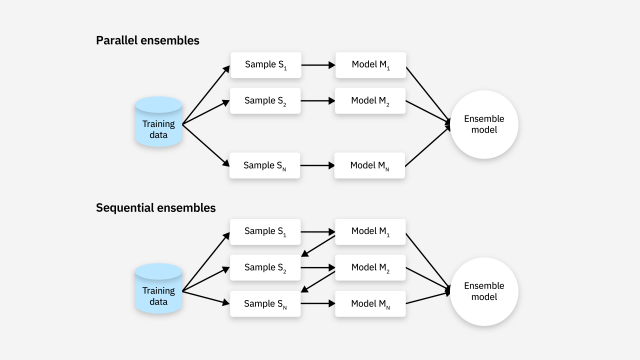
Ensemble learning

Ensemble learning is a [machine learning](https://www.ibm.com/topics/machine-learning) technique that aggregates two or more learners (e.g. [regression](https://www.ibm.com/topics/linear-regression) models, [neural networks](https://www.ibm.com/topics/neural-networks)) in order to produce better predictions.

Types of ensemble models

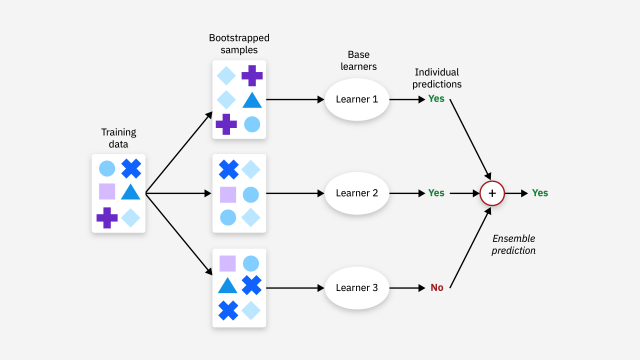
-**Parallel**methods train each base learner apart from the others of the others. Per its name, then, parallel ensembles train base learners in parallel and independent of one another.

- **Sequential** methods train a new base learner so that it minimizes errors made by the previous model trained in the preceding step. In other words, sequential methods construct base models sequentially in stages.



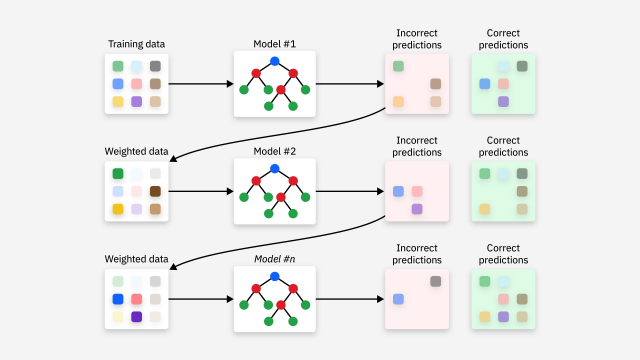
Ensemble learning techniques.

Bagging: [Bagging](https://www.ibm.com/topics/bagging) is a homogenous parallel method sometimes called *bootstrap aggregating*. It uses modified replicates of a given training data set to train multiple base learners with the same training algorithm.



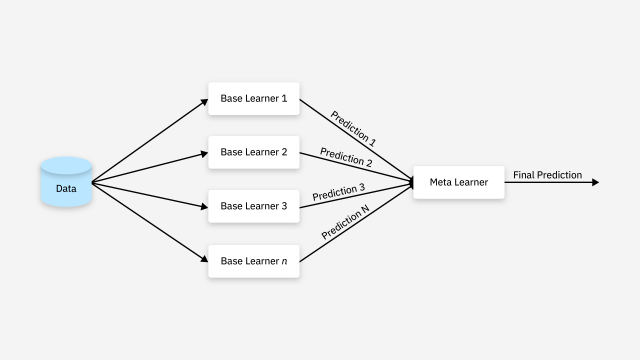
Boosting:

[Boosting](https://www.ibm.com/topics/boosting) algorithms are a sequential ensemble method. Boosting has many variations, but they all follow the same general procedure. Boosting trains a learner on some initial dataset, *d*. The resultant learner is typically weak, misclassifying many samples in the dataset. Much like bagging, boosting then samples instances from the initial dataset to create a new dataset (*d2*). Unlike bagging, however, boosting prioritizes misclassified data instances from the first model or learner. A new learner is trained on this new dataset *d2*. Then a third dataset (*d3*) is then compiled from *d1* and *d2*, prioritizes the second learner’s misclassified samples and instances in which *d1* and *d2* disagree. The process repeats *n* times to produce *n* learners. Boosting then combines and weights the all the learners together to produce final predictions.



**Stacking**

Stacking, or stacked generalization, is a heterogenous parallel method that exemplifies what is known as meta-learning. Meta-learning consists of training a meta-learner from the output of multiple base learners. Stacking specifically trains several base learners from the same dataset using a different training algorithm for each learner. Each base learner makes predictions on an unseen dataset. These first model predictions are then compiled and used to train a final model, being the meta-model.



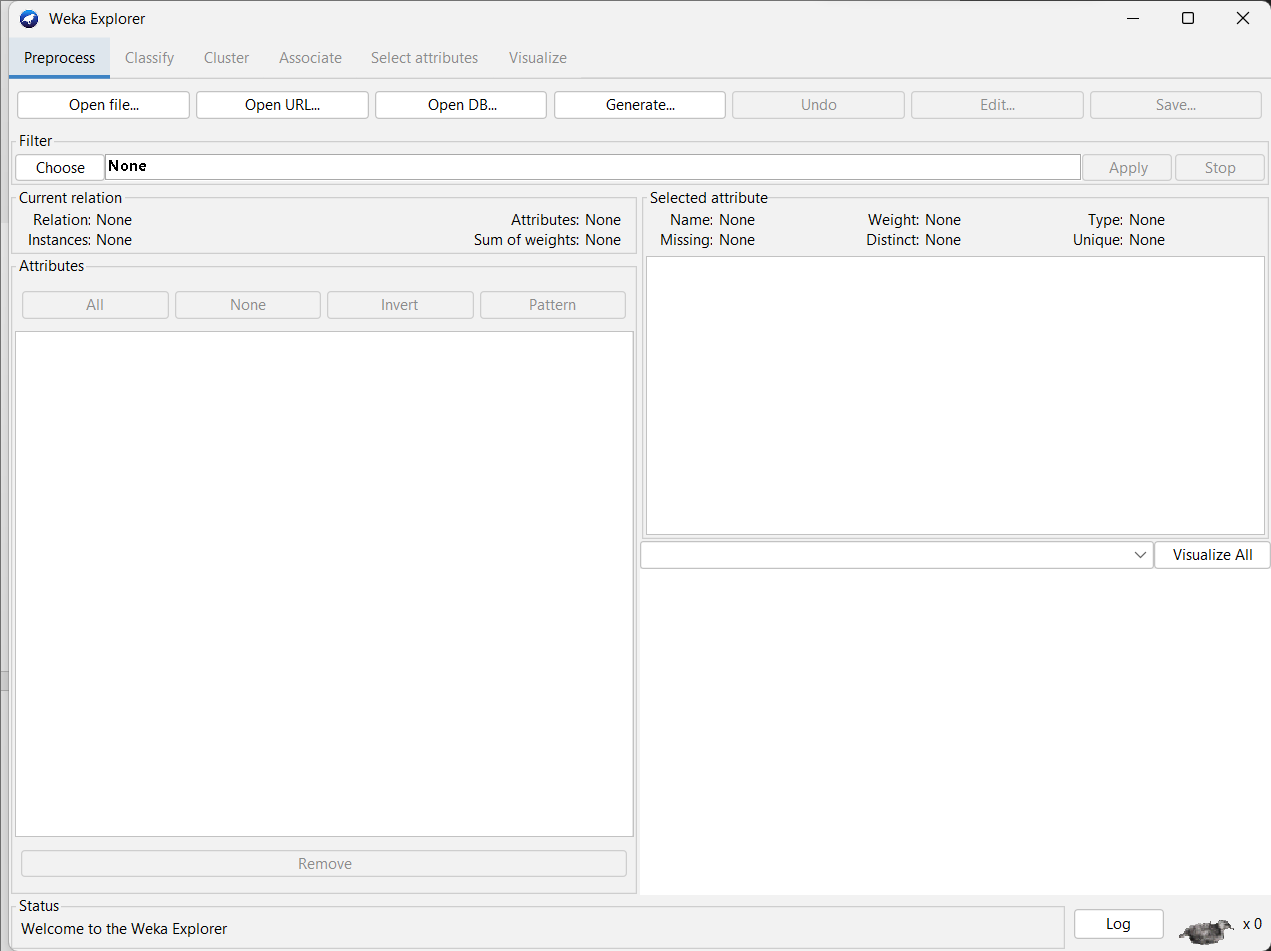
Ensemble model in weka

**1. Bagging (Bootstrap Aggregating)**

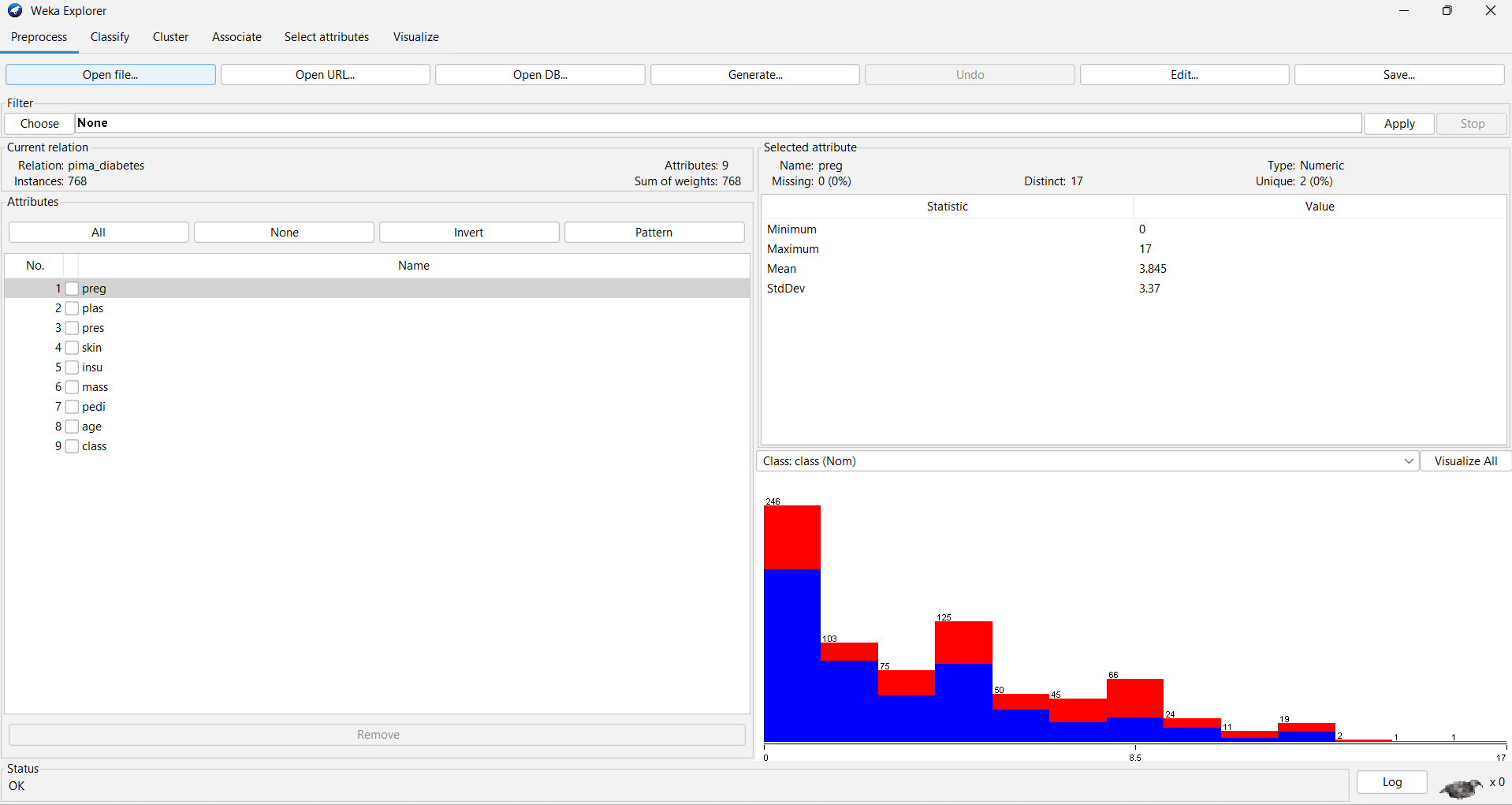
**Bagging** involves training multiple versions of a model on different subsets of the data (created by bootstrapping) and then aggregating their predictions.

**Steps:**

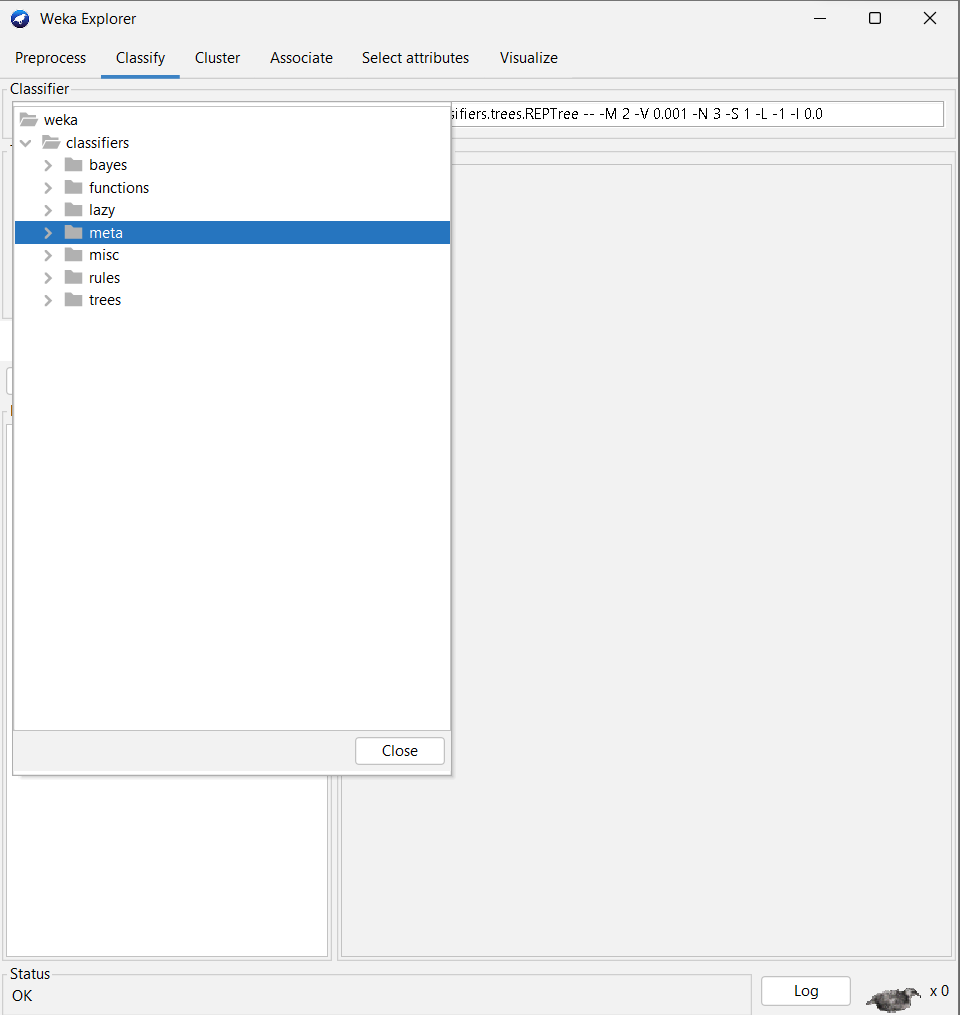
1. **Open WEKA** and go to the Explorer interface.



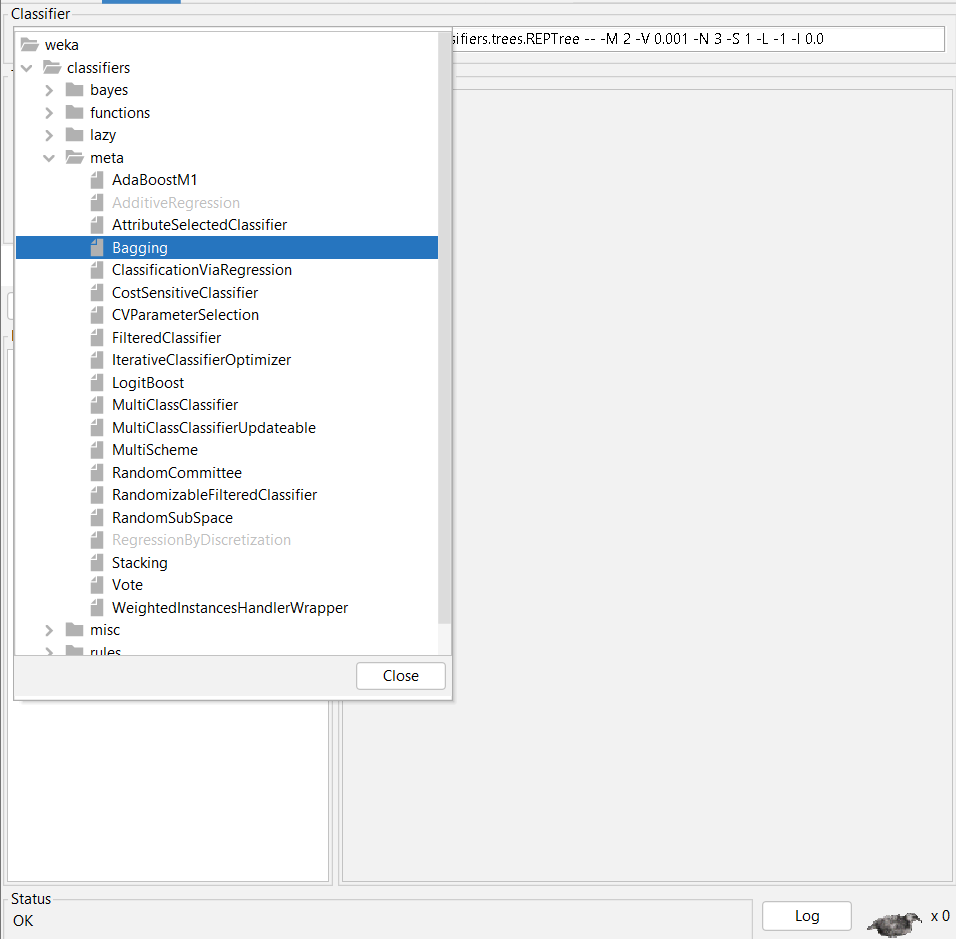
1. **Load your dataset** by clicking on the Open file button and selecting your data file.



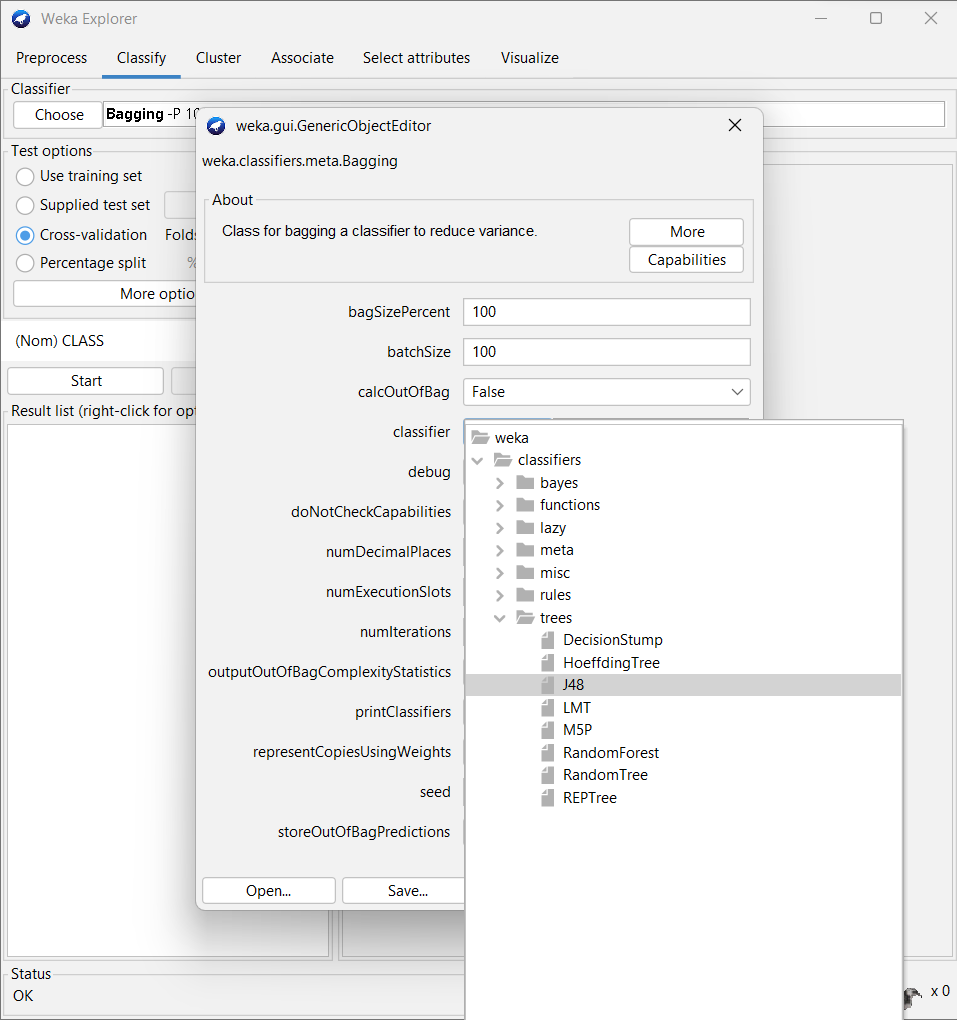
1. **Go to the Classify tab**.



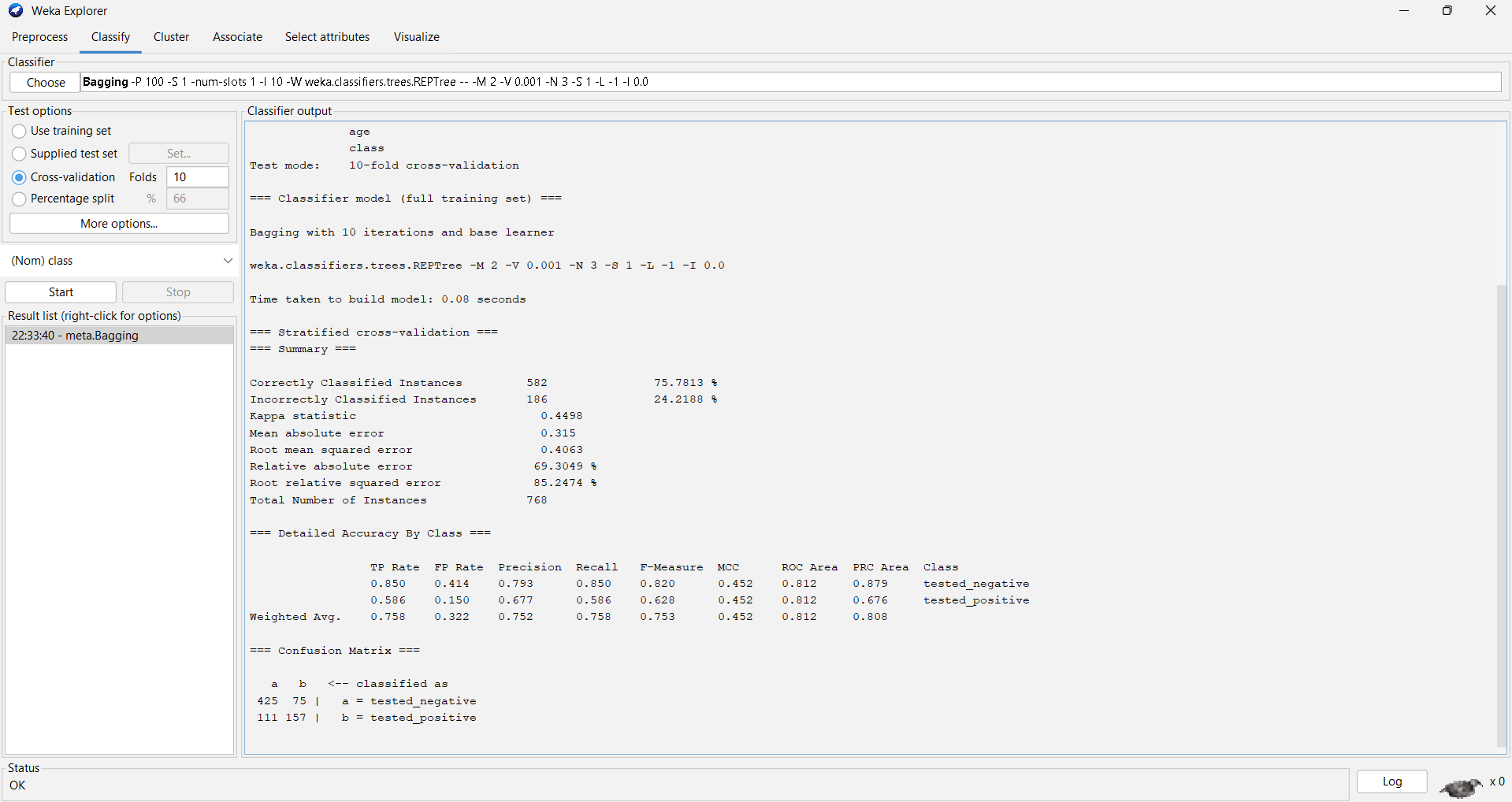
1. **Choose the Bagging algorithm:**
   * Click on the Choose button in the Classifier section.
   * Navigate to meta and select Bagging.

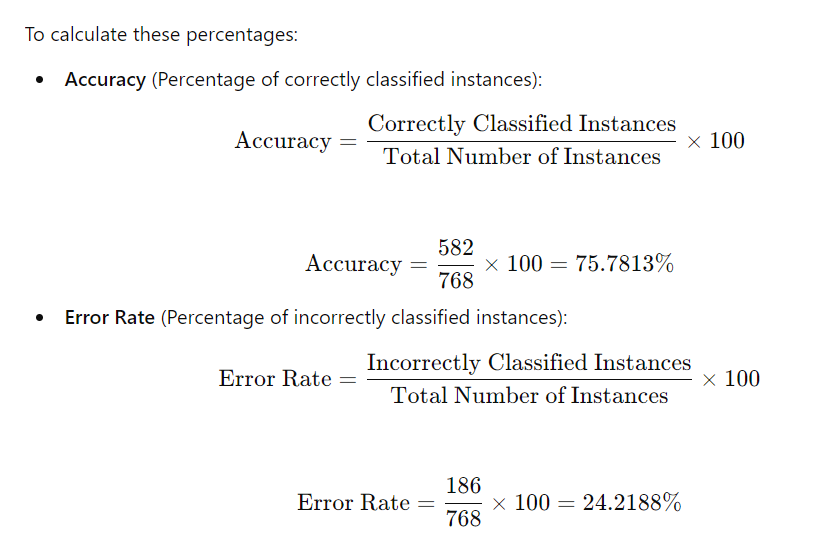


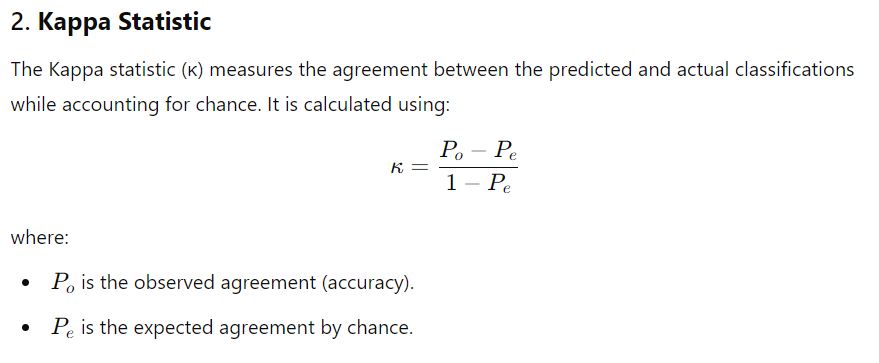
1. **Configure Bagging:**
   * Click on the Bagging entry in the Classifier list to open its configuration window.
   * Set the baselearner to the model you want to use (e.g., trees.J48 for a Decision Tree).
   * Adjust other parameters if needed (e.g., the number of iterations).



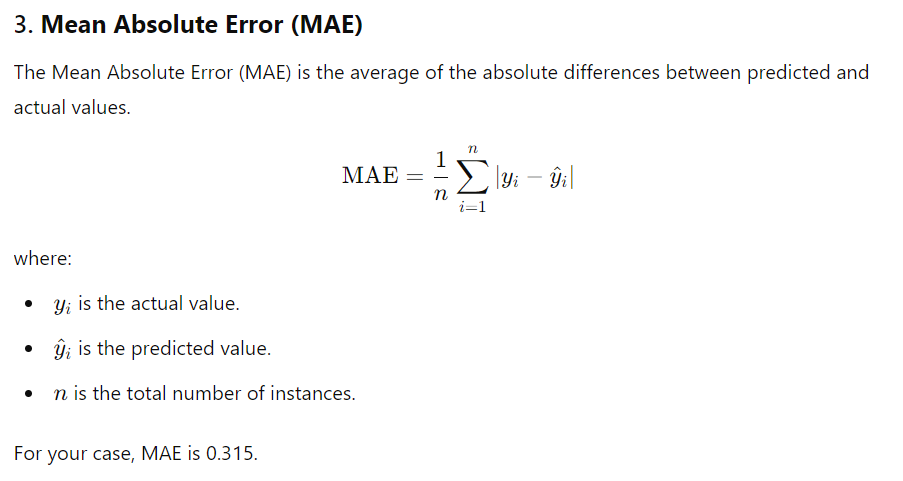
1. **Run the model** by clicking the Start button.

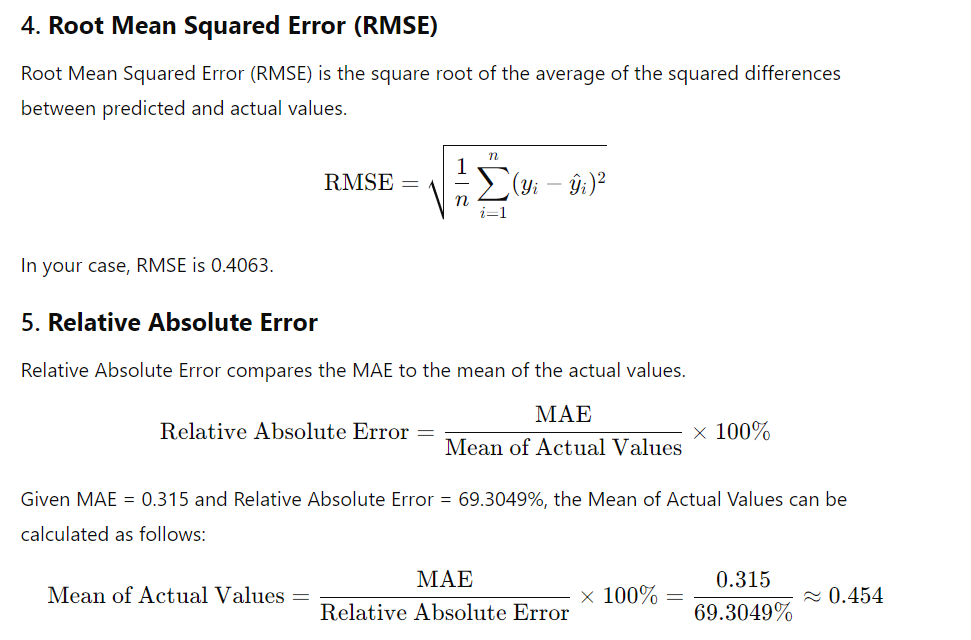


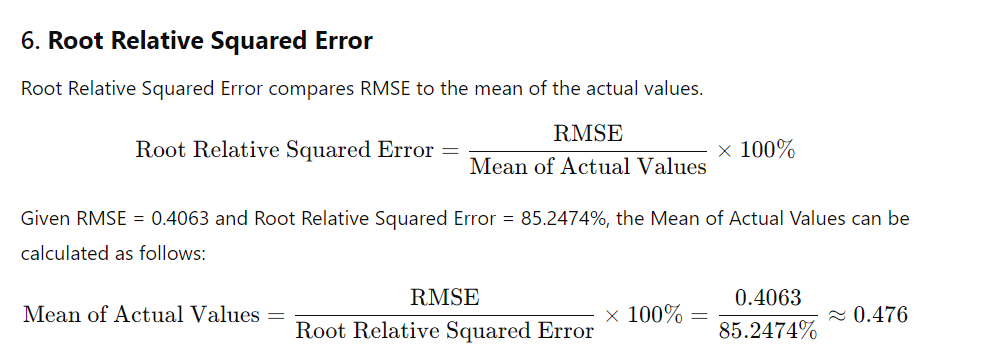




κ=0.4498.





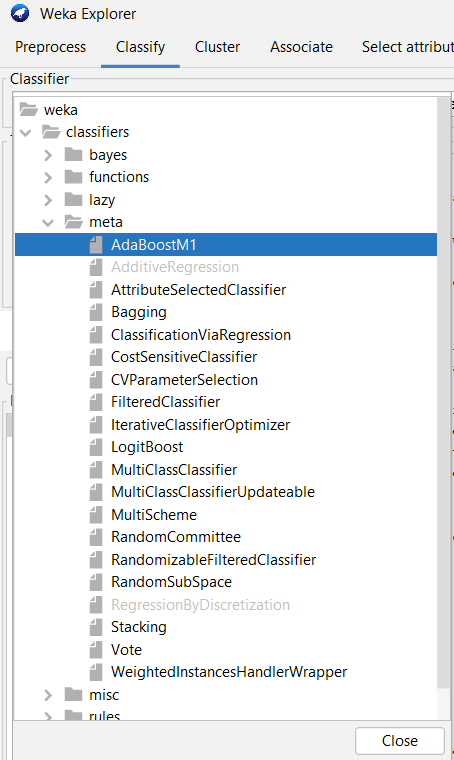


**2. Boosting**

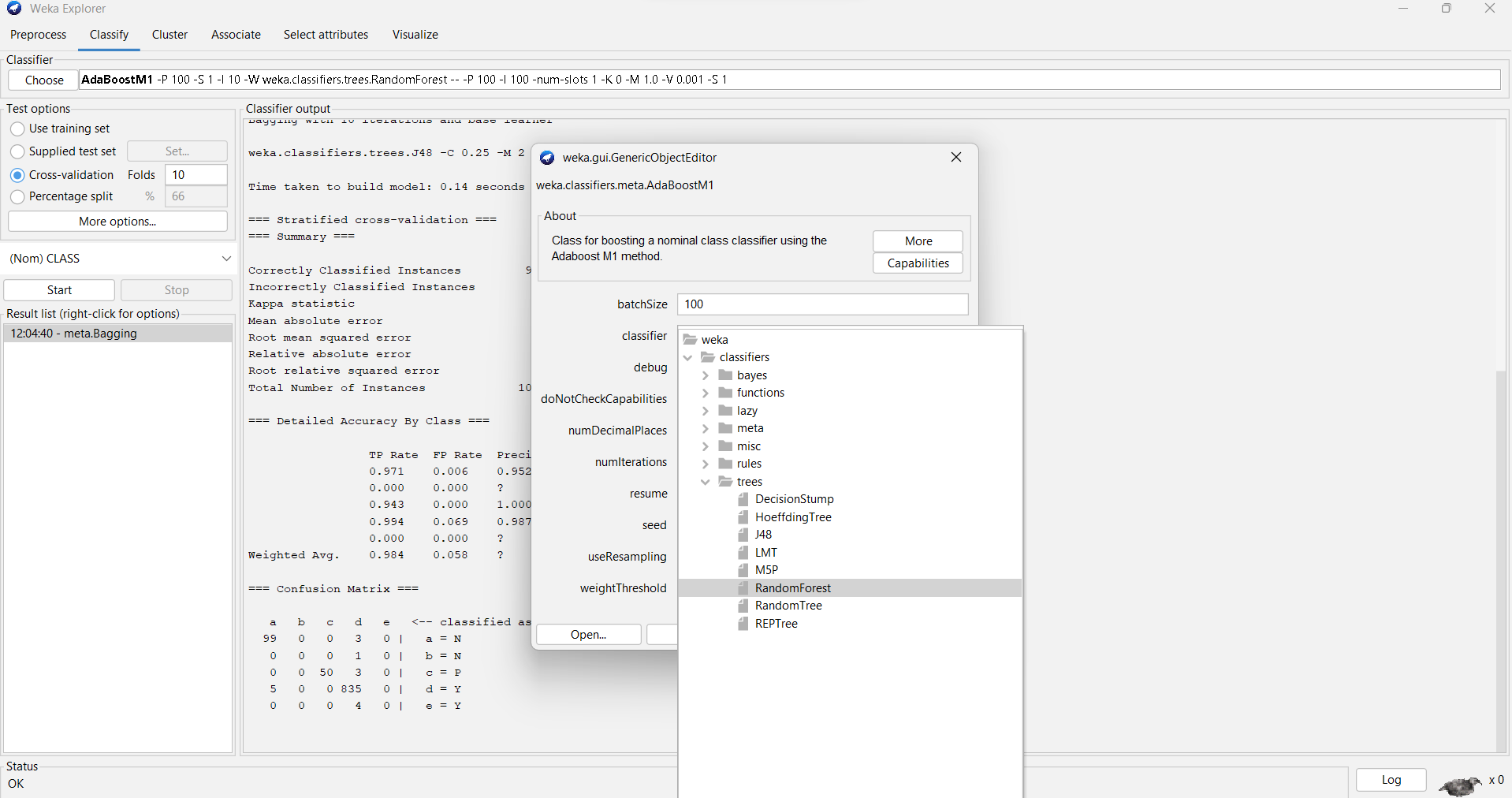
**Boosting** involves training a sequence of models where each model tries to correct the errors of the previous ones. WEKA uses AdaBoostM1 for boosting.

**Steps:**

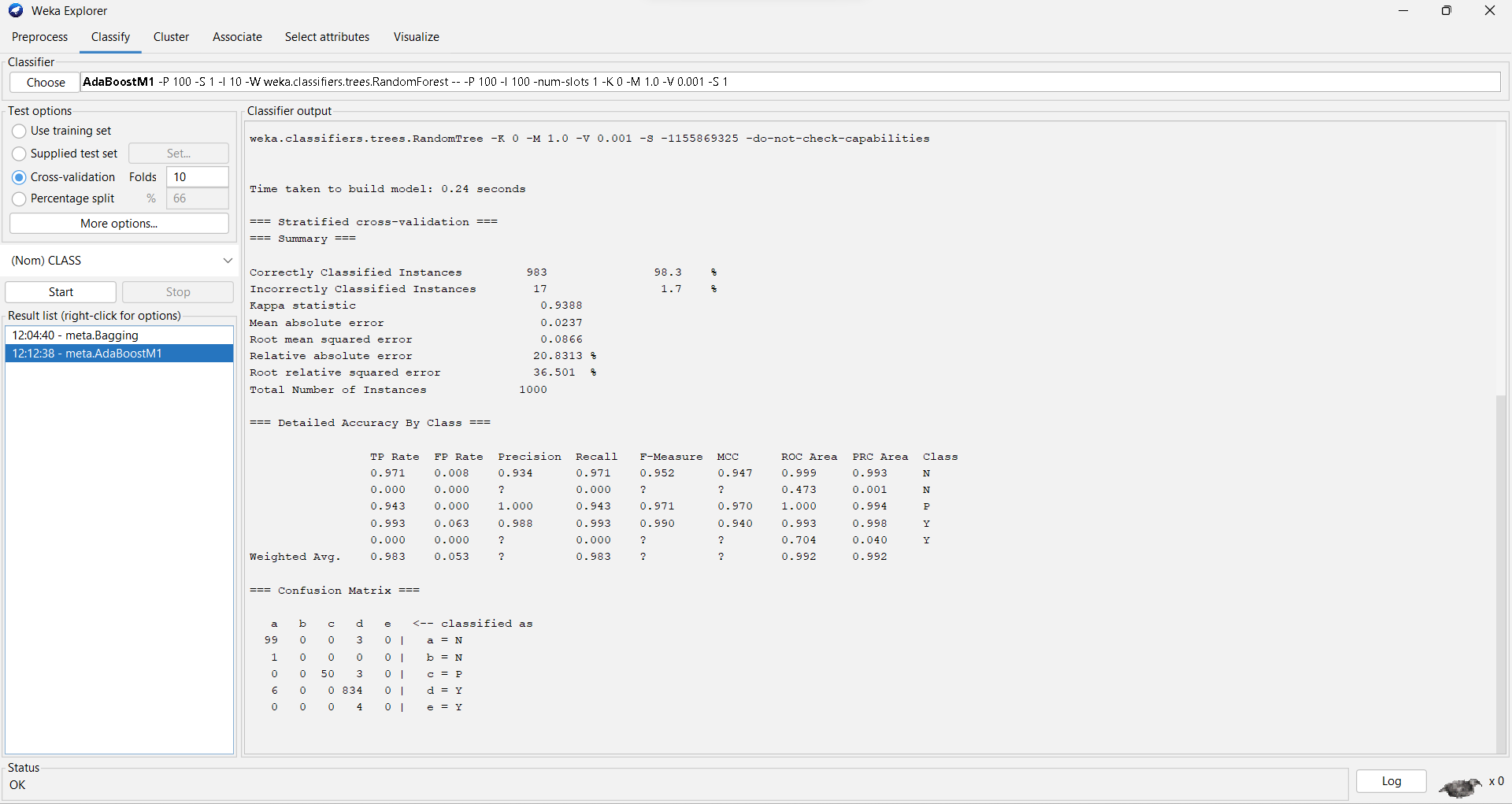
1. **Open WEKA** and navigate to the Explorer interface.
2. **Load your dataset** by clicking the Open file button.
3. **Go to the Classify tab**.
4. **Choose the AdaBoostM1 algorithm:**
   * Click the Choose button in the Classifier section.
   * Go to meta and select AdaBoostM1.



1. **Configure AdaBoostM1:**
   * Click on the AdaBoostM1 entry to open its configuration window.
   * Set the baseLearner to your chosen model (e.g., trees.J48).
   * Adjust the NumIterations parameter to control the number of boosting rounds.



1. **Run the model** by clicking the Start button.

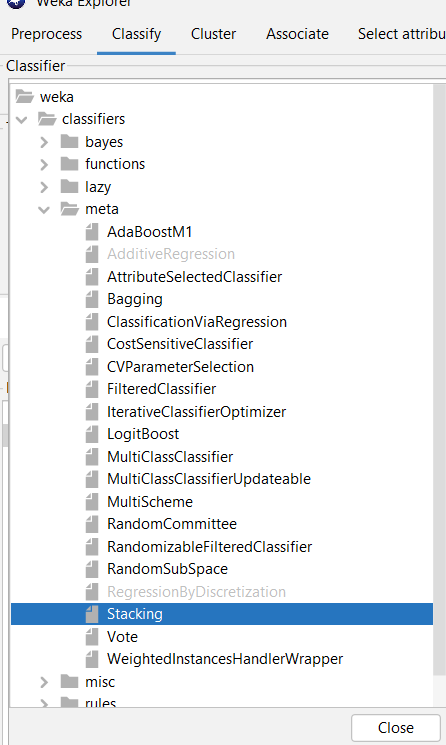


**3. Stacking**

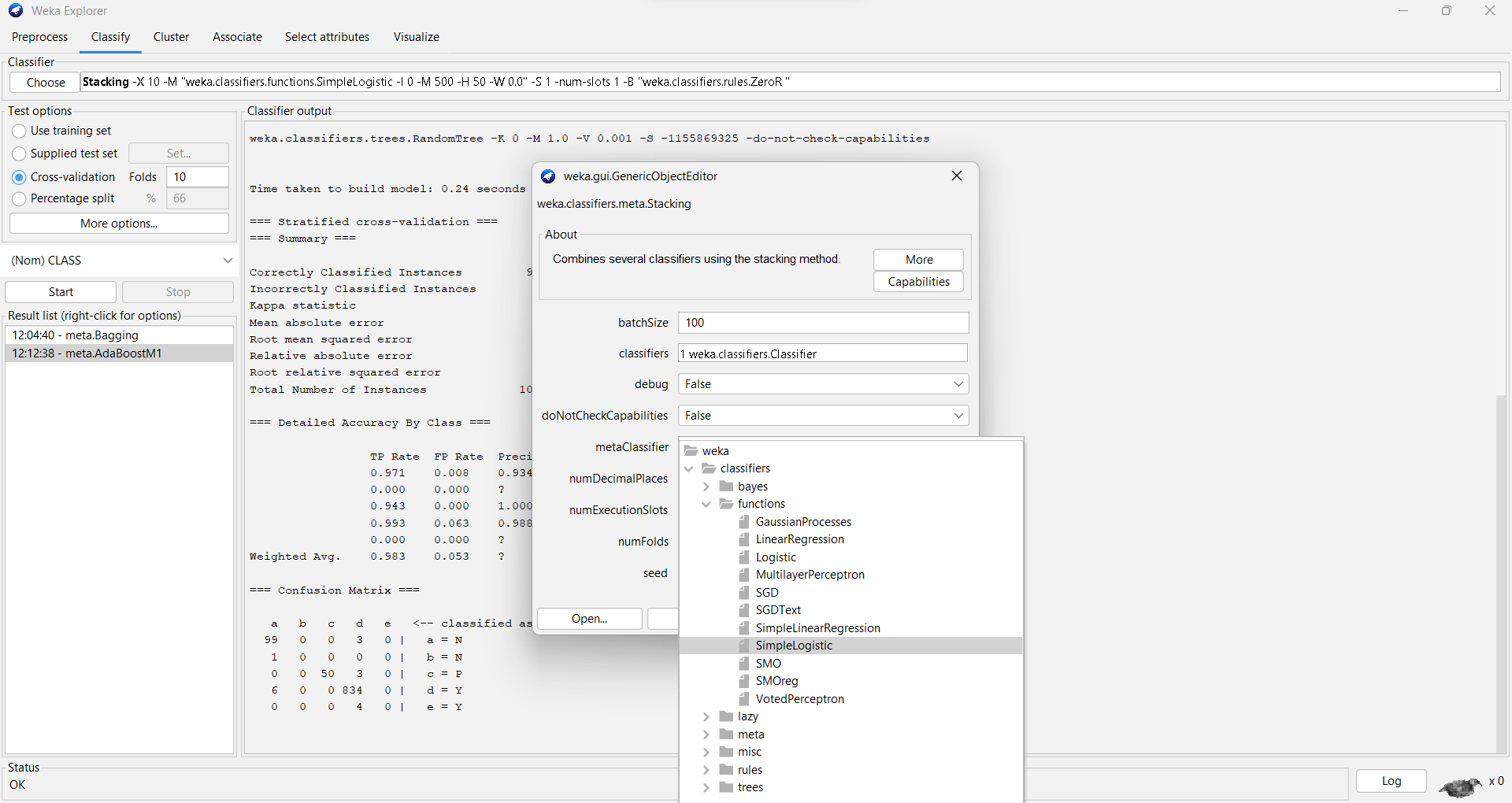
**Stacking** combines predictions from multiple base classifiers using a meta-classifier to improve overall performance.

**Steps:**

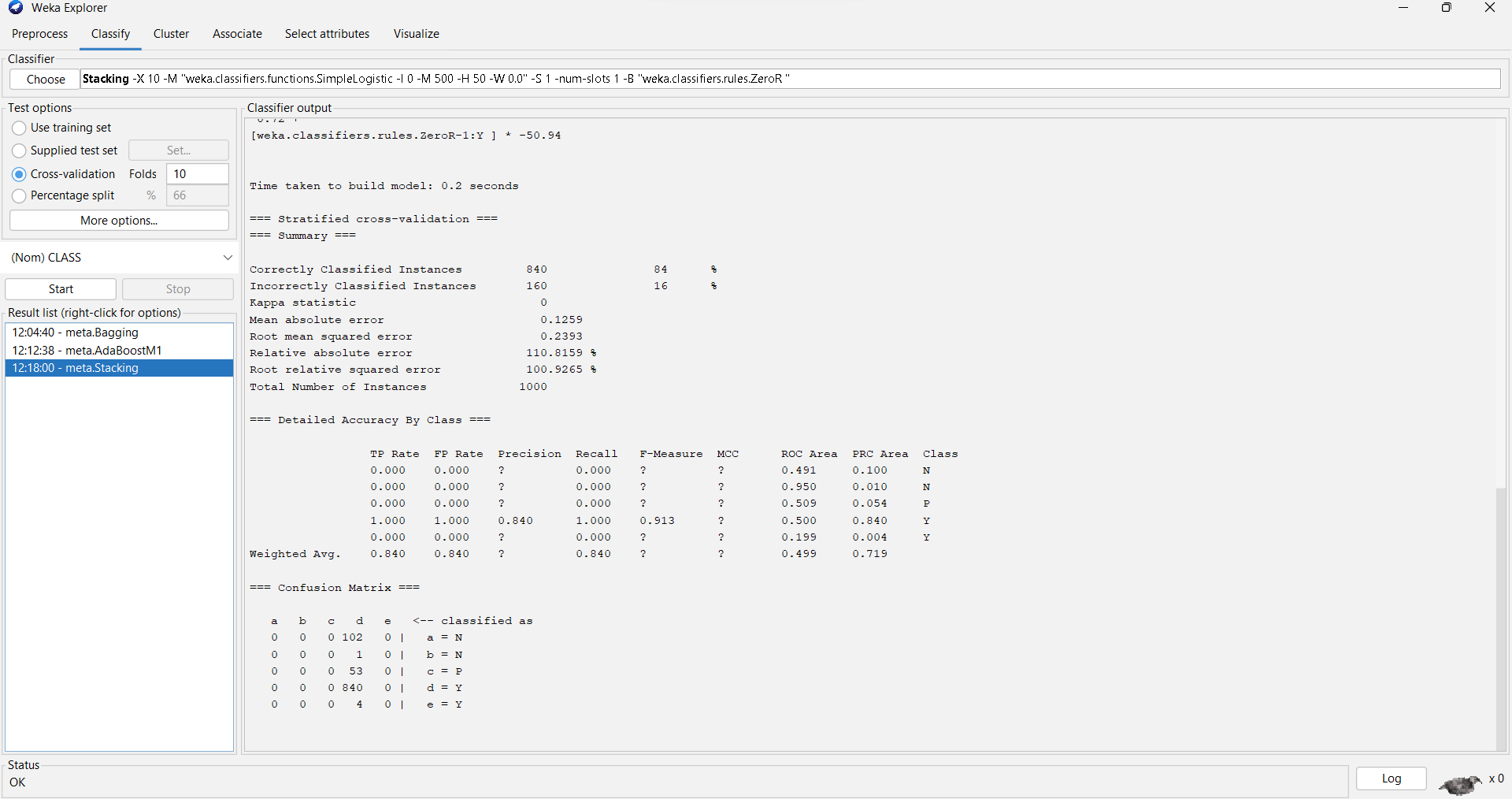
1. **Open WEKA** and go to the Explorer interface.
2. **Load your dataset** by clicking on the Open file button.
3. **Go to the Classify tab**.
4. **Choose the Stacking algorithm:**
   * Click the Choose button in the Classifier section.
   * Navigate to meta and select Stacking.



1. **Configure Stacking:**
   * Click on the Stacking entry to open its configuration window.
   * Define the classifiers (base models) to be used in the ensemble.
   * Choose a metaClassifier that will combine the predictions of the base models (e.g., functions.Logistic).



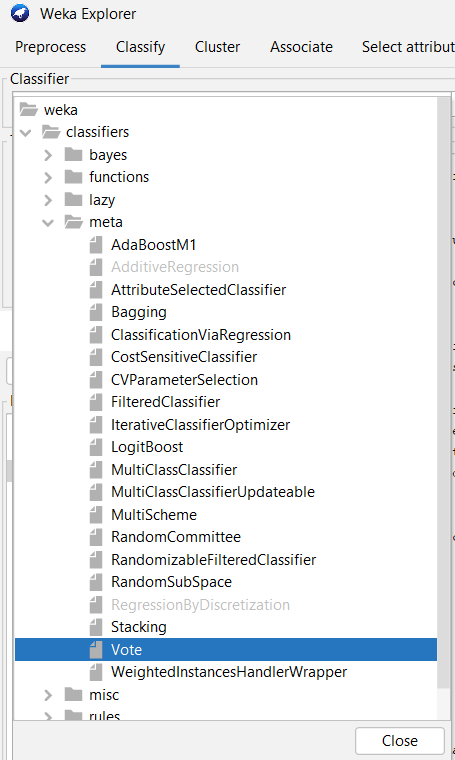
1. **Run the model** by clicking the Start button.

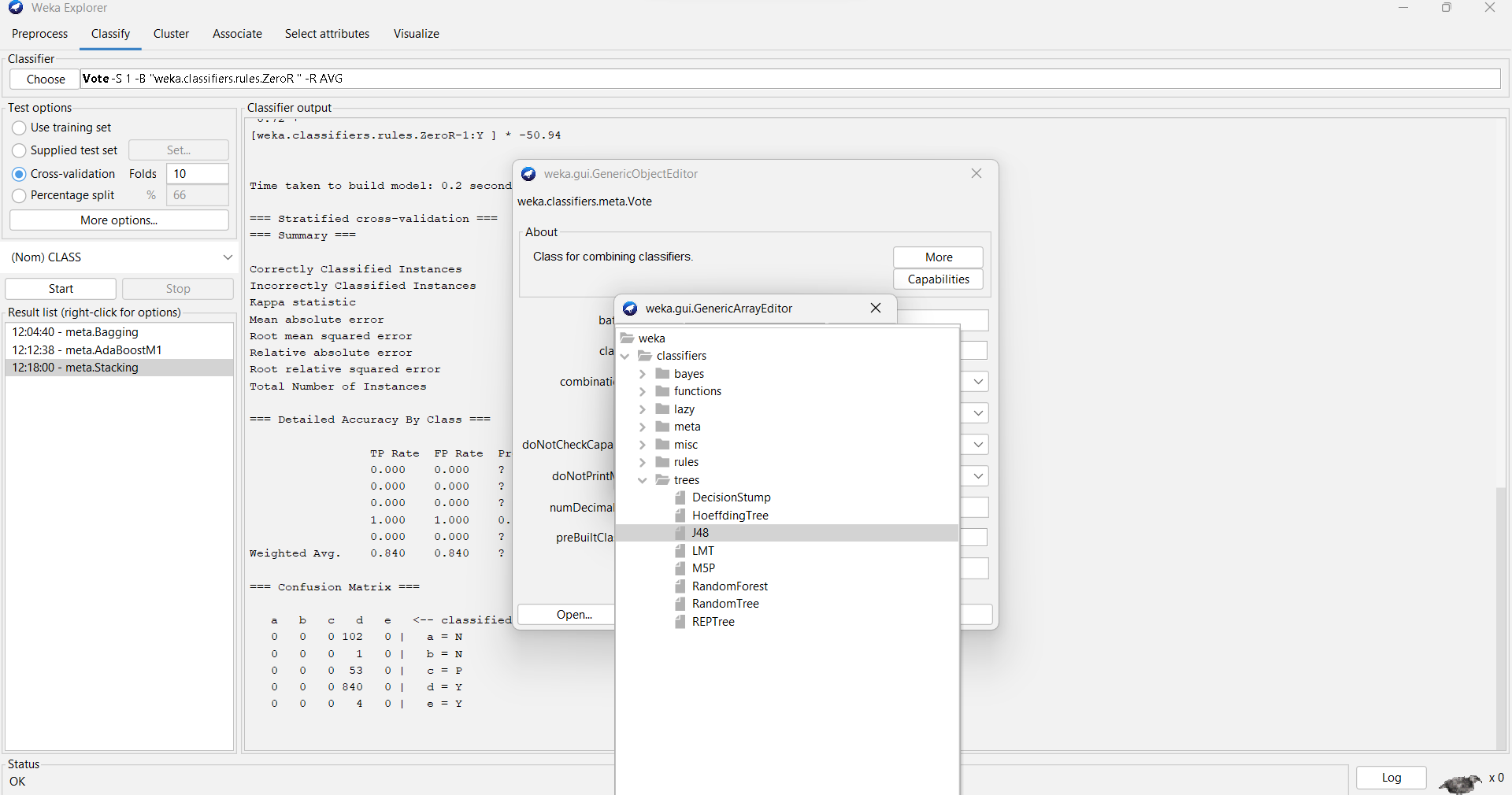


**4. Voting**

**Voting** involves combining the predictions of multiple models by majority voting (for classification) or averaging (for regression).

**Steps:**

1. **Open WEKA** and navigate to the Explorer interface.
2. **Load your dataset** by clicking on the Open file button.
3. **Go to the Classify tab**.
4. **Choose the Voting algorithm:**
   * Click the Choose button in the Classifier section.
   * Go to meta and select Vote.
   * 
5. **Configure Voting:**
   * Click on the Vote entry to open its configuration window.
   * Add the classifiers you want to use for voting (e.g., trees.J48, functions.SMO).



1. **Run the model** by clicking the Start button.

