AdaBoost

An ensemble meta-algorithm that combines weak learners and adapts to the 'hardness' of each training sample.

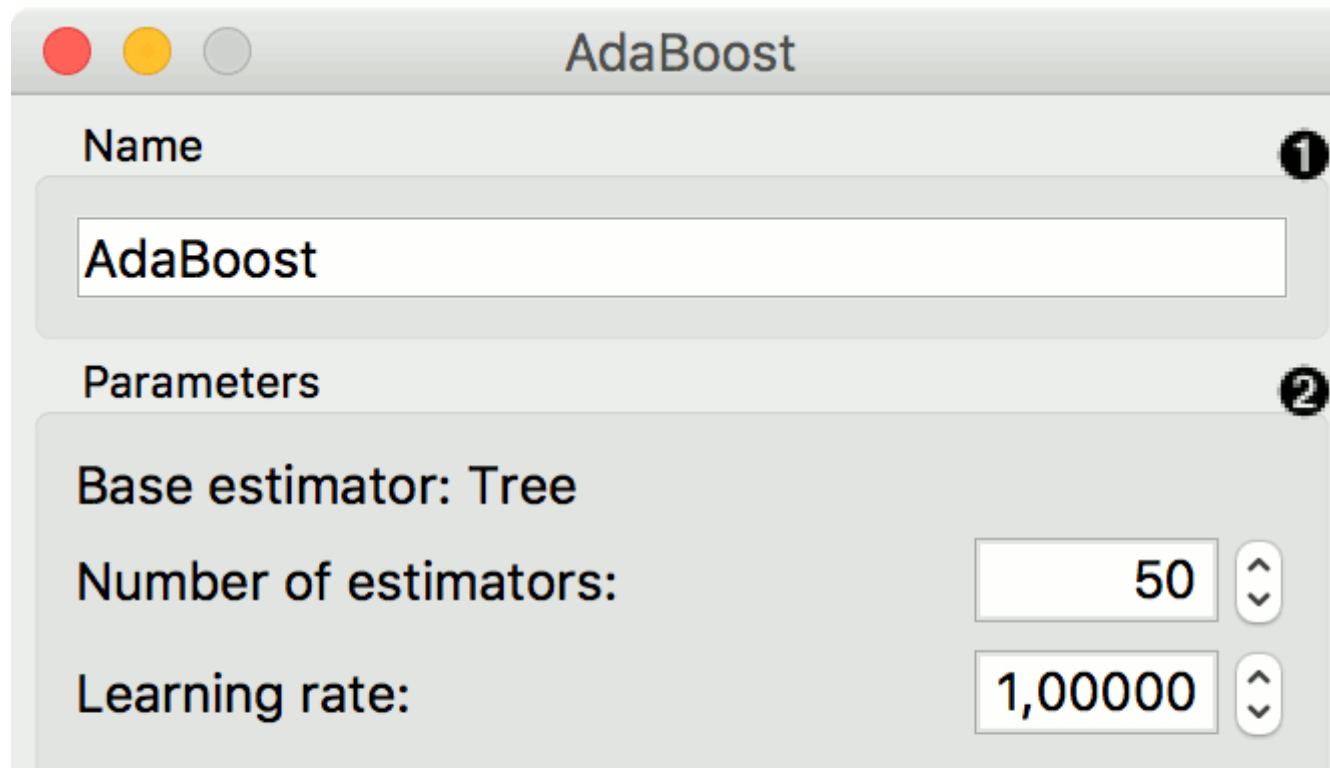**Inputs**

- Data: input dataset
- Preprocessor: preprocessing method(s)
- Learner: learning algorithm

Outputs

- Learner: AdaBoost learning algorithm
- Model: trained model

The AdaBoost (short for "Adaptive boosting") widget is a machine-learning algorithm, formulated by Yoav Freund and Robert Schapire. It can be used with other learning algorithms to boost their performance. It does so by tweaking the weak learners.

AdaBoost works for both classification and regression.

Fixed seed for random generator: 0

Boosting method   ❸

Classification algorithm: SAMME.R

Regression loss function: Linear

❹

❺

Report   ☑   Apply Automatically

1. The learner can be given a name under which it will appear in other widgets. The default name is "AdaBoost".
2. Set the parameters. The base estimator is a tree and you can set:
   - *Number of estimators*
   - *Learning rate*: it determines to what extent the newly acquired information will override the old information (0 = the agent will not learn anything, 1 = the agent considers only the most recent information)
   - *Fixed seed for random generator*: set a fixed seed to enable reproducing the results.
3. Boosting method.
   - *Classification algorithm* (if classification on input): SAMME (updates base estimator's weights with classification results) or SAMME.R (updates base estimator's weight with probability estimates).
   - *Regression loss function* (if regression on input): Linear (), Square (), Exponential ().
4. Produce a report.
5. Click *Apply* after changing the settings. That will put the new learner in the output and, if the training examples are given, construct a new model and output it as well. To communicate changes automatically tick *Apply Automatically*.

Examples

For classification, we loaded the *iris* dataset. We used *AdaBoost*, Tree and Logistic Regression and evaluated the models' performance in Test & Score.

AdaBoost

Name

AdaBoost

Parameters

Base estimator: Tree

Number of estimators:	50
Learning rate:	1,00000
☐ Fixed seed for random generator:	26

Boosting method

| Classification algorithm: | SAMME.R |
| Regression loss function: | Linear |

Test & Score

Sampling

◉ Cross validation

  Number of folds: 10

  ☑ Stratified

◯ Random sampling

  Repeat train/test: 10

  Training set size: 66 %

  ☑ Stratified

◯ Leave one out

◯ Test on train data

Evaluation Results

Method ▼	AUC	CA	F1	Precision	Recall
AdaBoost	0.965	0.953	0.953	0.953	0.953
Tree	0.975	0.960	0.960	0.960	0.960
Logistic Regression	0.990	0.960	0.960	0.962	0.960
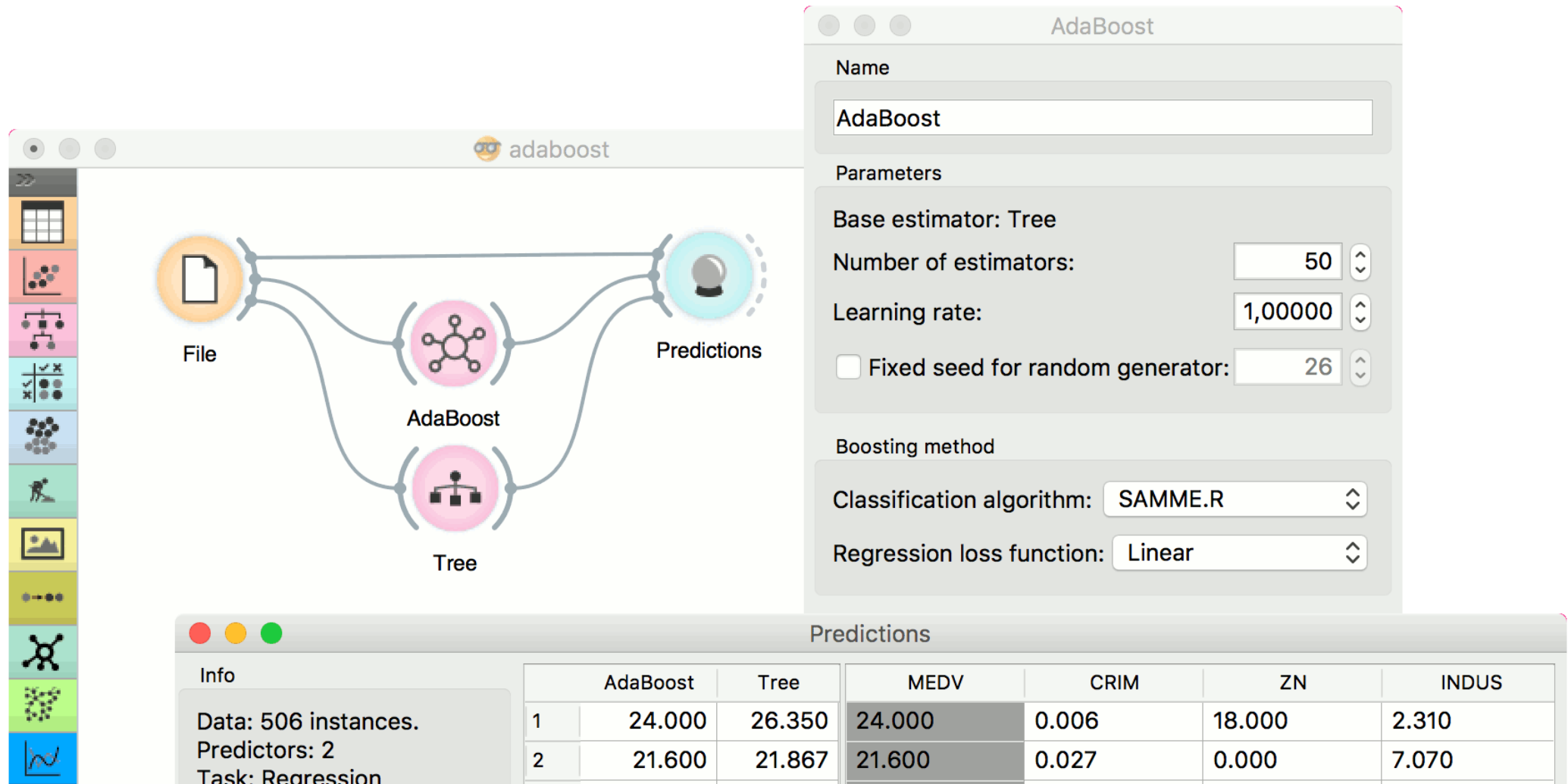
For regression, we loaded the *housing* dataset, sent the data instances to two different models (**AdaBoost** and Tree) and output them to the Predictions widget.

Restore Original Order

Data View

☑ Show full data set

Output

☑ Original data

☑ Predictions

☑ Probabilities

Report

3	34.700	34.800	34.700	0.027	0.000	7.070
4	33.400	33.200	33.400	0.032	0.000	2.180
5	36.100	37.150	36.200	0.069	0.000	2.180
6	28.700	28.900	28.700	0.030	0.000	2.180
7	22.600	22.300	22.900	0.088	12.500	7.870
8	27.100	22.100	27.100	0.145	12.500	7.870
9	16.500	15.475	16.500	0.211	12.500	7.870
10	18.900	18.350	18.900	0.170	12.500	7.870
11	15.000	15.475	15.000	0.225	12.500	7.870
12	18.900	19.167	18.900	0.117	12.500	7.870
13	21.700	22.425	21.700	0.094	12.500	7.870