|  |  |
| --- | --- |
|  | 1. To perform data cleaning and data transformation operations using Weka |

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student** | **Hardik Prajapati** | **Roll No.** | **9152** |
| **Sign here to indicate that you have read all relevant material provided /available on Moodle while performing and writing this experiment** | | **Sign:** | |

**Late Submission Details (if any)**

|  |  |  |
| --- | --- | --- |
| **Reason(s) of late submission** | **Date of practical performance** | **Date of practical submission** |
|  |  |  |

**References used**

|  |  |  |
| --- | --- | --- |
| 1 | Name and author of reference book(s) with page nos. |  |
| 2 | Name and roll nos. of the peers whose help you have taken (if any) |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Rubrics for assessment of Experiment:**   |  |  |  |  | | --- | --- | --- | --- | | Indicator | Poor | Average | Good | | Timeliness  Maintains Experiment deadline (3) | Experiment not done (0) | One or More than One week late (1-2) | Maintains deadline (3) | | Completeness and neatness  Complete all parts of Experiment (3) | N/A | < 80% complete (1-2) | 100% complete (3) | | Originality  Extent of plagiarism (2) | Copied it from someone else (0) | At least try to implement but could not succeed (1) | Implemented (2) | | Knowledge  In depth knowledge of the Experiment (2) | Unable to answer any questions (0) | Unable to answer few questions (1) | Able to answer all questions (2) | |
| **Assessment Marks:**   |  |  | | --- | --- | | Timeliness |  | | Completeness and neatness |  | | Originality |  | | Knowledge |  | | Total |  | |

**Signature of Teacher with date**

|  |  |
| --- | --- |
| **1.** | **Course, Subject & Experiment Details** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Course & Branch** | **T.E. (ECS)** | **Estimated Time** | **02 Hours Per Week** |
| **Current Semester** | **Semester V** | **Subject Name** | **DWM** |
| **Chapter No. & Unit** | **3** | **Chapter Title** | **Data pre-processing** |
| **Experiment Type** | **Software Performance** | **Subject Code** | **ECC 604** |

|  |  |
| --- | --- |
| **2.** | **Aim & Objective of Experiment** |

1. Understand data cleaning and data transformation operations

|  |  |
| --- | --- |
| **3.** | **Expected Outcome of Experiment** |

1. Demonstrate data cleaning operation using Weka
2. Demonstrate data transformation operation using Weka

|  |  |
| --- | --- |
| **4.** | **Theory and procedure of the experiment** |

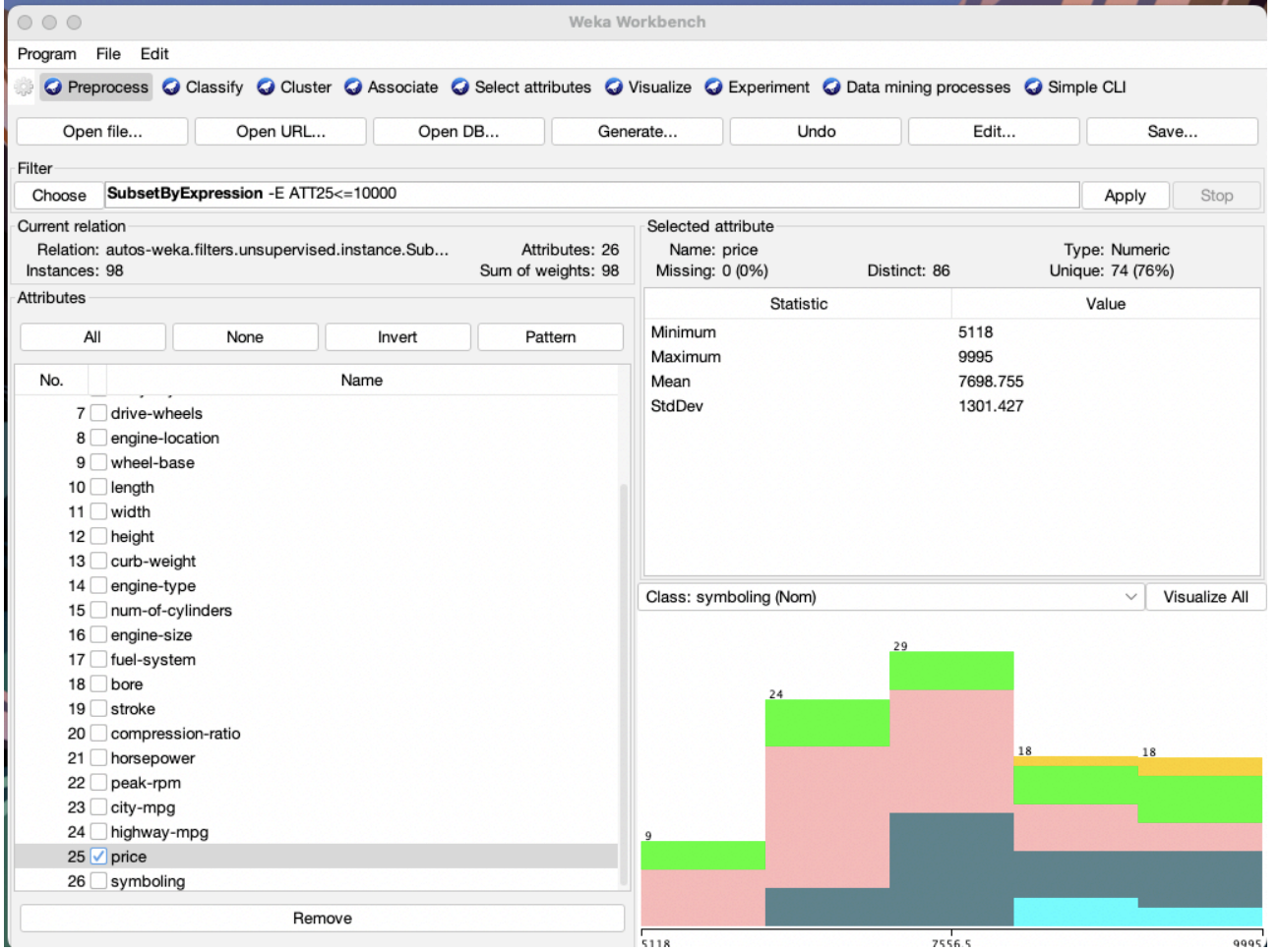
Solve Exercise 1, 2, 3A, 3B and 4 from the following document:

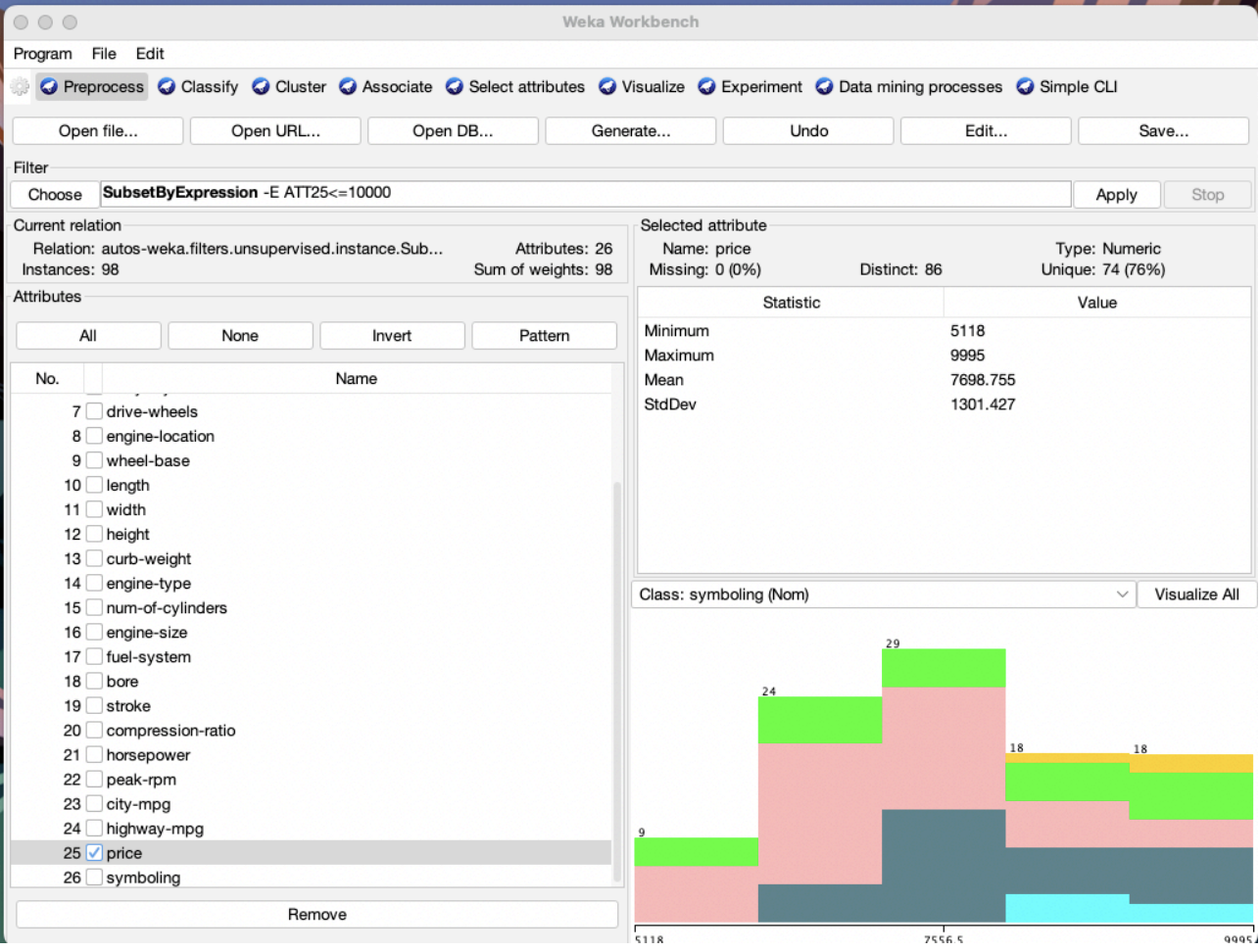
<https://drive.google.com/file/d/12uy7hEpYEUy5UdIJkRChWR0LUNmDYIjp/view?usp=share_link>

|  |  |
| --- | --- |
| **4.** | **Output** |

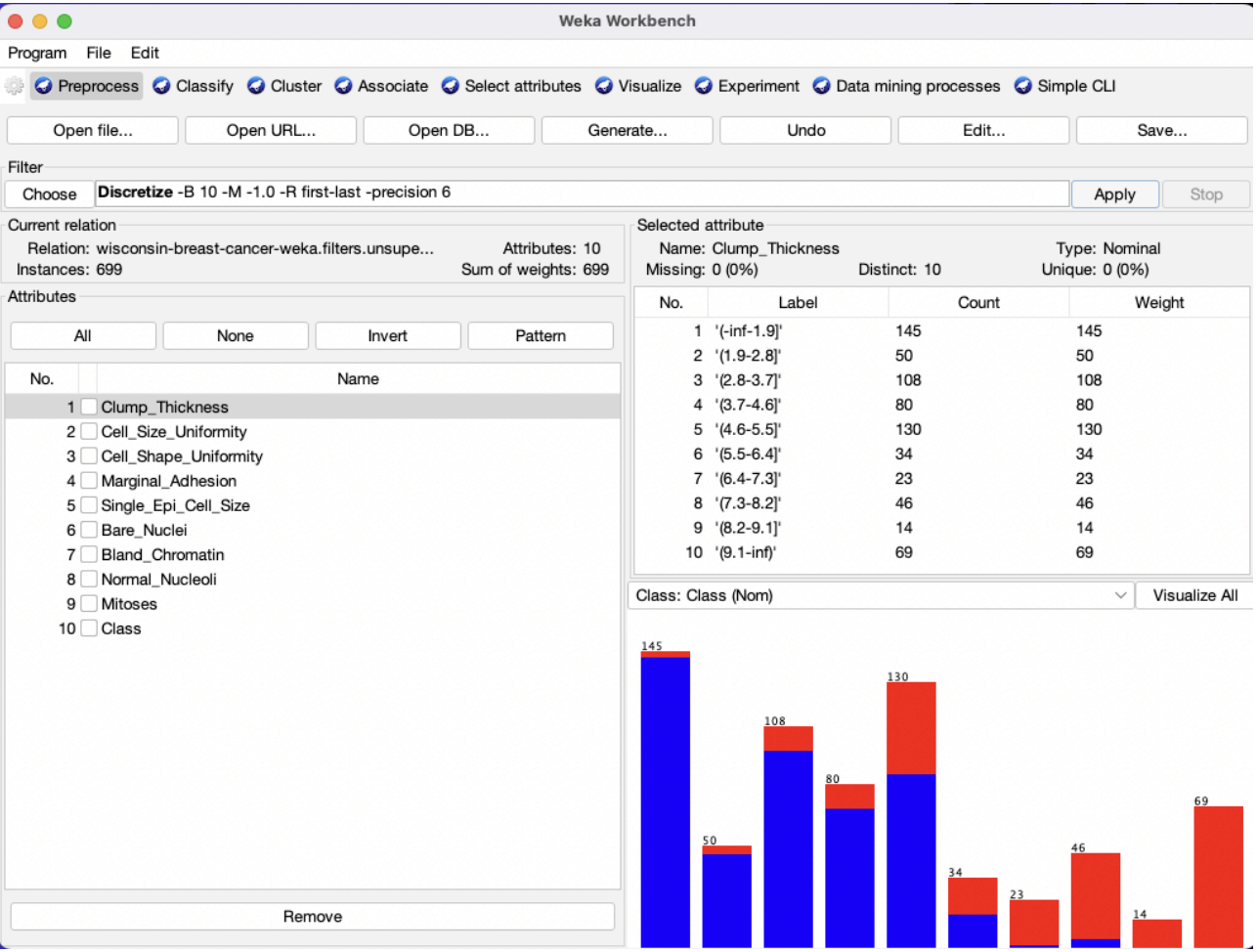
Attach the screenshots of output above exercises and comment on the result of each exercise

1A

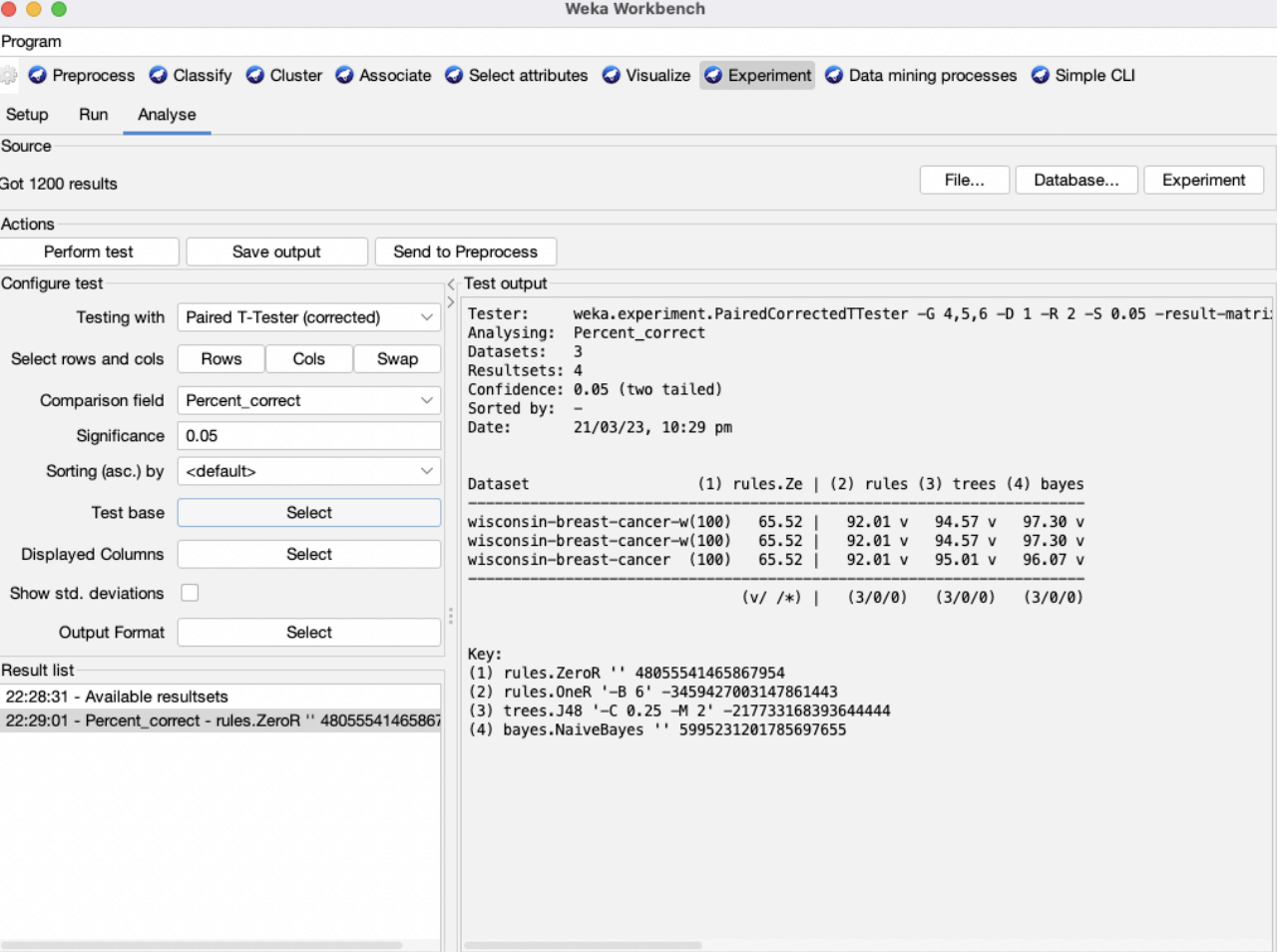




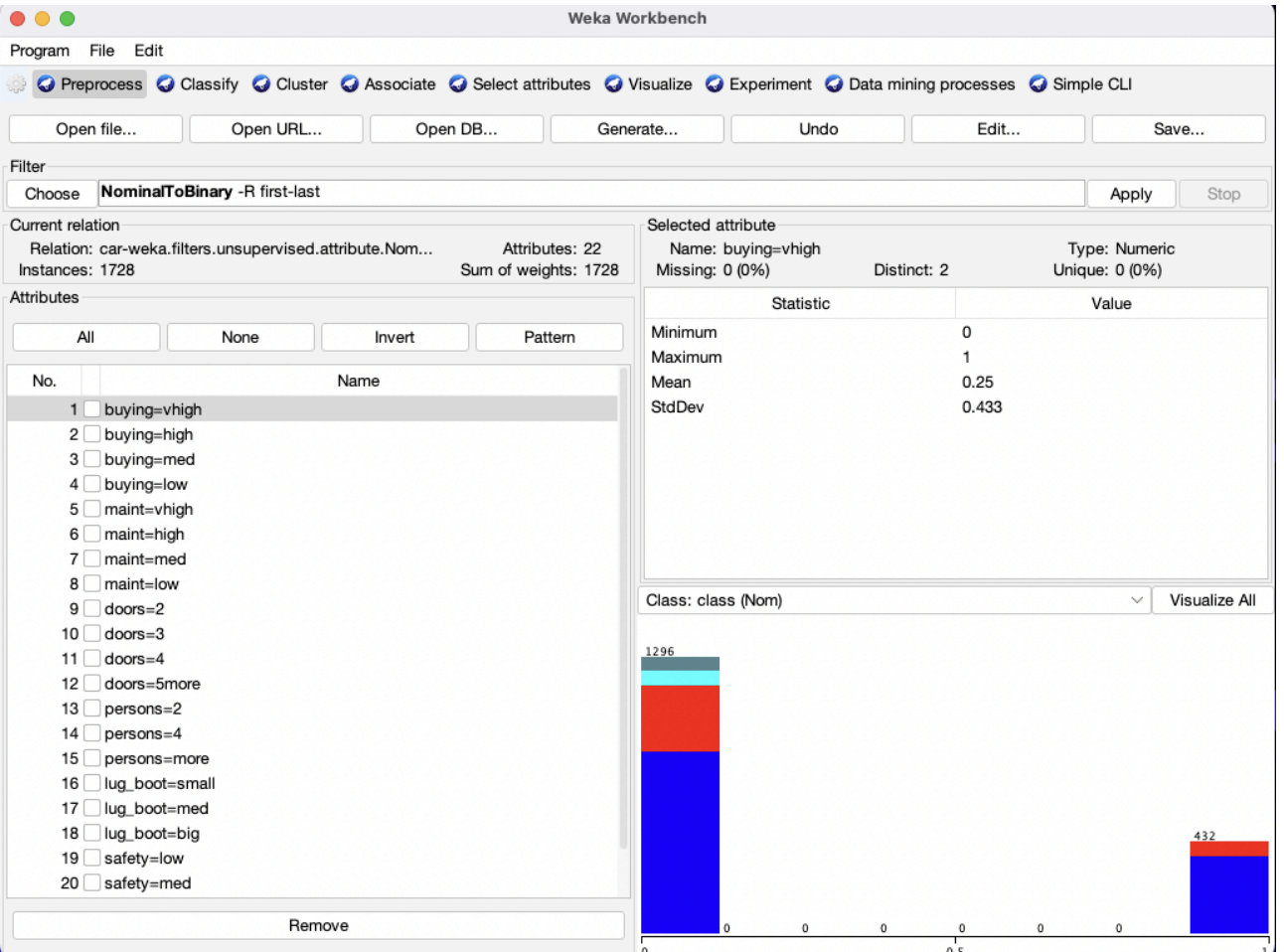
2A

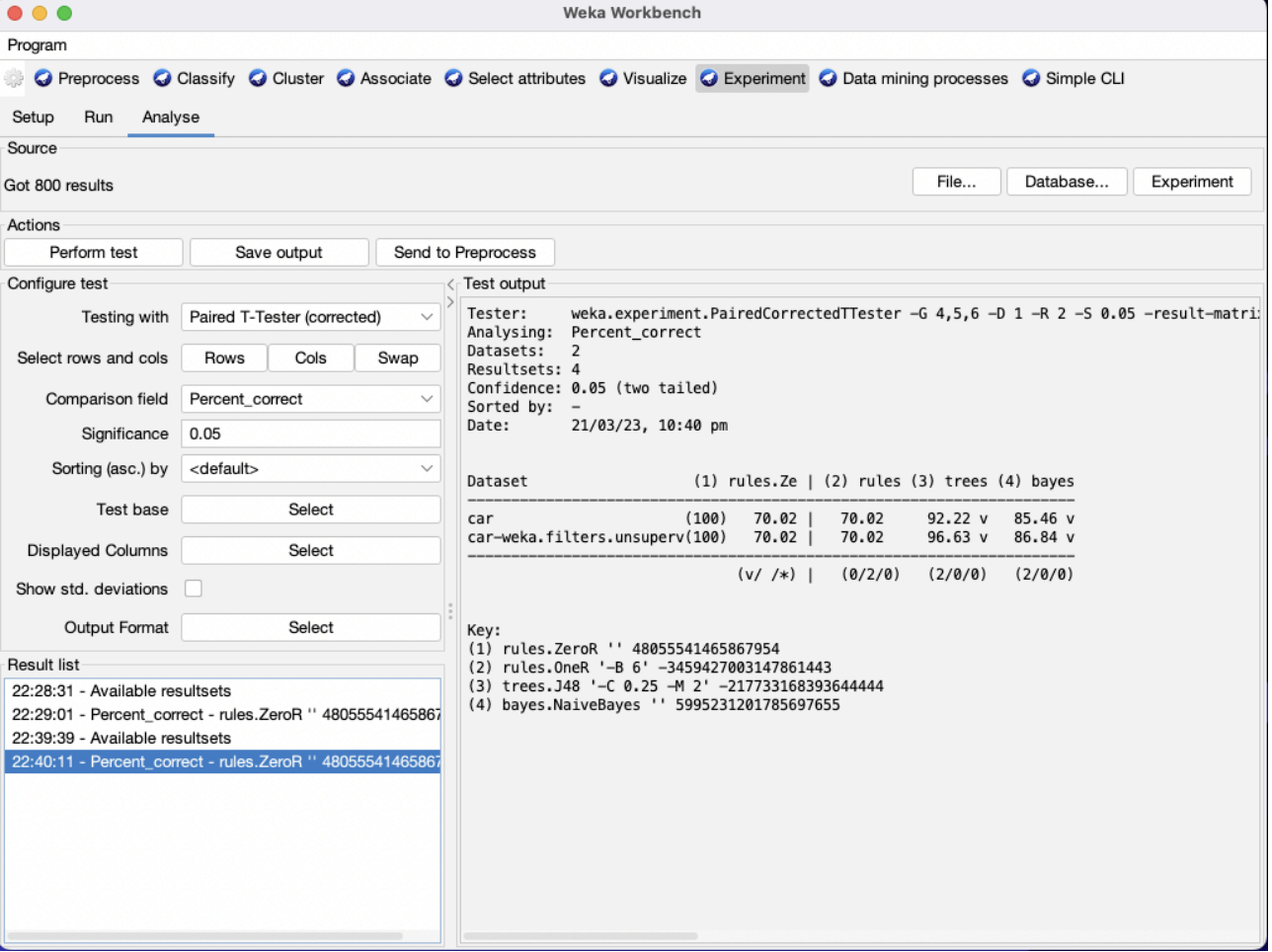


2 B

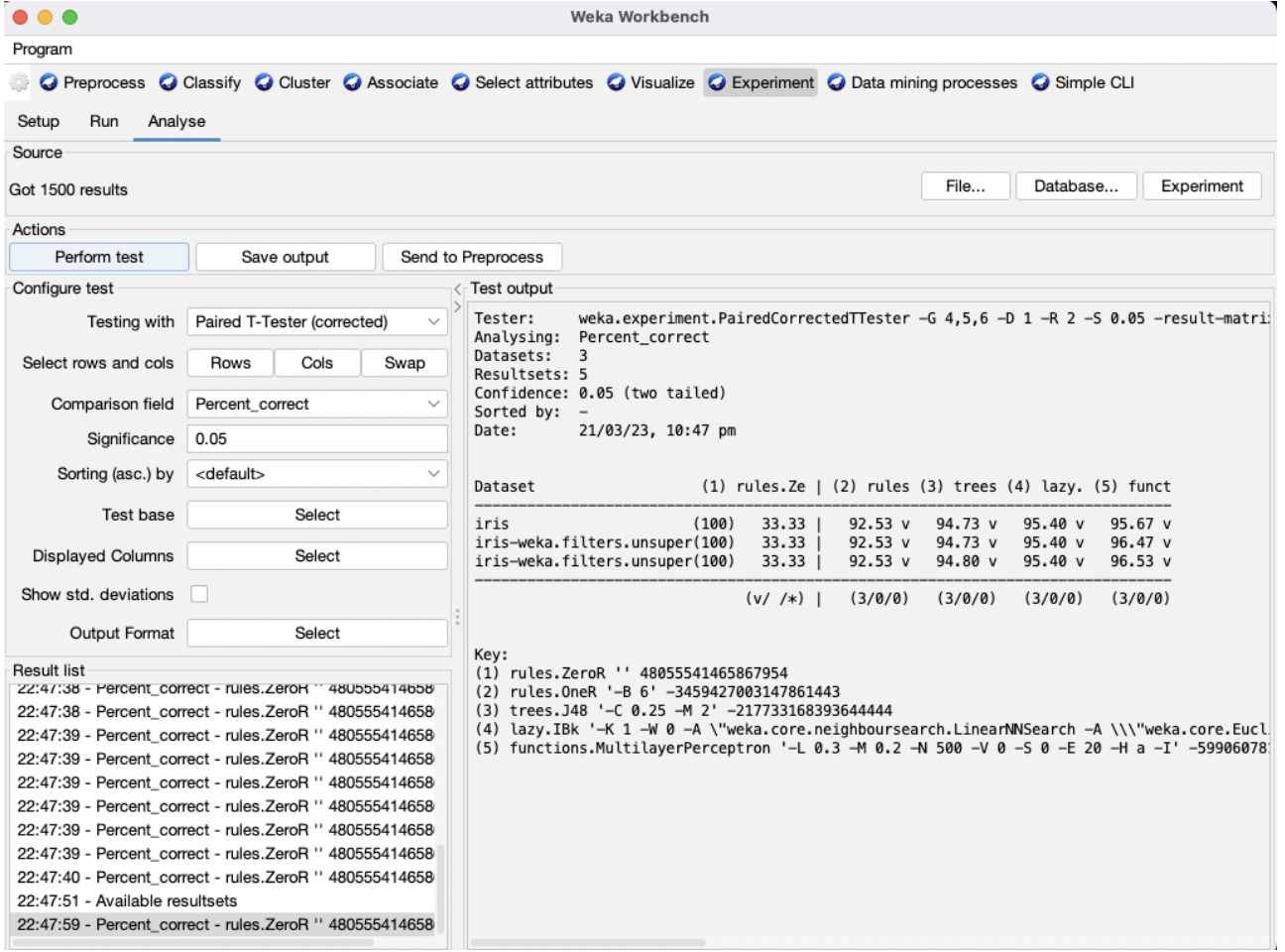


3A

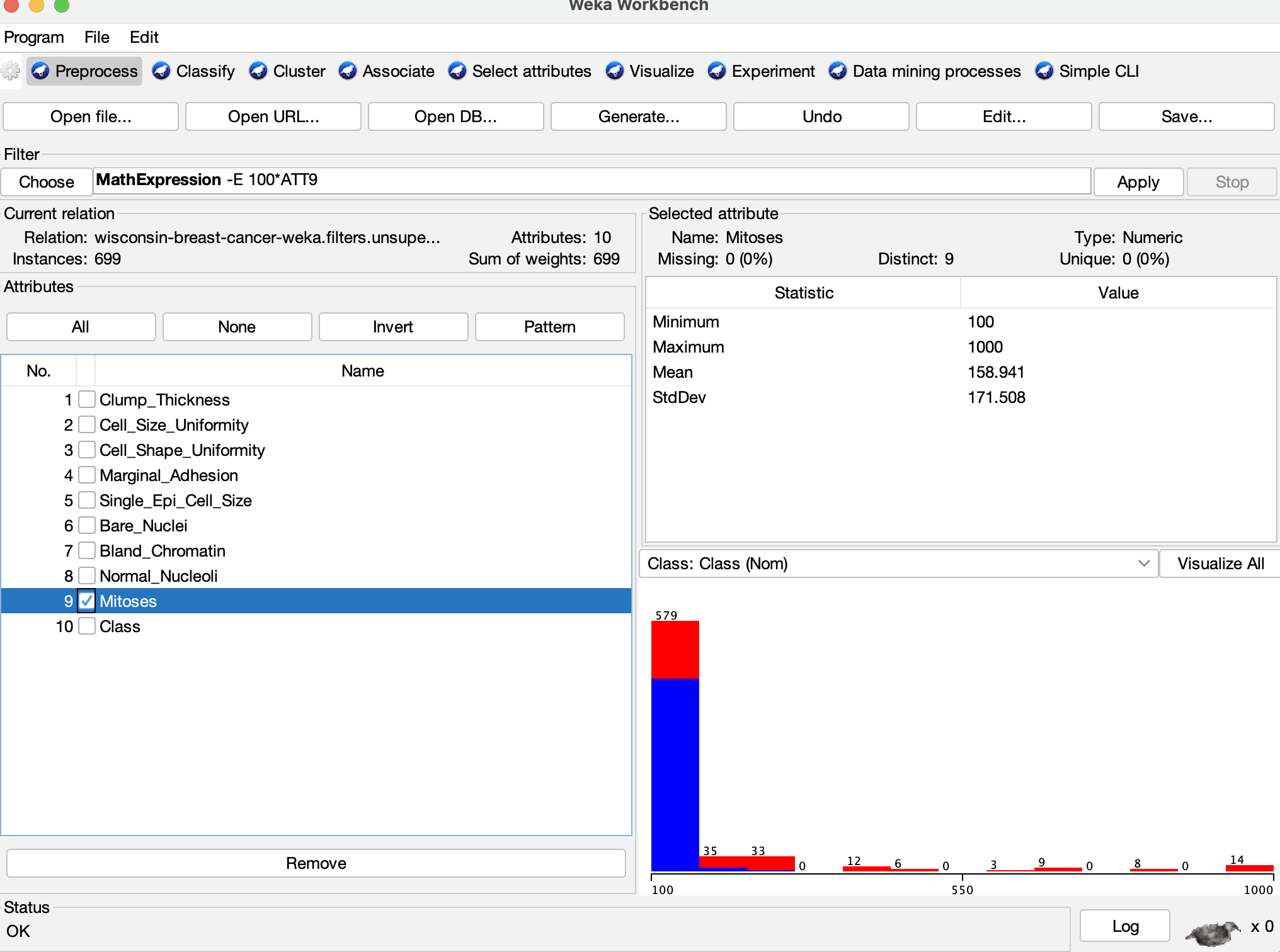




3B



4



|  |  |
| --- | --- |
| **5.** | **Conclusions & Inferences** |

From the above experiment, we have learnt how to use Weka for analysis of various data present using algorithms such as ZeroR, OneR, J48, IB1, MultilayerPerceptron

|  |  |
| --- | --- |
| **6.** | **Post Lab exercise** |

1. Describe the following classifiers used in this experiment:

**ZeroR, OneR, J48, IB1, MultilayerPerceptron**

ZeroR: ZeroR is a simple baseline classifier that always predicts the most frequent class in the training dataset. It does not take into account any input features and makes its predictions solely based on the class distribution of the training data. ZeroR is often used as a benchmark to compare the performance of more complex classifiers.

OneR: OneR is a simple rule-based classifier that selects a single feature and creates a decision rule based on that feature. It selects the feature that has the lowest error rate when used to predict the target class. The resulting rule is then used to classify new instances based on the value of the selected feature.

J48: J48 is a decision tree algorithm that builds a tree-like model of the data by recursively partitioning the feature space based on the target class. At each step, J48 selects the feature that provides the best split in terms of maximizing the information gain or minimizing the Gini index.

IB1: IB1 is an instance-based classifier that uses the entire training dataset as its model. To classify a new instance, it searches the training data for the k nearest neighbors based on a distance metric (e.g., Euclidean distance) and predicts the target class based on the majority class of those neighbors. In IB1, k is set to 1, which means that only the nearest neighbor is used to make predictions.

MultilayerPerceptron: MultilayerPerceptron is a type of artificial neural network that consists of multiple layers of interconnected nodes (neurons). Each neuron receives input from other neurons and applies a non-linear activation function to produce an output. The layers are typically organized into an input layer, one or more hidden layers, and an output layer.

1. List the tools used in industry to perform similar tasks, similar to the tasks in this experiment

Some of the commonly used tools in industry that work with Weka include:

KNIME: KNIME is an open-source data analytics platform that provides a graphical user interface for designing data processing workflows. KNIME integrates with Weka, allowing users to incorporate Weka models and algorithms into their workflows.

R: R is a popular programming language for statistical computing and data analysis. The RWeka package provides an interface between R and Weka, allowing R users to use Weka algorithms within their R code.

Python: Python is another popular programming language for data analysis and machine learning. The Python-weka-wrapper package provides a Python interface to Weka, allowing Python users to use Weka algorithms within their Python code.

Hadoop: Hadoop is a popular framework for distributed storage and processing of large datasets. The WekaHadoop package provides an interface between Weka and Hadoop, allowing users to run Weka algorithms on Hadoop clusters.

Mahout: Mahout is an open-source framework for scalable machine learning. The Mahout-Weka integration allows users to use Weka algorithms within Mahout, enabling the development of scalable machine learning applications.