

# ***Artificial Intelligence***

Open source analysis

Weka

Prof. Eunok Paek  
T.A. Seunghyuk Choi

# ***What is Weka ?***

---



Weka is a collection of machine learning algorithms for data mining tasks.

Weka contains lots of methods:

- 1) Data pre-processing
- 2) Classification
- 3) Regression
- 4) Clustering
- 5) Association Rules
- 6) Visualization

Weka is open source software issued under the GNU General Public License.

# How to Install ?



Machine Learning Group at the University of Waikato

[Project](#) [Software](#) [Book](#) [Publications](#) [People](#) [Related](#)

## Weka 3: Data Mining Software in Java

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

Found only on the islands of New Zealand, the Weka is a flightless bird with an inquisitive nature. The name is pronounced like **this**, and the bird sounds like **this**.

Weka is open source software issued under the **GNU General Public License**.

We have put together several free online courses that teach machine learning and data mining using Weka. Check out the **website for the courses** for details on when and how to enrol. The videos for the courses are available **on Youtube**.

Yes, it is possible to apply Weka to **big data**!

### Getting started

- [Requirements](#)
- [Download](#)
- [Documentation](#)
- [FAQ](#)
- [Getting Help](#)

### Further information

- [Citing Weka](#)
- [Datasets](#)
- [Related Projects](#)
- [Miscellaneous Code](#)
- [Other Literature](#)

### Developers

- [Development](#)
- [History](#)
- [Subversion](#)
- [Contributors](#)
- [Commercial licenses](#)

## 1. Go to "Download."

### • Snapshots

Every night a snapshot of the Subversion repository is taken, compiled and put together in ZIP files. For those who want to have the latest bugfixes, they can download these snapshots [here](#).

### • Stable version

Weka 3.8 is the latest stable version of Weka. This branch of Weka receives bug fixes only, although new features may become available in packages. There are different options for downloading and installing it on your system:

#### ◦ Windows

Click [here](#) to download a self-extracting executable for 64-bit Windows that includes Oracle's 64-bit Java VM 1.8 (weka-3-8-1jre-x64.exe; 112.0 MB)

Click [here](#) to download a self-extracting executable for 64-bit Windows without a Java VM (weka-3-8-1-x64.exe; 50.6 MB)

Click [here](#) to download a self-extracting executable for 32-bit Windows that includes Oracle's 32-bit Java VM 1.8 (weka-3-8-1jre.exe; 106.4 MB)

Click [here](#) to download a self-extracting executable for 32-bit Windows without a Java VM (weka-3-8-1.exe; 50.6 MB)

These executables will install Weka in your Program Menu. Download the version without the Java VM if you already have Java 1.7 (or later) on your system.

#### ◦ Mac OS X

Click [here](#) to download a disk image for OS X that contains a Mac application including Oracle's Java 1.8 JVM (weka-3-8-1-oracle-jvm.dmg; 123.8 MB)

#### ◦ Other platforms (Linux, etc.)

Click [here](#) to download a zip archive containing Weka (weka-3-8-1.zip; 50.9 MB)

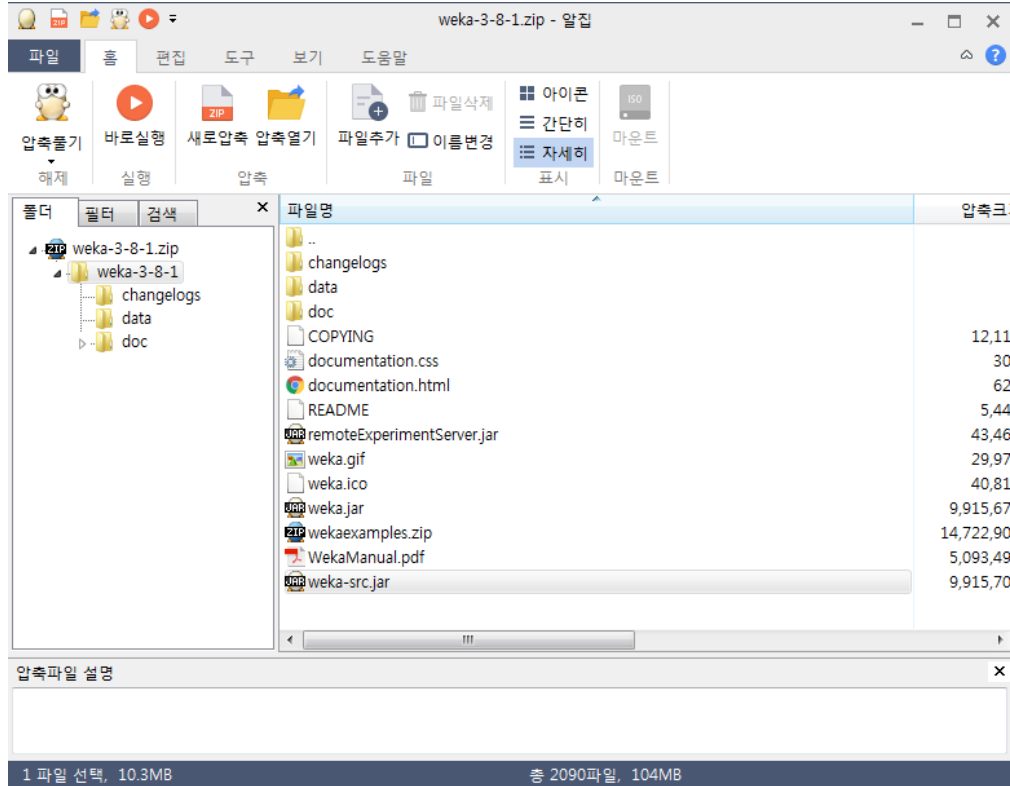
First unzip the zip file. This will create a new directory called weka-3-8-1. To run Weka, change into that directory and type

```
java -jar weka.jar
```

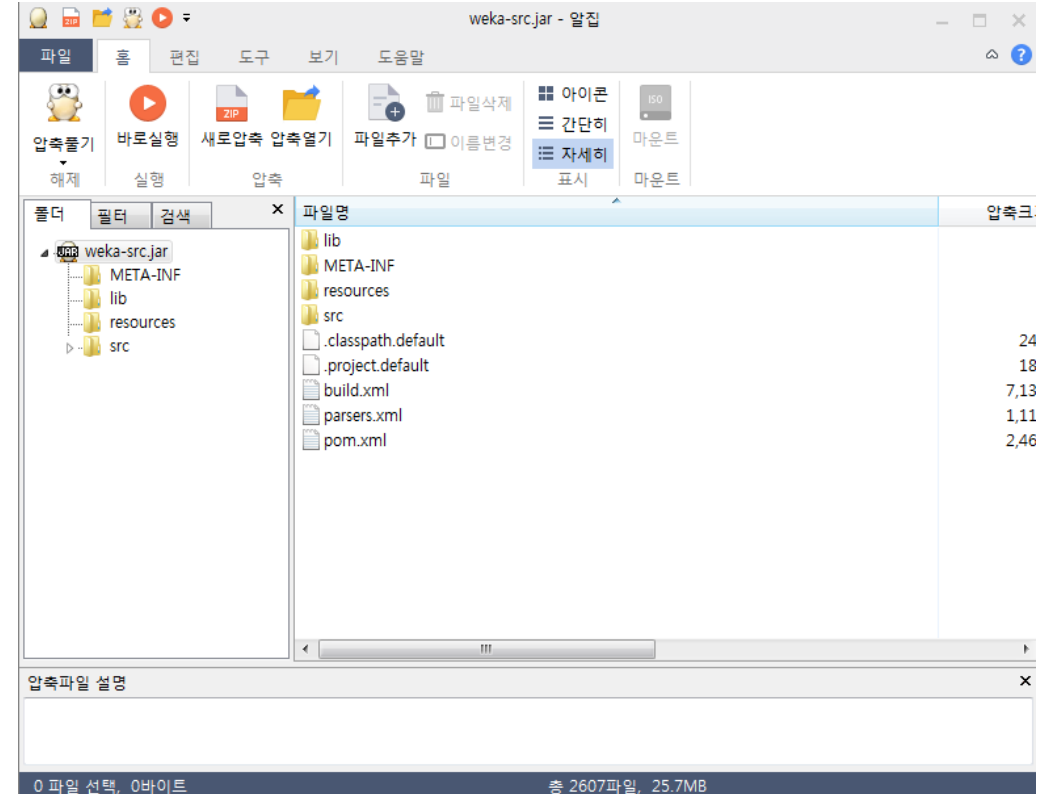
Note that Java needs to be installed on your system for this to work. Also note, that using `-jar` will override your current CLASSPATH variable and only use the `weka.jar`.

## 2. Download weka-3-8-1.zip.

# How to Install ?



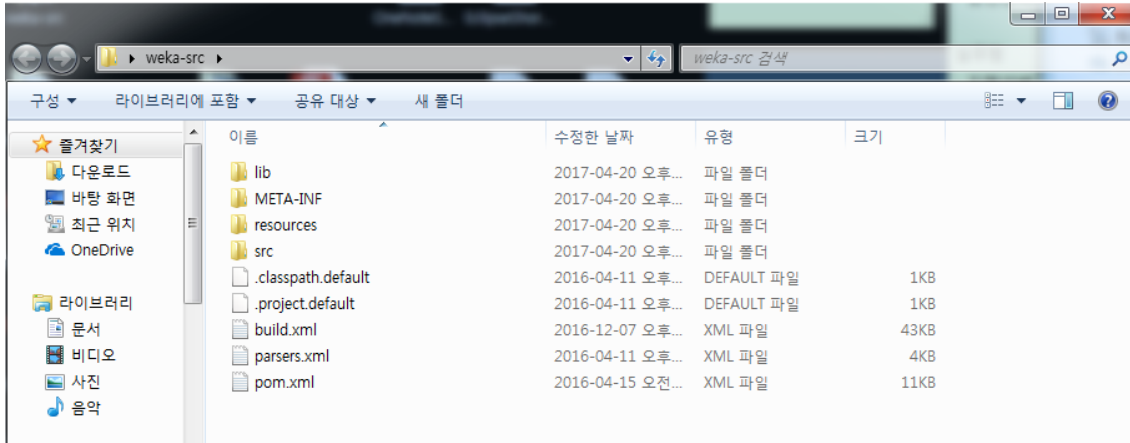
3. Extract weka-src.jar from weka-3-8-1.zip.



4. Extract all from weka-src.jar

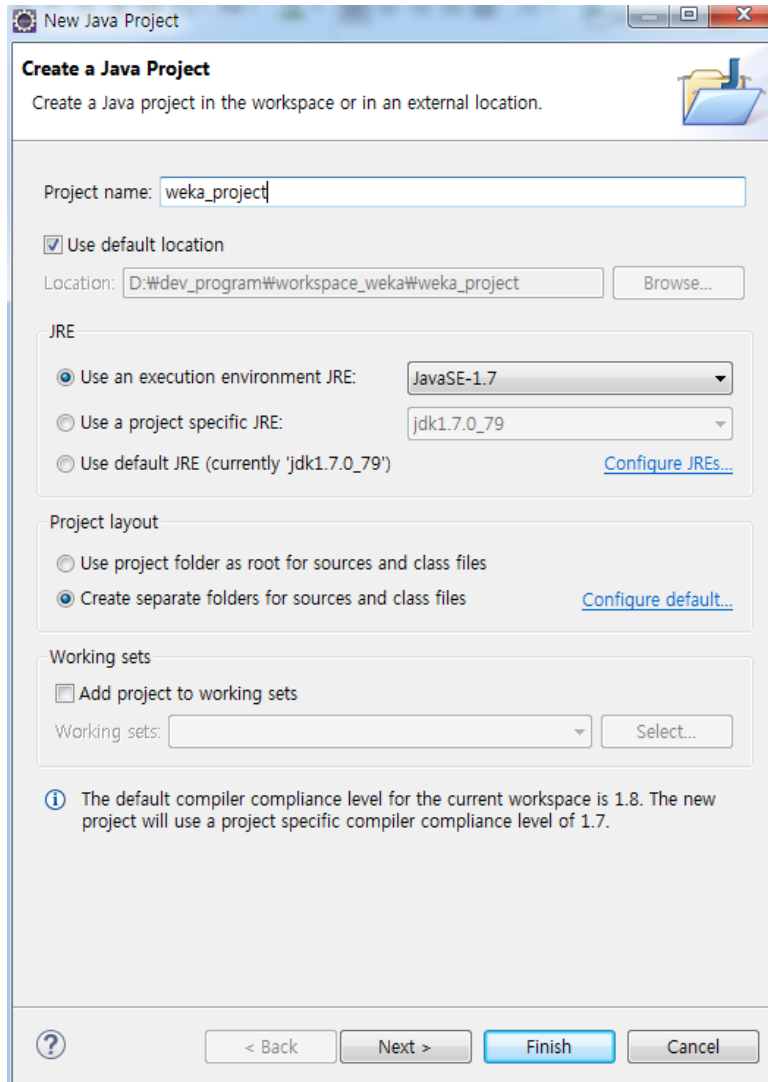
# How to Install ?

---

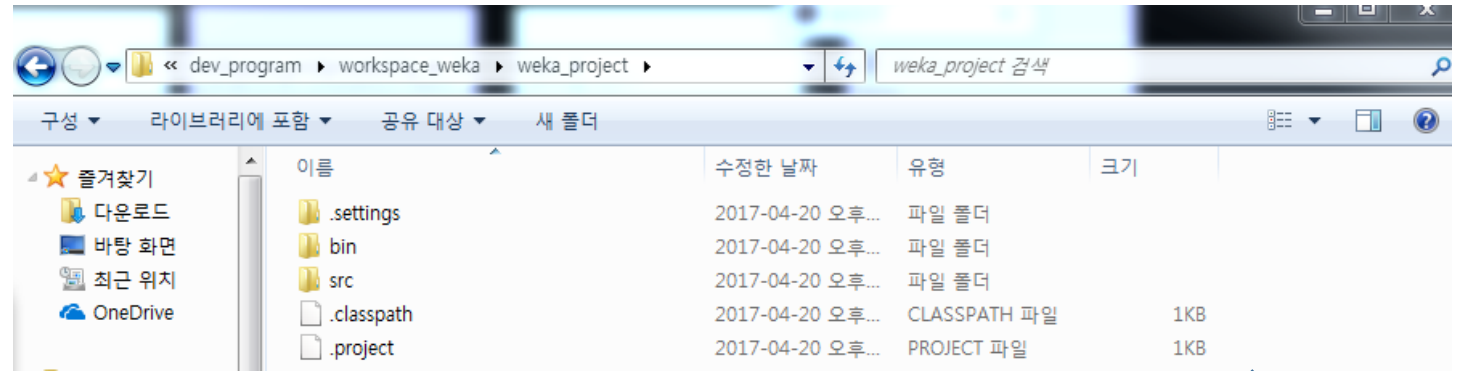


Now you got the above weka-src.

# How to Install ?

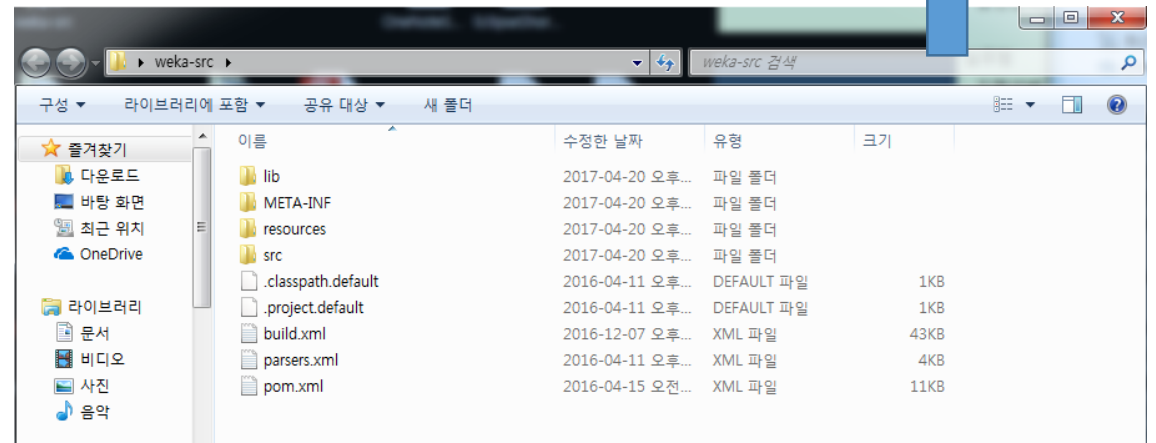


5. Open Eclipse and make a weka project.

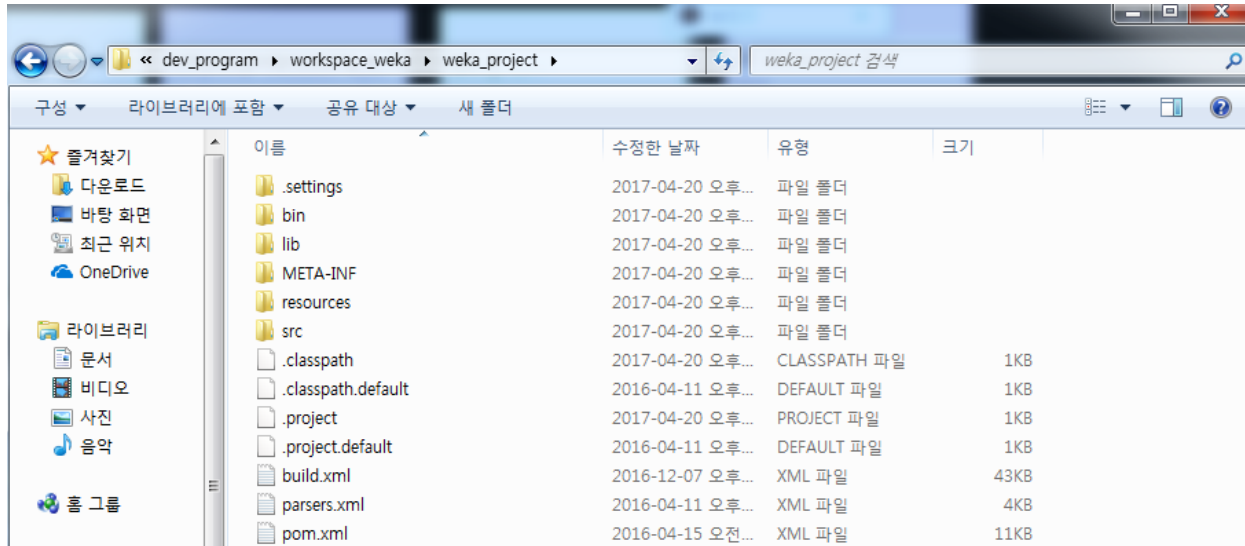


6. Go to the weka project folder and paste the weka-src into the weka project.

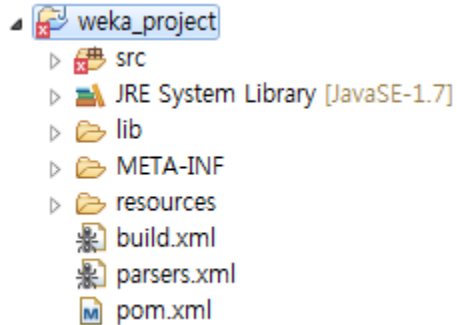
Paste all!



# How to Install ?

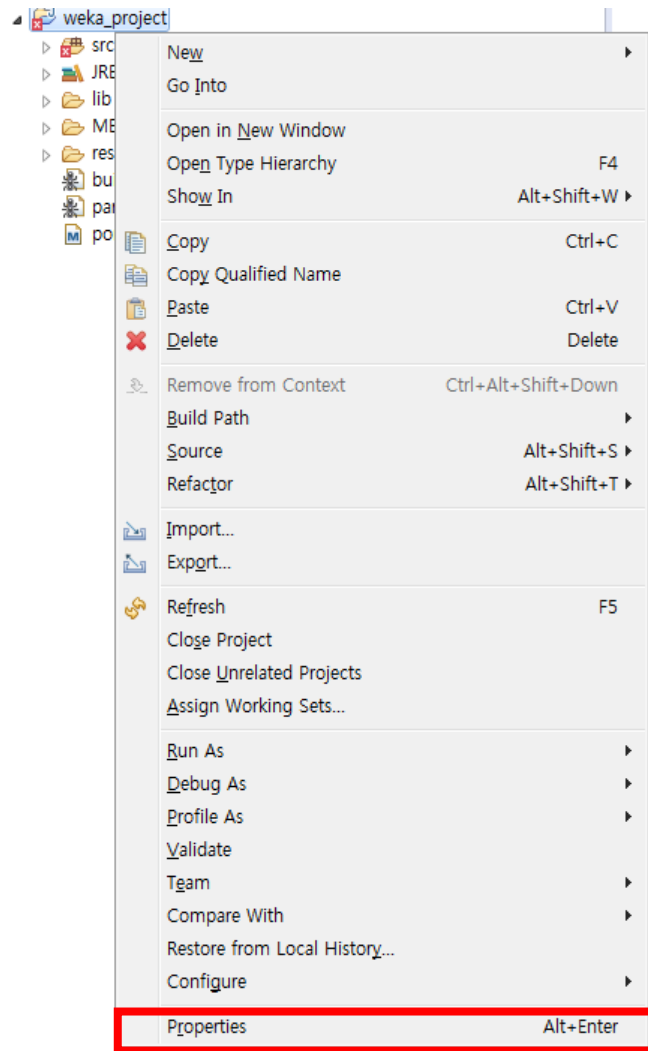


Now you got the weka project folder!

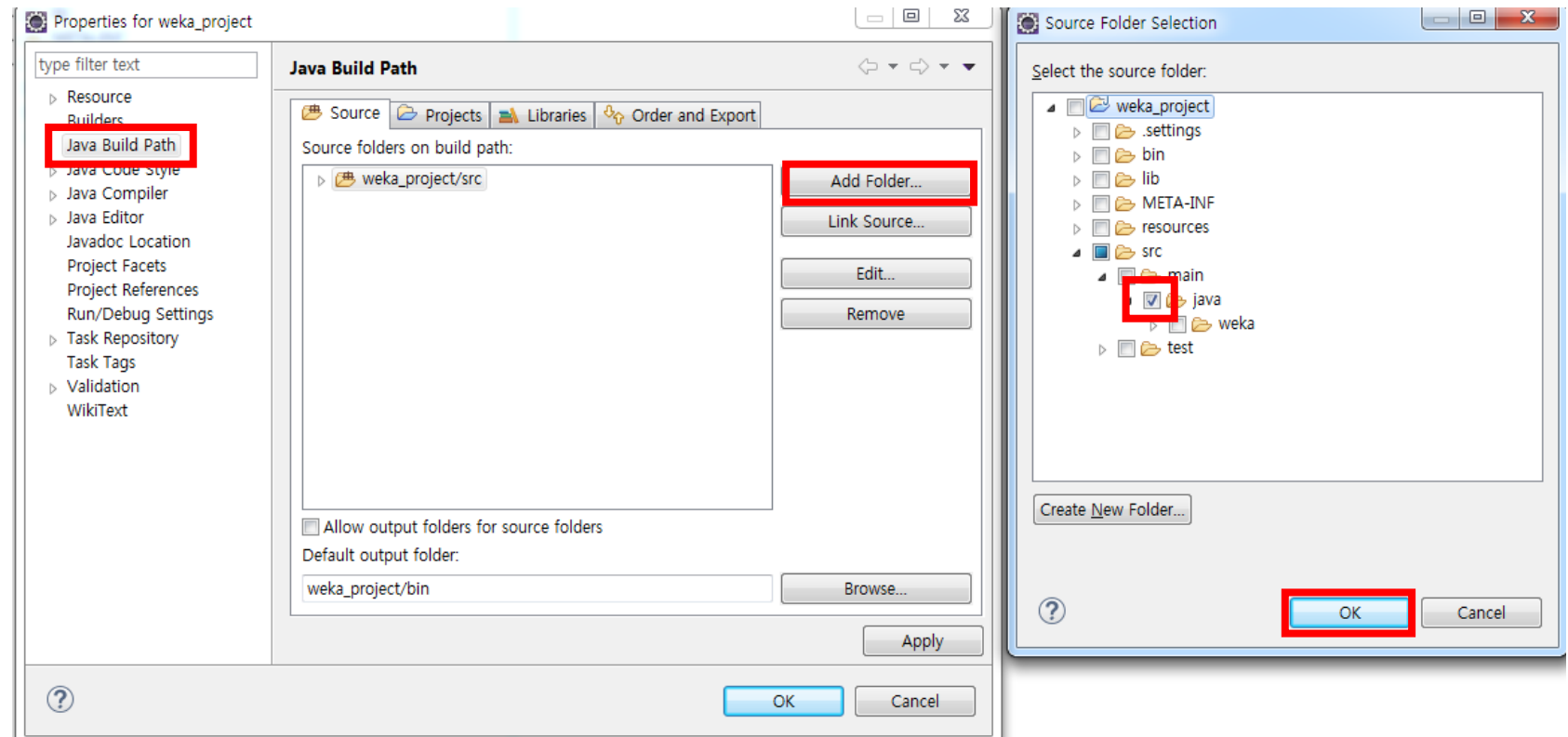


However, your project is something wrong with a path settings...

# How to Install ?



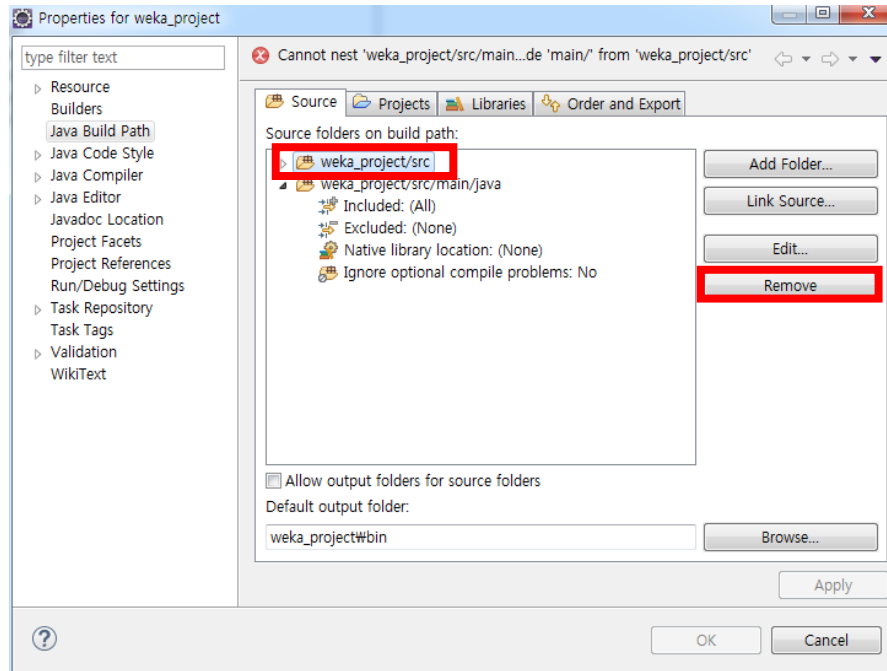
7. Go to "Properties."



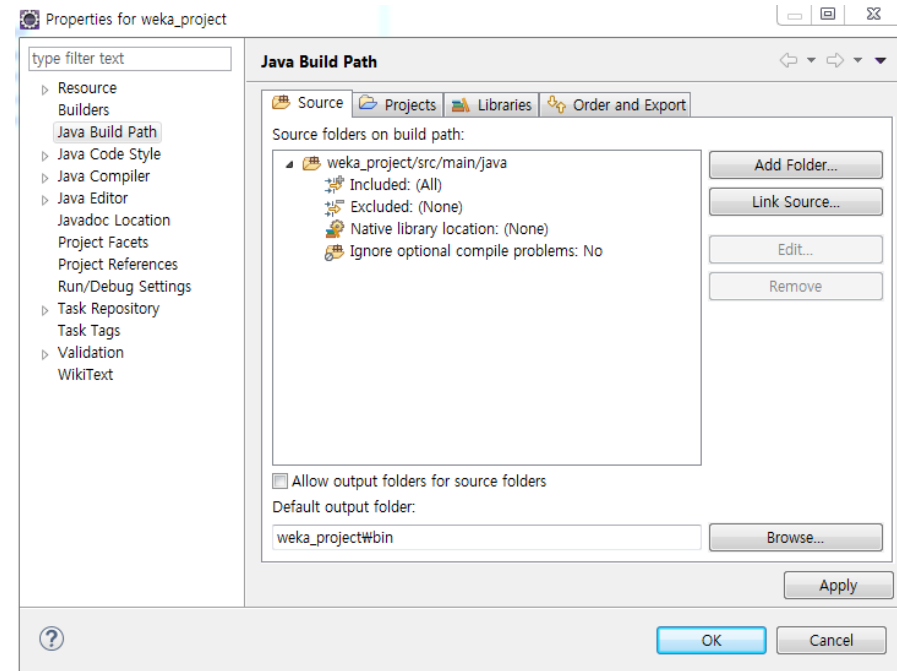
8. Go to "Java Build Path," and then click "Add Folder." check "java" and click "OK."



# How to Install ?

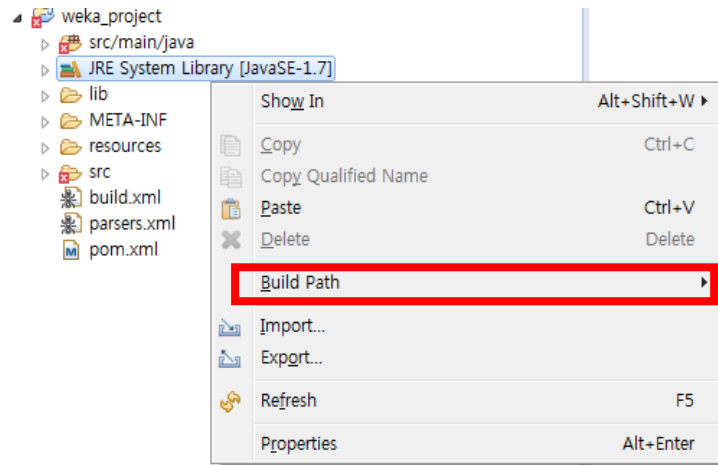


9. Click weka\_project/src, and remove it.

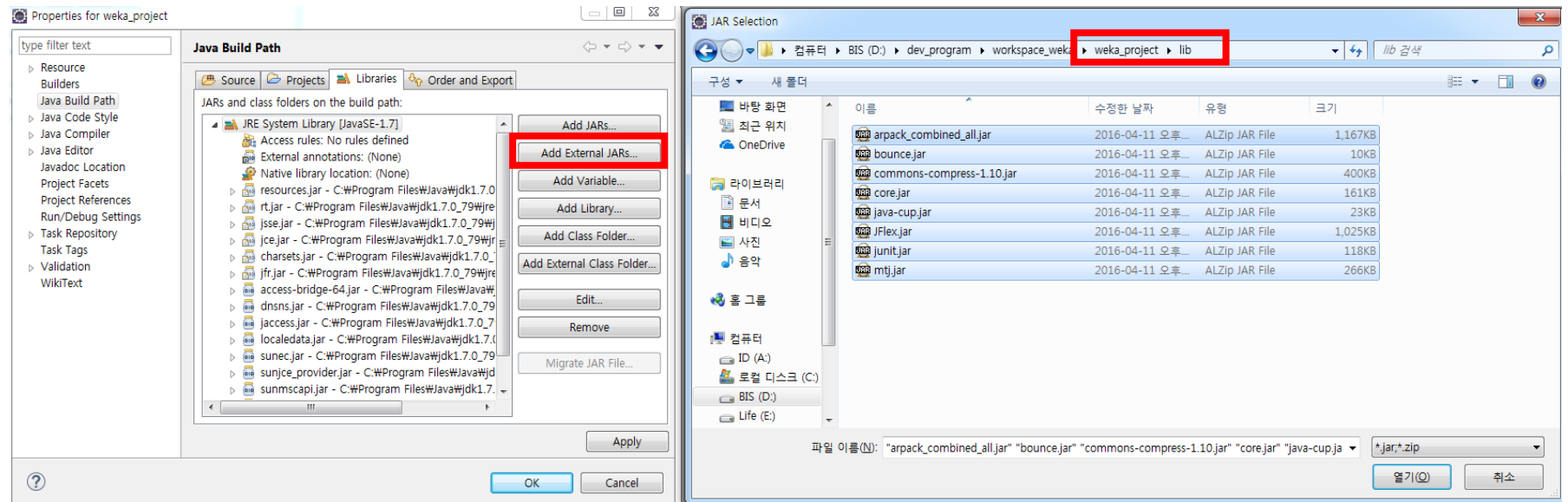


10. Click "OK."

# How to Install ?



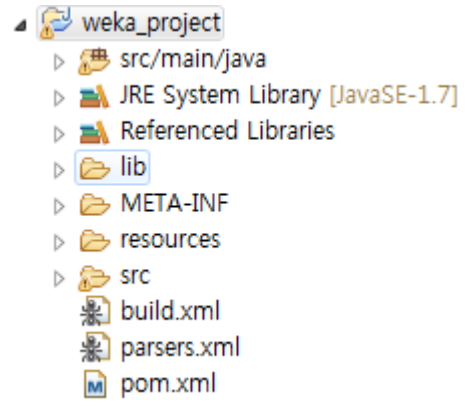
11. Click "Properties," and go to "Configure Build Path."



12. Click "Add External JARs," and select all of wek libs (in the lib folder).

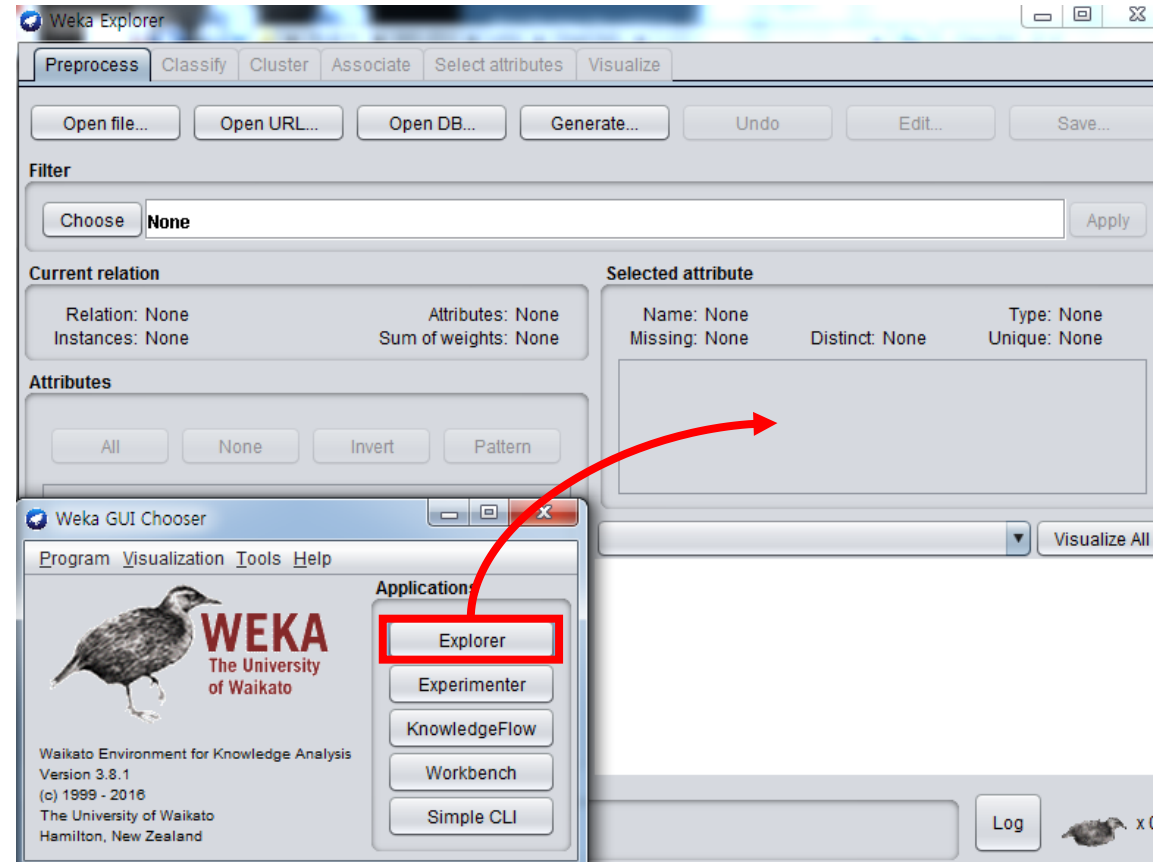
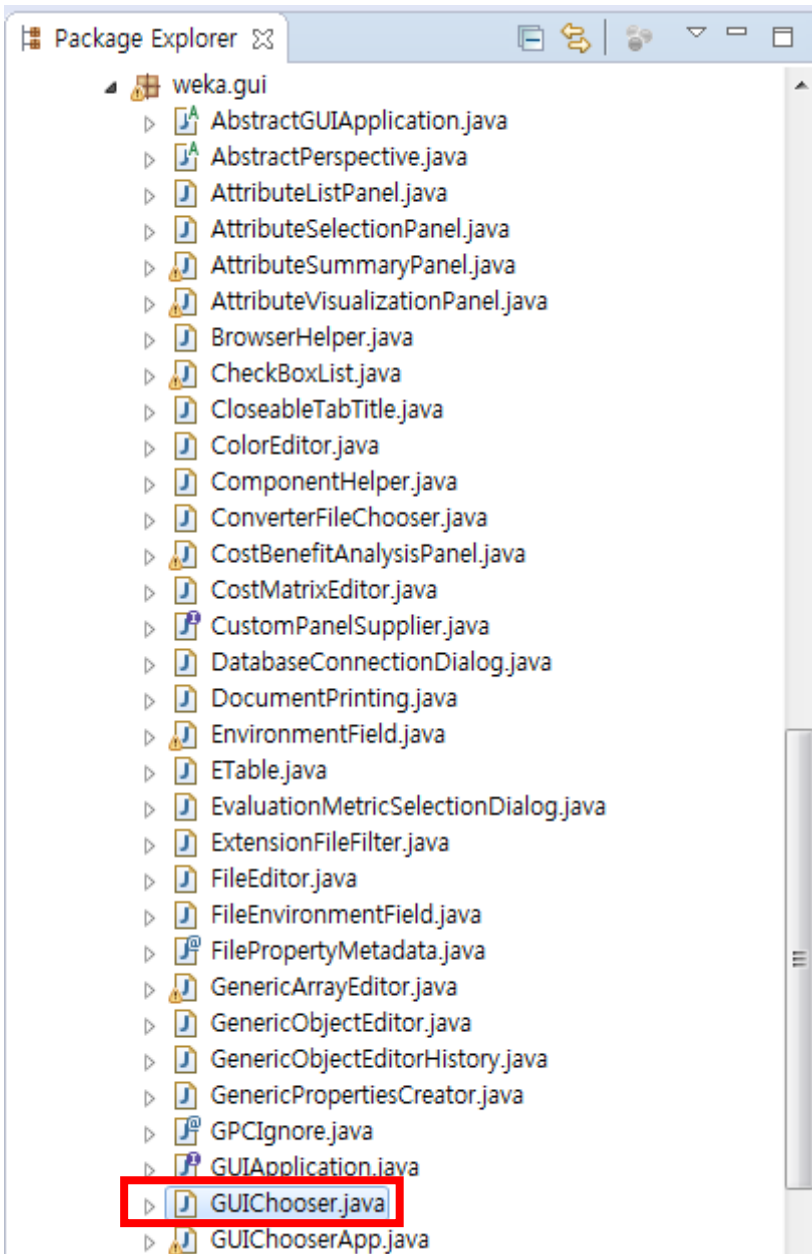
# ***How to Install ?***

---



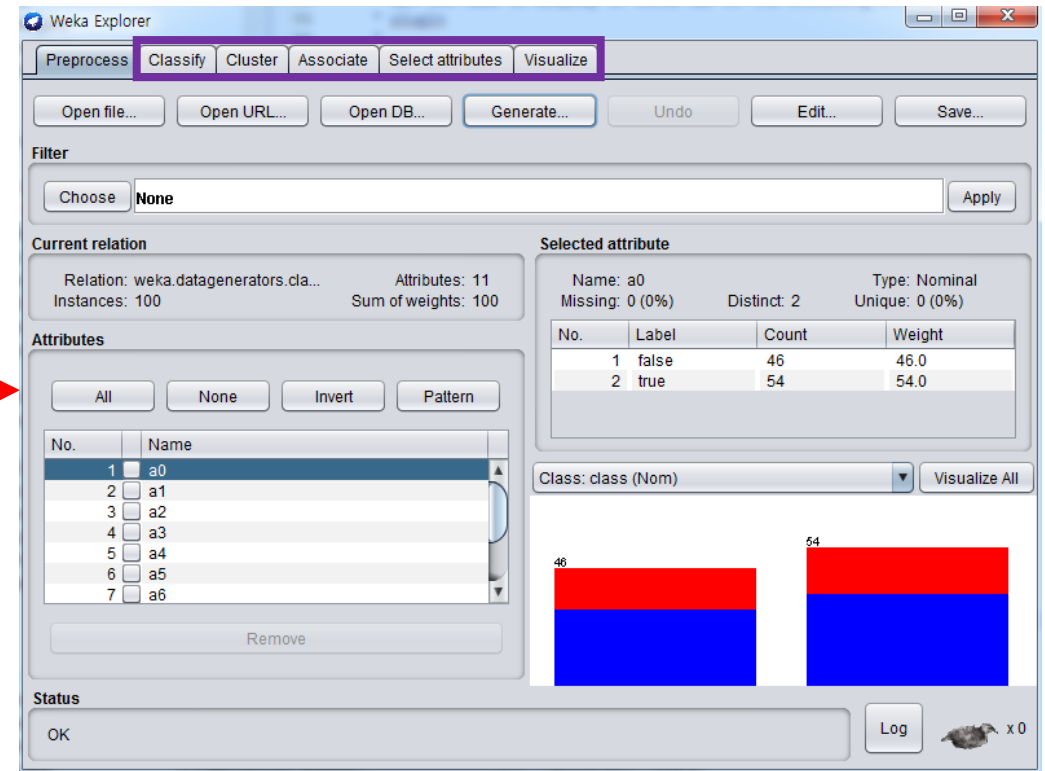
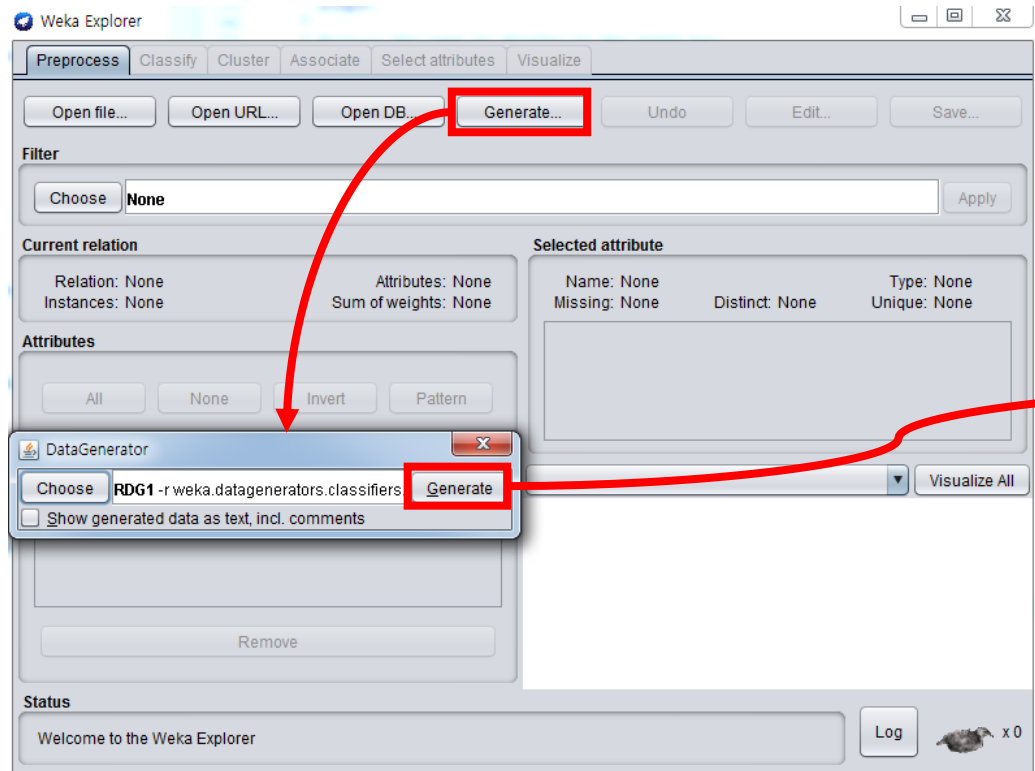
Finally, you are ready to do **HOMEWORK**.

# Weka



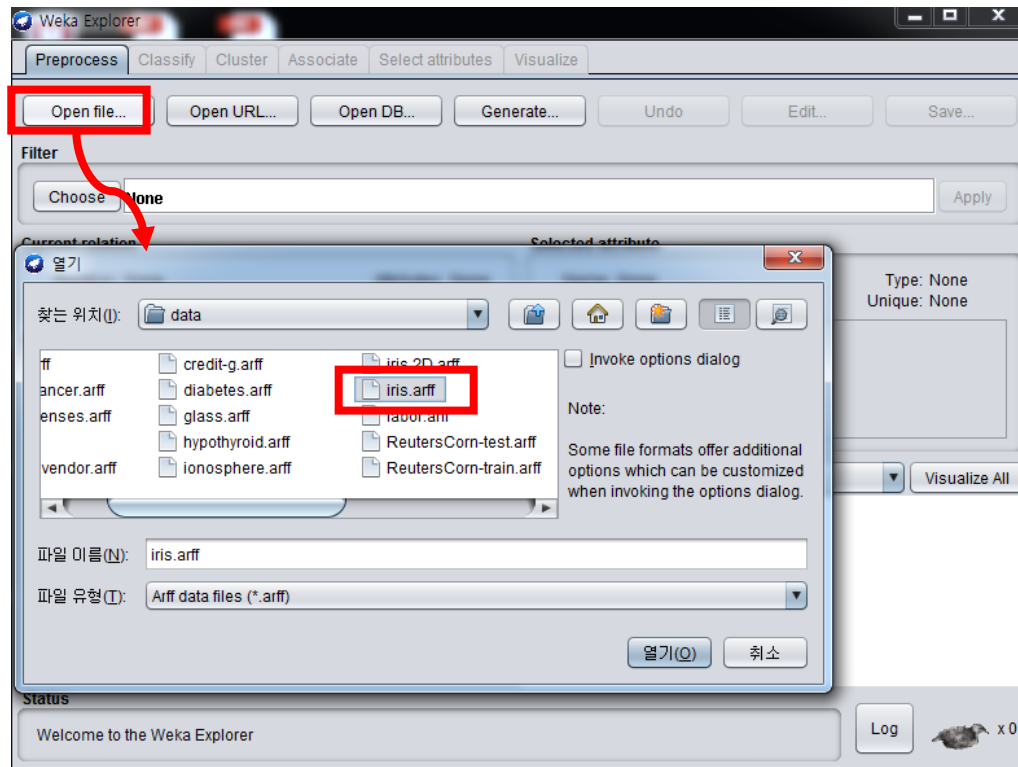
The main class of weka is "GUIChooser.java" in weka.gui package. If you run the class, you will see the above weka GUI. After clicking "Explorer" button, you can face with the main process.

# Weka: Loading File



Clicking "Generate" button makes a training set automatically.  
Next, you can see that the "Classify," "Cluster," "Associate," "Select attributes," and "Visualize" tab are activated.

# Weka: Loading File



Click "Open file" button and select data file.

**@RELATION iris** Data name

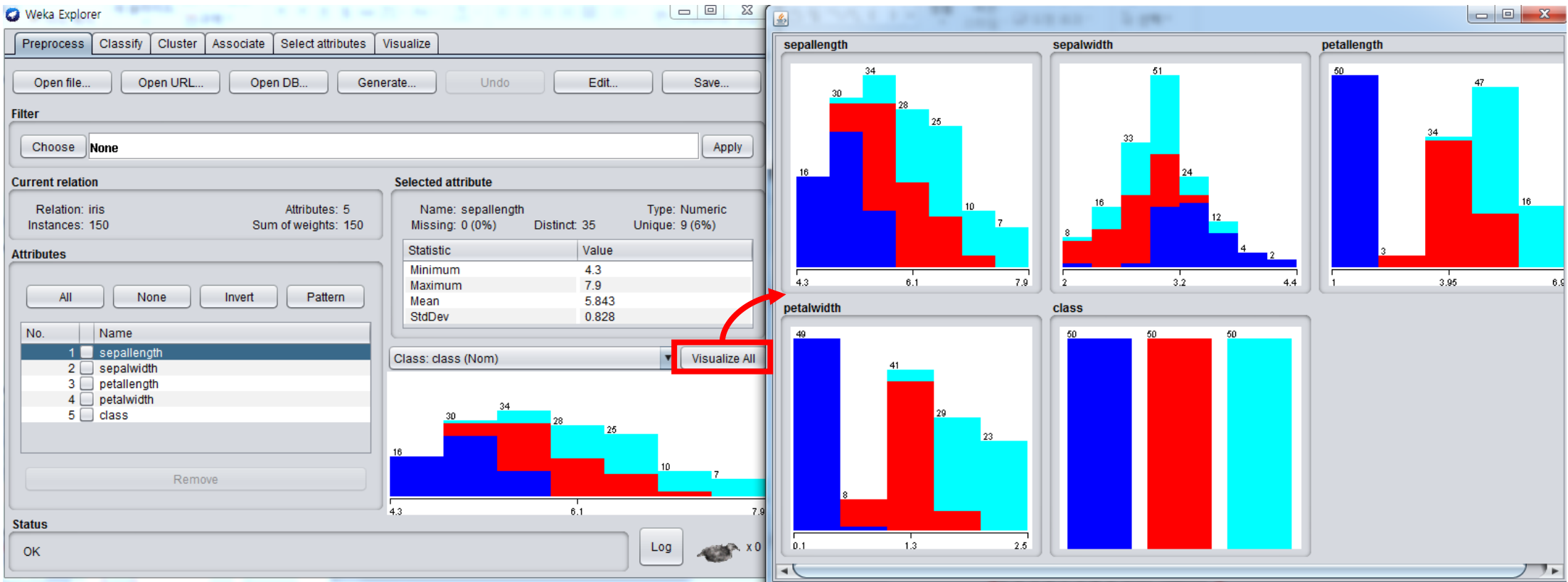
**@ATTRIBUTE** sepalength REAL  
**@ATTRIBUTE** sepalwidth REAL  
**@ATTRIBUTE** petallength REAL  
**@ATTRIBUTE** petalwidth REAL  
**@ATTRIBUTE** class {Iris-setosa,Iris-versicolor,Iris-virginica}

Attribute setting

**@DATA**  
5.1,3.5,1.4,0.2,Iris-setosa  
4.9,3.0,1.4,0.2,Iris-setosa  
4.7,3.2,1.3,0.2,Iris-setosa  
4.6,3.1,1.5,0.2,Iris-setosa  
5.0,3.6,1.4,0.2,Iris-setosa  
5.4,3.9,1.7,0.4,Iris-setosa  
4.6,3.4,1.4,0.3,Iris-setosa  
5.0,3.4,1.5,0.2,Iris-setosa  
4.4,2.9,1.4,0.2,Iris-setosa  
4.9,3.1,1.5,0.1,Iris-setosa  
5.4,3.7,1.5,0.2,Iris-setosa  
4.8,3.4,1.6,0.2,Iris-setosa  
4.8,3.0,1.4,0.1,Iris-setosa  
4.3,3.0,1.1,0.1,Iris-setosa  
5.8,4.0,1.2,0.2,Iris-setosa  
5.7,4.4,1.5,0.4,Iris-setosa  
5.4,3.9,1.3,0.4,Iris-setosa  
5.1,3.5,1.4,0.3,Iris-setosa  
5.7,3.8,1.7,0.3,Iris-setosa  
5.1,3.8,1.5,0.3,Iris-setosa

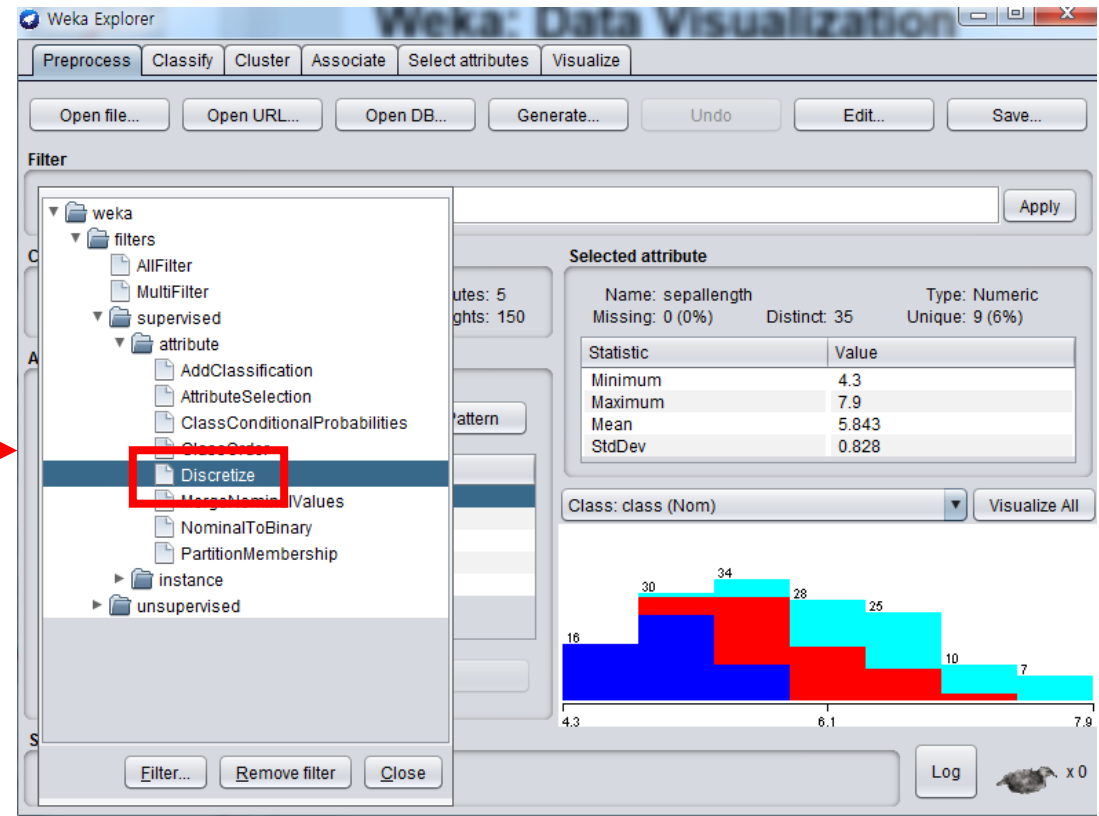
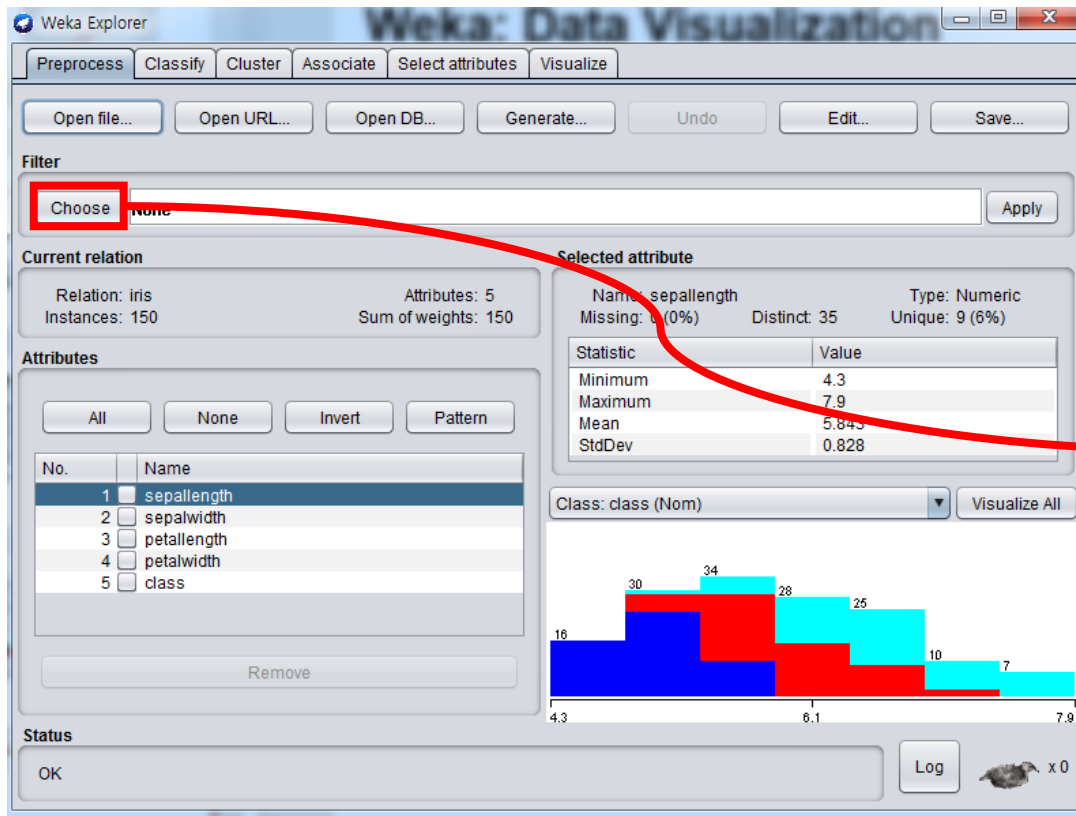
Instances

# Weka: Data Visualization



Clicking "Visualize All" button shows overall data distributions.

# Weka: Data Preprocessing



If you want to preprocess the given data set, then clicking "Choose" button serves several data preprocessing methods.



# Weka: Data Preprocessing

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Open file... | Open URL... | Open DB... | Generate... | Undo | Edit... | Save...

Filter: Choose **Discretize - R first-last - precision 6** **Apply**

Current relation: Relation: iris-weka.filters.supervis... Attributes: 5 Instances: 150 Sum of weights: 150

Selected attribute: Name: sepalength Missing: 0 (0%) Distinct: 3 Type: Nominal Unique: 0 (0%)

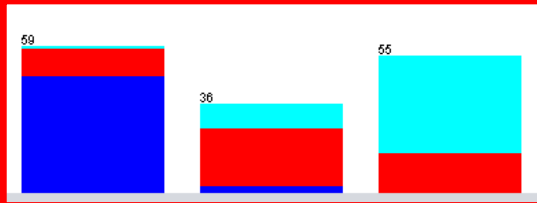
No.	Label	Count	Weight
1	'(-inf-5.55]'	59	59.0
2	'(5.55-6.15]'	36	36.0
3	'(6.15-inf)'	55	55.0

Attributes: All | None | Invert | Pattern

No.	Name
1	<input checked="" type="checkbox"/> sepalength
2	<input type="checkbox"/> sepalwidth
3	<input type="checkbox"/> petalength
4	<input type="checkbox"/> petalwidth
5	<input type="checkbox"/> class

Remove

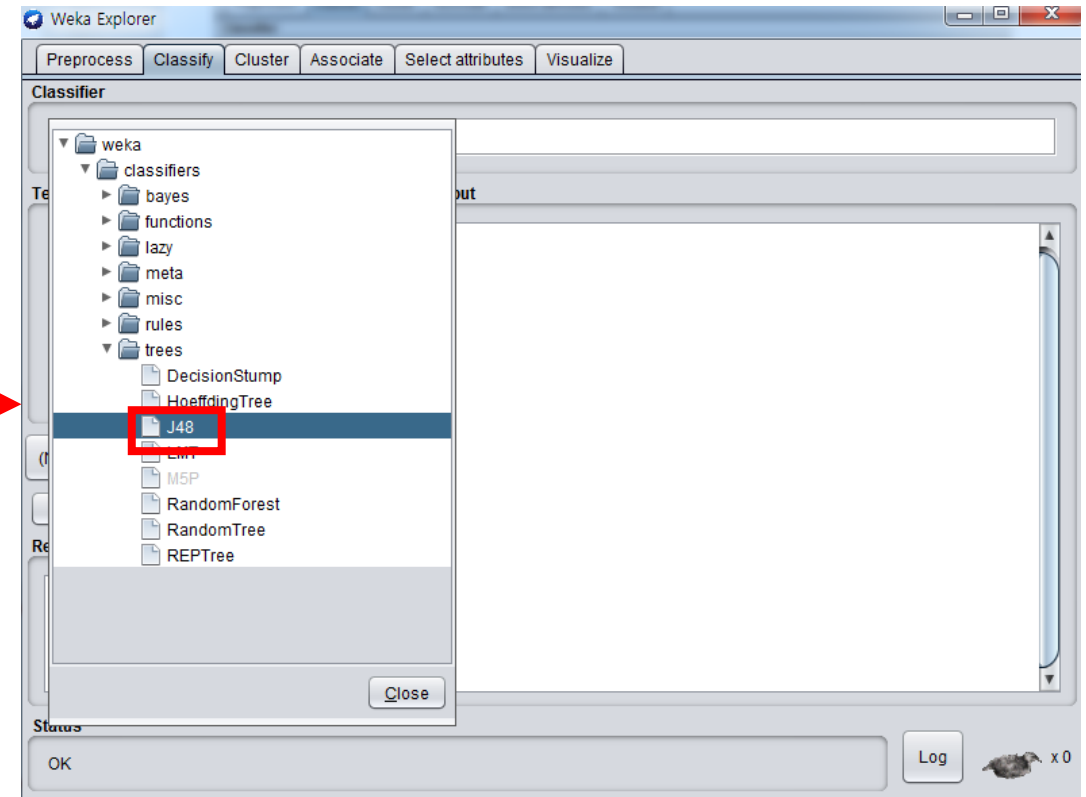
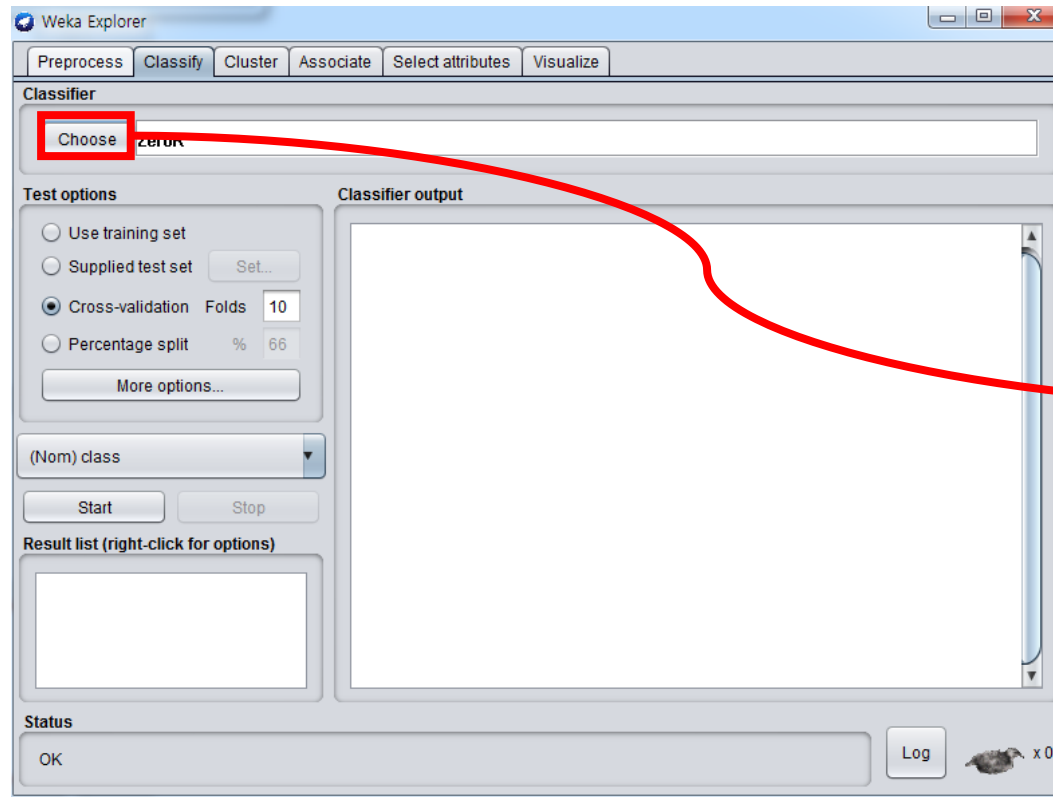
Class: class (Nom) Visualize All



Status: OK Log x 0

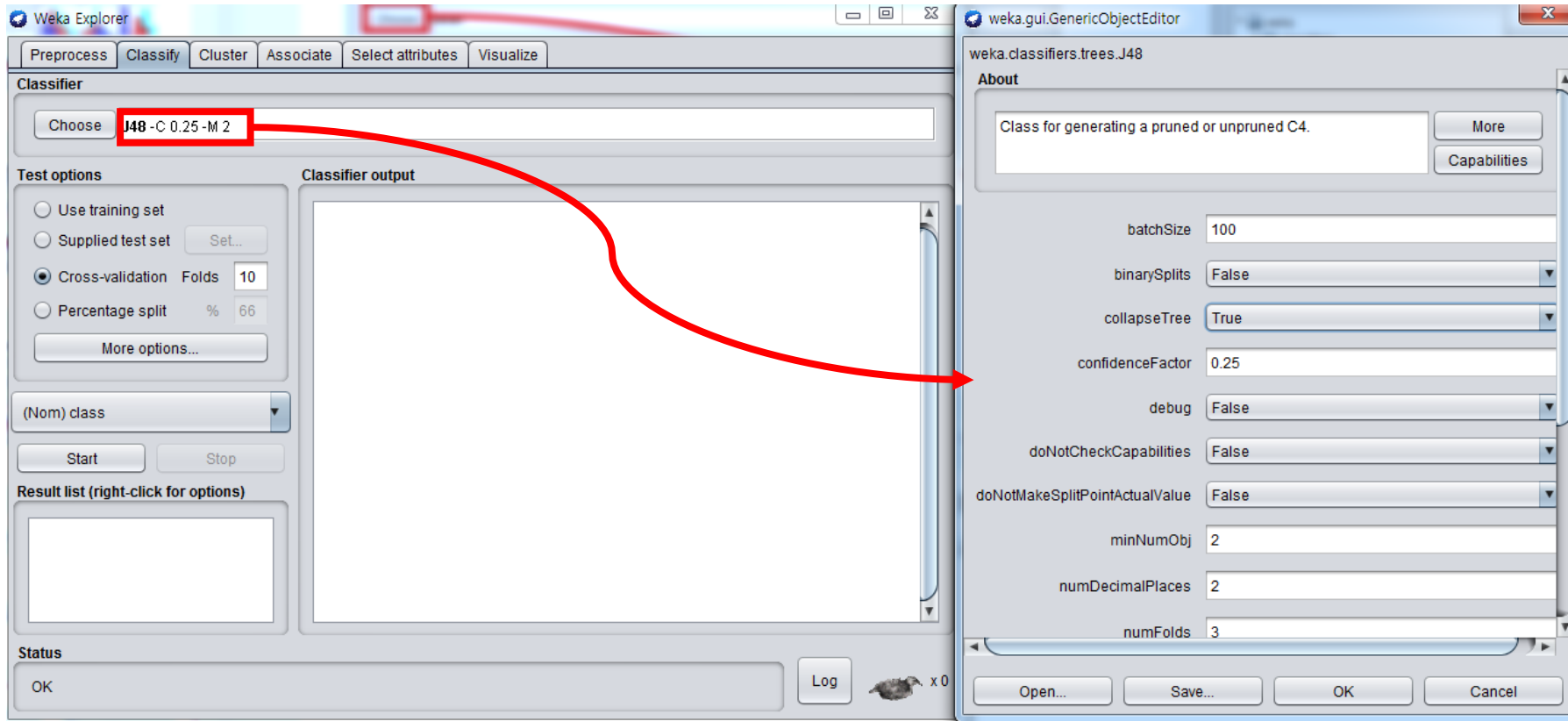
Clicking "Apply" button shows applied data set.

# ***Weka: Classifier***



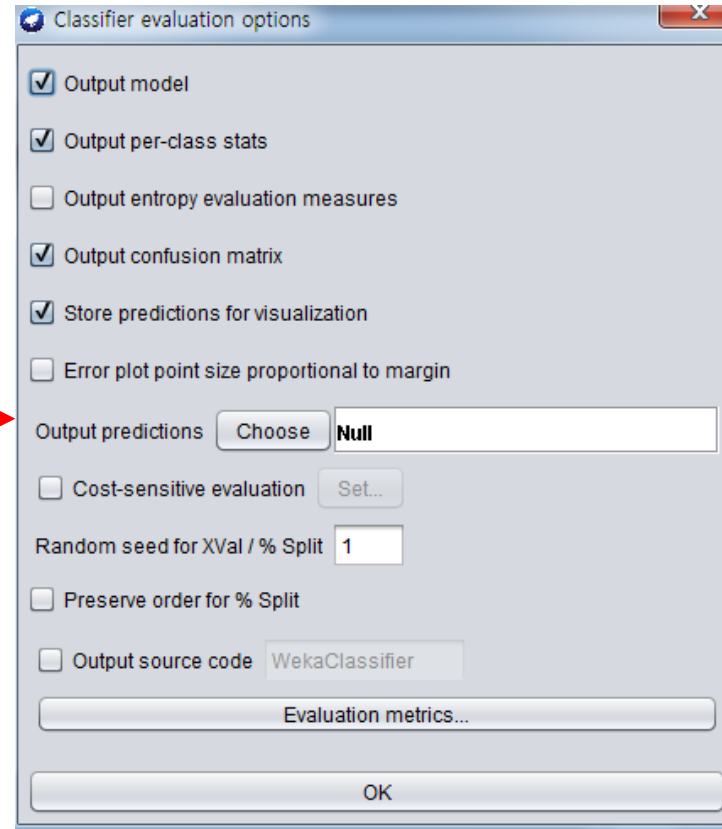
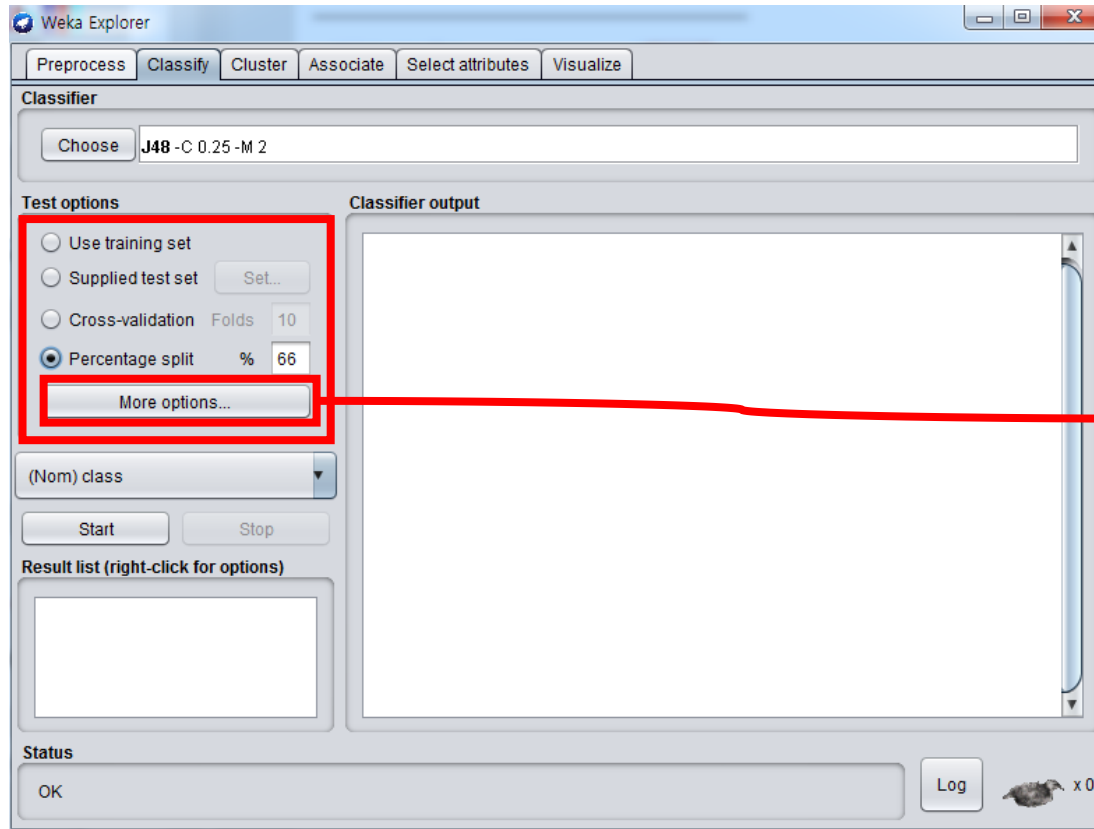
Weka provides several classifiers, for example, decision tree, and multilayer perceptron and so on. Clicking "Choose" button will show you lots of classifiers.

# Weka: Classifier



Clicking the above red mark shows detail options.  
You can set the details that you want to apply the given classifier.

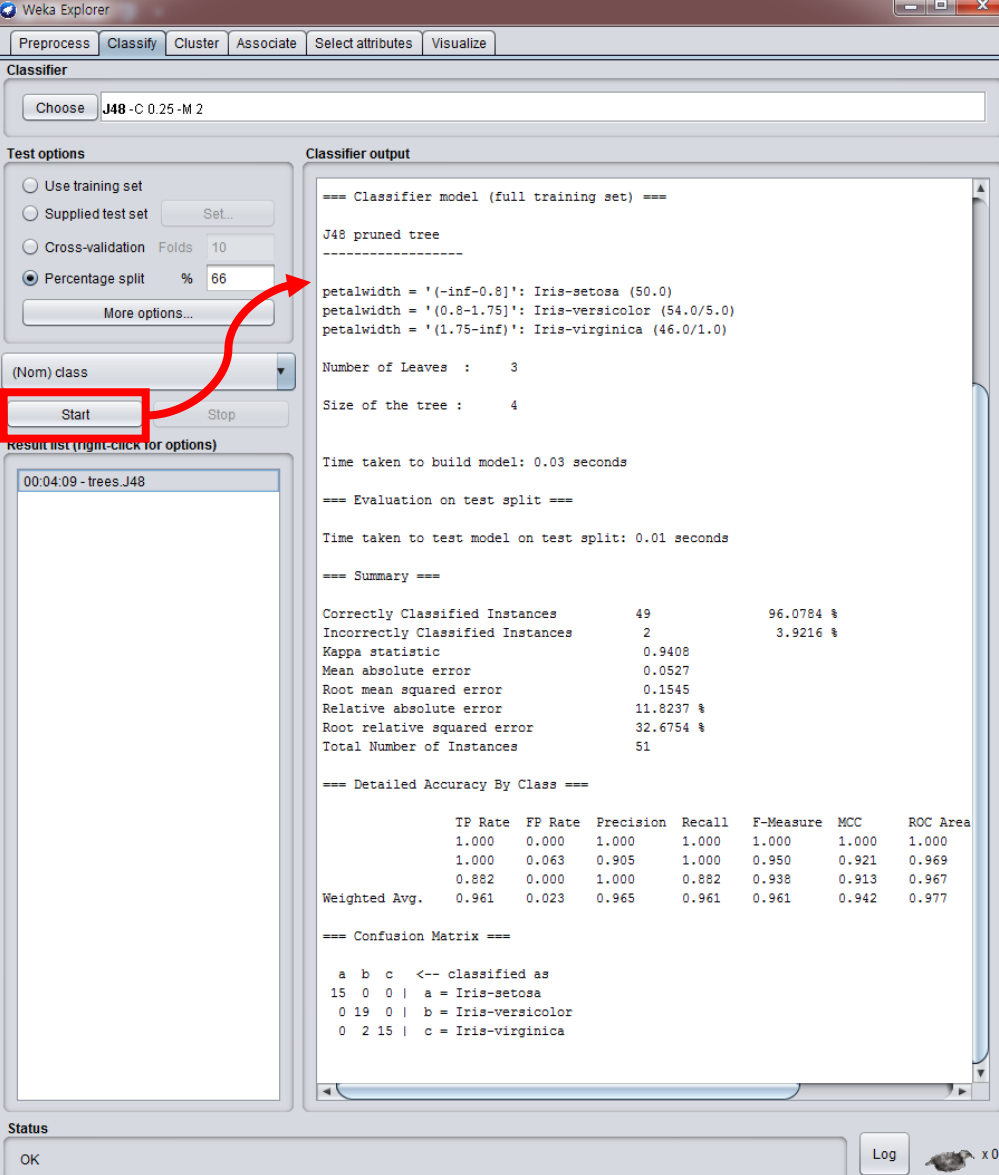
# Weka: Classifier



When training the classifier, it provides test option using the given training data. If you want to set more details, than clicking "More options" derives you detail setting window.

# Weka: Classifier

Clicking "Start" button runs the algorithm and prints options (you sat before), processing, and results.



The screenshot shows the Weka Explorer window with the 'Classify' tab selected. The classifier chosen is 'J48 - C 0.25 - M 2'. In the 'Test options' section, 'Percentage split' is selected with a value of 66. The 'Start' button is highlighted with a red box, and a red arrow points to it from the 'More options...' button. The 'Classifier output' pane displays the following results:

```
=== Classifier model (full training set) ===  
J48 pruned tree  
-----  
petalwidth = '(-inf-0.8]': Iris-setosa (50.0)  
petalwidth = '[0.8-1.75]': Iris-versicolor (54.0/5.0)  
petalwidth = '(1.75-inf)': Iris-virginica (46.0/1.0)  
  
Number of Leaves :    3  
Size of the tree :    4  
  
Time taken to build model: 0.03 seconds  
  
=== Evaluation on test split ===  
  
Time taken to test model on test split: 0.01 seconds  
  
=== Summary ===  
  
Correctly Classified Instances      49           96.0784 %  
Incorrectly Classified Instances    2            3.9216 %  
Kappa statistic                    0.9408  
Mean absolute error                 0.0527  
Root mean squared error            0.1545  
Relative absolute error            11.8237 %  
Root relative squared error        32.6754 %  
Total Number of Instances         51  
  
=== Detailed Accuracy By Class ===  
  
          TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  
1.000    0.000    1.000    1.000    1.000    1.000    1.000  
1.000    0.063    0.905    1.000    0.950    0.921    0.969  
0.882    0.000    1.000    0.882    0.938    0.913    0.967  
Weighted Avg.   0.961    0.023    0.965    0.961    0.961    0.942    0.977  
  
=== Confusion Matrix ===  
  
  a  b  c  <-- classified as  
15  0  0 | a = Iris-setosa  
 0 19  0 | b = Iris-versicolor  
 0  2 15 | c = Iris-virginica
```

The 'Result list' pane shows a single entry: '00:04:09 - trees.J48'. The 'Status' bar at the bottom shows 'OK' and a 'Log' button.

# Weka: Classifier

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set  
☐ Supplied test set Set...  
☐ Cross-validation Folds 10  
☒ Percentage split % 66  
More options...

(Nom) class

Start Stop

Result list (right-click for options)

00:04:09 - trees.J48

- View in main window
- View in separate window
- Save result buffer
- Delete result buffer(s)
- Load model
- Save model
- Re-evaluate model on current test set
- Re-apply this model's configuration
- Visualize classifier errors
- Visualize tree
- Visualize margin curve
- Visualize threshold curve
- Cost/Benefit analysis
- Visualize cost curve

Classifier output

=== Classifier model (full training set) ===

J48 pruned tree

-----

petalwidth = '(-inf-0.8]': Iris-setosa (50.0)  
petalwidth = '(0.8-1.75]': Iris-versicolor (54.0/5.0)  
petalwidth = '(1.75-inf)': Iris-virginica (46.0/1.0)

Number of Leaves : 3  
Size of the tree : 4

Time taken to build model: 0.03 seconds

Test split ==

Test model on test split: 0.01 seconds

	Classified Instances		
	49	96.0784 %	
misclassified Instance	2	3.9216 %	
error	0.9408		
std error	0.0527		
std error	0.1545		
std error	11.8237 %		
squared error	32.6754 %		
Instances	51		

Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area
	1.000	0.000	1.000	1.000	1.000	1.000	1.000
	1.000	0.063	0.905	1.000	0.921	0.921	0.969
	0.882	0.000	1.000	0.882	0.938	0.813	0.967
Weighted Avg.	0.961	0.023	0.965	0.961	0.961	0.942	0.977

=== Confusion Matrix ===

a b c <-- classified as

15 0 0 | a = Iris-setosa

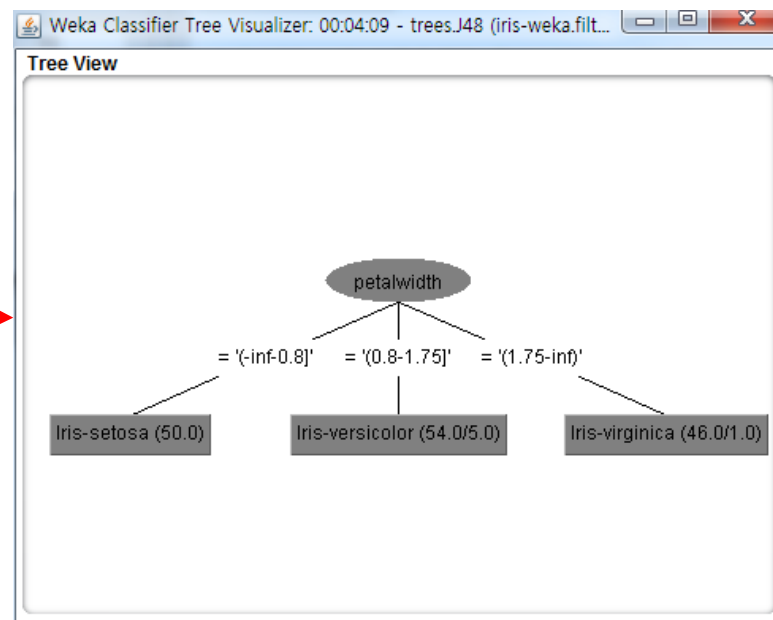
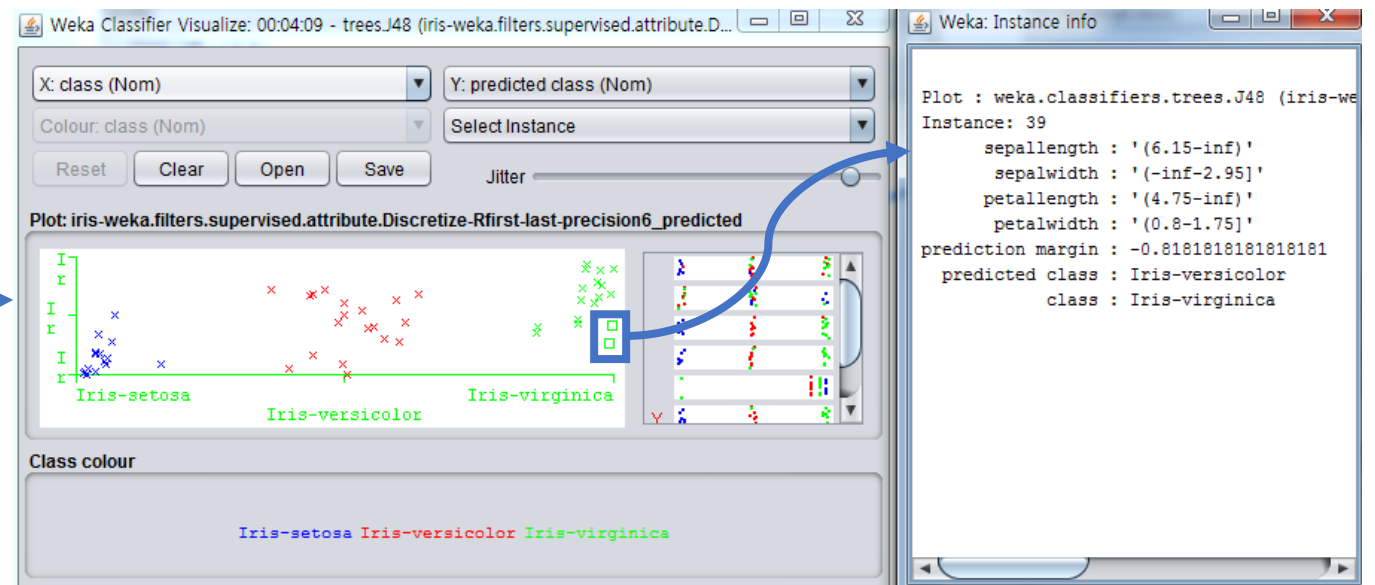
0 19 0 | b = Iris-versicolor

0 2 15 | c = Iris-virginica

Status

OK

Log



***Thanks-***