating attribute of dataset. WEKA has 11 methods to evaluate.



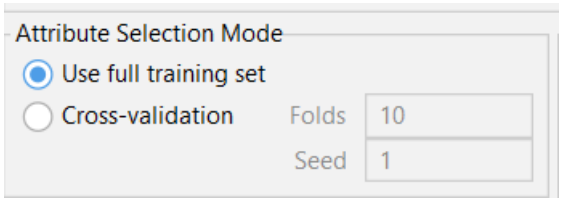
Name	Description
CfsSubsetEval	Evaluate attribute by prediction of every attributes
ClassifierAttributeEval	Evaluate attribute by using classifier of class
ClassifierSubsetEval	Evaluate child attribute in training set.
CorrelationAttributeEval	You can calculate the correlation between each attribute
GainRatioAttributeEval	You can calculate the increased ratio of attribute
InfoGainAttributeEval	You can calculate the information gain (also called entropy) for each attribute for the output variable.
OneRAttributeEval	Evaluate attribute by using classifier OneR
PrincipalComponents	Analyze main part and transfer data
ReliefAttributeEval	Evaluate attribute by how they showing
SummetricalUncertAttributeEval	Evaluate attribute by dissymmetrical
WrapperSubsetEval	Evaluate attribute by 1 classifer with cross-validation

– Search Method: choose search method to do. WEKA has 3 methods to search.

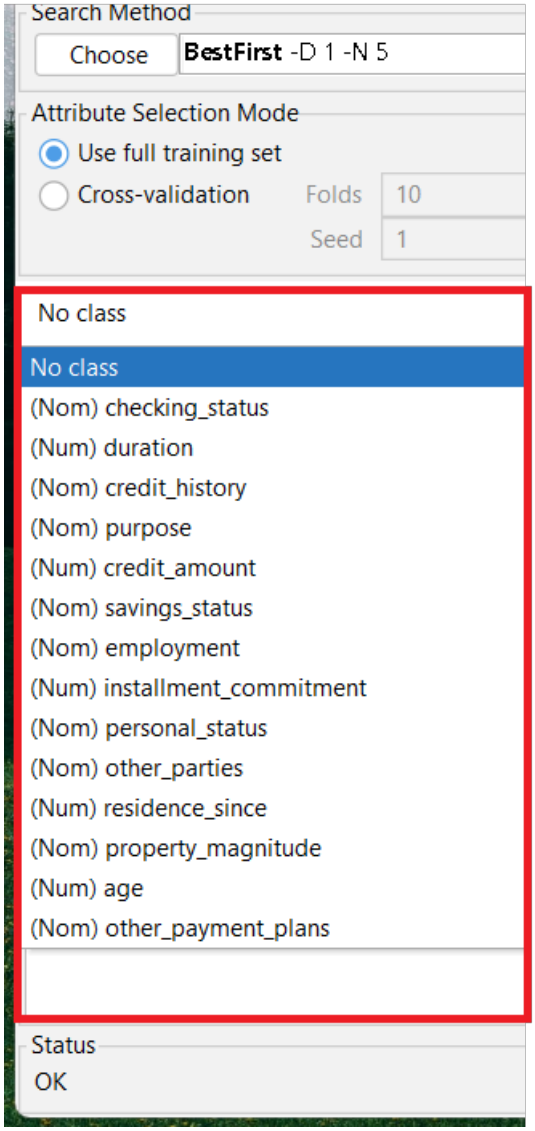


Method	Description
BestFirst	Using greedy hill climbing with backtracking. It can search forward from none attribute, backward from all attributes.
GreedyStepwise	Use Greedy Algorithm in space of attributes. It can search forward and backward. But it not backtracking, it will stop when add or delete the best attribute.
Ranker	Ratings attributes and get higher rating attribute, remove lower rating attributes

– Attribute Selection Mode: WEKA has 2 options: use full training set or cross-validation. When choose option cross-validation, you have to choose folds and seed.



– Drop down to choose attribute for predicting/classifying.

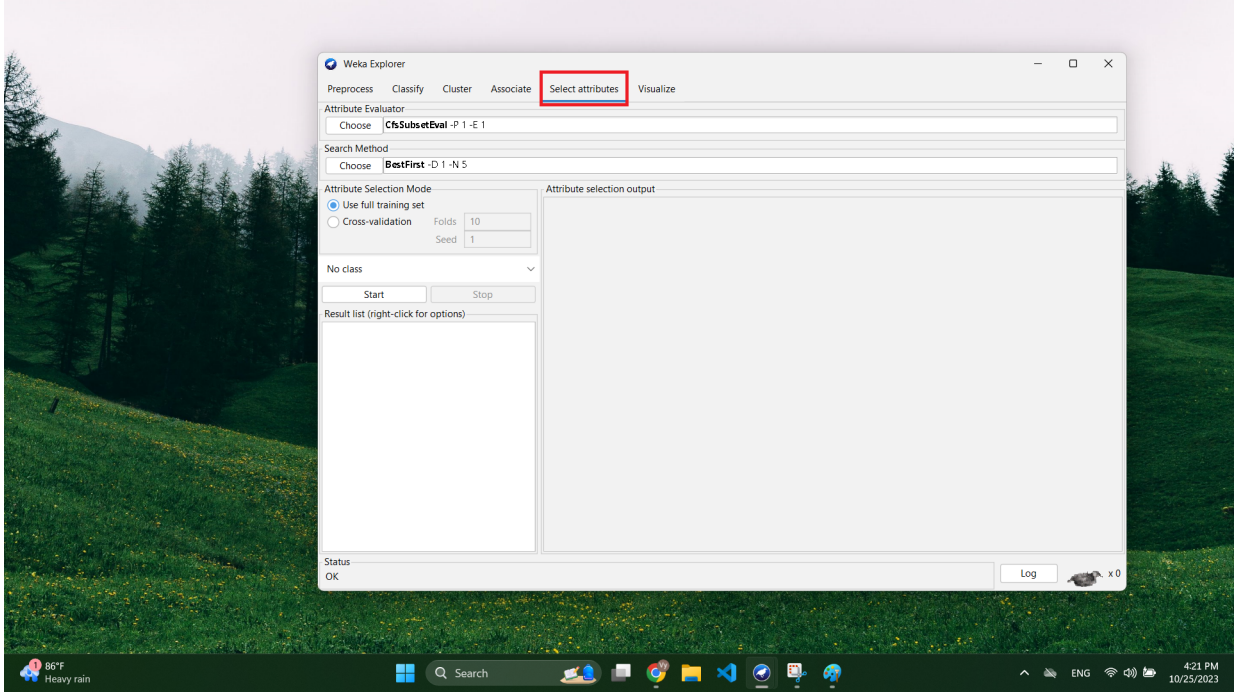


- Which options should be used to select the 5 attributes with the highest correlation?(Step-

by-step description, with step-by-step photos and final results)

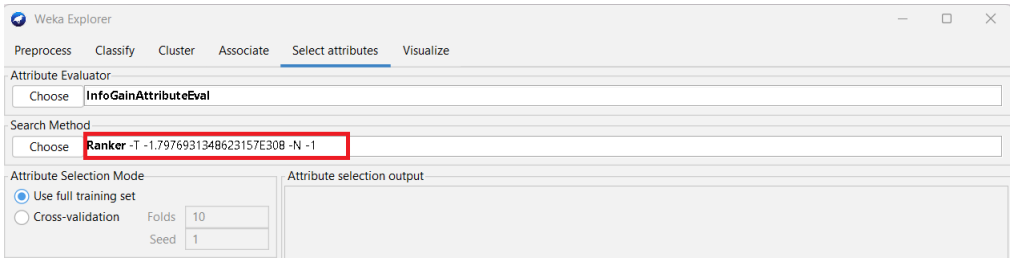
Answer : We choose attribute **class** for this question. Following these step we get the result:

– Select **Select attributes**

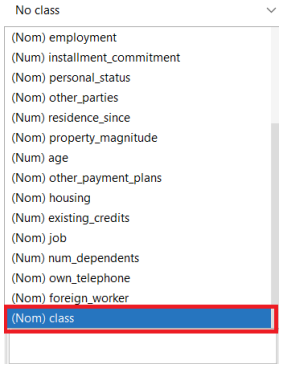


– Select **InfoGainAttributeEval** at **Attribute Evaluator**

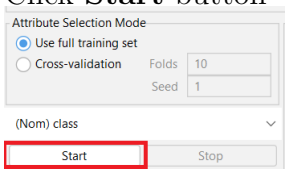
– Select **Ranker** at **Search Method**



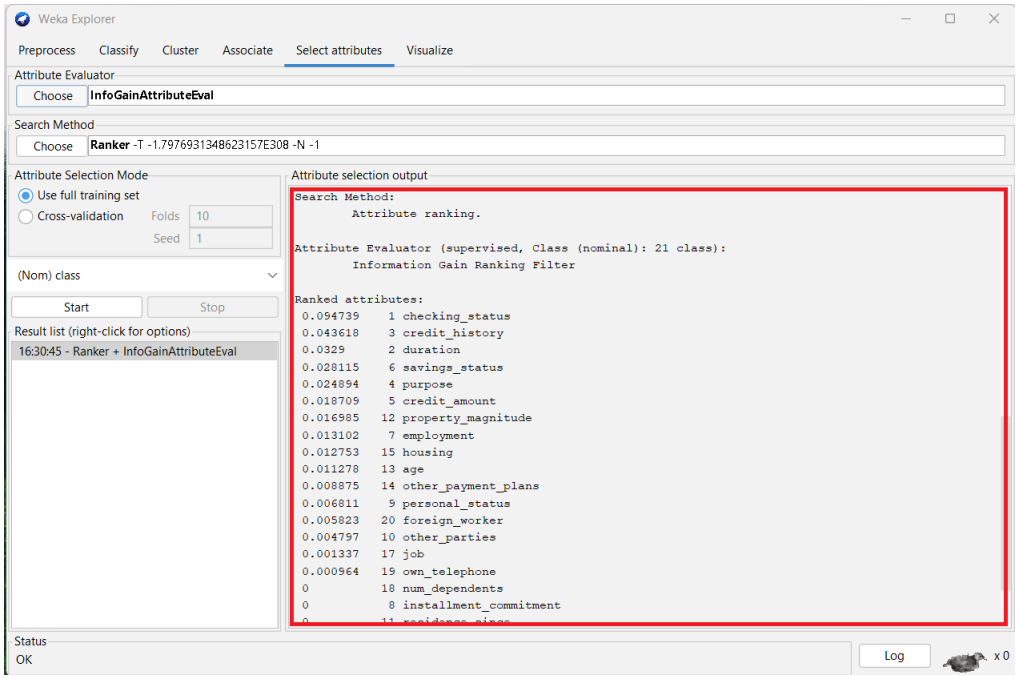
– Select attribute **class** at drop down



– Click **Start** button



– See result at **Attribute selection output**



We can see 5 attributes has the highest correlation in order from highest to lowest: checking_status, credit_history, duration, savings_status and purpose.

```
Ranked attributes:
 0.094739    1 checking_status
 0.043618    3 credit_history
 0.0329      2 duration
 0.028115    6 savings_status
 0.024894    4 purpose
```

3.3 Preprocessing Data in Python

3.3.1 Description

- The program operates according to the console mechanism and user requirements are specified via command line parameters.
- Some rules about command line parameters of the program:

- First parameter is executable file name, default name is main.py
- Second parameter is name of data set file for testing, default name is house-prices.csv
- Third parameter is name of function preprocessing, including:
  * extract_columns_with_missing_values: Extract columns with missing values.
  * count_lines_with_missing_data: Count the number of lines with missing data.
  * fill_in_missing_values: Fill in the missing value using mean, median (for numeric properties) and mode (for the categorical attribute).
  * delete_row_has_more_than_particular_num: Deleting rows containing more than a particular number of missing values.
  * delete_column_has_more_than_particular_num: Deleting columns containing more than a particular number of missing values.
  * delete_duplicate: Delete duplicate samples.
  * normalize_numeric_attribute: Normalize a numeric attribute using min-max and Z-score methods.
  * handle_calculation: Performing addition, subtraction, multiplication, and division between two numerical attributes.

3.3.2 Extract columns with missing values

- Syntax: python3 lab01.py file_path=house-prices.csv function=extract_columns_with_missing_values
- Result: Number of missing values of each column:

```
Columns with missing values: LotFrontage - 173 missing values
Alley - 941 missing values
MasVnrType - 593 missing values
MasVnrArea - 10 missing values
BsmtQual - 27 missing values
BsmtCond - 27 missing values
BsmtExposure - 28 missing values
BsmtFinType1 - 27 missing values
BsmtFinType2 - 29 missing values
FireplaceQu - 501 missing values
GarageType - 60 missing values
GarageYrBlt - 60 missing values
GarageFinish - 60 missing values
GarageQual - 60 missing values
GarageCond - 60 missing values
PoolQC - 1000 missing values
Fence - 815 missing values
MiscFeature - 963 missing values
```

3.3.3 Count the number of lines with missing data

- Syntax: python3 lab01.py file_path=house-prices.csv function=count_lines_with _missing_data

- Result: Print number of lines have missing data

```
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=house-prices.csv function=count_lines_with_missing_data
Numbers of lines with missing data:  1000
```


3.3.4 Fill in the missing value using mean, median (for numeric properties) and mode (for the categorical attribute)

- Syntax: python3 lab01.py file_path=house-prices.csv function=fill_in_missing_values output_file=q3.csv

- Result: We use function to fill number of missing value and check it by using extract columns of missing values. Number of columns have missing data (none - because we filled it all)

```
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=house-prices.csv function=fill_in_missing_values output_file=q3.csv
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=q3.csv function=extract_columns_with_missing_values
Columns with missing values:
```

3.3.5 Deleting rows containing more than a particular number of missing values

- Syntax: python3 lab01.py file_path=house-prices.csv function=delete_row_has_more_than_particular_num particular_num=10 output_file=q4.csv
- Result:
  - To test the results, we will count the number of rows containing missing values in the file q4.csv is obtained after executing the delete_row_has_more_than_particular_num function. Number of rows after delete is printed on console.

```
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=house-prices.csv function=delete_row_has_more_than_particular_num part
icular_num=10 output_file=q4.csv
Length of matrix after deleting rows have more than_particular number with missing values: 920
```

3.3.6 Deleting columns containing more than a particular number of missing values

- Syntax: python3 lab01.py file_path=house-prices.csv function=delete_column_has_more_than_particular_num particular_num=10 output_file=q5.csv
- Result:
  - To test the results, we will count the values of each columns containing missing values in the file q5.csv is obtained after executing the delete_column_has_more_than_particular_num function. Missing values of columns after delete is printed on console.

```
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=house-prices.csv function=delete_column_has_more_than_particular_num p
articular_num=10 output_file=q5.csv
Deleted!
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=q5.csv function=extract_columns_with_missing_values
Columns with missing values:
MasVnrArea - 10 missing values
BsmtQual - 27 missing values
BsmtCond - 27 missing values
BsmtExposure - 28 missing values
BsmtFinType1 - 27 missing values
BsmtFinType2 - 29 missing values
GarageType - 60 missing values
GarageYrBlt - 60 missing values
GarageFinish - 60 missing values
GarageQual - 60 missing values
GarageCond - 60 missing values
```

3.3.7 Delete duplicate samples

- Syntax: python3 lab01.py file_path=house-prices.csv function=delete_duplicate output_file=q6.csv
- Result:
  - To test the results, we will count the number of rows containing duplicate values in the file q6.csv is obtained after executing the delete_duplicate function. Number of rows after delete is printed on console.

```
PS D:\Visual Studio Code\python\Data mining\Lab01> python3 lab01.py file_path=house-prices.csv function=delete_duplicate output_file=q6.csv
Data after delete dupplicates samples: 716
```

3.3.8 Normalize a numeric attribute using min-max and Z-score methods

- Syntax: python3 lab01.py file_path=house-prices.csv function=normalize_numeric_attribute method=z-score output_file=q7.csv
- Result (z-score):
  - Values of "Id" column is normalized z-score.

Id
1.196263761
1.174349486
1.583415957
1.52497789
-1.321442971
1.561501682
0.55831486
-0.649405197
-0.873417789
-0.050415008
-1.207001756
0.83589568
0.658146558
1.437320789
0.667886236
0.23690549
-0.715148023
1.254701829
-0.135637189
-1.769468154
1.371577963
-0.576357613
0.772587773
1.636984186
1.235222473

- Syntax: python3 lab01.py file_path=house-prices.csv function=normalize_numeric_attribute method=min-max output_file=q7.csv
- Result (min-max):
  - Values of "Id" column is normalized min-max in [0,1].

Id
0.85048
0.844307
0.959534
0.943073
0.141289
0.953361
0.670782
0.33059
0.26749
0.499314
0.173525
0.748971
0.698903
0.918381
0.701646
0.580247
0.312071
0.866941
0.475309
0.015089
0.899863
0.351166

3.3.9 Performing addition, subtraction, multiplication, and division between two numerical attributes

- Syntax: python3 lab01.py file_path=house-prices.csv function=handle_calculation attr_1=Id attr_2=MSSubClass operation=addition output_file=q8.csv

- Result:

  In "q8.csv" has column of result of subtraction between 2 attributes.

Id	MSSubClass	subtraction
1242	20	1222
1233	90	1143
1401	50	1351
1377	30	1347
208	20	188
1392	90	1302
980	20	960
484	120	364
392	60	332
730	30	700
255	20	235
1094	20	1074
1021	20	1001
1341	20	1321
1025	20	1005
848	20	828
457	70	387
1266	160	1106
695	50	645
24	120	-96

In "q8.csv" has column of result of multiplication between 2 attributes.

Id	MSSubClass	multiplication
1242	20	24840
1233	90	110970
1401	50	70050
1377	30	41310
208	20	4160
1392	90	125280
980	20	19600
484	120	58080
392	60	23520
730	30	21900
255	20	5100
1094	20	21880
1021	20	20420
1341	20	26820
1025	20	20500
848	20	16960
457	70	31990
1266	160	202560
695	50	34750
24	120	2880
1314	60	78840
514	20	10280
1068	60	64080
1423	120	170760
1258	30	37740
620	60	37200

4 Reference

https://www.tutorialspoint.com/weka/weka_preprocessing_data.html