

AdaBoost

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

Inputs

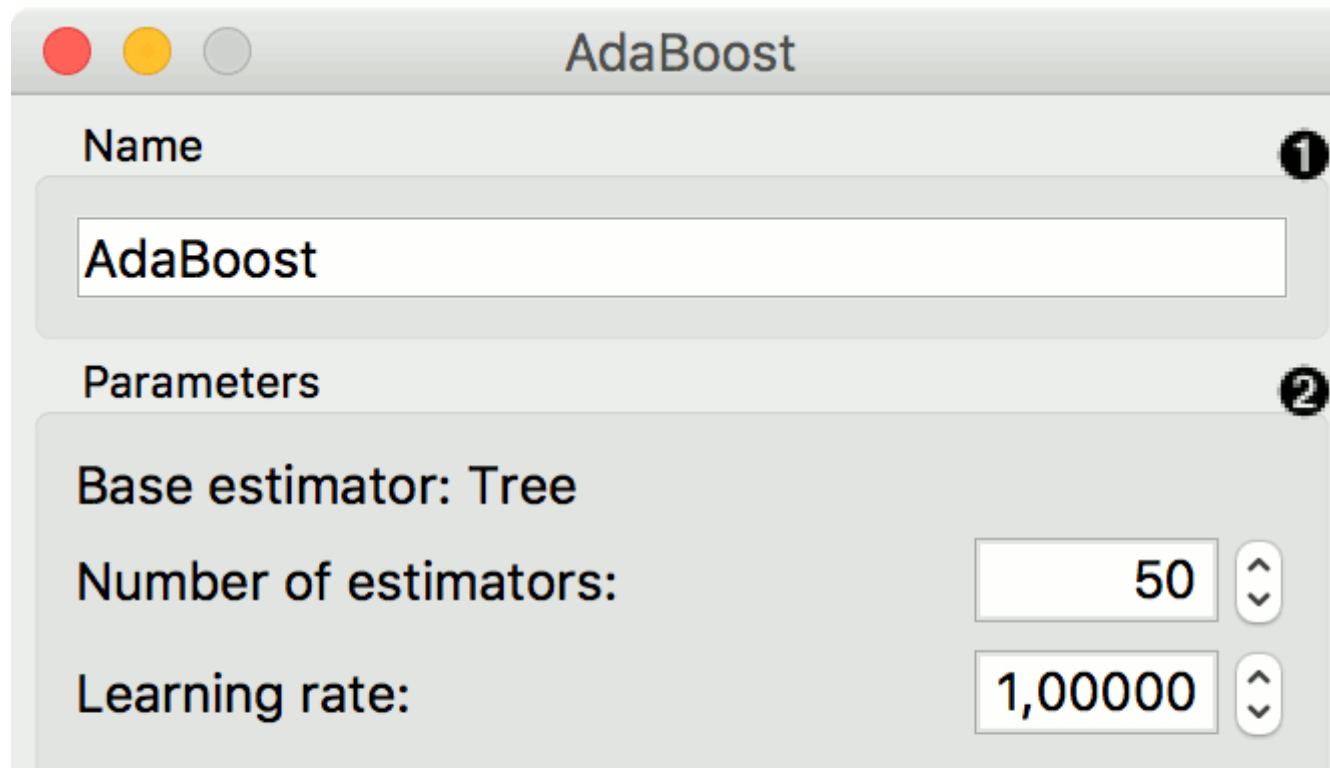
- Data: input dataset
- Preprocessor: preprocessing method(s)
- Learner: learning algorithm

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.



The screenshot shows the 'AdaBoost' widget interface. It has a title bar with three colored buttons (red, yellow, grey) and the text 'AdaBoost'. Below the title bar, there are two main sections: 'Name' and 'Parameters'. The 'Name' section has a text box containing 'AdaBoost'. The 'Parameters' section has three rows: 'Base estimator: Tree', 'Number of estimators: 50', and 'Learning rate: 1,00000'. Each row has a corresponding input field or a set of up/down arrows for adjustment. The 'Number of estimators' and 'Learning rate' fields have up/down arrows on the right side.

☐ Fixed seed for random generator:

Boosting method 3

Classification algorithm:

Regression loss function:

4 5

☒

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*.

The screenshot displays the Orange Data Mining software interface. On the left, a workflow is visible with a 'File' widget connected to three model widgets: 'AdaBoost', 'Tree', and 'Logistic Regression'. These three models are then connected to a 'Test & Score' widget. A vertical toolbar on the far left contains various data mining widgets. On the right, the 'AdaBoost' widget's configuration window is open, showing the following settings:

- 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

Below the configuration window, the 'Test & Score' widget's results are shown. It includes a 'Sampling' section and an 'Evaluation Results' table.

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

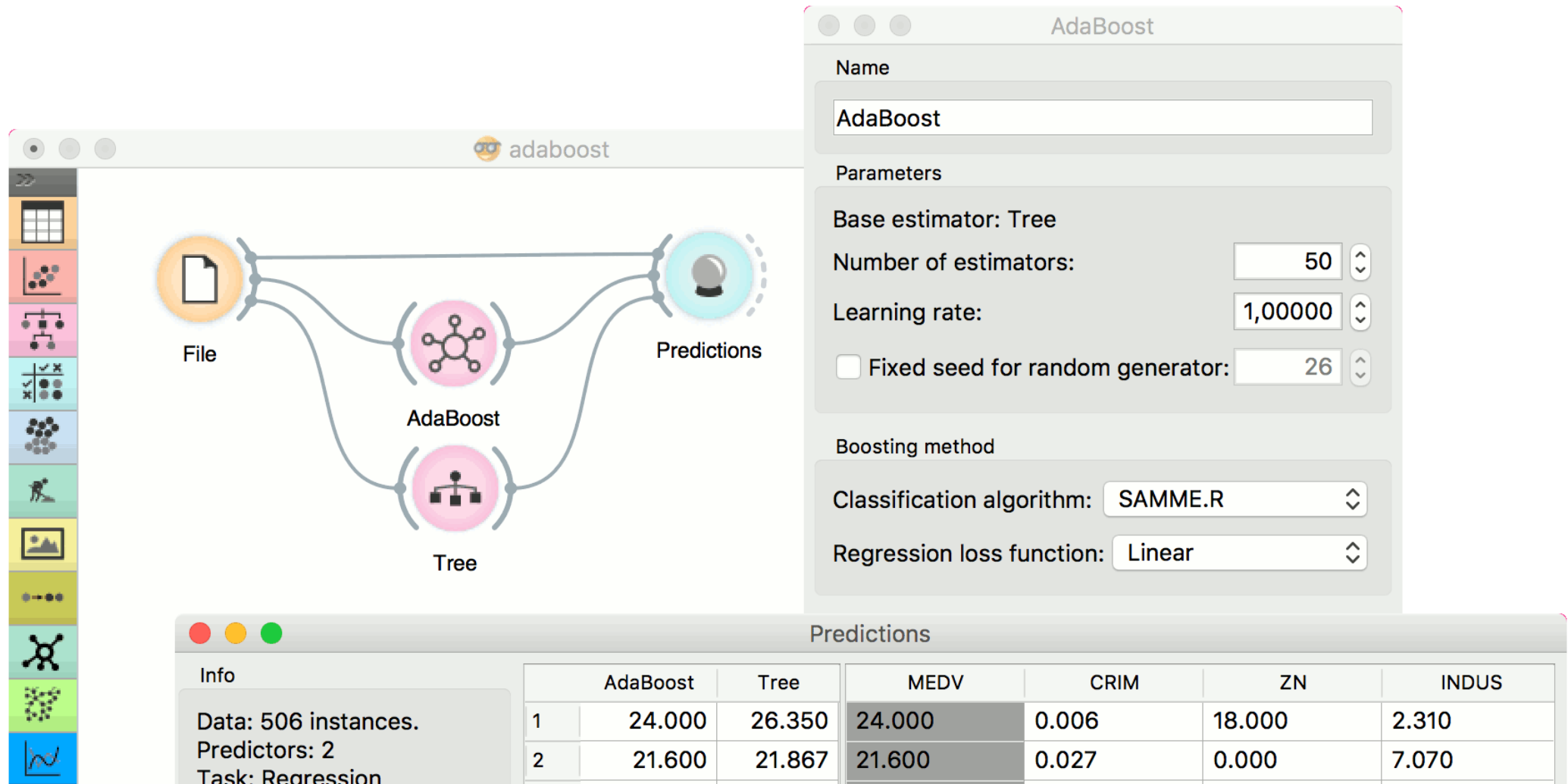
☐ Test on test data

Target Class

(Average over classes) 

Report

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.



The screenshot shows the Orange Data Mining workflow editor with the following components:

- File** widget: The data source.
- AdaBoost** widget: A pink circle with a network icon.
- Tree** widget: A pink circle with a tree icon.
- Predictions** widget: A blue circle with a lightbulb icon.

The workflow connects the **File** widget to both the **AdaBoost** and **Tree** widgets, and then both model widgets to the **Predictions** widget.

The **AdaBoost** widget settings are shown in the right panel:

- 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

The **Predictions** widget output is shown in the bottom panel:

Info	AdaBoost	Tree	MEDV	CRIM	ZN	INDUS
Data: 506 instances.						
Predictors: 2						
Task: Regression						
	24.000	26.350	24.000	0.006	18.000	2.310
	21.600	21.867	21.600	0.027	0.000	7.070



Restore Original Order		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
Data View		5	36.100	37.150	36.200	0.069	0.000	2.180
<input checked="" type="checkbox"/> Show full data set		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
Output		8	27.100	22.100	27.100	0.145	12.500	7.870
<input checked="" type="checkbox"/> Original data		9	16.500	15.475	16.500	0.211	12.500	7.870
<input checked="" type="checkbox"/> Predictions		10	18.900	18.350	18.900	0.170	12.500	7.870
<input checked="" type="checkbox"/> Probabilities		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
Report		13	21.700	22.425	21.700	0.094	12.500	7.870