


Wekaの導入と実行

Wekaのダウンロード

- 公式ページにアクセスします

<http://www.cs.waikato.ac.nz/ml/weka/>



Machine Learning

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

Home

- Getting started
- Requirements
- Download
- Documentation
- FAQ
- Citing Weka

Further information

- Datasets
- Related Projects
- Miscellaneous Code
- Other Literature

Developers

- Development
- History
- Subversion
- Contributors

Various

- Wekalist stats

Weka 3: Data Mining Software in Java

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

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

Pentaho's live forum for Weka

The open-source BI software company Pentaho is a major sponsor of Weka development and provides a forum for interaction among Weka project community members.

The Weka mailing list

Please post Weka-related questions, comments, and bug reports to the **Weka** mailing list (the **online documentation** first, before posting to the list). There is also the **Weka** mailing list (Mirrors: **news.gmane.org**, **Nabble**). Please do not email individual members with problems.

Also, please have in mind that your message will be sent to several thousand members of the **Mailing List Etiquette**. The administrator also removes members from the mailing list if they do not follow the mailing list rules.

Wekaのダウンロード

- 左の欄からDownloadを選ぶと、下の方にDeveloper versionがあるので、使ってるパソコンのbit数に合ったもののJava VM付きのものをダウンロードします

- Developer version

This is the trunk of Weka and continues from the stable-3-6 code line. It receives both bug fixes and new features.

- Windows x86

Click [here](#) to download a self-extracting executable that includes Java VM 1.6 (weka-3-7-7/jre.exe; 37.4 MB)

Click [here](#) to download a self-extracting executable without the Java VM (weka-3-7-7.exe; 22.6 MB)

These executables will install Weka in your Program Menu. Download the second version if you already have Java 1.6 (or later) on your system.

- Windows x64

Click [here](#) to download a self-extracting executable that includes 64 bit Java VM 1.6 (weka-3-7-7/jre-x64.exe; 37.7 MB)

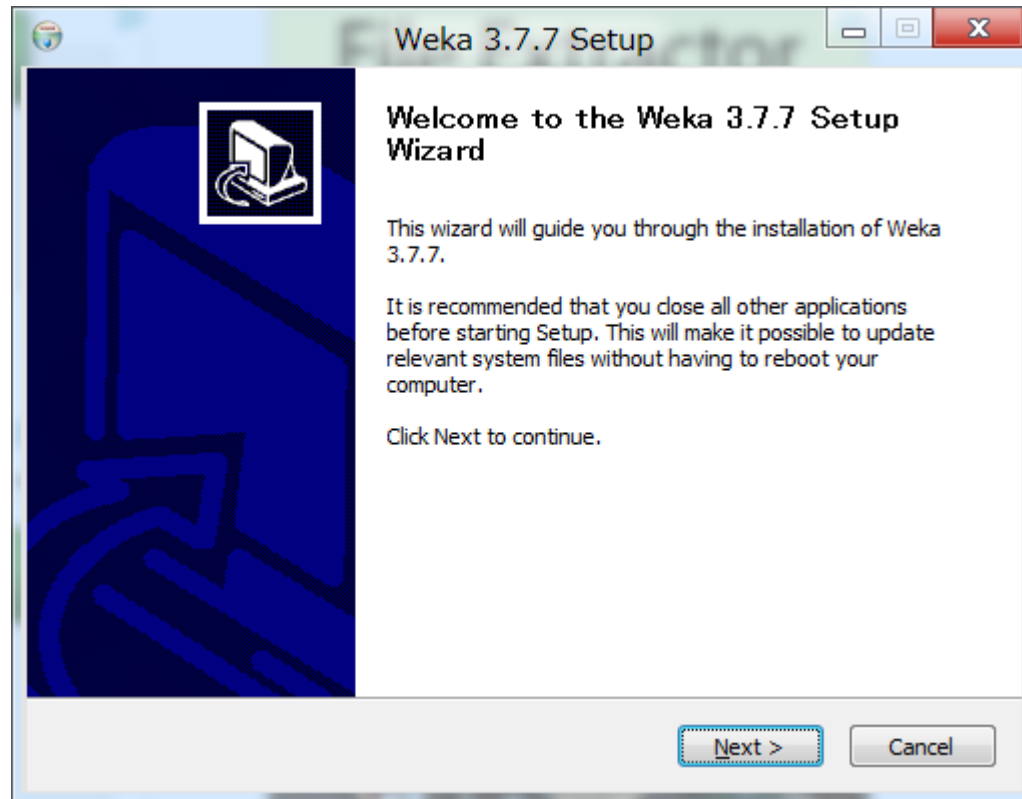
Click [here](#) to download a self-extracting executable without the Java VM (weka-3-7-7-x64.exe; 22.6 MB)

These executables will install Weka in your Program Menu. Download the second version if you already have Java 1.6 (or later) on your system.

※上にある3rd ed. Versionは日本語対応ですが不具合があります(日本語ロケールの問題?)

Wekaのインストール

- ダウンロードが済んだらexeファイルを実行します



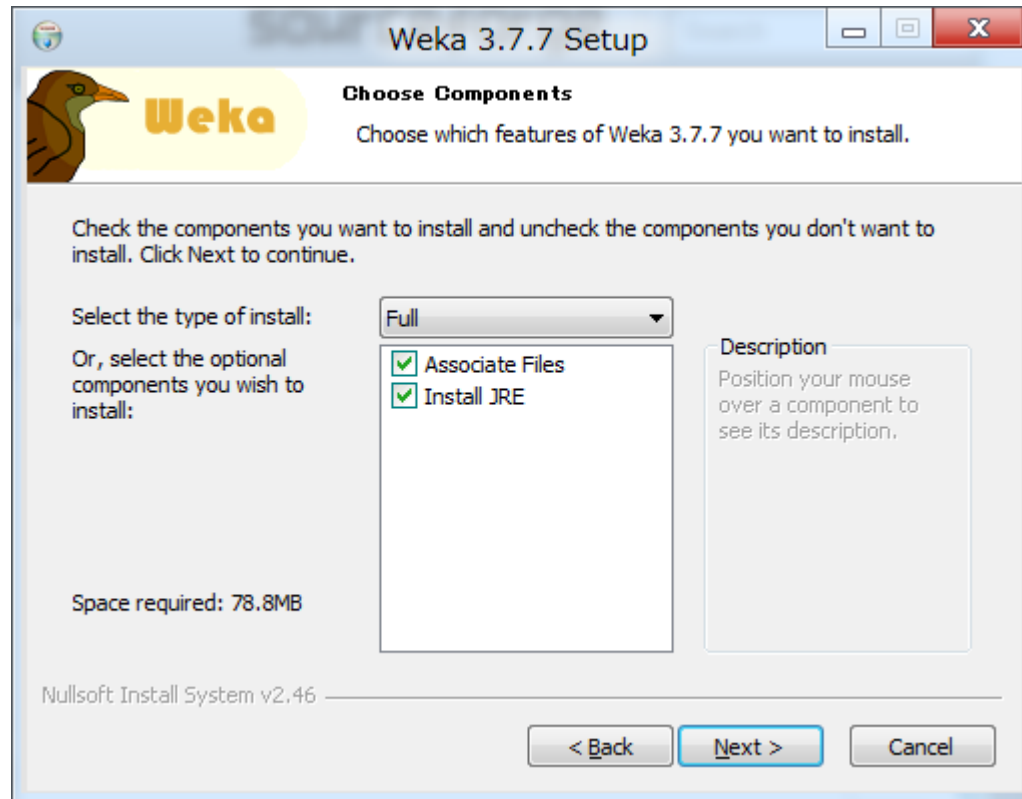
Wekaのインストール

- ライセンスに同意します



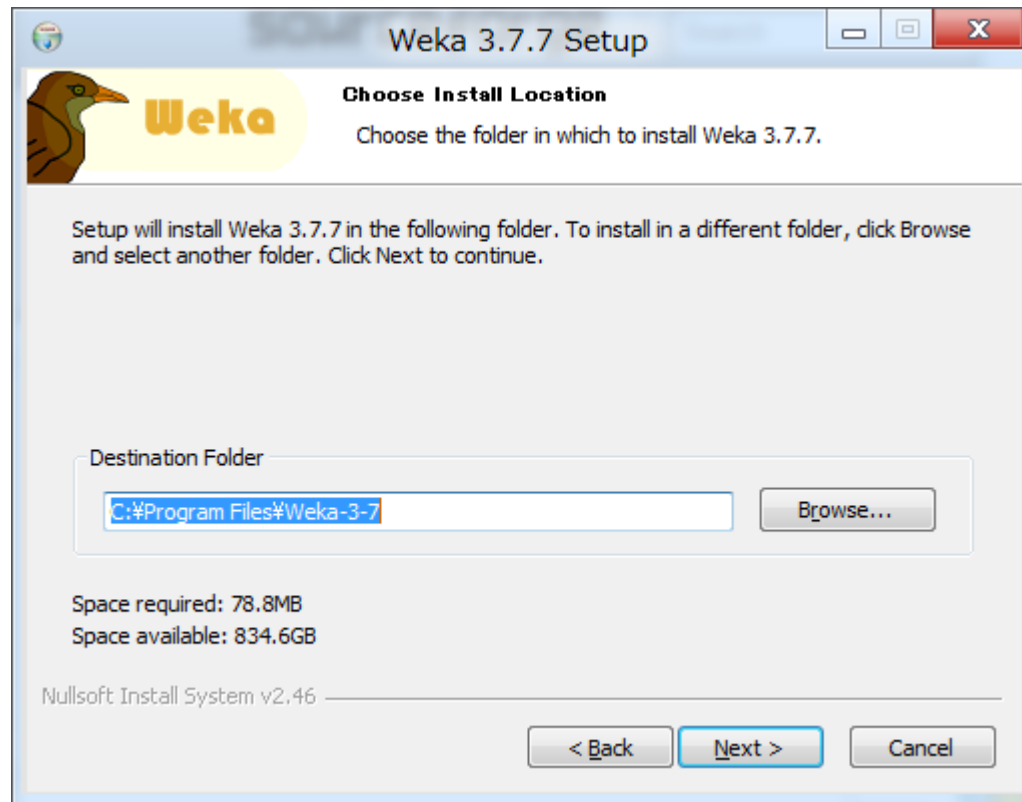
Wekaのインストール

- “全てインストール”にして先に進みます



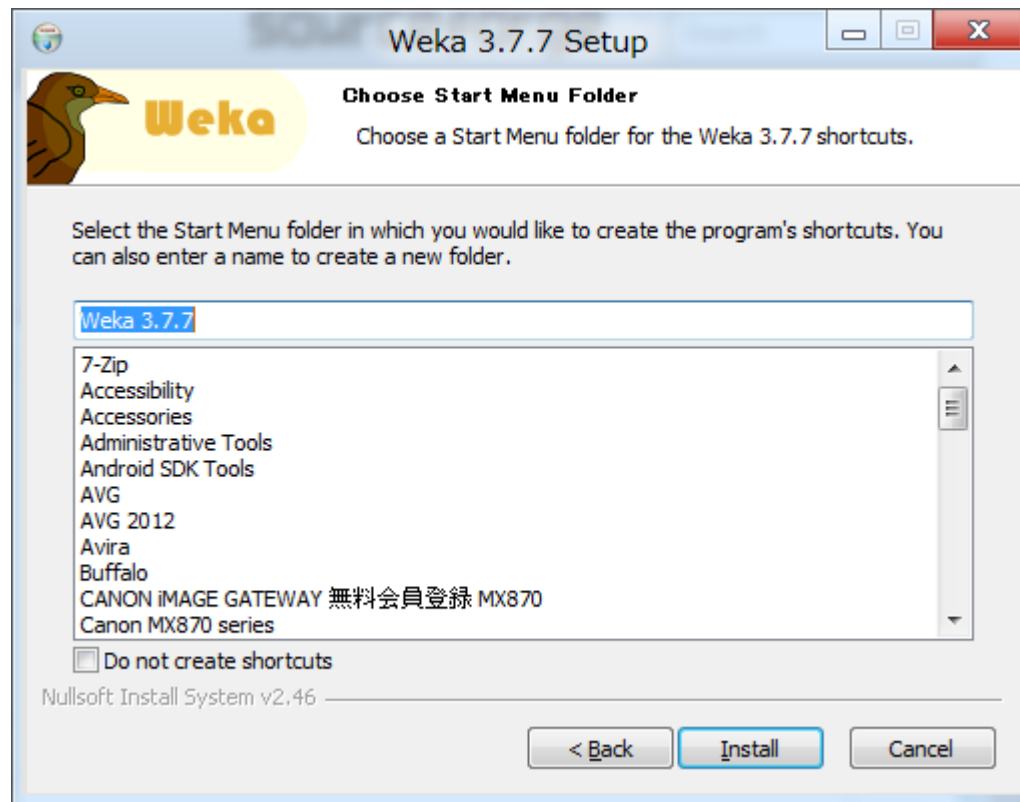
Wekaのインストール

- インストール場所を設定します



Wekaのインストール

- スタートメニュー・ショートカットの設定をします



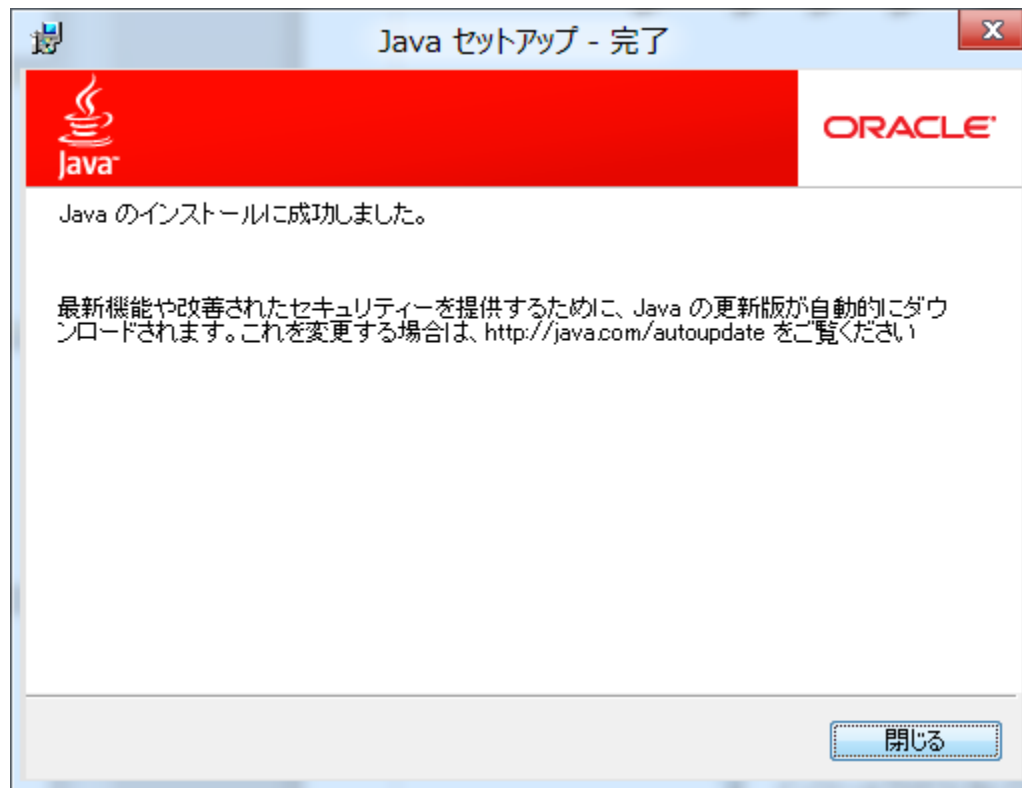
Javaのインストール

- インストール途中でJavaのインストールが始まります。



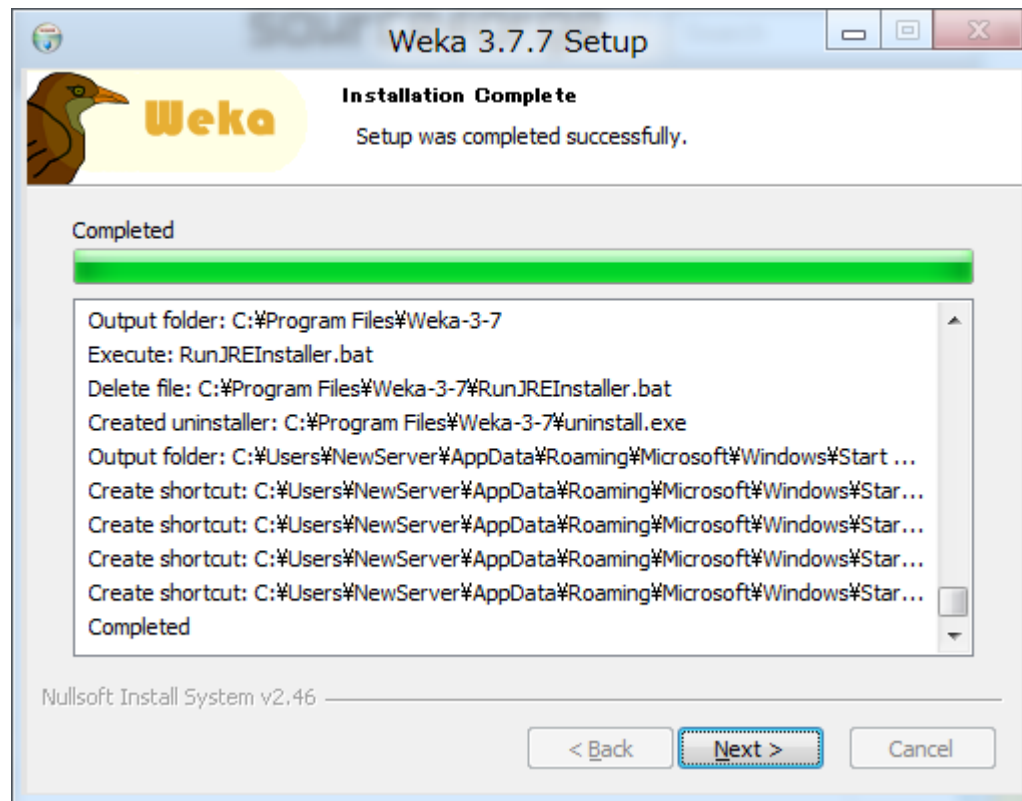
Javaのインストール

- インストールが終わったらウィンドウを閉じます



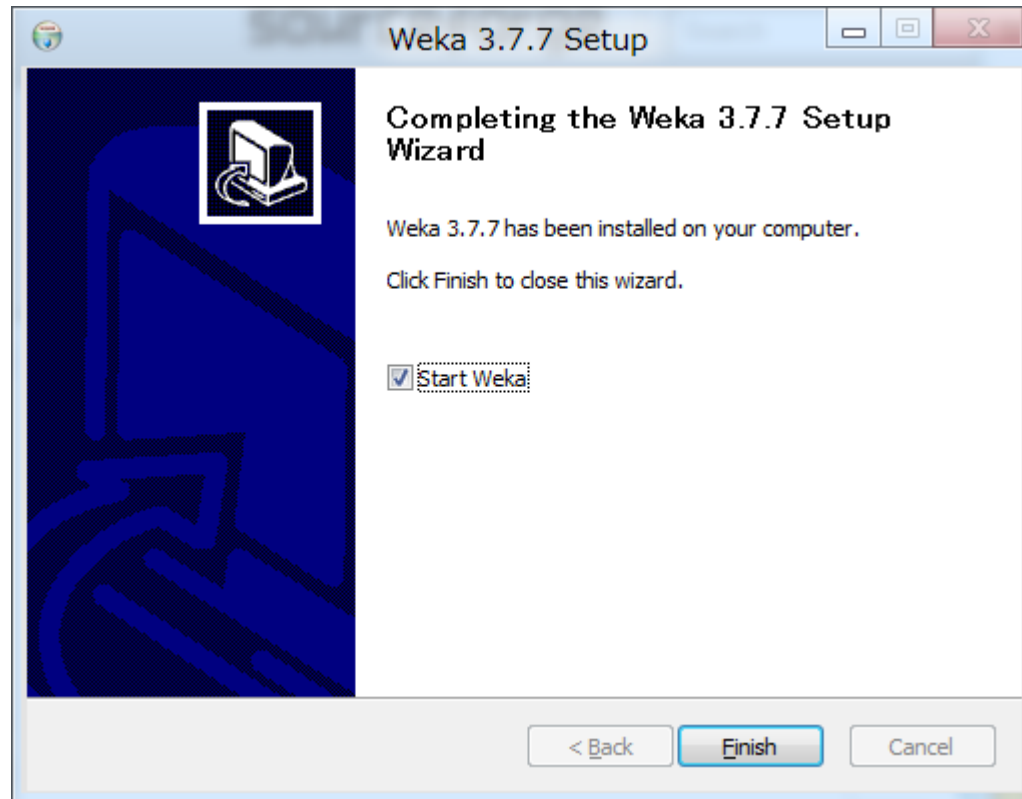
Wekaのインストール

- インストールが終わったらNextを押して先に進みます



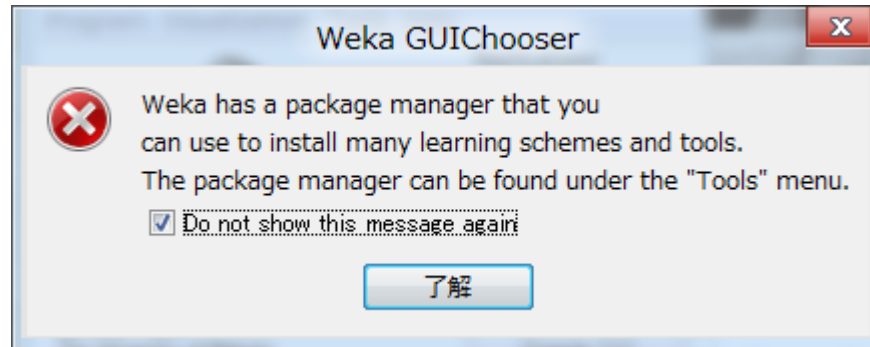
Wekaのインストール

- Start Wekaにチェックを入れてFinishを押します。うっかり閉じた場合は(C:\Program Files\Weka-3-7にあるWekaa3.7を実行します)



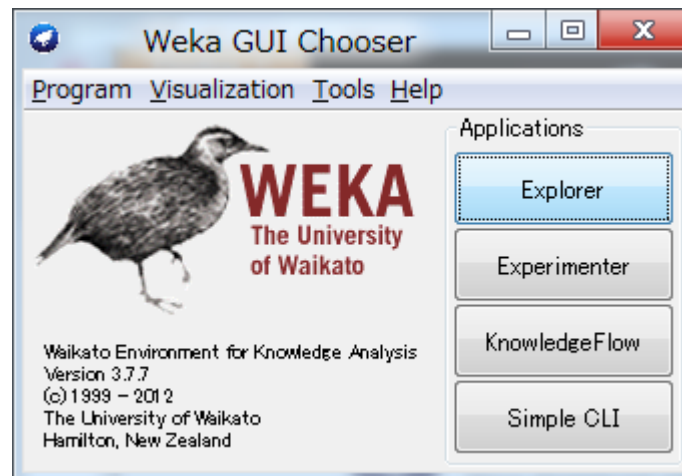
Wekaの実行

- Wekaを実行すると警告が出ますが、チェックを入れて了解を押します



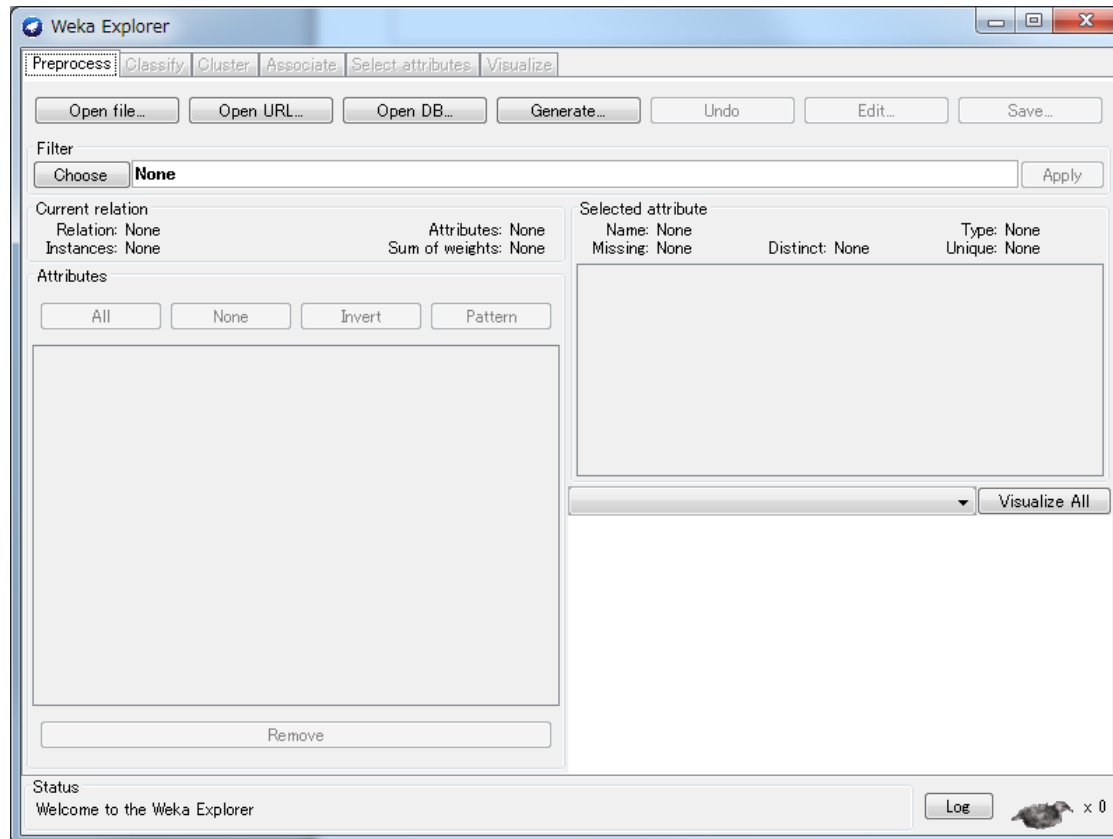
Wekaの実行

- これがWekaの起動画面です。
- Explorerを押します



Wekaの実行

- この画面からデータを開いたりデータマイニングをしたりします



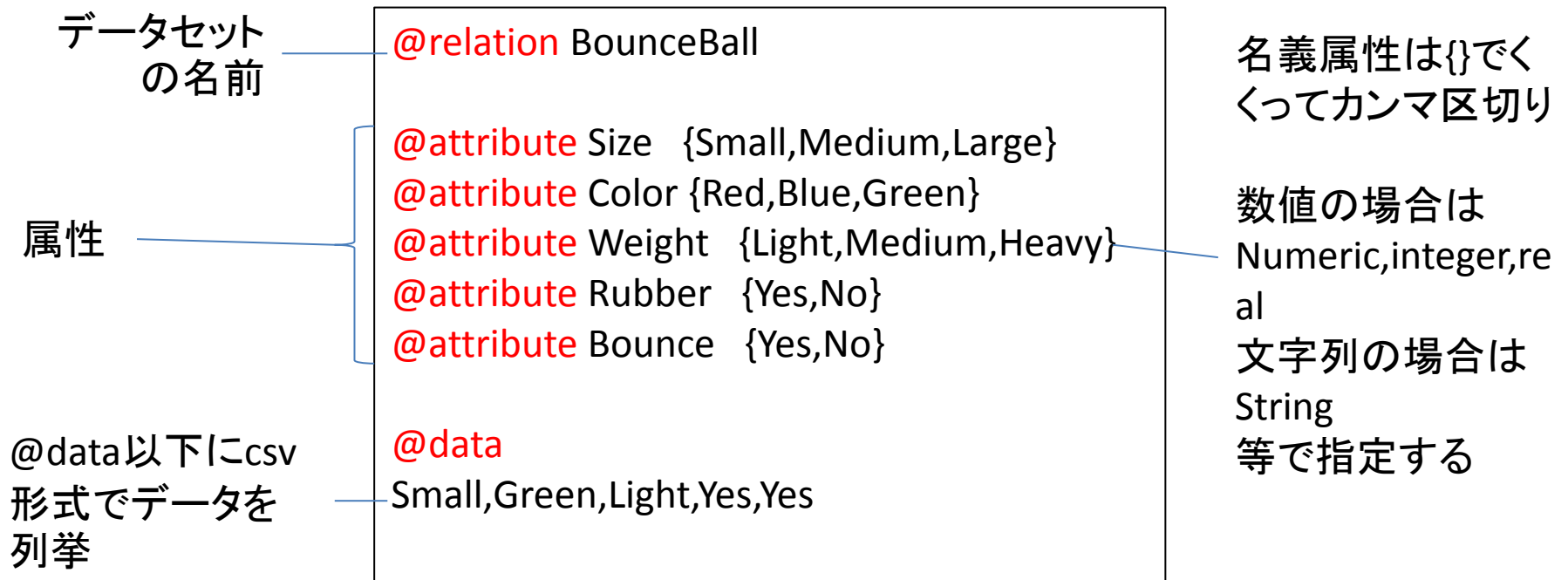
データセットの作成

- 例としてbounce ballのデータセットを作成し、データマイニングを試みましょう

Ball	Size	Color	Weight	Rubber?	Result(Bounce?)
1	Small	Green	Light	Yes	Yes
2	Small	Blue	Medium	No	No
3	Medium	Red	Medium	No	No
4	Small	Red	Medium	Yes	Yes
5	Large	Green	Heavy	Yes	Yes
6	Medium	Blue	Heavy	Yes	No
7	Medium	Green	Heavy	Yes	No
8	Small	Red	Light	No	No

データセットの書式

- Wekaではarff形式のファイルが推奨されている



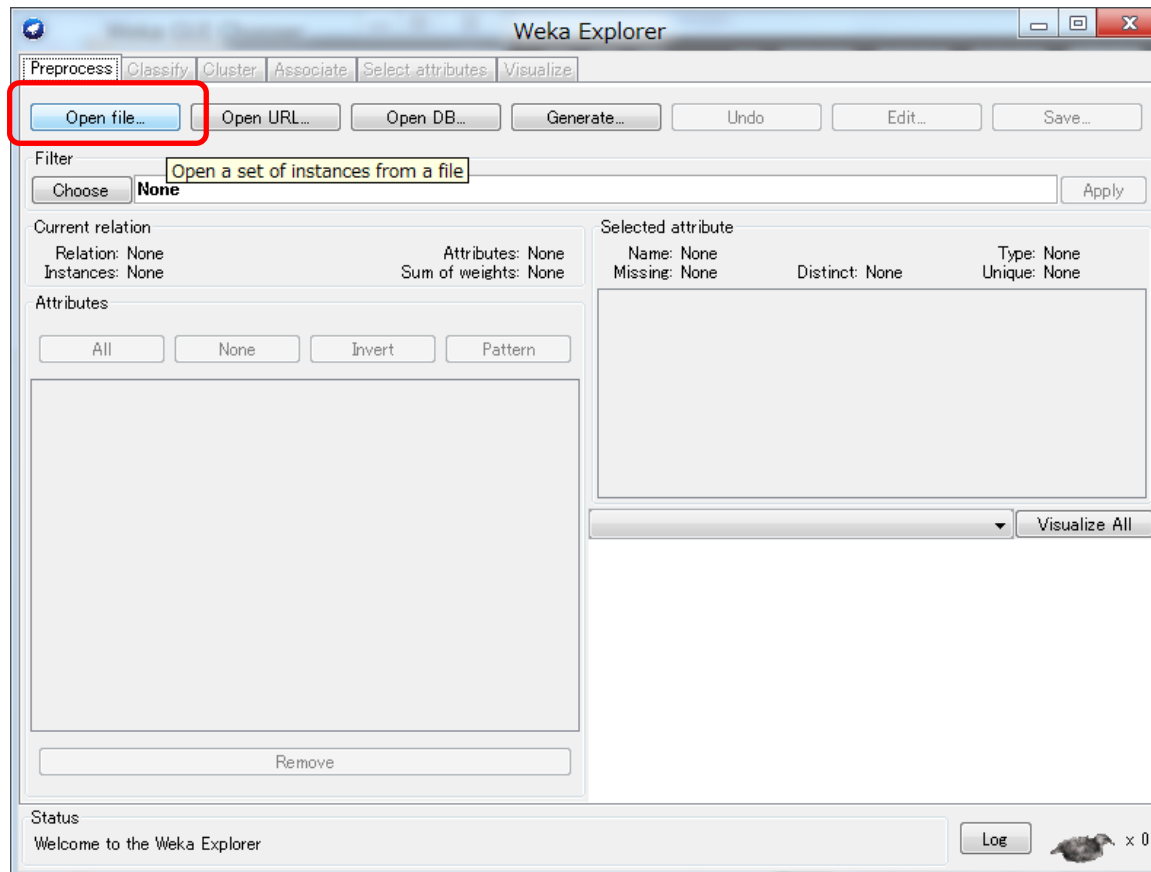
※dataフォルダ内のサンプルも参考にしてもよい

※Wekaでは大文字小文字を区別します

※csv形式のファイルも読み込めますが、推奨はされません

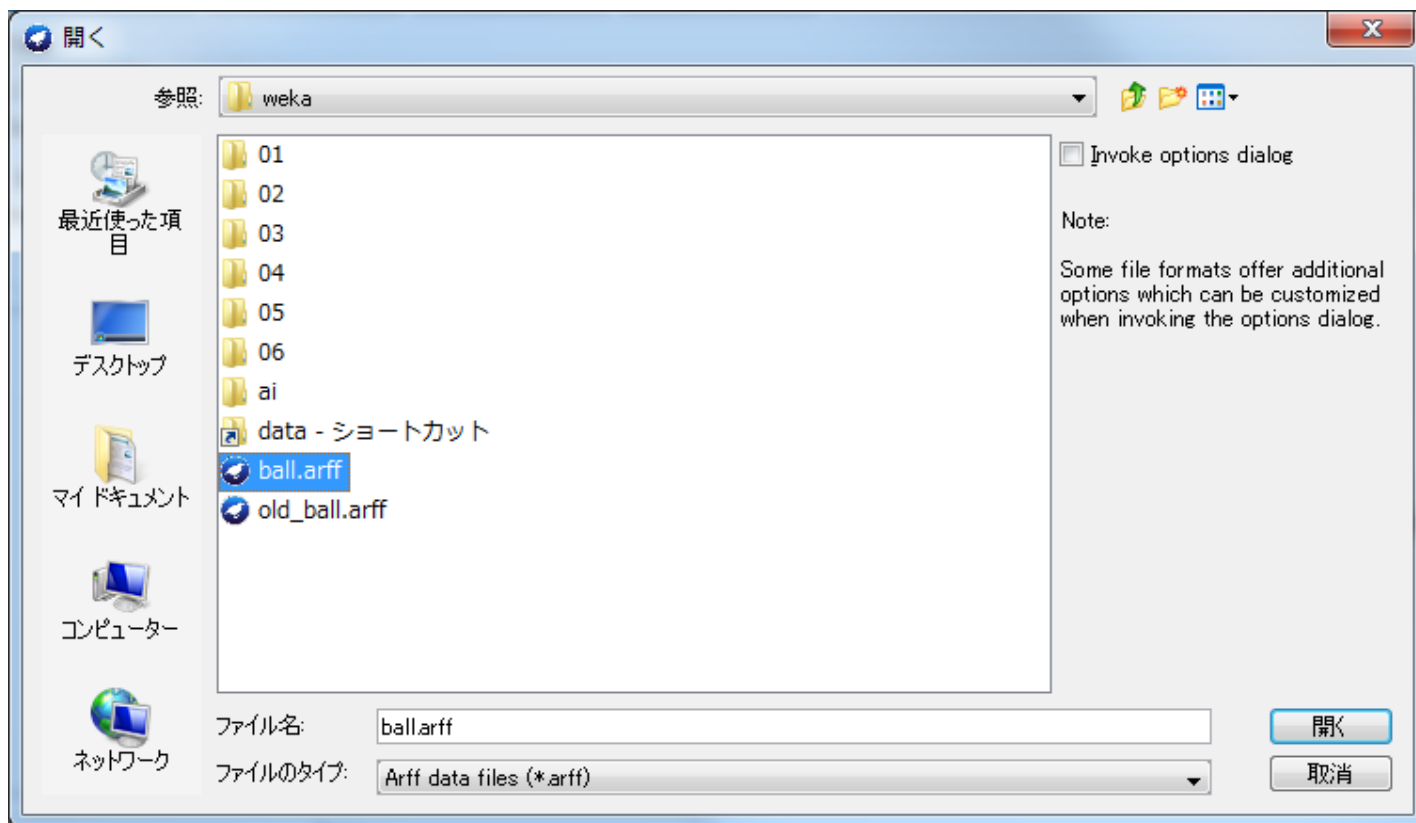
データセットの読み込み

- Preprocessタブを開き、Open fileを押します



データセットの読み込み

- いま、作成したBounceBallのデータセットを読み込みます



データセットの読み込み

- 書式が間違っている場合は、以下の様なエラーが出るので、間違っている部分を修正しましょう



Wekaの実行

- データの内容が反映されていることを確認します

The screenshot shows the Weka Explorer interface. The 'Preprocess' tab is active, and the 'weather' relation is loaded. The 'Attributes' list shows 'outlook' selected. The 'Selected attribute' table shows the distribution of 'outlook' values: sunny (5 instances, weight 5.0), overcast (4 instances, weight 4.0), and rainy (5 instances, weight 5.0). The 'Class: play (Nom)' is set to 'play (Nom)', and the 'Visualize All' button is visible. The status bar at the bottom shows 'OK' and a 'Log' button.

Weka Explorer

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

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

Filter: Choose **None** Apply

Current relation: Relation: weather, Instances: 14, Attributes: 5, Sum of weights: 14

Attributes: All | None | Invert | Pattern

No.	Name
1	<input checked="" type="checkbox"/> outlook
2	<input type="checkbox"/> temperature
3	<input type="checkbox"/> humidity
4	<input type="checkbox"/> windy
5	<input type="checkbox"/> play

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

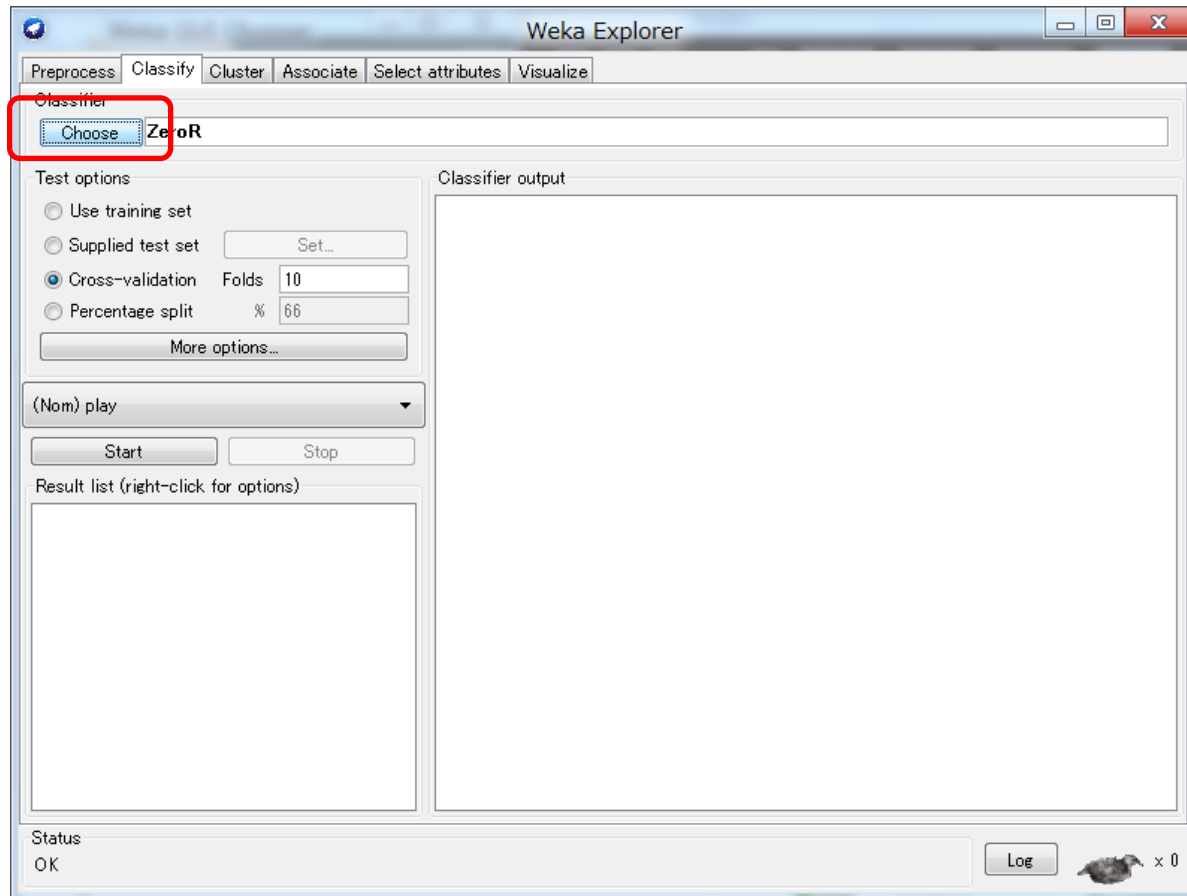
No.	Label	Count	Weight
1	sunny	5	5.0
2	overcast	4	4.0
3	rainy	5	5.0

Class: play (Nom) Visualize All

Status: OK Log x 0

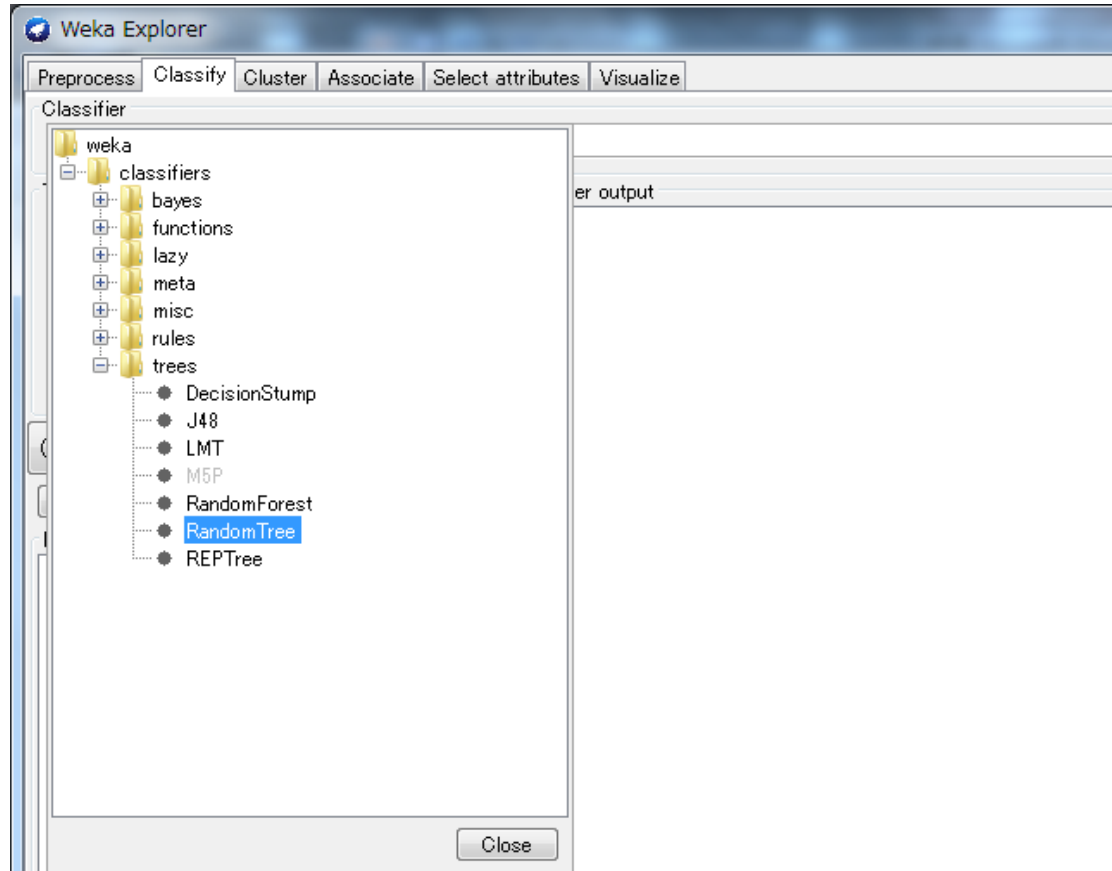
Wekaの実行

- Classifyタブを開き、Chooseを押します



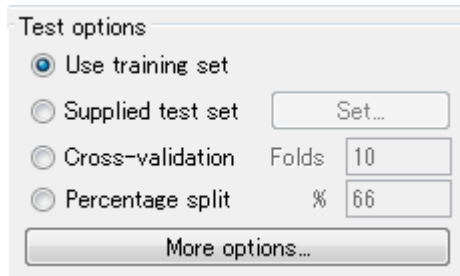
アルゴリズムの指定

- アルゴリズムの一覧が表示されるので、treesフォルダの中にあるRandom Treeを選択します



評価方法の指定

- 初期設定ではCross-validation(交差検定)になっていますが、Use training setに変更します(全てを学習データとする)



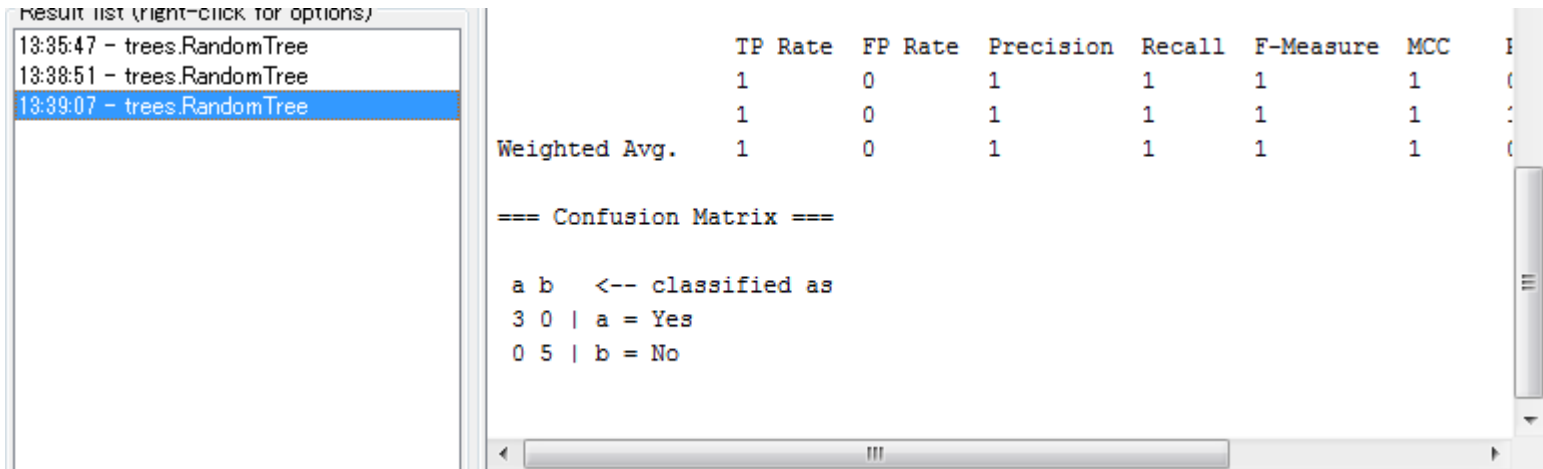
The image shows a 'Test options' dialog box with the following settings:

- Use training set
- Supplied test set (with a 'Set...' button)
- Cross-validation (with 'Folds' set to 10)
- Percentage split (with '%' set to 66)

At the bottom, there is a 'More options...' button.

Wekaの実行

- Startを押すと解析を実行します



The screenshot shows the 'Results' window in Weka. On the left, a list of results is shown, with the most recent one, '13:39:07 - trees.RandomTree', selected. The main area displays performance metrics for this model:

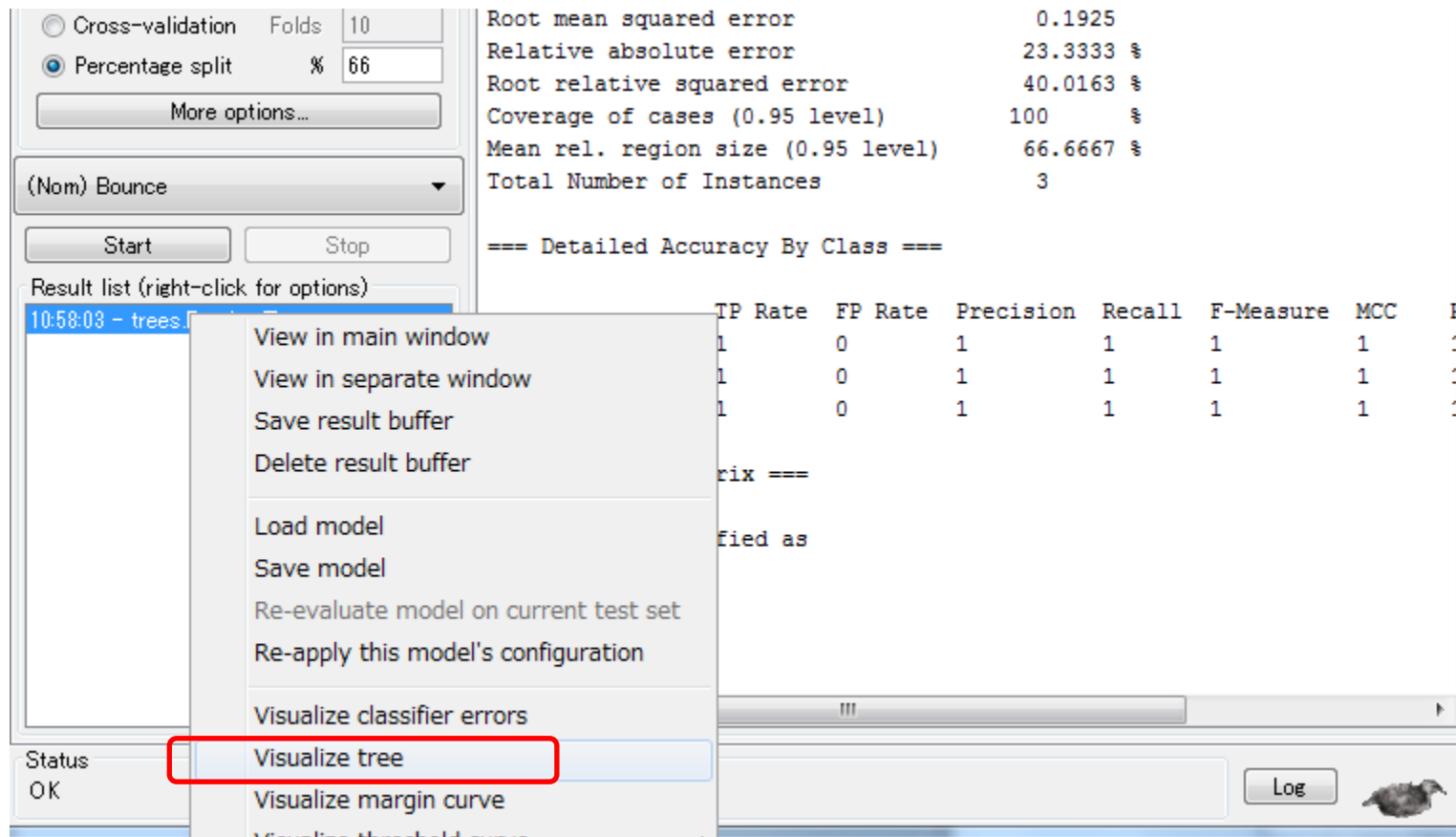
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	
	1	0	1	1	1	1	0
	1	0	1	1	1	1	0
Weighted Avg.	1	0	1	1	1	1	0

Below the metrics, a confusion matrix is displayed:

```
=== Confusion Matrix ===  
  
 a b  <-- classified as  
 3 0 | a = Yes  
 0 5 | b = No
```

Wekaの実行

- 結果が右側に表示されたら、結果のリストを右クリックし、Visualize treeを選択します



The screenshot shows the Weka GUI interface. On the left, the 'Cross-validation' section is set to 'Percentage split' with 'Folds' set to 10 and 'Percentage split' set to 66%. Below this, the '(Nom) Bounce' dropdown is visible, along with 'Start' and 'Stop' buttons. The 'Result list (right-click for options)' is open, showing a list of results. The first result, '10:58:03 - trees', is selected, and a context menu is displayed over it. The menu options include 'View in main window', 'View in separate window', 'Save result buffer', 'Delete result buffer', 'Load model', 'Save model', 'Re-evaluate model on current test set', 'Re-apply this model's configuration', 'Visualize classifier errors', 'Visualize tree' (highlighted with a red box), 'Visualize margin curve', and 'Visualize threshold curve'. The status bar at the bottom shows 'Status OK' and a 'Log' button.

Root mean squared error 0.1925
Relative absolute error 23.3333 %
Root relative squared error 40.0163 %
Coverage of cases (0.95 level) 100 %
Mean rel. region size (0.95 level) 66.6667 %
Total Number of Instances 3

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	
1	0	1	1	1	1	1	:
1	0	1	1	1	1	1	:
1	0	1	1	1	1	1	:

rix ===
fied as

Wekaの実行

- データのbounceにおける決定木が生成されました

RandomTree

=====

Size = Small

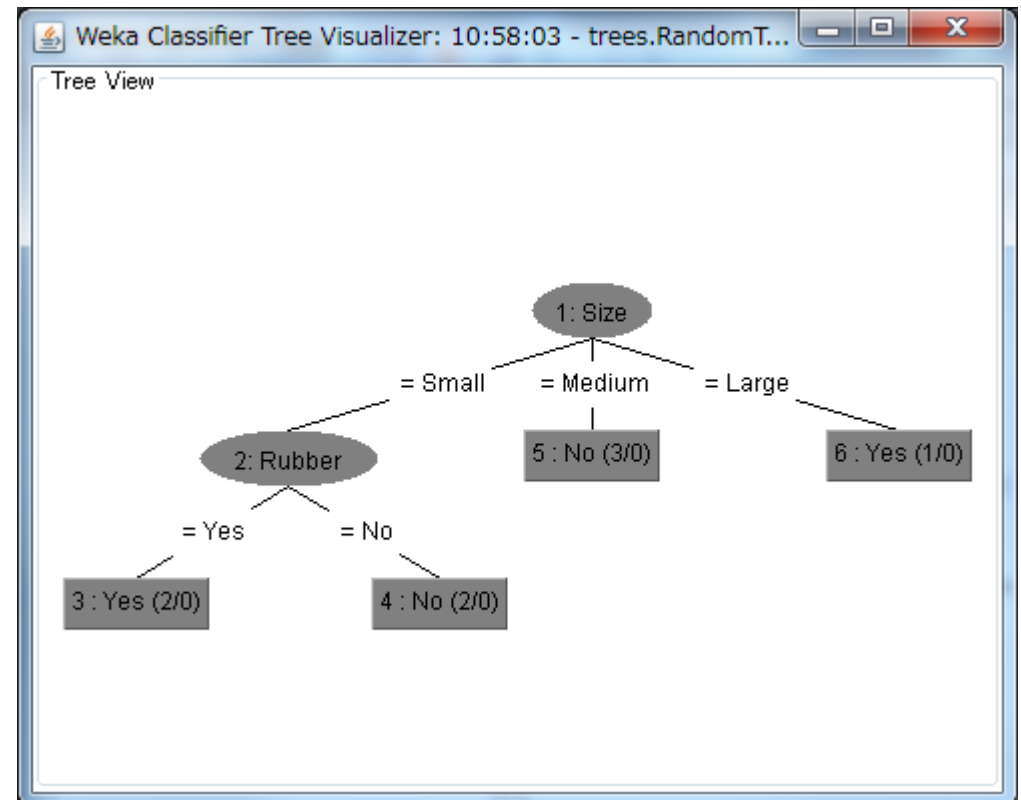
| Rubber = Yes : Yes (2/0)

| Rubber = No : No (2/0)

Size = Medium : No (3/0)

Size = Large : Yes (1/0)

Size of the tree : 6



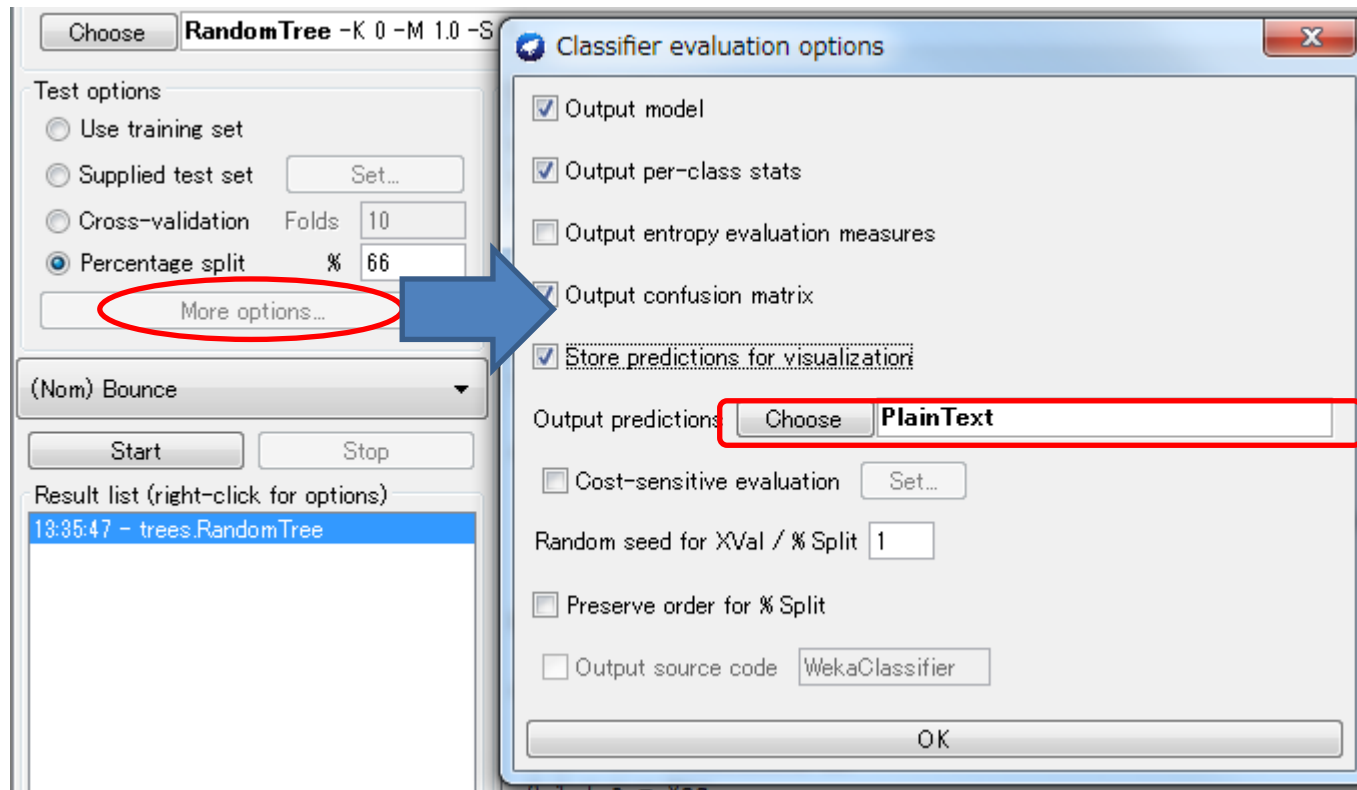
新しいデータの予測

- データセットに、不明なデータを追加します。
不明なデータは?と書く

```
@data↓  
Small,Green,Light,Yes,Yes↓  
Small,Blue,Medium,No,No↓  
Medium,Red,Medium,No,No↓  
Small,Red,Medium,Yes,Yes↓  
Large,Green,Heavy,Yes,Yes↓  
Medium,Blue,Heavy,Yes,No↓  
Medium,Green,Heavy,Yes,No↓  
Small,Red,Light,No,No↓  
Small,Blue,Light,Yes,?←
```

設定の追加

- More Optionsを開き、Output predictionsをPlainTextに変えます



予測結果

- マイニングを実行すると、不明な部分がYesと予測されました

```
=== Predictions on training set ===  
  
inst#   actual   predicted error prediction  
1      1:Yes   1:Yes     0           1  
2      2:No    2:No     0           1  
3      2:No    2:No     0           1  
4      1:Yes   1:Yes     0           1  
5      1:Yes   1:Yes     0           1  
6      2:No    2:No     0           1  
7      2:No    2:No     0           1  
8      2:No    2:No     0           1  
9      1:?     1:Yes    1           1
```