



**Amal Jyothi College of Engineering
Kanjirappally, Kerala**

**PREDICTION OF EDIBLE AND POISONOUS
MUSHROOMS USING WEKA**

INTEGRATED MCA SEMINAR REPORT

*Submitted in the partial fulfillment of the requirements for the
Award of the Degree in*

Integrated Master of Computer Applications By

SIJIMOL CYRIAC

Reg No: AJC16MCA-I52

Under The Guidance Of

Ms. MEERA ROSE MATHEW

January 2021
DEPARTMENT OF COMPUTER
APPLICATIONS AMAL JYOTHI COLLEGE OF
ENGINEERING KANJIRAPPALLY



CERTIFICATE

*This is to certify that the seminar report, "**Prediction of Edible and Poisonous Mushrooms Using Weka Tool**" is the bonafide work of **SIJIMOL CYRIAC (Reg.No:AJC16MCA-I52)** in partial fulfillment of the requirements for the award of the Degree of Integrated Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2020-21.*

Ms. MEERA ROSE MATHEW
Internal Guide

Ms. SONA MARIA SEBASTIAN
Coordinator

Fr. Rubin Thottupuram
Head of the Department

ACKNOWLEDGEMENT

First and foremost, I thank God almighty for his eternal love and protection throughout the seminar. I take this opportunity to express my gratitude to all who helped me in completing this seminar successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our manager **Rev. Fr. Dr. Mathew Paikatt** and Principal **Dr. Z V Lakaparampil** for providing good faculty for guidance.

I owe a great depth of gratitude towards our Head of the Department Fr. Rubin Thottupuram for helping us. I extend my whole hearted thanks to the seminar coordinators Fr. Rubin Thottupuram and Ms. Sona Maria Sebastian for their valuable suggestions and for overwhelming concern and guidance from the beginning to the end of the seminar. I would also like to express sincere gratitude to my guide, Ms. Meera Rose Mathew for her inspiration and helping hand.

I thank our beloved teachers for their cooperation and suggestions that helped me throughout the seminar. I express my thanks to all my friends and classmates for their interest, dedication, and encouragement shown towards the seminar. I convey my hearty thanks to my family for the moral support, suggestions, and encouragement to make this venture a success.

ABSTRACT

Mushrooms are actually fungi kingdom, but come under the category of vegetable for cooking purposes. They are fat-free, low-sodium, low-calorie, and cholesterol-free. Mushrooms are of different types both edible and poisonous, each mushroom has a unique look and taste. But nutritional benefits vary depending on the type of mushroom. They contain essential nutrients such as proteins, vitamins, minerals, amino acids, antibiotics and antioxidants.

But all species of mushrooms are not edible. So before eating, it should be checked whether it is edible or not. Eating mushrooms collected in the wild is risky and should only be undertaken by individuals knowledgeable in mushroom recognition. Actual determination and proper identification of species are the only safe way to ensure edibility and safeguard against possible accident of consuming poisonous one. This review paper aims in identifying edible and poisonous mushrooms based on certain attributes like shape, size, color, etc. A large volume of data is downloaded from Kaggle repository. The data have been formed into two groups, one for training and another for testing. We use one of the Supervised Learning Algorithms for training. This classification model is saved and then used later to predict whether a given mushroom is edible or not. This is done with the help of ***weka tool*** (Waikato Environment for Knowledge Analysis).

CONTENTS

1	INTRODUCTION	1
1.1	WAIKATO ENVIRONMENT FOR KNOWLEDGE ANALYSIS (WEKA)	
1.2	HISTORY OF WEKA TOOL	
1.3	WORKING OF WEKA TOOL	
2	PREDICTION OF EDIBLE AND POISONOUS MUSHROOMS USING DECISION TREE ALGORITHM WITH THE HELP OF WEKA	8
2.1	INTRODUCTION TO DECISION TREE ALGORITHM(J48 ALGORITHM)	
2.2	STEPS	
2.3	FEATURES	
2.4	LIMITATIONS	
3	WORKING OF PREDICTION OF EDIBLE AND POISONOUS MUSHROOMS USING DECISION TREE ALGORITHM WITH THE HELP OF WEKA	11
4	NEW PROBLEM CONCEPT	28
5	CONCLUSION	28
6	REFERENCES	29

1. INTRODUCTION

1.1 Waikato Environment for Knowledge Analysis (WEKA)

Data mining is a process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems. Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD. Data mining includes the utilization of refined data analysis tools to find previously unknown, valid patterns and relationships in huge data sets. Thus, data mining incorporates analysis and prediction. In recent data mining projects, various major data mining techniques have been developed and used, including association, classification, clustering, prediction, sequential patterns, and regression. **Data Mining tools** have the objective of discovering patterns/trends/groupings among large sets of data and transforming data into more refined information. It is a framework that allows you to perform different types of data mining analysis. We can perform various algorithms such as clustering or classification on your data set and visualize the results itself.

WEKA is a data mining tool developed by the University of Waikato in New Zealand that implements data mining algorithms. WEKA is open source software issued under the GNU General Public License. It provides tools for data preprocessing, implementation of several machine learning algorithms, and visualization tools so that we can develop machine learning techniques and apply them to real-world data mining problems. Weka supports several standard data mining tasks, more specifically, data preprocessing, clustering, classification, regression, visualization, and feature selection. All of Weka's techniques are based on the assumption that the data is available as one flat file or relation, where each data point is described by a

fixed number of attributes (numeric or nominal attributes, some other supported attribute types). Weka provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query. Weka provides access to deep learning with Deeplearning4j. It is not capable of multi-relational data mining, but there is separate software for converting a collection of linked database tables into a single table that is suitable for processing using Weka.

Features and other information of Data Mining includes

- **Open Source:** It is released as open source software under the GNU GPL. It is dual licensed and Pentaho Corporation owns the exclusive license to use the platform for business intelligence in their own product.
- **Graphical Interface:** It has a Graphical User Interface (GUI). This allows you to complete your machine learning projects without programming.
- **Command Line Interface:** All features of the software can be used from the command line. This can be very useful for scripting large jobs.
- **Java API:** It is written in Java and provides a API that is well documented and promotes integration into your own applications. Note that the GNU GPL means that in turn your software would also have to be released as GPL.
- **Documentation:** There are books, manuals, wikis and MOOC courses that can train you how to use the platform effectively.
- **Portability:** Since it is fully implemented in the Java programming language and thus runs on almost any modern computing platform.
- A comprehensive collection of data preprocessing and modelling techniques.
- It contains 49 data preprocessing tools, 76 classification and regressions algorithms, 8 clustering algorithms, 3 algorithms for finding association rule, 15 attribute selection algorithms

and 10 feature selection algorithms are present in WEKA.

- Using WEKA, users can develop custom code for machine learning.
- Platform Independent
- Flexibility for scripting experiments
- Ease of use due to its graphical user interfaces.

1.2 HISTORY OF WEKA

- In 1993, the University of Waikato in New Zealand began development of the original version of Weka, which became a mix of Tcl/Tk, C, and Makefiles.
- The original non-Java version of Weka was a Tcl/Tk front-end to (mostly third-party) modeling algorithms implemented in other programming languages, plus data preprocessing utilities in C, and a Makefile-based system for running machine learning experiments. This original version was primarily designed as a tool for analyzing data from agricultural domains
- In 1997, the decision was made to redevelop Weka from scratch in Java, including implementations of modeling algorithms.
- In 2005, Weka team received the SIGKDD Data Mining and Knowledge Discovery Service Award for their development of the freely-available Weka Data Mining Software.
- In 2006, Pentaho Corporation acquired an exclusive licence to use Weka for business intelligence. It forms the data mining and predictive analytics component of the Pentaho business intelligence suite. Pentaho has since been acquired by Hitachi Vantara, and Weka now underpins the PMI (Plugin for Machine Intelligence) open source component.

1.3 WORKING OF WEKA TOOL

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms are applied directly to a dataset. WEKA implements algorithms for data pre-processing, classification, regression, clustering, association rules; it also includes a visualization tool and I is made use of for the prediction. It is also well-suited for developing new machine learning schemes.

The raw data collected from the field may contain several null values and irrelevant fields. The data preprocessing tools provided in WEKA helps to cleanse the data and save the preprocessed data in your local storage for applying ML algorithms.

Next, depending on the kind of ML model that are trying to develop we can select one of the options such as **Classify, Cluster, Associate** etc. The **Attributes Selection** allows the automatic selection of features to create a reduced dataset. Under each category, WEKA provides the implementation of several algorithms. We can select an algorithm of our choice, set the desired parameters and run it on the dataset.

The type of algorithms that you apply is based largely on your domain knowledge. Even within the same type, for example classification, there are several algorithms available. You may like to test the different algorithms under the same class to build an efficient machine learning model. While doing so, you would prefer visualization of the processed data and thus you also require visualization tools. Each entry in a dataset is an instance of the java class: – `weka.core.Instance`.

Then, WEKA would give you the statistical output of the model processing. It provides you a visualization tool to inspect the data. The various models can be applied on the same dataset. We can then compare the outputs of different models and select the best that meets our purpose. Thus, the use of WEKA results in a quicker development of machine learning models on the whole.

Weka Data Formats

Weka uses the Attribute Relation File Format for data analysis, by default. But listed below are some formats that Weka supports, from where data can be imported:

- CSV
- ARFF
- Database using ODBC

Each instance consists of a number of attributes:

- Nominal: one of a predefined list of values – e.g. red, green, blue
- Numeric: A real or integer number
- String: Enclosed in “double quotes”
- Date
- Relational

The external representation of an Instances class consists of:

- A header: Describes the attribute types
- Data section: Comma separated list of data

Different WEKA Tool GUI Chooser

The WEKA GUI Chooser application allows you to run five different types of applications as listed below:

- **Explorer** for data preparation, feature selection and evaluating algorithms.
- **Experiment** Environment for designing, running and analyzing the results from controlled experiments.
- **KnowledgeFlow** Environment for graphically designing and executing machine learning pipelines.
- **Workbench** that incorporates all of the Weka tools into a single convenient interface.

- **Simple CLI** for using the Weka API from the command line.

Explorer

The Weka Explorer is designed to investigate your machine learning dataset. It is an environment for exploring data. The explorer is where you play around with your data and think about what transforms to apply to your data, what algorithms you want to run in experiments. It is useful when you are thinking about different data transforms and modeling algorithms that you could investigate with a controlled experiment later. It is excellent for getting ideas and playing what-if scenarios. The interface is divided into 6 tabs, each with a specific function.

Experimenter

The Weka Experiment Environment is for designing controlled experiments, running them, then analyzing the results collected. It is an environment for performing experiments and conducting statistical tests between learning schemes. This interface is for designing experiments with your selection of algorithms and datasets, running experiments and analyzing the results. The tools for analyzing results are very powerful, allowing you to consider and compare results that are statistically significant over multiple runs.

KnowledgeFlow

It is an environment supports essentially the same functions as the Explorer but with a drag-and-drop interface. One advantage is that it supports incremental learning. Applied machine learning is a process and the Knowledge Flow interface allows you to graphically design that process and run the designs that you create. This includes the loading and transforming of input data, running of algorithms and the presentation of results.

The Weka KnowledgeFlow Environment is a graphical workflow tool for designing a machine learning pipeline from data source to results summary, and much more. Once designed, the pipeline can be executed and evaluated within the tool. The KnowledgeFlow Environment is a powerful tool.

Workbench

It is an environment that combines all of the GUI interfaces into a single interface. It provides three main ways to work on your problem: The Explorer for playing around and trying things out, the Experimenter for controlled experiments, and the KnowledgeFlow for graphically designing a pipeline for your problem.

The Weka Workbench environment that combines all of the GUI interfaces into a single interface. It is useful if you find yourself jumping a lot between two or more different interfaces, such as between the Explorer and the Experiment Environment.

Simple CLI

Weka can be used from a simple Command Line Interface (CLI). This is powerful because you can write shell scripts to use the full API from command line calls with parameters, allowing you to build models, run experiments and make predictions without a graphical user interface.

The Simple CLI provides an environment where you can quickly and easily experiment with the Weka command line interface commands. It provides a simple command-line interface that allows direct execution of WEKA commands for operating systems that do not provide their own command line interface.

2. PREDICTION OF EDIBLE AND POISONOUS MUSHROOMS USING DECISION TREE ALGORITHM WITH THE HELP OF WEKA

Consumption of poisonous mushrooms may vary from gastrointestinal upset to life-threatening organ failure resulting in death. Therefore care must be taken to classify edible mushrooms with that of poisonous ones. The objective of this paper is to do this classification using the features of mushroom extracted from Kaggle repository. WEKA also offers many classification algorithms for decision tree. Here, we use J48 algorithm, one of the popular classification algorithms which outputs a decision tree.

2.1 INTRODUCTION TO DECISION TREE ALGORITHM (J48 ALGORITHM)

Decision Tree is the classification technique that consists of three components root node, branch (edge or link), and leaf node. Root represents the test condition for different attributes, the branch represents all possible outcomes that can be there in the test, and leaf nodes contain the label of the class to which it belongs. The root node is at the starting of the tree which is also called the top of the tree.

Another more advanced decision tree algorithm that we use here is the C4.5 algorithm, called J48 in Weka. J48 classifier is an algorithm to generate a decision tree that is generated by C4.5 (an extension of ID3). It is also known as a statistical classifier. Decision tree J48 is the implementation of algorithm ID3 (Iterative Dichotomiser 3) developed by the WEKA project team. You can review a visualization of a decision tree prepared on the entire training data set by right clicking on the "Result list" and clicking "Visualize Tree".

Decision trees can support classification and regression problems. Decision trees are more recently referred to as Classification And Regression Trees (CART). They work by creating a tree to evaluate an instance of data, start at the root of the tree and moving town to the leaves (roots) until a prediction can be made. The process of creating a decision tree works by greedily selecting the best split point in order to make predictions and repeating the process until the tree is a fixed depth. After the tree is constructed, it is pruned in order to improve the model's ability to generalize to new data. Decision tree can be constructed moderately quick compare to other methods of classification. Decision tree classifiers obtain like or better accuracy when compare with other classification methods.

- The topmost node in the Decision tree is called the **Root node**
- The bottom-most node is called the **Leaf node**
- A node divided into sub-nodes is called a **Parent node**. The sub-nodes are called **Child nodes**

2.2 STEPS

Step 1: Download Weka and Install if it is not pre-installed.

Visit the Weka Download page and locate a version of Weka suitable for your computer (Windows, Mac, or Linux).

Step 2: Start Weka. This may involve finding it in program launcher or double clicking on the weka.jar file. This will start the Weka GUI Chooser. The Weka GUI Chooser lets you choose one of the Explorer, Experimenter, KnowledgeFlow and the Simple CLI (command line interface).

Step 3: Open the data/mushroom.arff Dataset. Click the "**Open file...**" button to open a data set and double click on the "**data**" directory. Weka provides a number of small common machine learning datasets that you can use to practice on. Select the "**mushroom.arff**" file to load the Iris dataset.

Step 4: Select and Run an Algorithm. Now that you have loaded a dataset, it's time to choose a machine learning algorithm to model the problem and make predictions. Click the "**Classify**" tab. This is the area for running algorithms against a loaded dataset in Weka. You will note that the "**ZeroR**" algorithm is selected by default. Here we use "**J48**" algorithm, so select the "**J48**" algorithm from tree option. Click the "**Start**" button to run this algorithm.

Step 5: Review Results. After running the "J48" algorithm, you can note the results in the "Classifier output" section and the presented result is a summary of those predictions.

2.3 FEATURES

- It handles classification with the missing values in the data.
- It can be applied to both discrete and continuous variables.
- It also performs the pruning of the tree.
- It can handle high dimensional data.
- It replaces internal node with a leaf node and thus reduces the error rate.

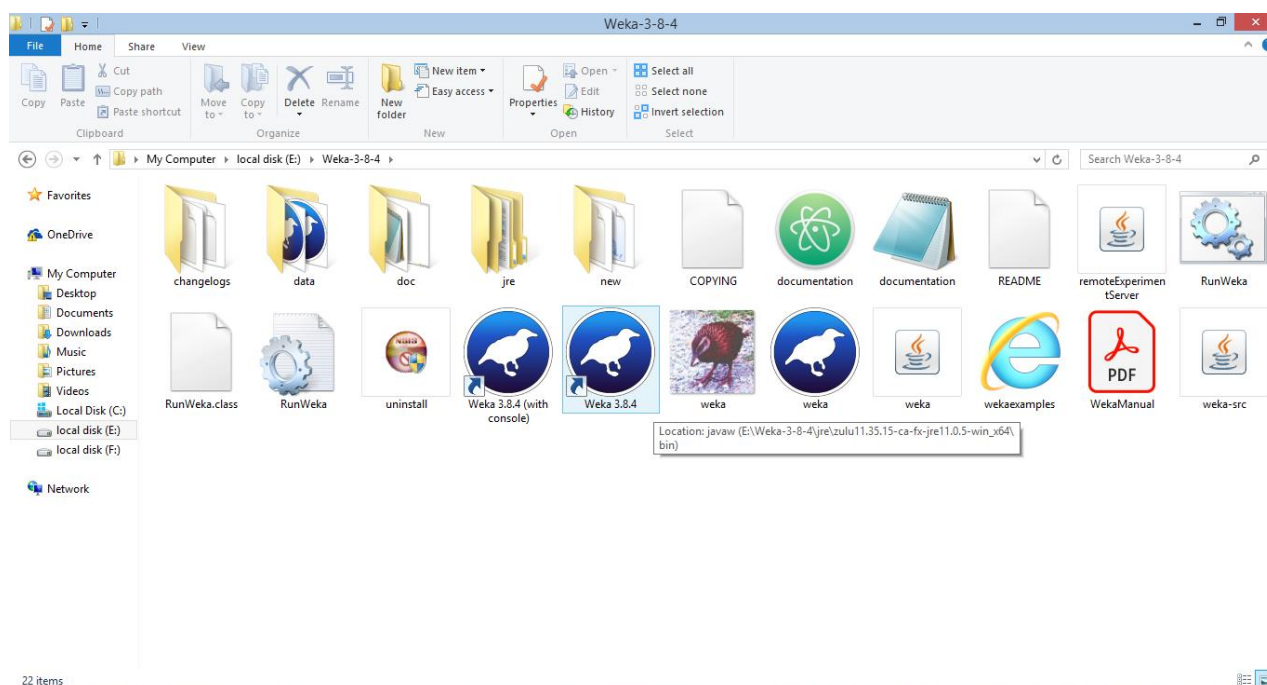
2.4 LIMITATIONS

- A small change in the data can cause a large change in the structure of the decision tree causing instability.
- For a Decision tree sometimes calculation can go far more complex compared to other algorithms.
- Decision tree often involves higher time to train the model.
- Decision tree training is relatively expensive as the complexity and time has taken are more.
- The Decision Tree algorithm is inadequate for applying regression and predicting continuous values.

3. WORKING OF PREDICTION OF EDIBLE AND POISONOUS MUSHROOMS USING DECISION TREE ALGORITHM WITH THE HELP OF WEKA TOOL

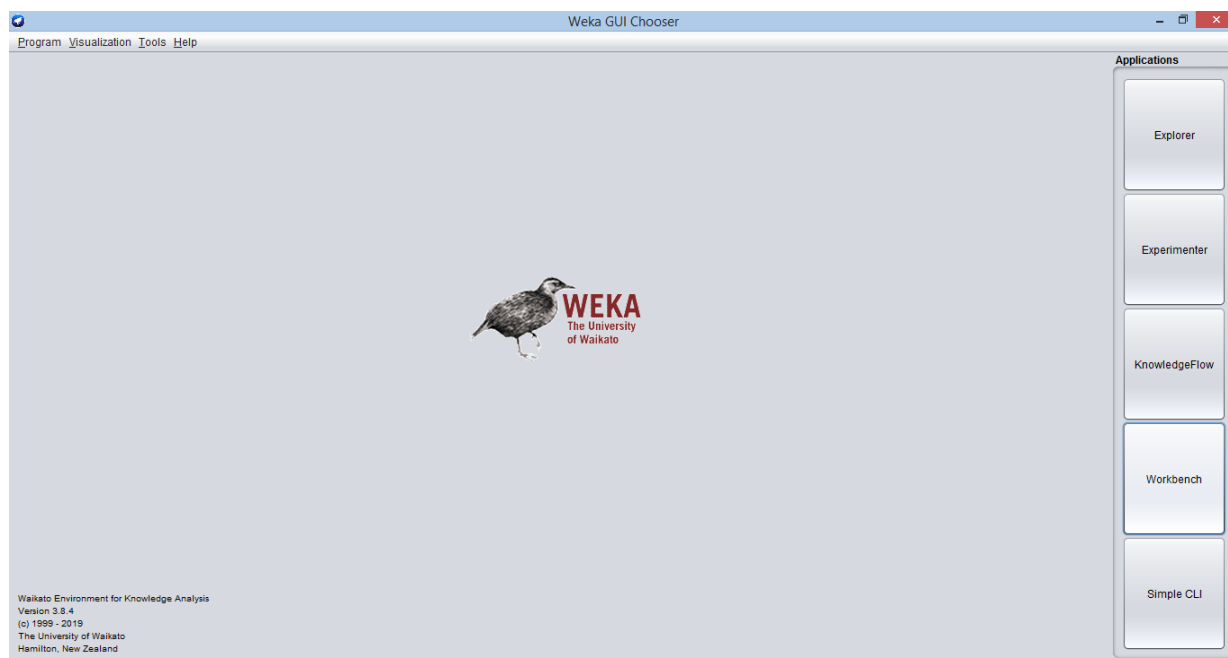
1. Download and install Weka if it is not installed.
2. Once the install is finished, start the Weka GUI Chooser. The Weka GUI Chooser lets you choose one of the following:
 - Explorer
 - Experimenter
 - KnowledgeFlow
 - Workbench
 - Simple CLI

You will see the following screen on successful installation.



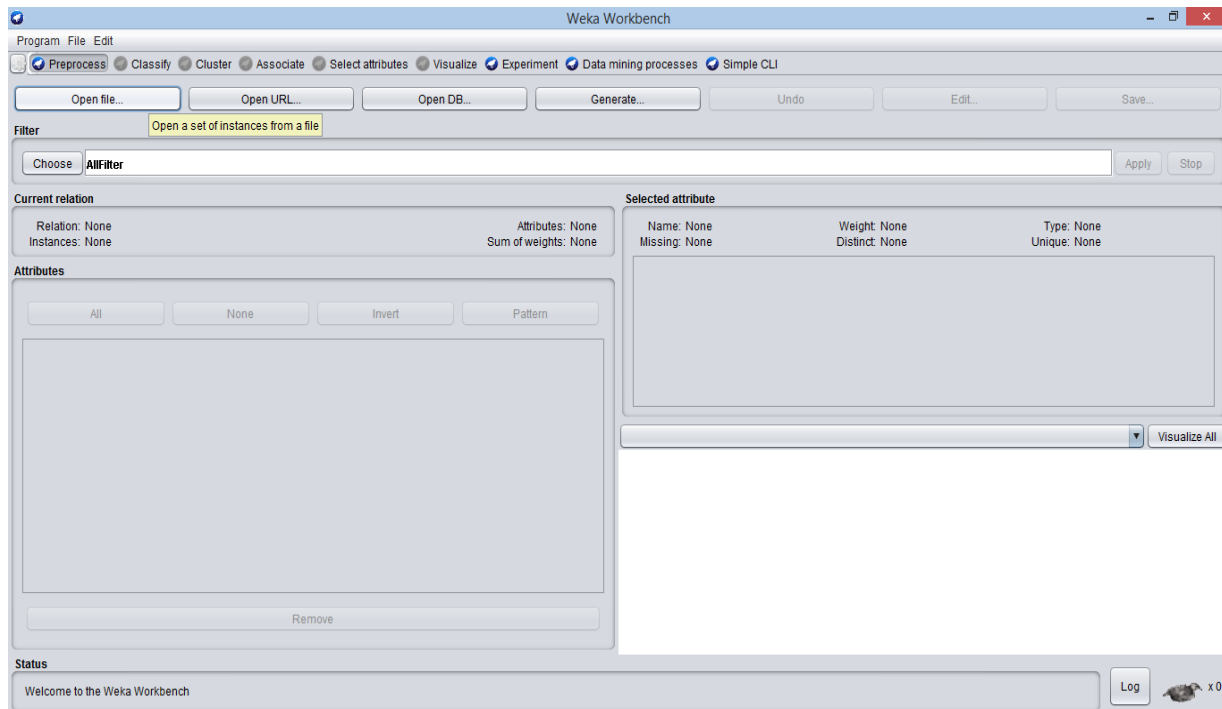
3. Click on the **weka 3.8.4** icon to start Weka.

The WEKA GUI Chooser application will start and you would see the following screen.



The GUI Chooser application allows you to run five different types of applications as listed above. Here I use **Workbench**.

4. When you click on the **Workbench** button in the **Applications** selector, it opens the following screen.



On the top, you will see several tabs as listed below.

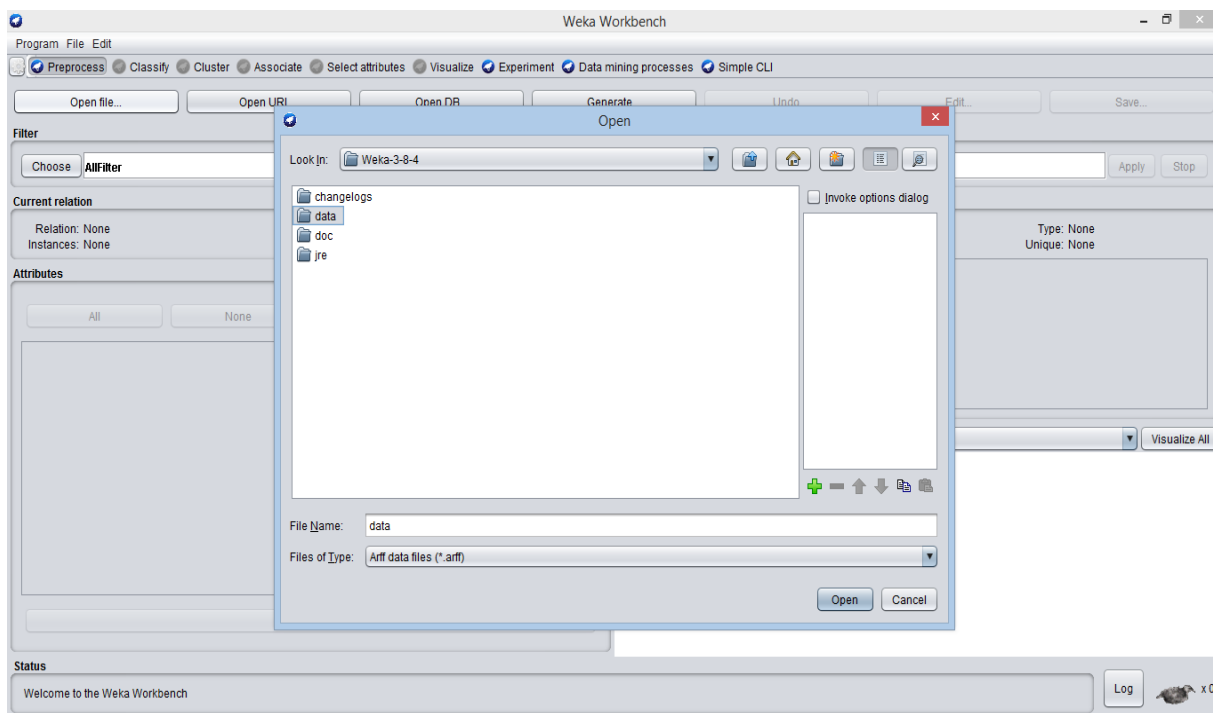
- Preprocess: This allows us to choose the data file.
- Classify: This allows us to apply and experiment with different algorithms on preprocessed data files.
- Cluster: This allows us to apply different clustering tools, which identify clusters within the data file.
- Associate: This allows us to apply association rules, which identify the association within the data.
- Select Attributes: These allow us to see the changes on the inclusion and exclusion of attributes from the experiment.
- Visualize: This allows us to see the possible visualisation produced on the data set in a 2D format, in scatter plot and bar graph output.

Under these tabs, there are several pre-implemented machine learning algorithms.

Data preprocessing is a must. There are three ways to inject the data for preprocessing:

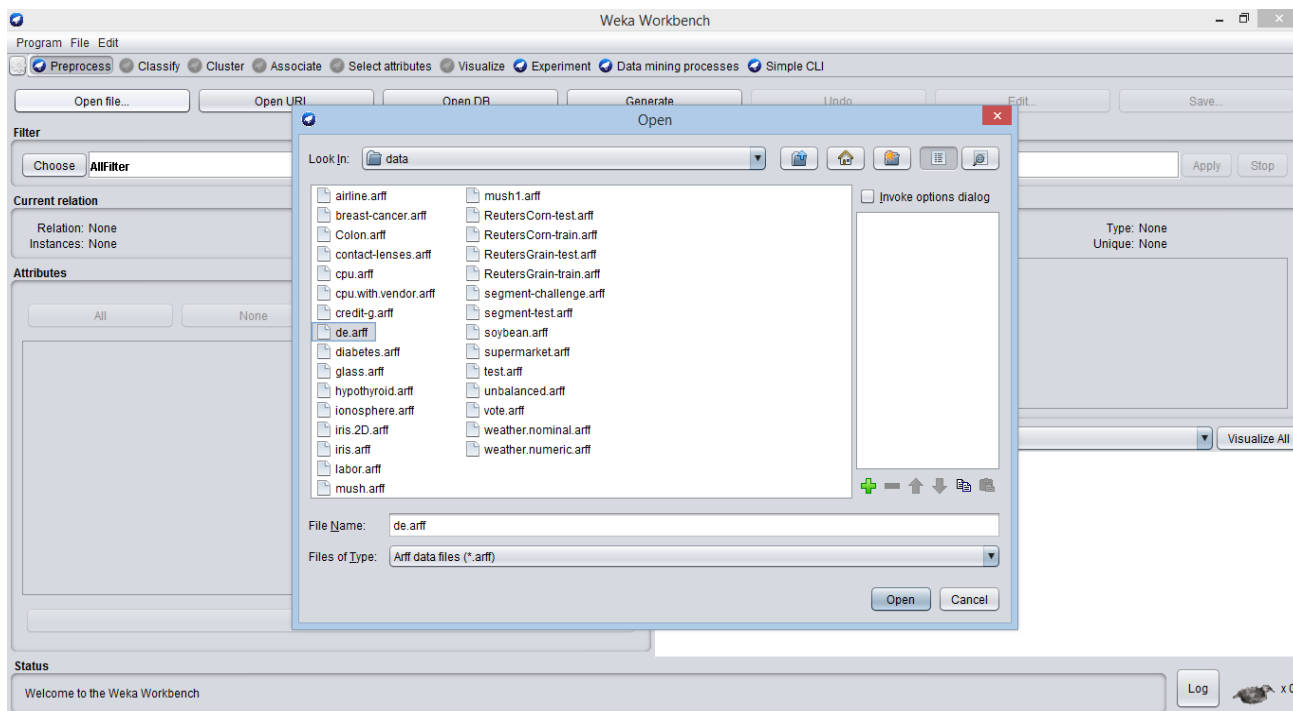
- Open File: enables the user to select the file from the local machine
- Open URL: enables the user to select the data file from different locations
- Open Database: enables users to retrieve a data file from a database source

5. A screen for selecting a file from the local machine to be preprocessed is shown below. After loading the data in Workbench, we can refine the data by selecting different options. We can also select or remove the attributes as per our need and even apply filters on data to refine the result.



Now, navigate to the folder where your data files are stored. WEKA installation comes up with many sample databases for you to experiment. These are available in the **data** folder of the WEKA installation.

Open the mushroom dataset from the data folder as shown below.

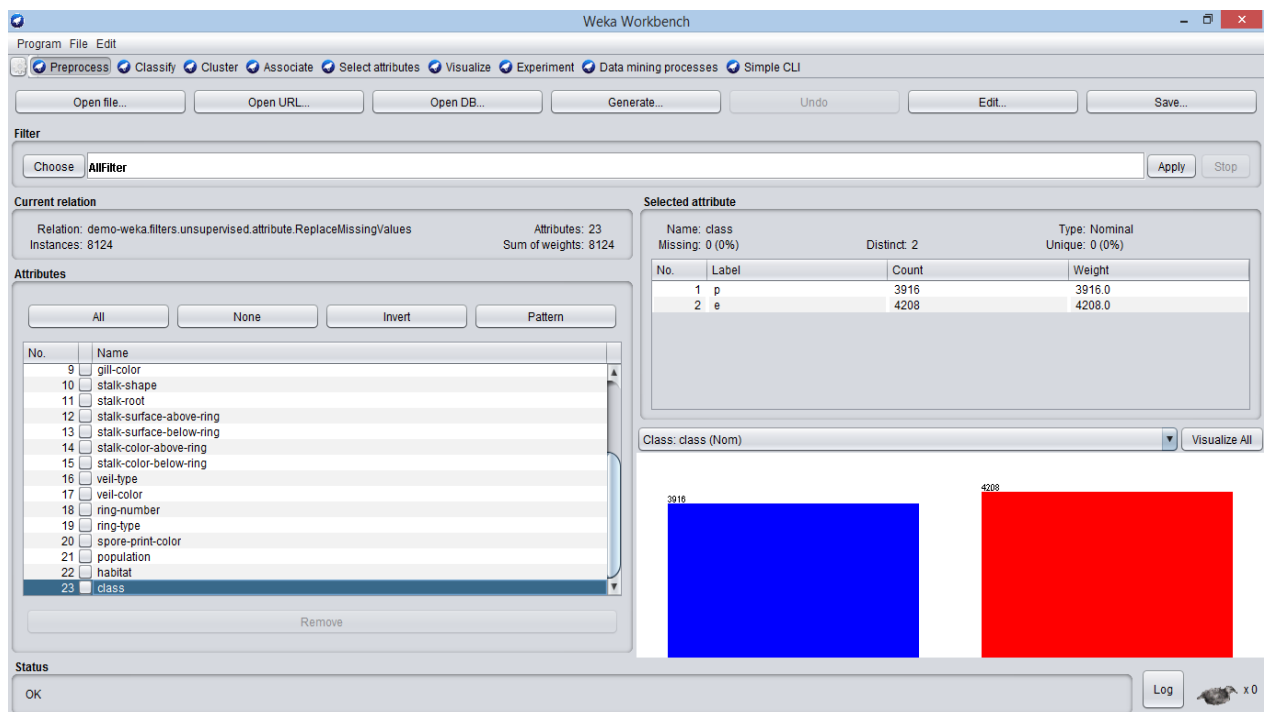


When you open the file, your screen looks like as shown below. This screen tells us several things about the loaded data.

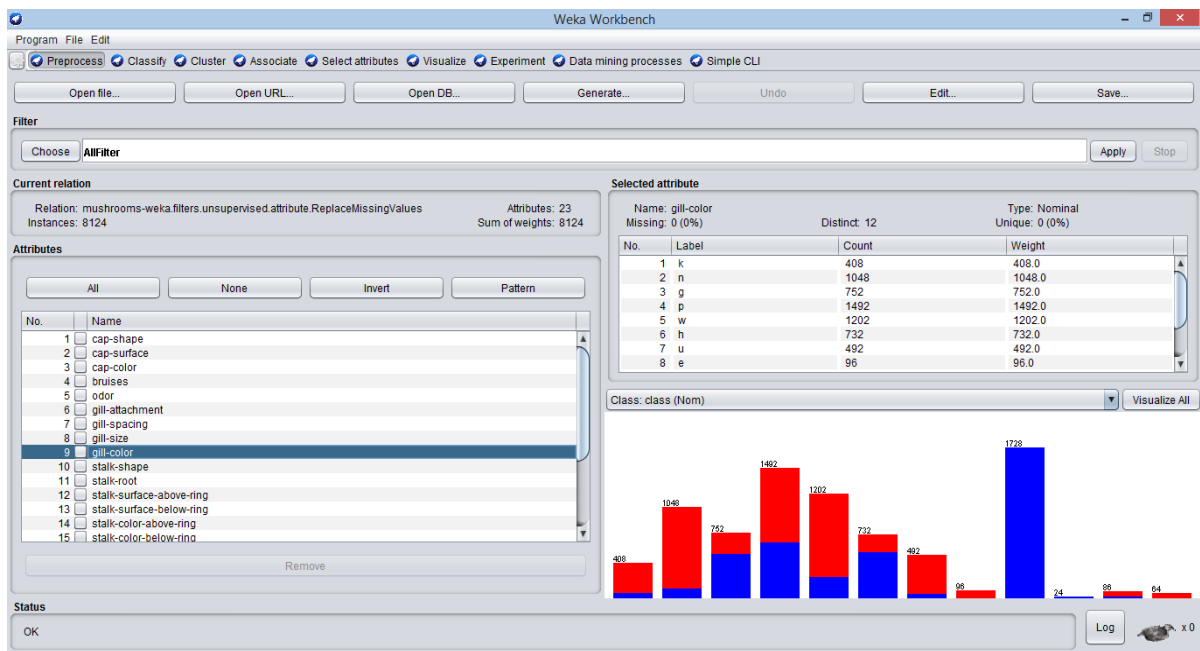
Let us first look at the highlighted **Current relation** sub window. It shows the name of the database that is currently loaded. You can infer two points from this sub window –

- There are 8124 instances - the number of rows in the table.
- The table contains 23 attributes - the fields.

On the left side, notice the **Attributes** sub window that displays the various fields in the database.



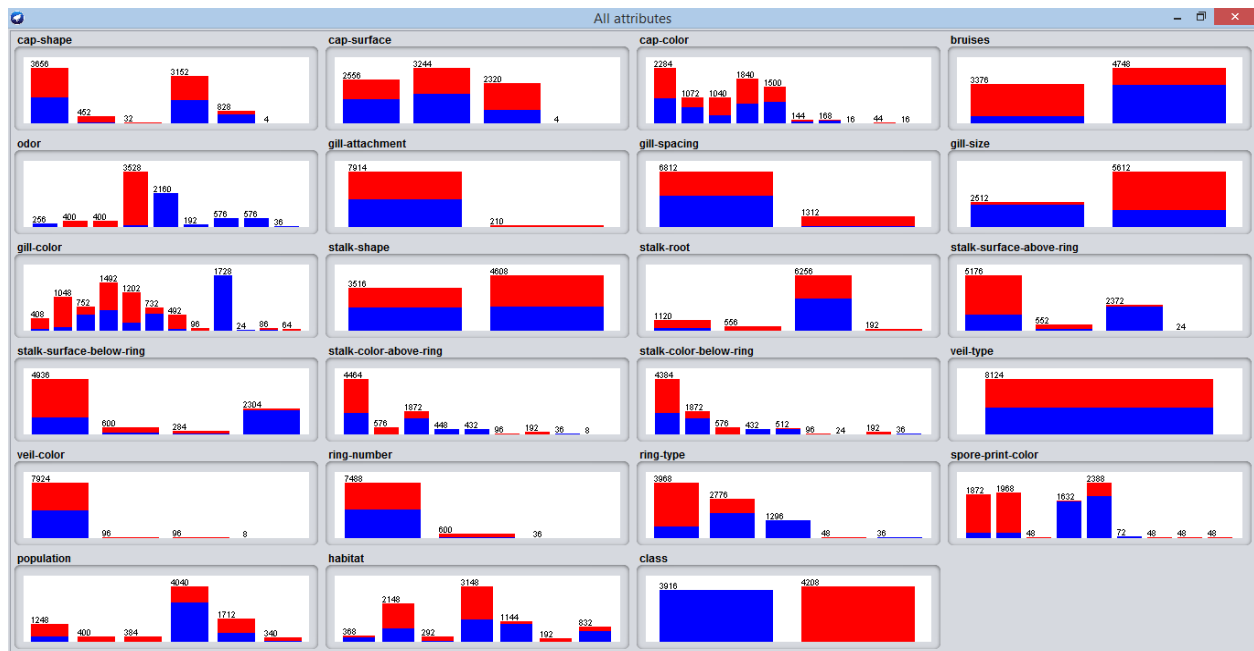
The **mushroom** database contains twenty three fields. When you select an attribute from this list by clicking on it, further details on the attribute itself are displayed on the right hand side. Let us select the **gill-color** attribute first. When you click on it, you would see the following screen.



In the **Selected Attribute** subwindow, you can observe the following –

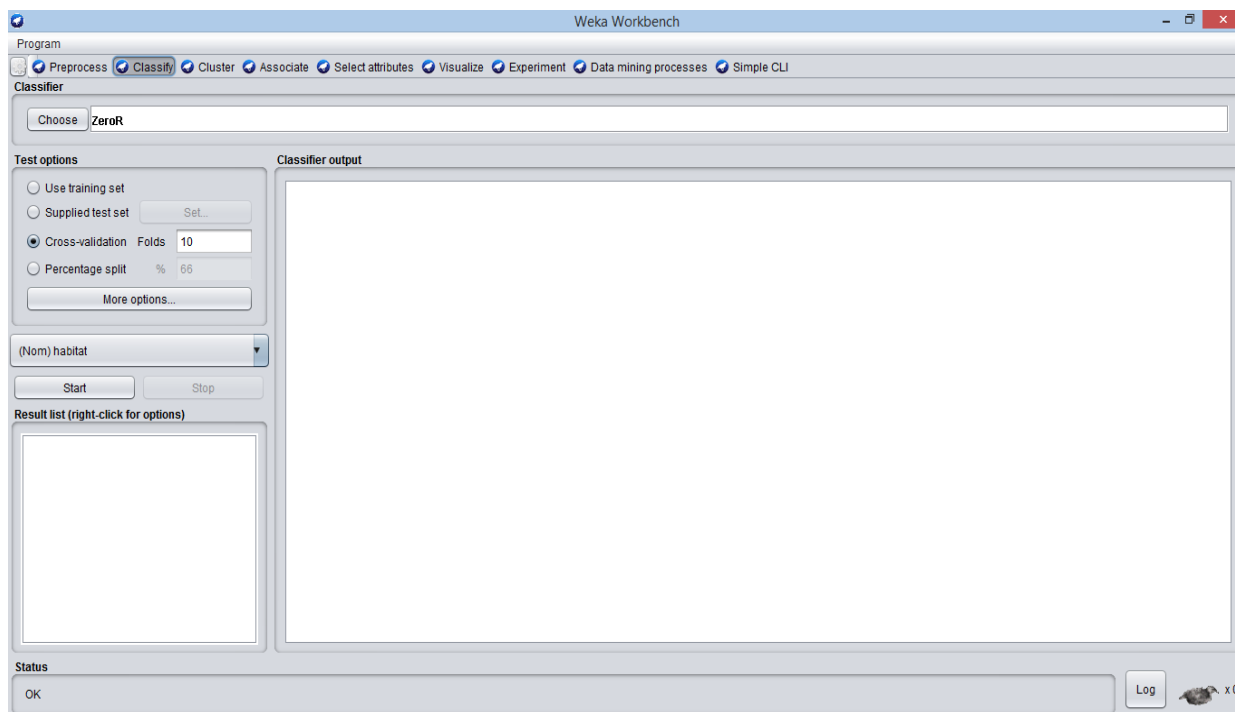
- The name and the type of the attribute are displayed.
- The type for the **gill-color** attribute is **Nominal**.
- The number of **Missing** values is zero.
- There are twelve **Distinct** values with no unique value.
- The table underneath this information shows the nominal values for this field as black =b ,brown= n,buff=b ,chocolate=h ,gray=g, green=r,orange=o,pink=p,red=e,white=w.
- It also shows the count and weight in terms of a percentage for each nominal value.

6. At the bottom of the window, you see the visual representation of the **class** values by clicking on the **Visualize All** button, you will be able to see all features in one single window as shown below.



To predict nominal or numeric quantities, we have classifiers in Weka. Available learning schemes are decision-trees and lists, support vector machines, instance- based classifiers, logistic regression and Bayes' nets. Once the data has been loaded, all the tabs are enabled. Based on the requirements and by trial and error, we can find out the most suitable algorithm to produce an easily understandable representation of data. To classify the data set based on the characteristics of attributes, Weka uses classifiers.

7. Click on the **Classify** tab and click on the Choose button, and you would see the following screen.



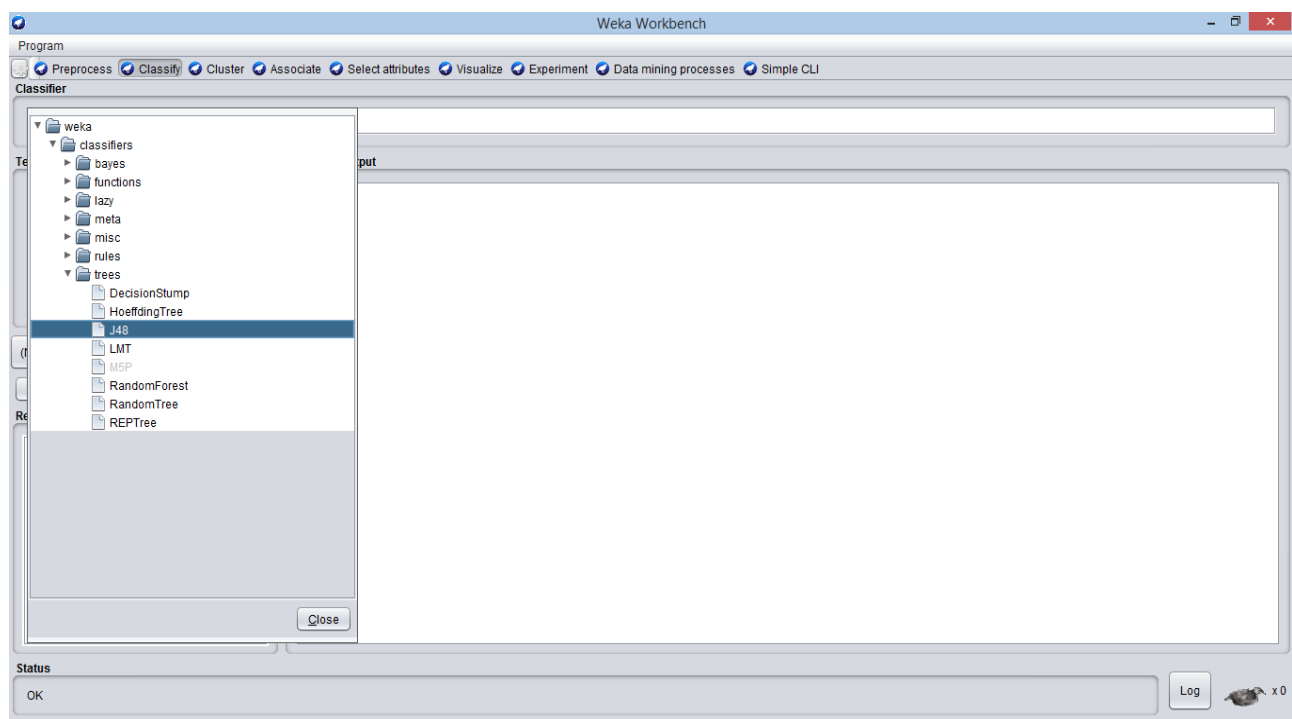
Unless you have your own training set or a client supplied test set, you would use cross-validation or percentage split options. Under cross-validation, you can set the number of folds in which entire data would be split and used during each iteration of training. In the percentage split, you will split the data between training and testing using the set split percentage. Before running any classification algorithm, we need to set test options. Available test options are listed below.

- *Use training set:* Evaluation is based on how well it can predict the class of the instances it was trained on.
- *Supplied test set:* Evaluation is based on how well it can predict the class of a set of instances loaded from a file.
- *Cross-validation:* Evaluation is based on cross-validation by using the number of folds entered in the 'Folds' text field.
- *Split percentage:* Evaluation is based on how well it can predict a certain percentage of the data, held out for testing by using the values entered in the '%' field.

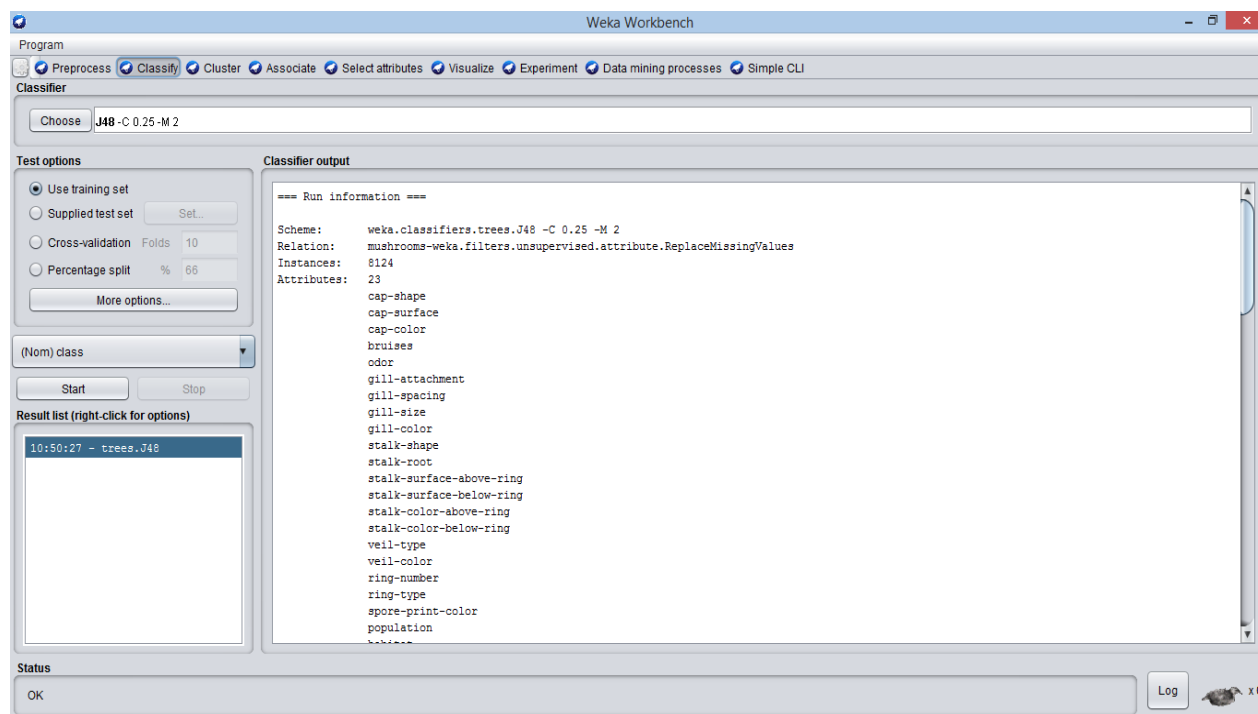
8. Next, you will select the classifier. Click on the Choose button and select the following classifier.

Weka > Classify > trees > J48

Here we choose J48 algorithm for training our dataset and generate decision tree. This is shown in the screenshot below.

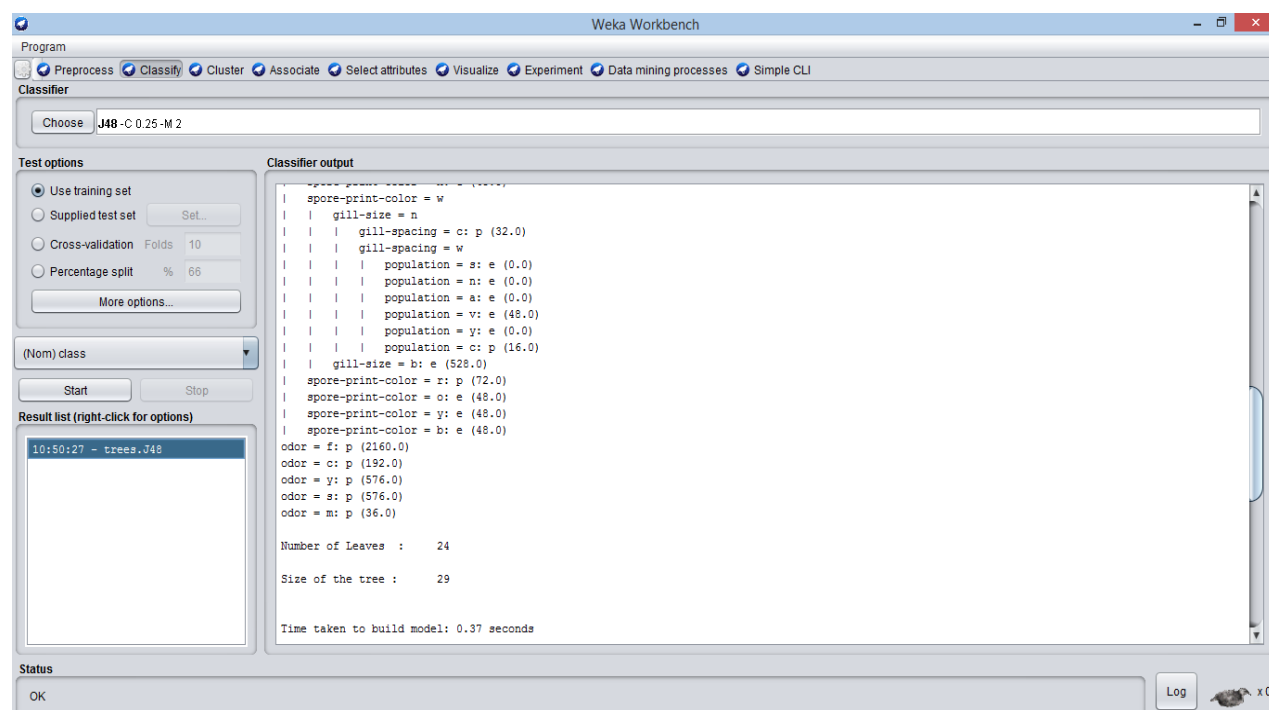


9. Click on the **Start** button to start the classification process. After a while, the classification results would be presented on your screen as shown below.

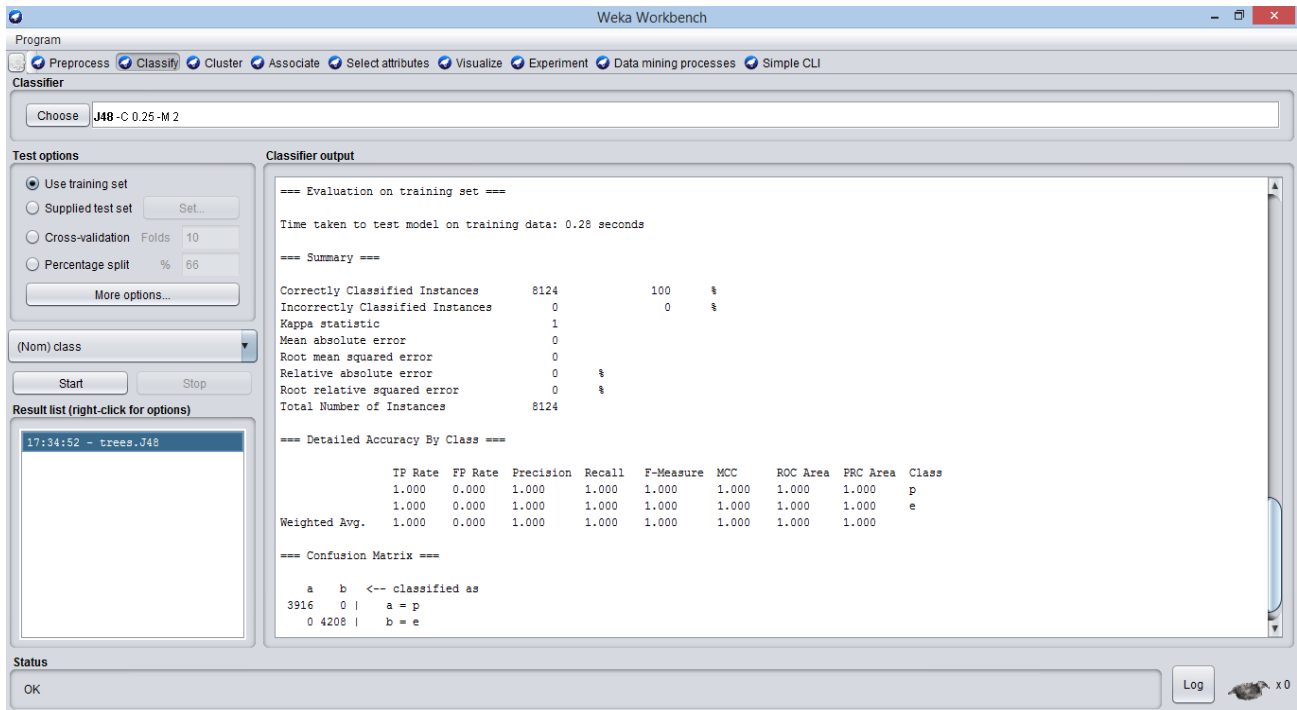


Let us examine the output shown on the right hand side of the screen.

Scheme: The classification algorithm used. **Instances:** Number of data rows in the dataset. **Attributes:** The dataset has 8124 attributes. Full classification of the J48 pruned with the attributes and number of instances.

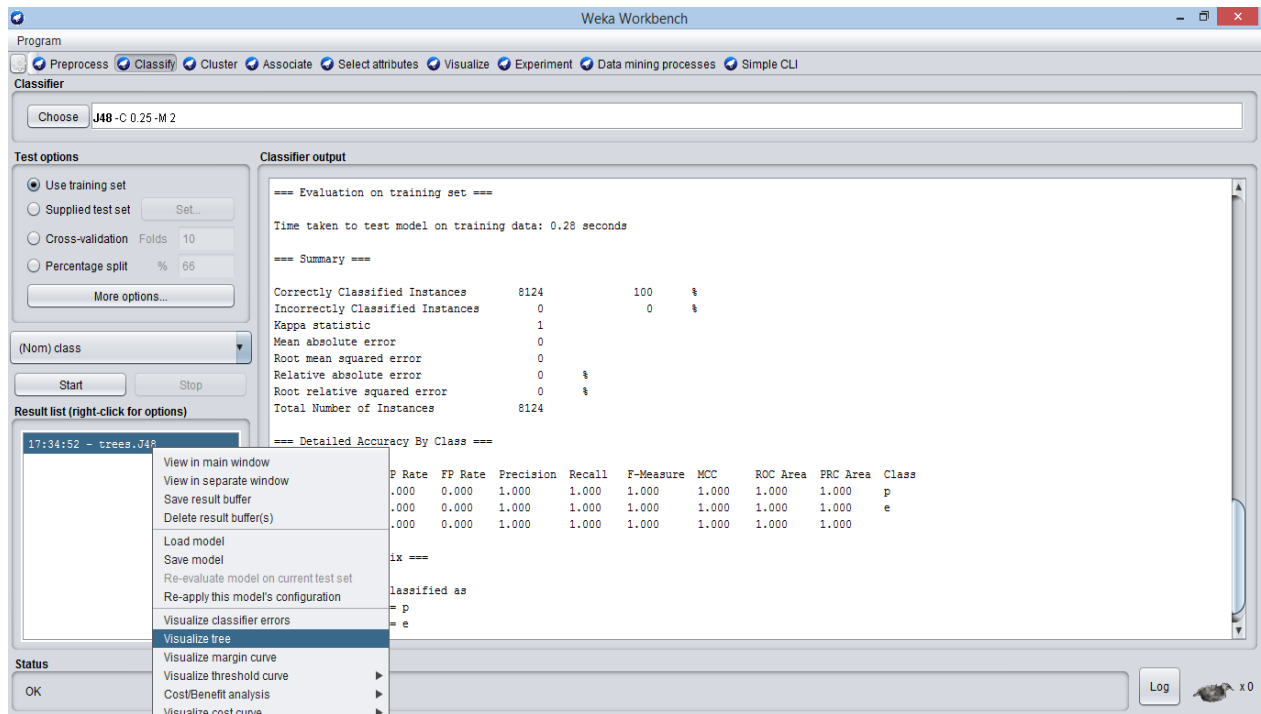


The number of leaves and the size of the tree describes the decision tree. It says the size of the tree is 29 and number of leaves is 24. **Time taken to build the model:** Time for the output.



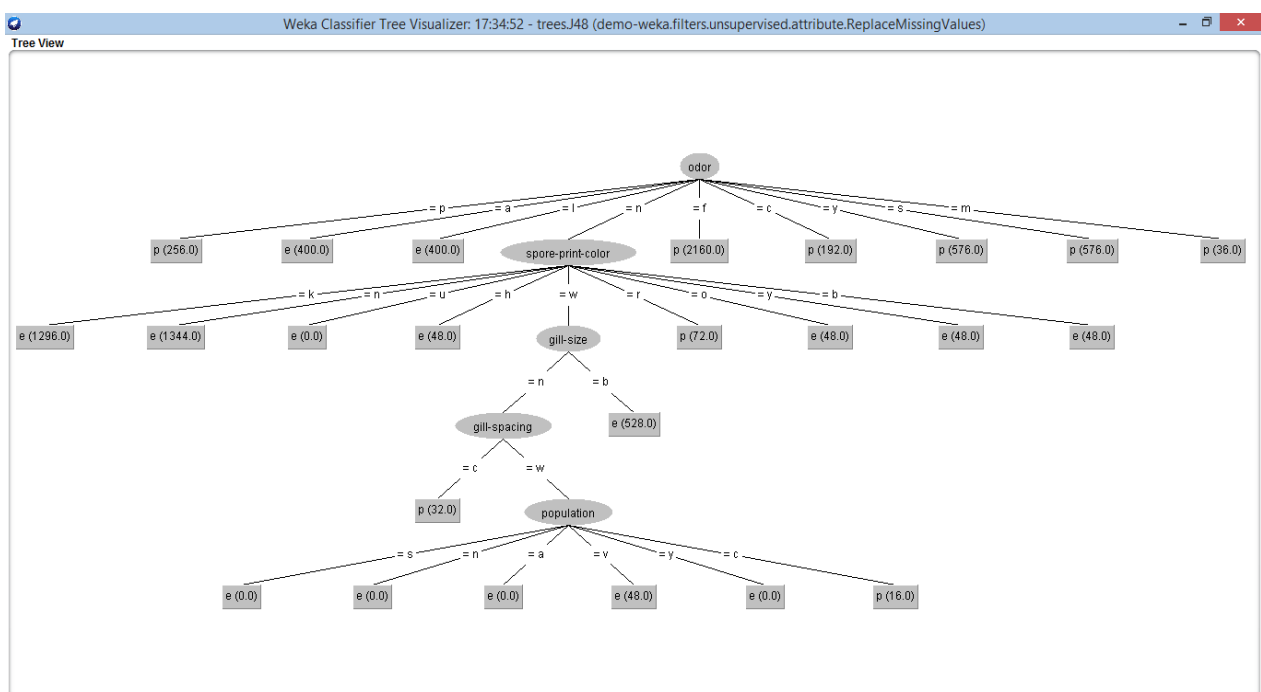
In the summary, it says that the accuracy is 100%, kappa statistics is 1, mean absolute error is 0. In the Confusion Matrix, it says that the correctly classified instances as 3916 that is poisonous and the incorrectly classified instances as 4208 that is edible.

10. To see the visual representation of the results, right click on the result in the **Result list** box. Several options would pop up on the screen as shown below.

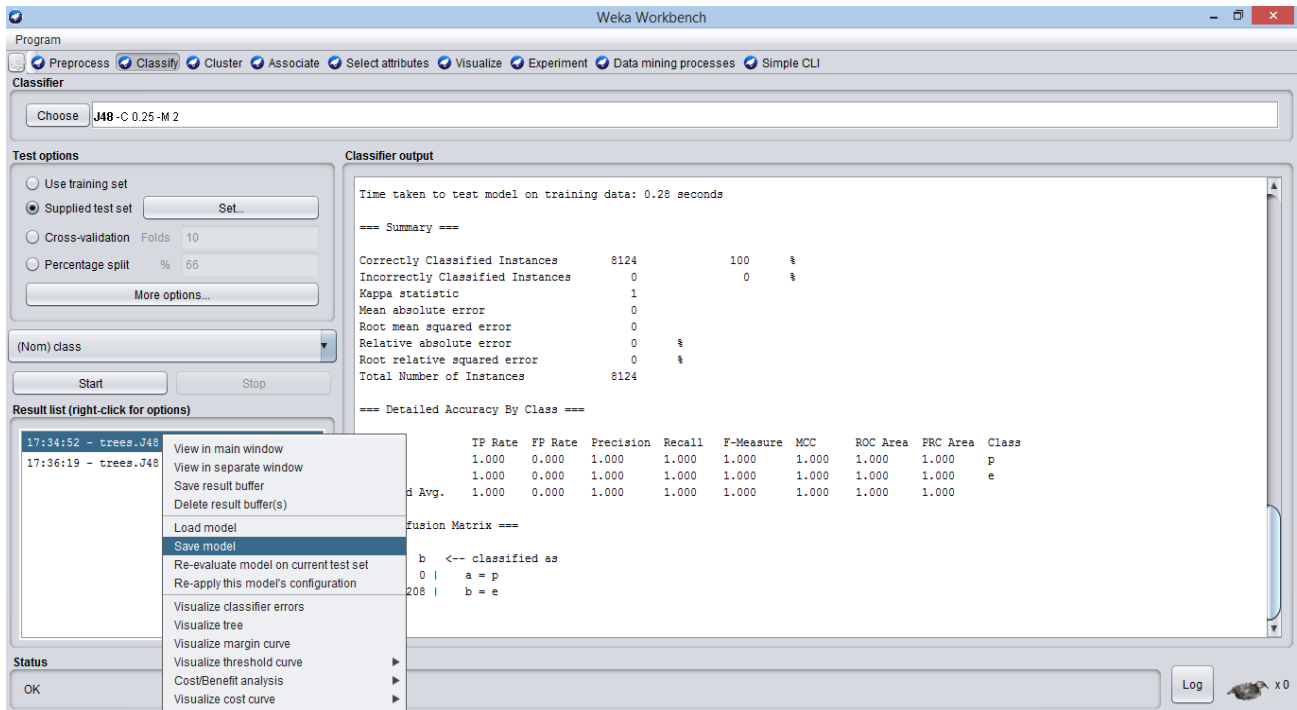


10. Select **Visualize tree** to get a visual representation of the traversal tree as seen in the screenshot below.

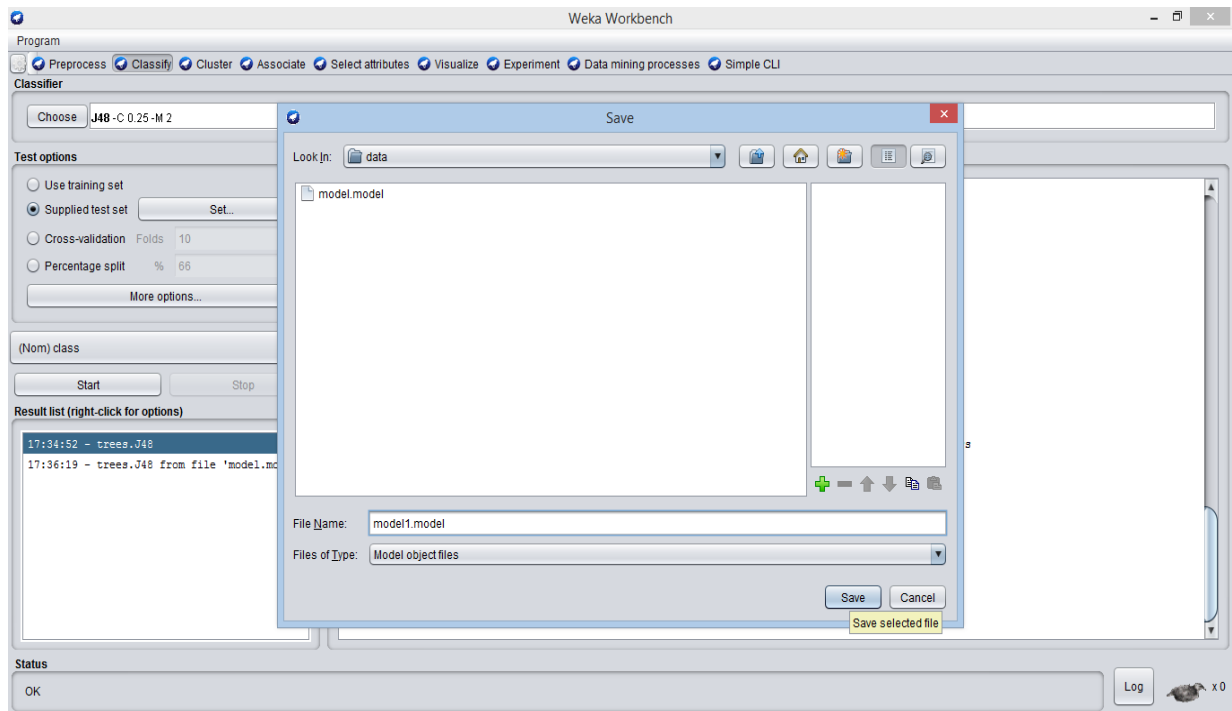
Here odor attribute is the root node, because it have the highest information gain. Information gain measures the amount of information contained in a set of data. It gives the idea of importance of an attribute in a dataset.



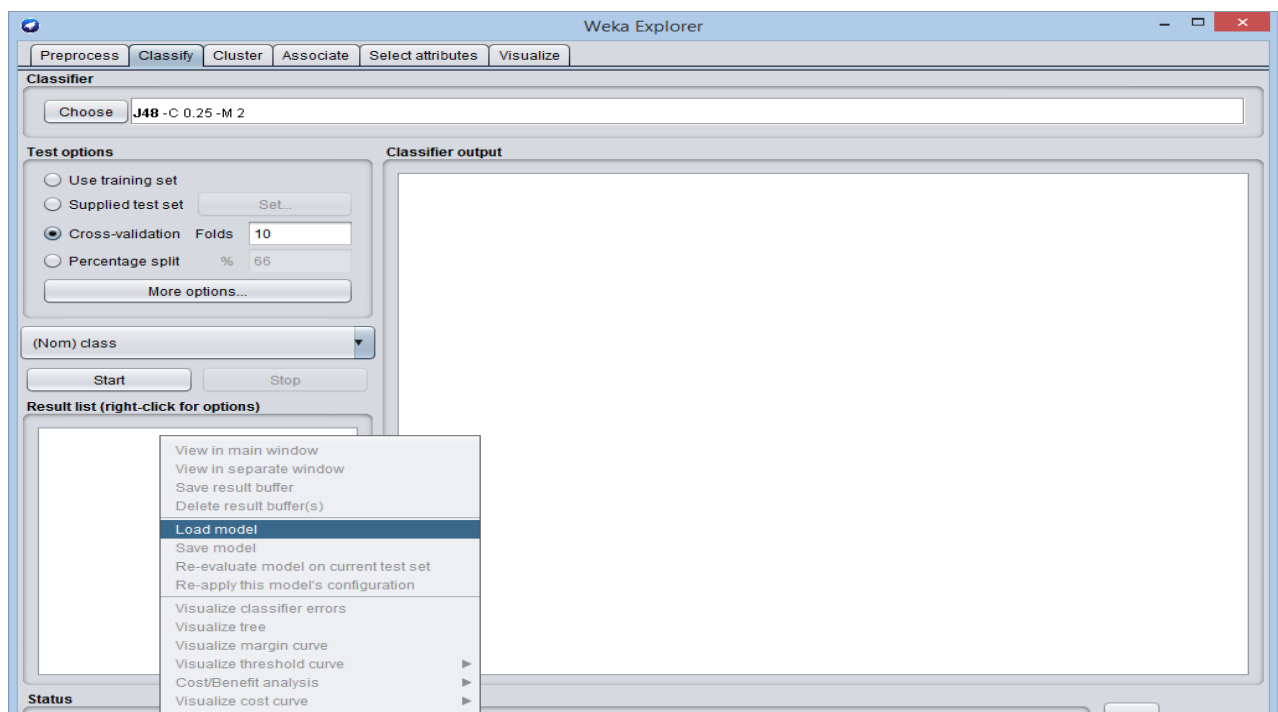
The performance of classification model is evaluated using accuracy, mean absolute error, kappa statistics, confusion matrix etc. Based on the accuracy of the model, we choose J48 for prediction.



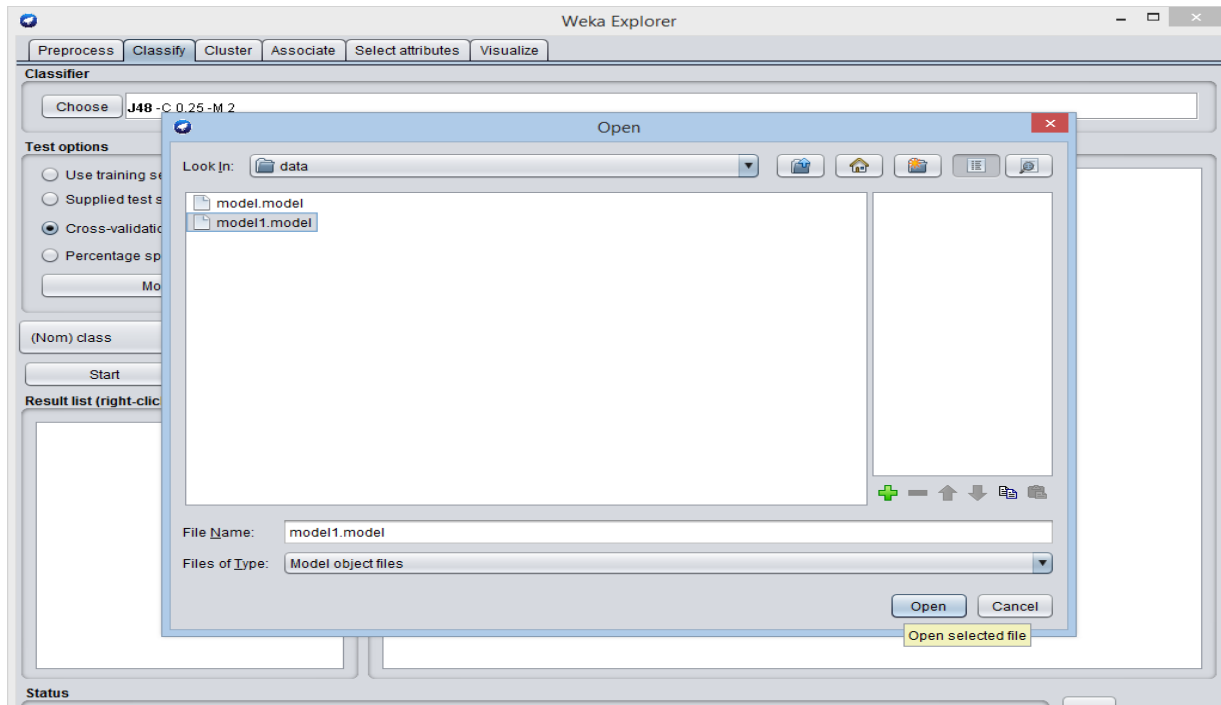
11. This model is then applied to the test dataset for predicting the class of target(unknown) dataset. For that, firstly we have to save the model in the corresponding folder. Below screenshot shows that how to save that model in the corresponding folder.



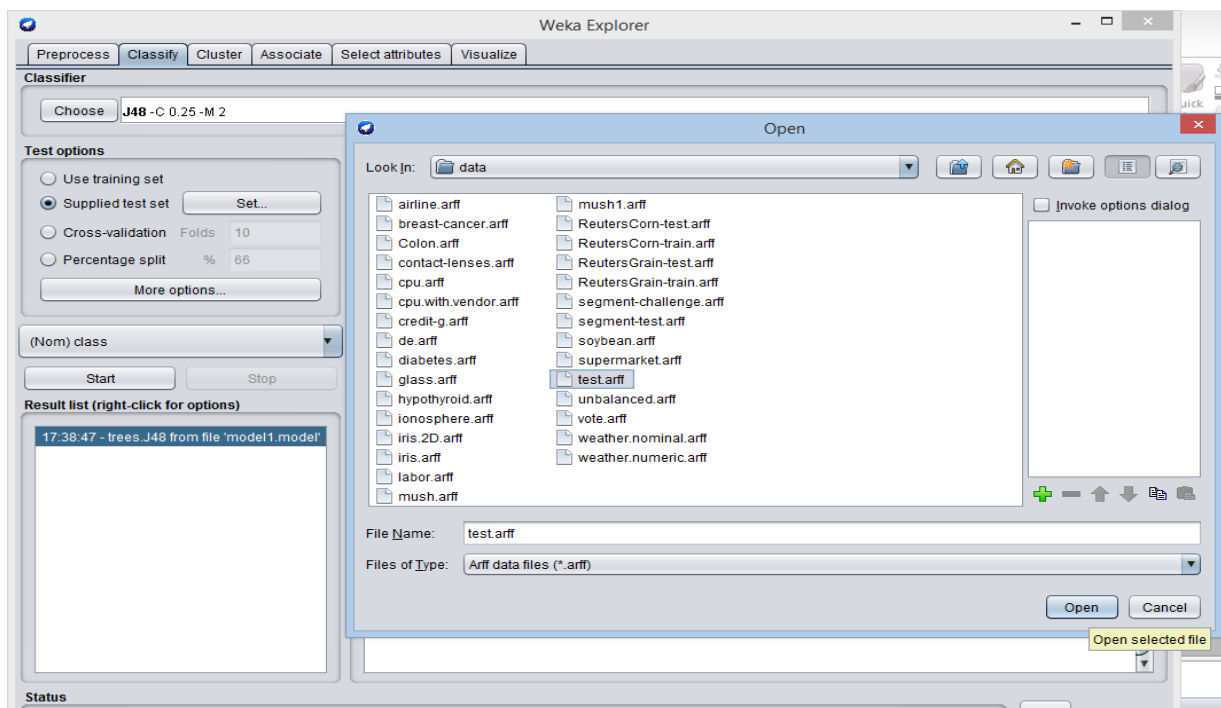
12. After saving the model, again load that model for prediction. For that, right click for options in the Result list and click on Load model option as shown below.



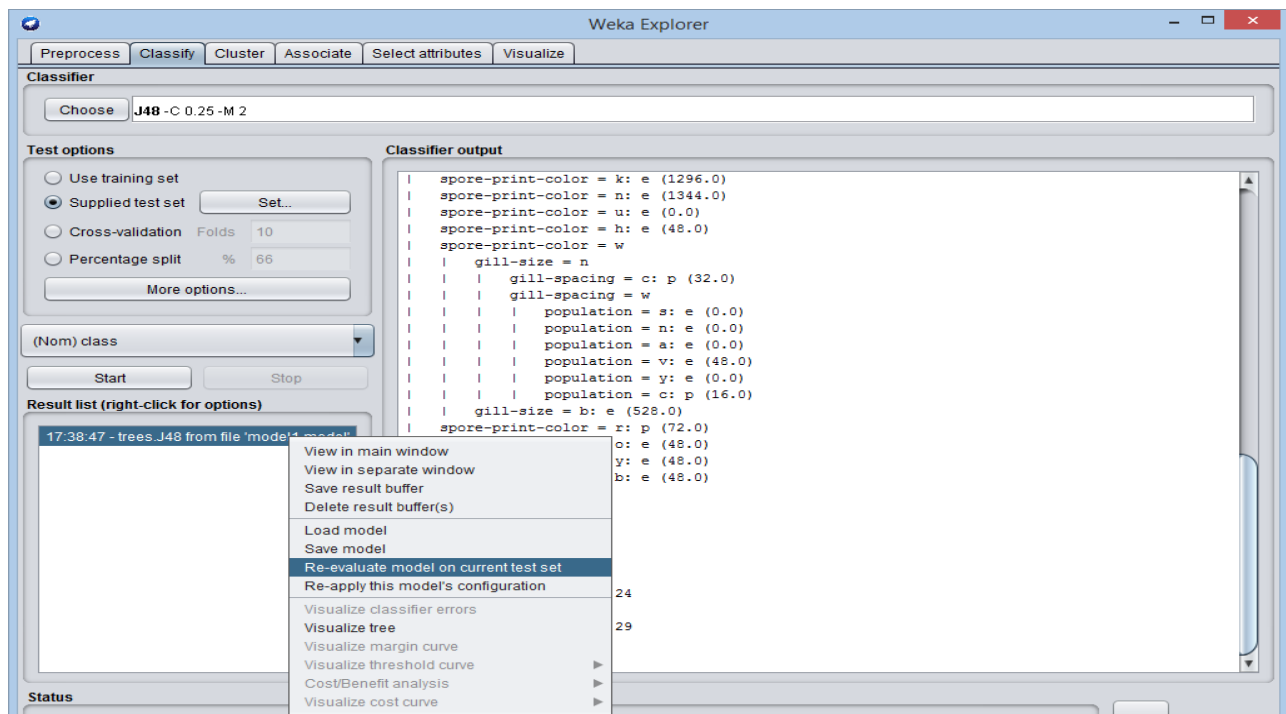
13. Select model1 file which is saved in the corresponding folder and open that file for re-evaluation as shown below.



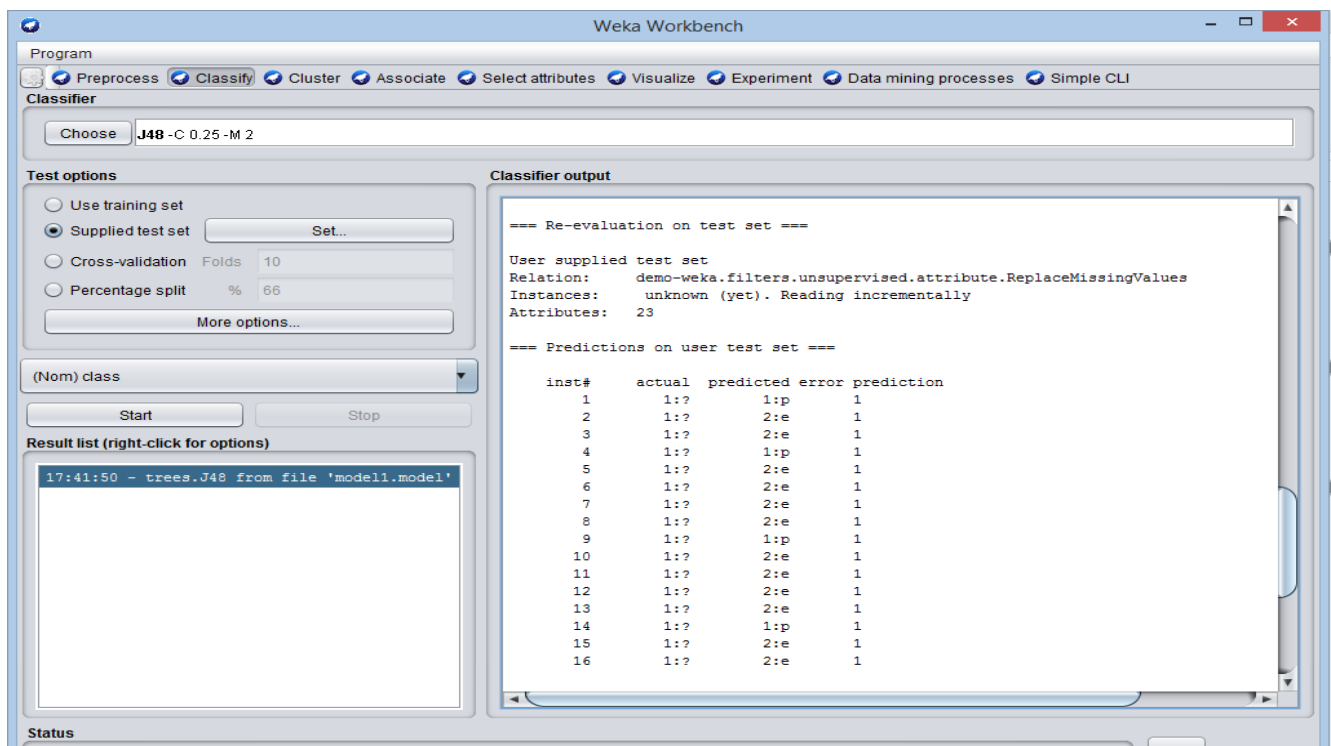
14. Then, select supplied test set option and open the test dataset for prediction, given as input to the Weka classifier, which classifies and gives the result as shown below.



15. Then again right click for options from Result list and choose Re-evaluate the model on current dataset option as shown below.



If the random target class and predicted target class matches, then it gives the prediction margin as +1 else -1.



Here, we got error prediction value as positive value, which means the prediction is correct. Therefore, we can make sure that the model is able to predict whether the mushroom is poisonous or edible.

4. NEW PROBLEM CONCEPT

5. CONCLUSION

There are innumerable benefits of mushroom, but at the same time ingestion of poisonous mushrooms can result fatal. Correct identification of edible mushroom is thus important for consumption. The mushroom data set is obtained from Kaggle repository. WEKA also offers many classification algorithms for decision tree. J48 is one of the popular classification algorithms which outputs a decision tree. Using the Classify tab the user can visualize the decision tree. If the decision tree is too populated, tree pruning can be applied from the Preprocess tab by removing the attributes which are not required and start the classification process again.

The data is trained using J48 algorithm and a model is developed. Given a new set of attributes, the model is able to predict the category of mushroom. The target class of trained data set comes along with the attribute set, whereas the target class of test data is a fictitious one. If the fictitious target matches with the actual target, then the prediction margin is a positive value else negative. If the prediction margin is a positive value, it implies the prediction is correct.

WEKA offers a wide range of sample datasets to apply machine learning algorithms. The users can perform machine learning tasks such as classification, regression, attribute selection, association on these sample datasets, and can also learn the tool using them. WEKA explorer/workbench is used for

performing several functions, starting from preprocessing. Preprocessing takes input as a .arff file, processes the input, and gives an output that can be used by other computer programs. In WEKA the output of preprocessing gives the attributes present in the dataset which can be further used for statistical analysis and comparison with class labels.

6. REFERENCES

- [https://www.academia.edu/34903775/Prediction of Edible and Poisonous Mushrooms using Decision tree](https://www.academia.edu/34903775/Prediction_of_Edible_and_Poisonous_Mushrooms_using_Decision_tree)
- <https://www.opensourceforu.com/2017/01/an-introduction-to-weka/>
- <https://arxiv.org/ftp/arxiv/papers/1310/1310.4647.pdf>
- <https://www.softwaretestinghelp.com/weka-datasets/>
- <http://www.journalglobal.in/uploads/paper/35.pdf>
- <https://ieeexplore.ieee.org/document/8350746>
- https://www.researchgate.net/publication/282740534_Mushroom_classification_using_data_mining_techniques
- <https://www.youtube.com/watch?reload=9&v=m7kpIBGEdkI>
- <https://www.youtube.com/watch?v=ydEbw2z8xaw>
- https://www.youtube.com/watch?v=gEGJvv5_pRw