

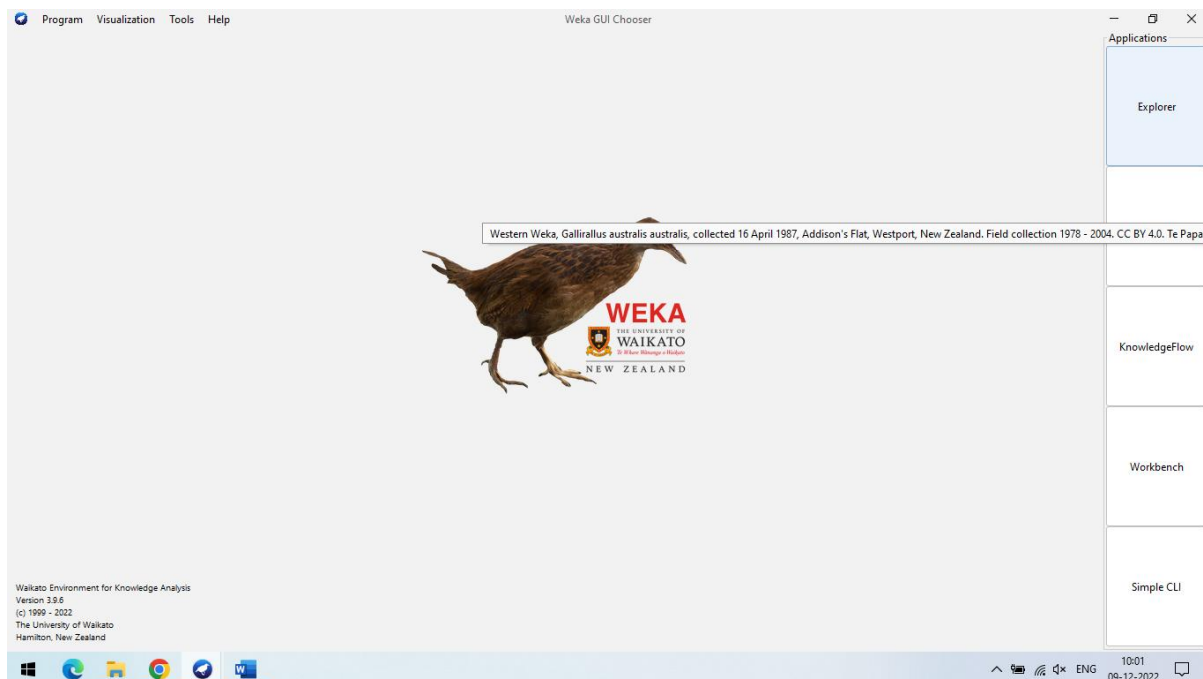
PRACTICAL-

AIM: Study about the tools Weka tool for Data Mining.

Theory:

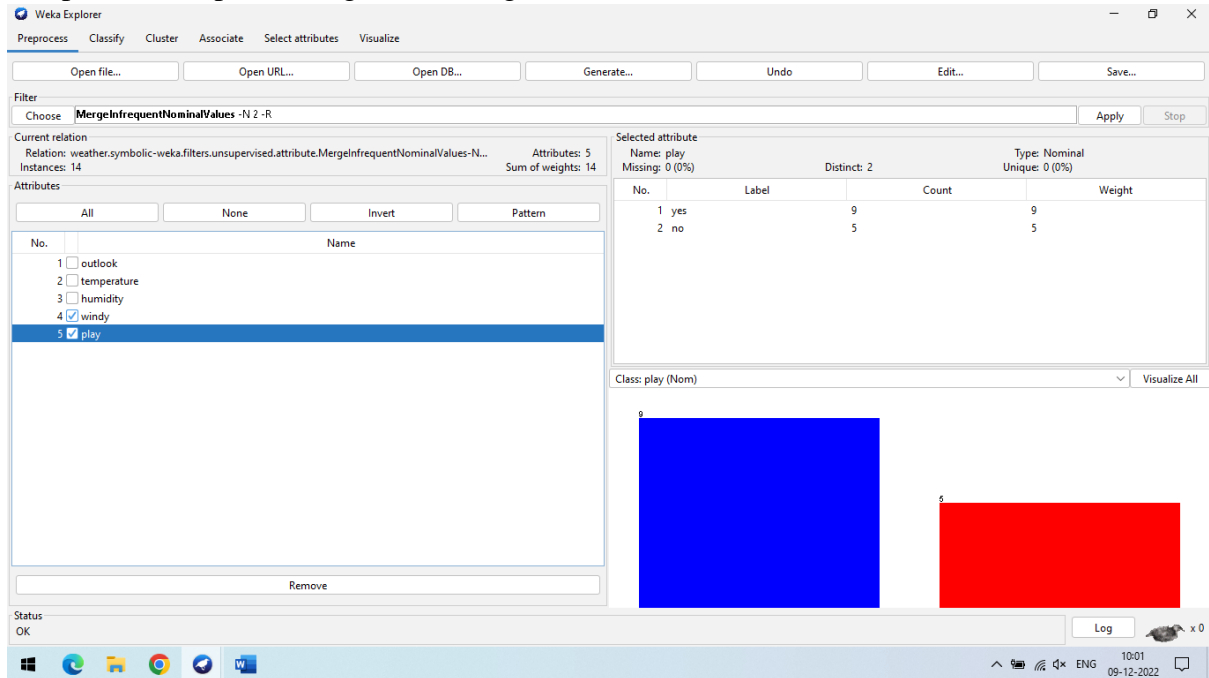
Weka Tool:

Weka is one of the very popular open source data mining tools developed at the University of Waikato in New Zealand in 1992. It is a Java based tool and can be used to implement various machine learning and data mining algorithms written in Java. The simplicity of using Weka has made it a landmark for machine learning and data mining implementation. Weka supports reading of files from several different databases and also allows importing the data from the internet, from web pages or from a remotely located SQL database server by entering the URL of resource. Among all the available data mining tools, Weka is the most commonly used of all due to its fast performance and support for major classification and clustering algorithm. Weka can be easily downloaded and deployed. Weka provides both, a GUI and CLI for performing data mining and does a good job of providing support for all the data mining tasks. Weka supports a variety of data formats like CSV (Comma-separated Value), ARFF and Binary. Weka focuses more on textual representation of the data rather than visualization although it does provide support to display some visualization but those are very generic. Also, Weka does not provide visual representation of results of processing in an effective and understanding manner like Rapid Miner. Weka performs accurately when the size of the data set is not large. If the size is large, then Weka does experience some performance issues. Weka provides support for filtering out data or attributes. Weka supports the following three graphical user interfaces.



1.The Explorer:

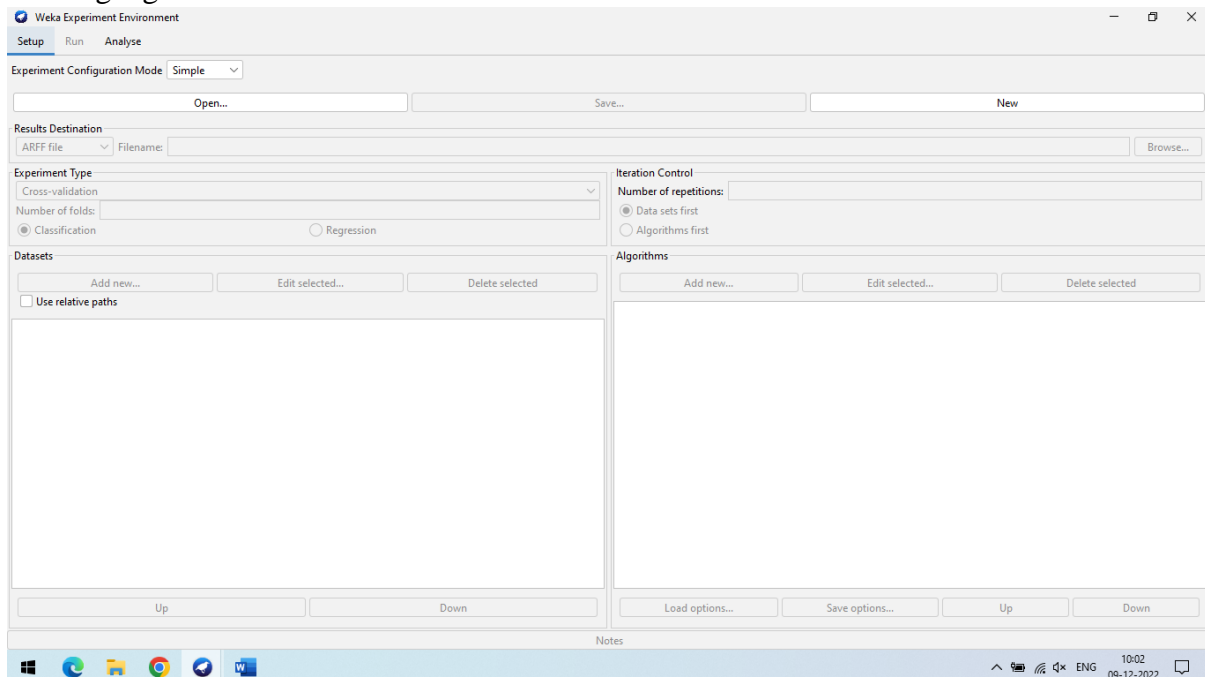
It is the most commonly used graphical user interface in Weka to implement data mining algorithms⁸. It supports exploratory data analysis to perform preprocessing, attribute selection, learning and visualization. This interface consists of different tabs to access various components for performing data mining. The different tabs are-



- A) **Preprocessing** Using this tab, we can load input data files and perform preprocessing on this data using filters.
- B) **Classify** This tab is used to implement different classification and regression algorithms. We can do this by selecting a particular classifier from this tab. For example, the K-NN or Naïve Bayesian algorithm can be implemented by using this tab.
- C) **Associate** This tab is used to find out all association rules between different attributes of the data and which can be used for further mining. For example, Association rule mining, etc.
- D) **Cluster** Using this tab, we can select a particular clustering algorithm to implement for our data set. Clustering algorithms like K-means can be implemented using this tab.
- E) **Select attributes** This tab is used to select particular attributes from the data set useful for implementing the algorithm.
- F) **Visualize** This tab is used to visualize the data whenever available or supported by a particular algorithm in the form of scatter plot matrix.

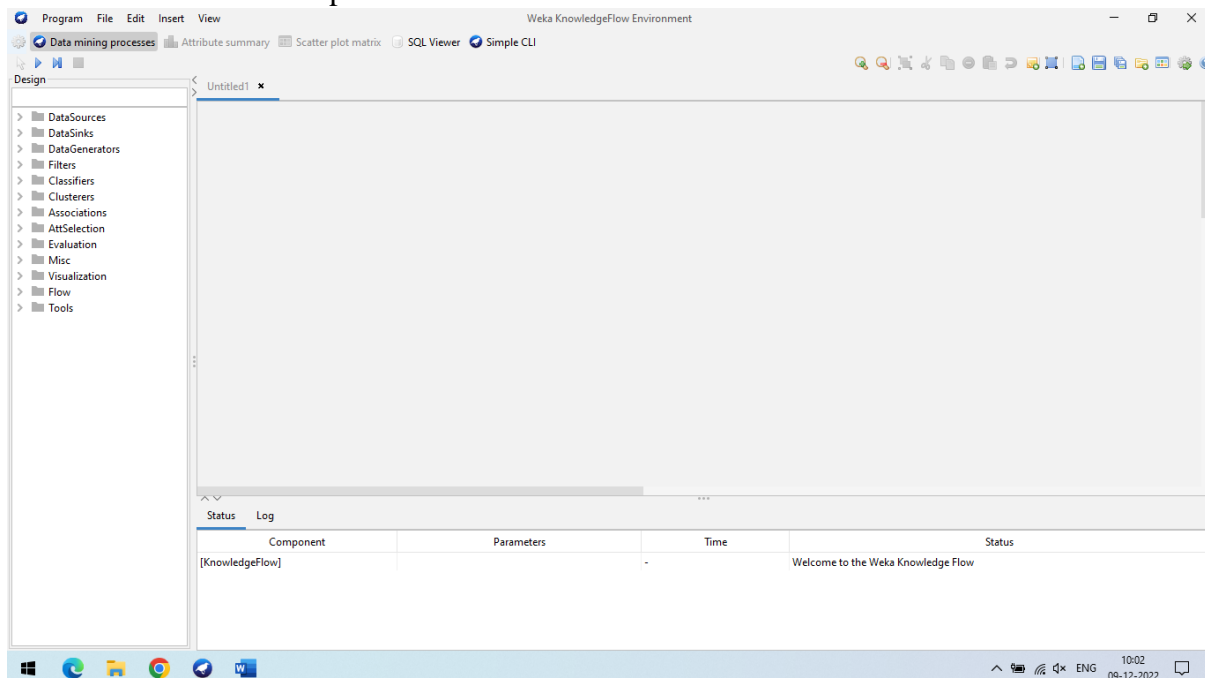
2. The Experimenter

This user interface provides experimental environment for testing and evaluating machine learning algorithms.



3. The Knowledge

Flow Knowledge flow is basically a component based interface similar to explorer. This interface is used for new process evaluations.



PRACTICAL-5

AIM: Perform Pre-processing on a dataset. Apply various Filters and discuss the effect of each filter applied.

a. Handle Missing Values

b. Handle Infrequent Nominal Values

c. Derive an attribute from the existing attribute

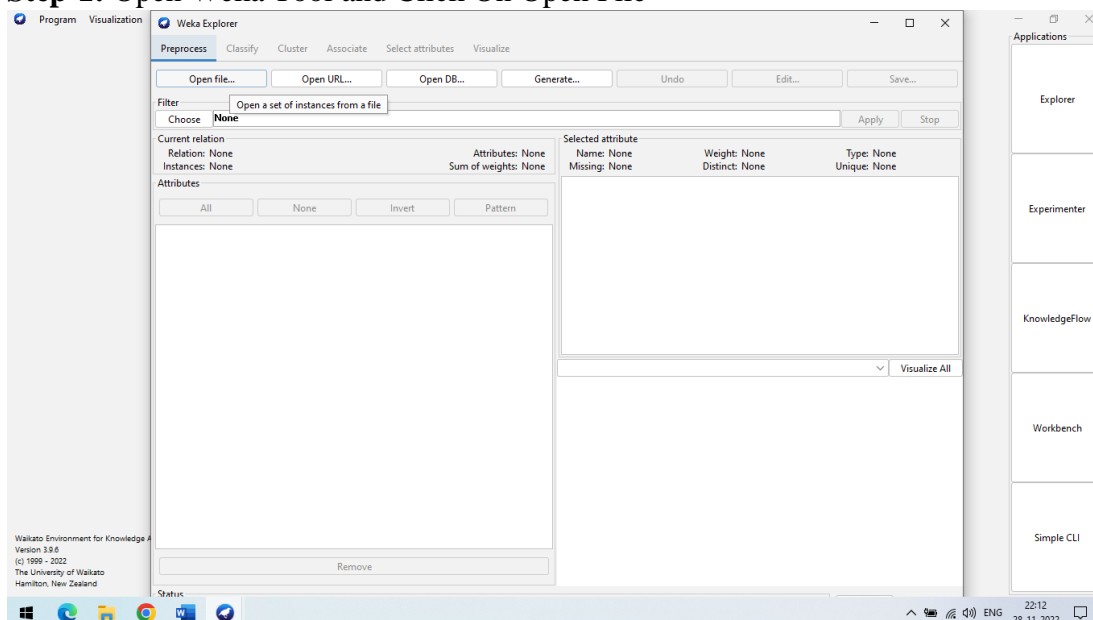
d. Sampling

e. Discretization

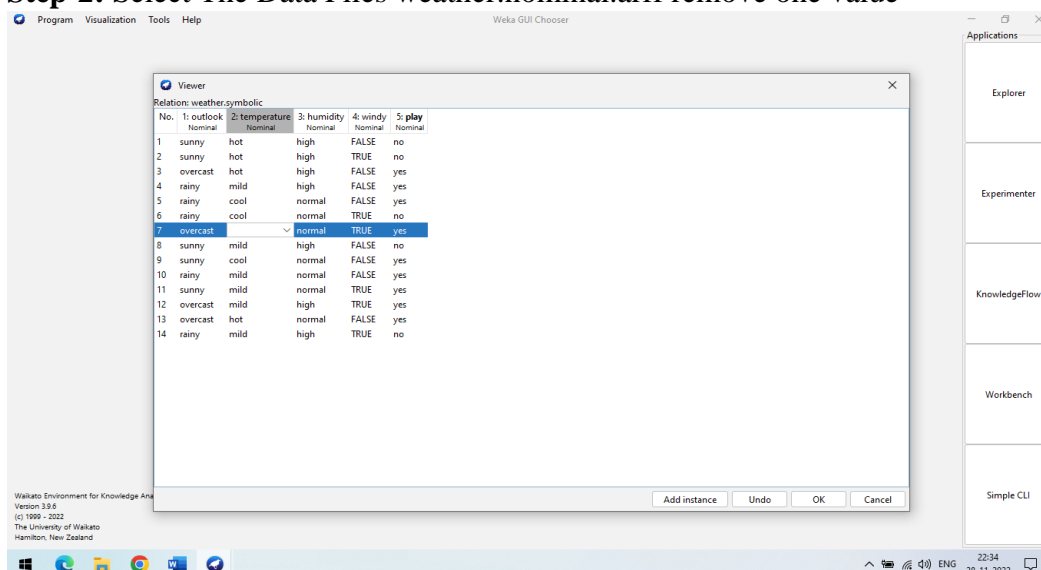
1) Use Weka Tool 2) Use XL Miner Tool.

a. Handle Missing Values

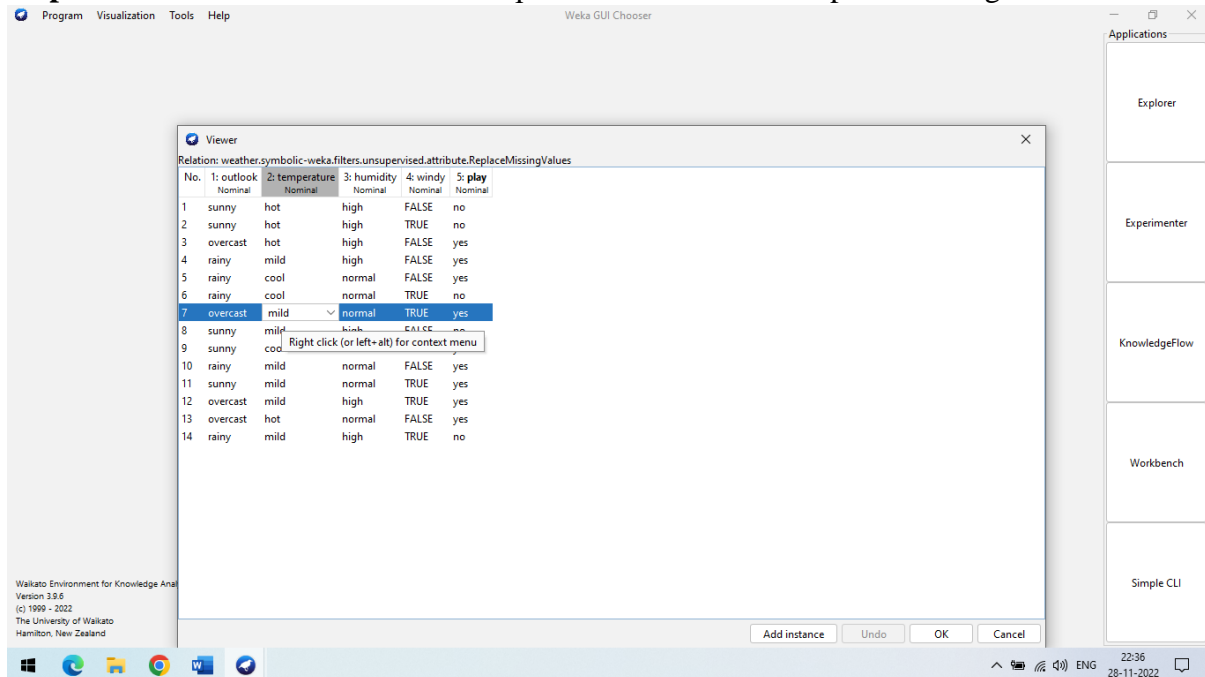
Step-1: Open Weka Tool and Click On Open File



Step-2: Select The Data Files weather.nominal.arff remove one value

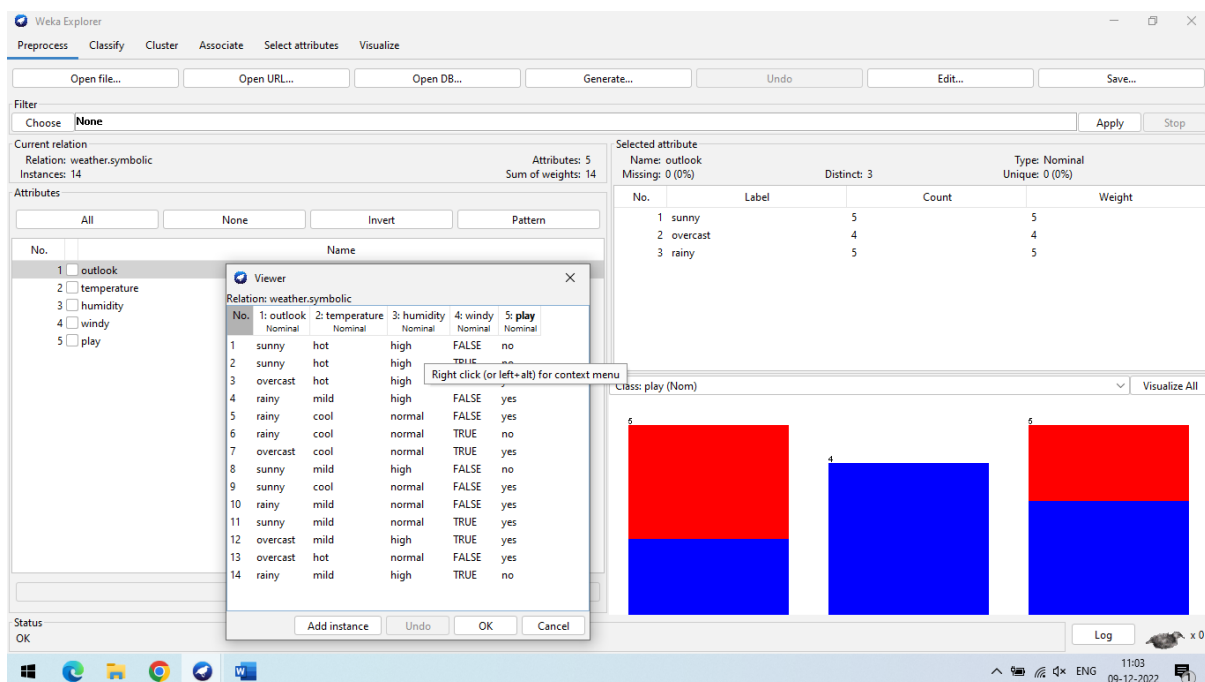


Step-3 : Choose→weka→filters→unsupervised→attribute→ReplaceMissingValue

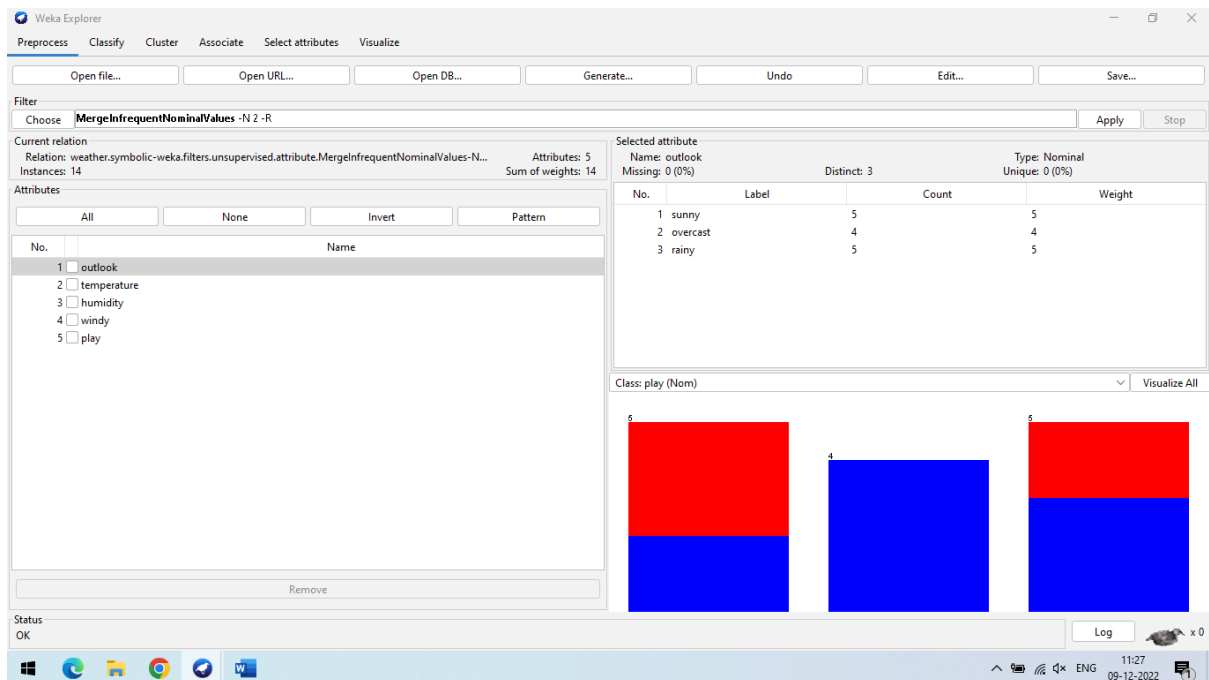


b. Handle Infrequent Nominal Values

Step 1: First of all here we will use the weather.nominal.arff dataset.



Step-2 : After that we have to choose mergeInfrequentNominal for that we have to follow this path. weka→filters→unsupervised→attribute→mergeInfrequentNominalValues and then click on Apply button.

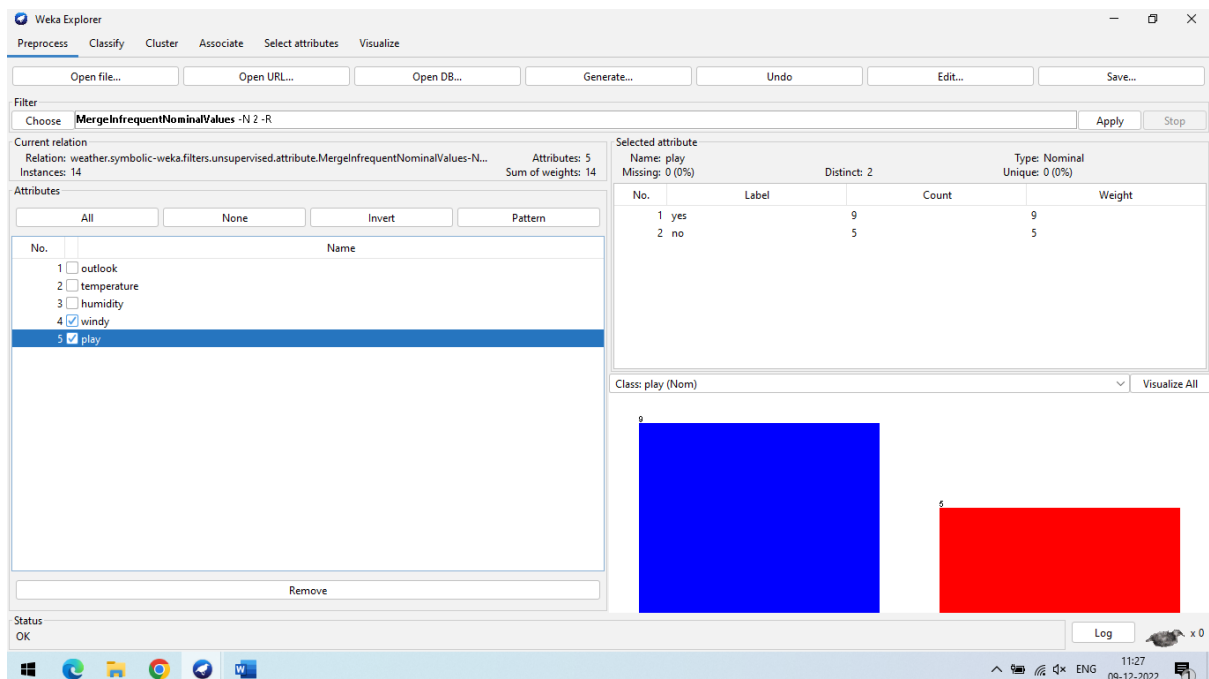


The screenshot shows the Weka Explorer interface with the 'Merge Infrequent Nominal Values' filter applied to the 'outlook' attribute. The 'Attributes' list on the left includes 'outlook', 'temperature', 'humidity', 'windy', and 'play'. The 'Selected attribute' table on the right shows the following data:

No.	Label	Count	Type: Nominal	Weight
1	sunny	5	Unique: 0 (0%)	5
2	overcast	4		4
3	rainy	5		5

Below the table, a bar chart visualizes the distribution of the 'play' class (Nom) for the 'outlook' attribute. The bars are colored red and blue, representing the 'yes' and 'no' classes respectively.

Step 3: After click on apply button we can get the output like this.



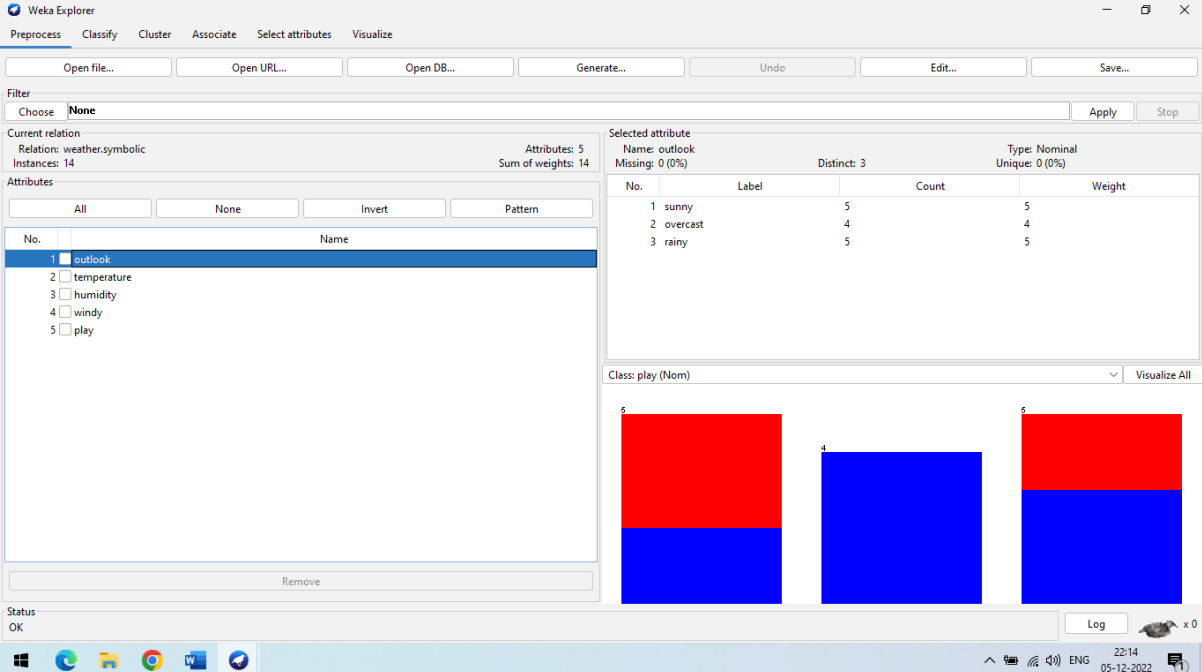
The screenshot shows the Weka Explorer interface with the 'Merge Infrequent Nominal Values' filter applied to the 'play' attribute. The 'Attributes' list on the left includes 'outlook', 'temperature', 'humidity', 'windy', and 'play'. The 'Selected attribute' table on the right shows the following data:

No.	Label	Count	Type: Nominal	Weight
1	yes	9	Unique: 0 (0%)	9
2	no	5		5

Below the table, a bar chart visualizes the distribution of the 'play' class (Nom) for the 'play' attribute. The bars are colored red and blue, representing the 'yes' and 'no' classes respectively.

c. Derive an attribute from the existing attribute

Step-1 : we will use **weather-numeric.arff** database



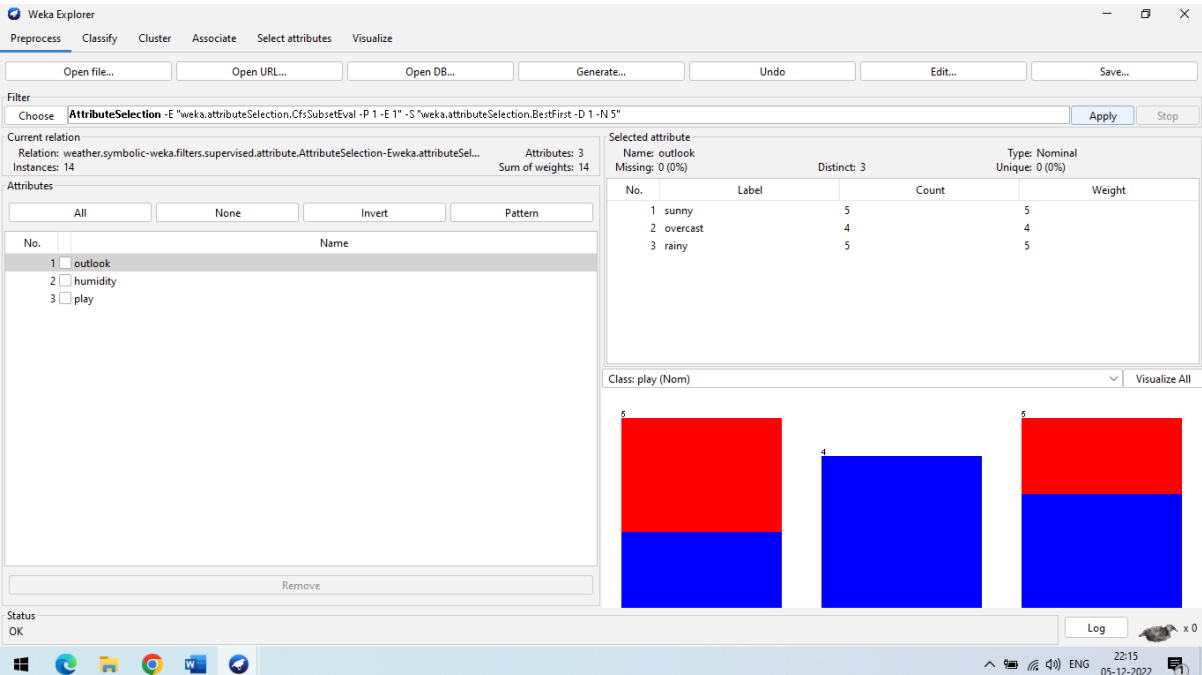
Weka Explorer interface showing the 'weather.symbolic' dataset. The 'Attributes' list on the left includes outlook, temperature, humidity, windy, and play. The 'Selected attribute' table on the right shows the distribution of the 'outlook' attribute.

No.	Label	Count	Weight
1	sunny	5	5
2	overcast	4	4
3	rainy	5	5

Visualize All

Step-2 : weka→filters→supervised→attribute→AttributeSelection

We notice that temperature and windy will be remove.



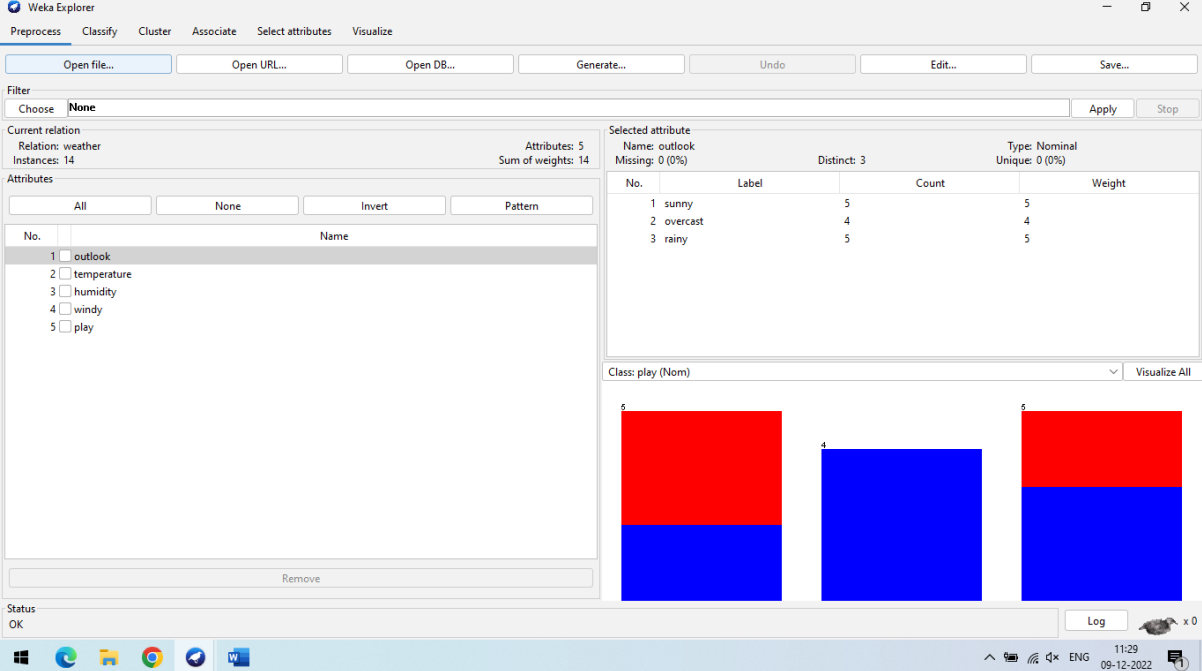
Weka Explorer interface showing the 'AttributeSelection' filter applied. The 'Attributes' list on the left now only includes outlook, humidity, and play. The 'Selected attribute' table on the right remains the same as in Step 1.

No.	Label	Count	Weight
1	sunny	5	5
2	overcast	4	4
3	rainy	5	5

Visualize All

d. Sampling

Step 1: First of all we are using weather-numeric.arff dataset and select the humidity data.

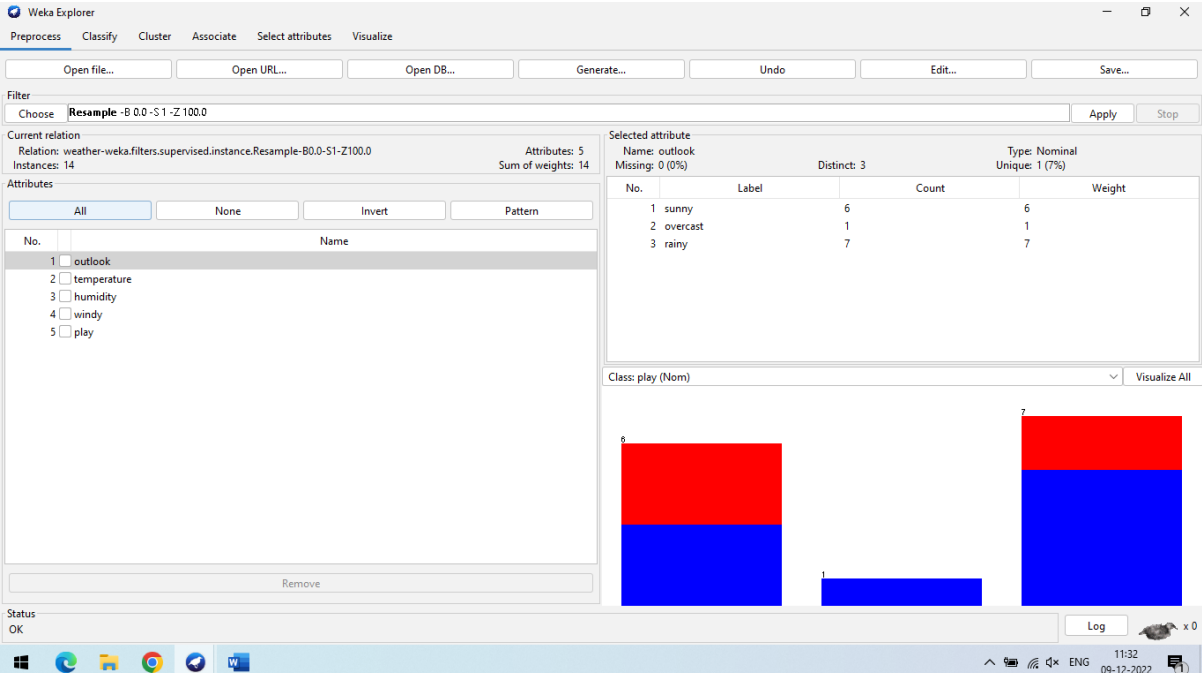


Weka Explorer interface showing the 'weather' dataset. The 'Attributes' list on the left includes outlook, temperature, humidity, windy, and play. The 'Selected attribute' table on the right shows the distribution of the 'outlook' attribute.

No.	Label	Count	Weight
1	sunny	5	5
2	overcast	4	4
3	rainy	5	5

Class: play (Nom) Visualize All

Step-2 : After that we have to follow this path for select the Resample
 Choose→weka→filters→supervised→attribute→instance→Resample and then click on applu button.



Weka Explorer interface showing the 'Resample' filter applied to the 'weather' dataset. The 'Selected attribute' table on the right shows the distribution of the 'outlook' attribute after resampling.

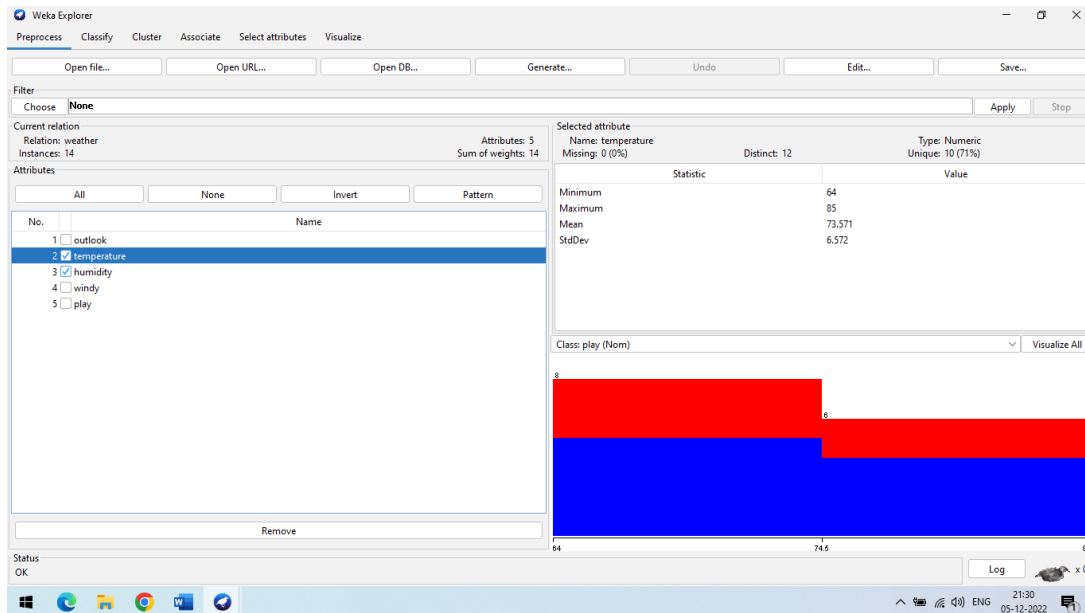
No.	Label	Count	Weight
1	sunny	6	6
2	overcast	1	1
3	rainy	7	7

Class: play (Nom) Visualize All

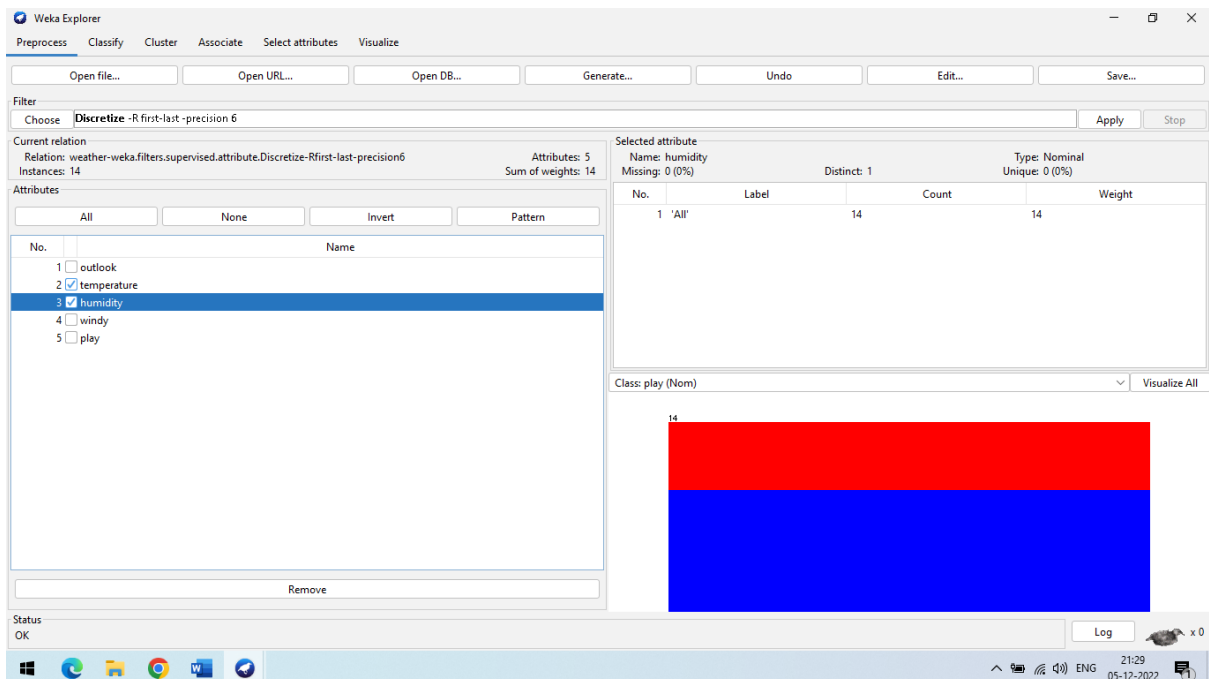
Note: Here you will notice that rainy, overcast and sunny will change.

e. Discretization

Step-1: we will use **weather-numeric.arff** database that contains two **numeric** attributes - **temperature** and **humidity**



Step-2 : Choose→weka→filters→supervised→attribute→Discretize



we will notice that these have changed from numeric to nominal types