Linear Regression

A linear regression algorithm with optional L1 (LASSO), L2 (ridge) or L1L2 (elastic net) regularization.

Inputs

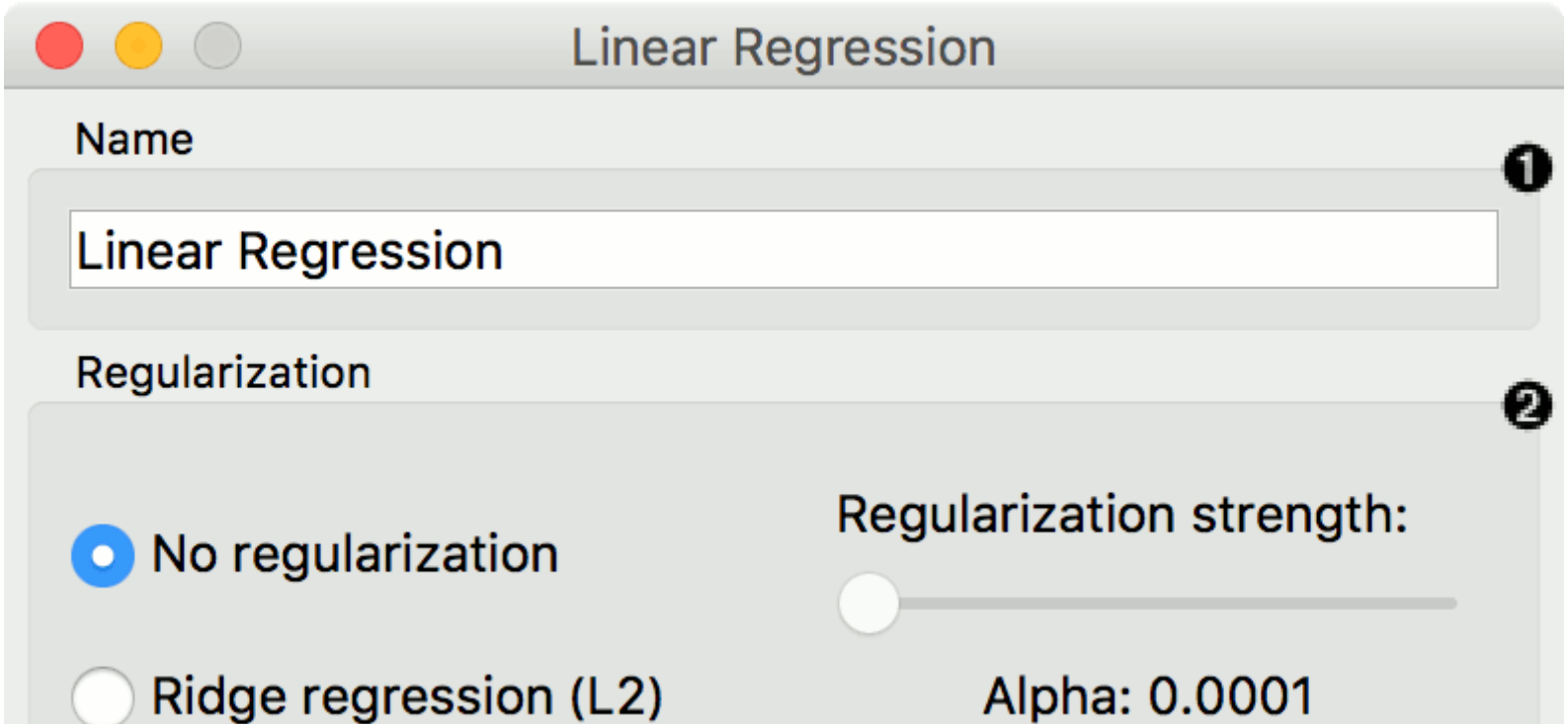
- Data: input dataset
- Preprocessor: preprocessing method(s)

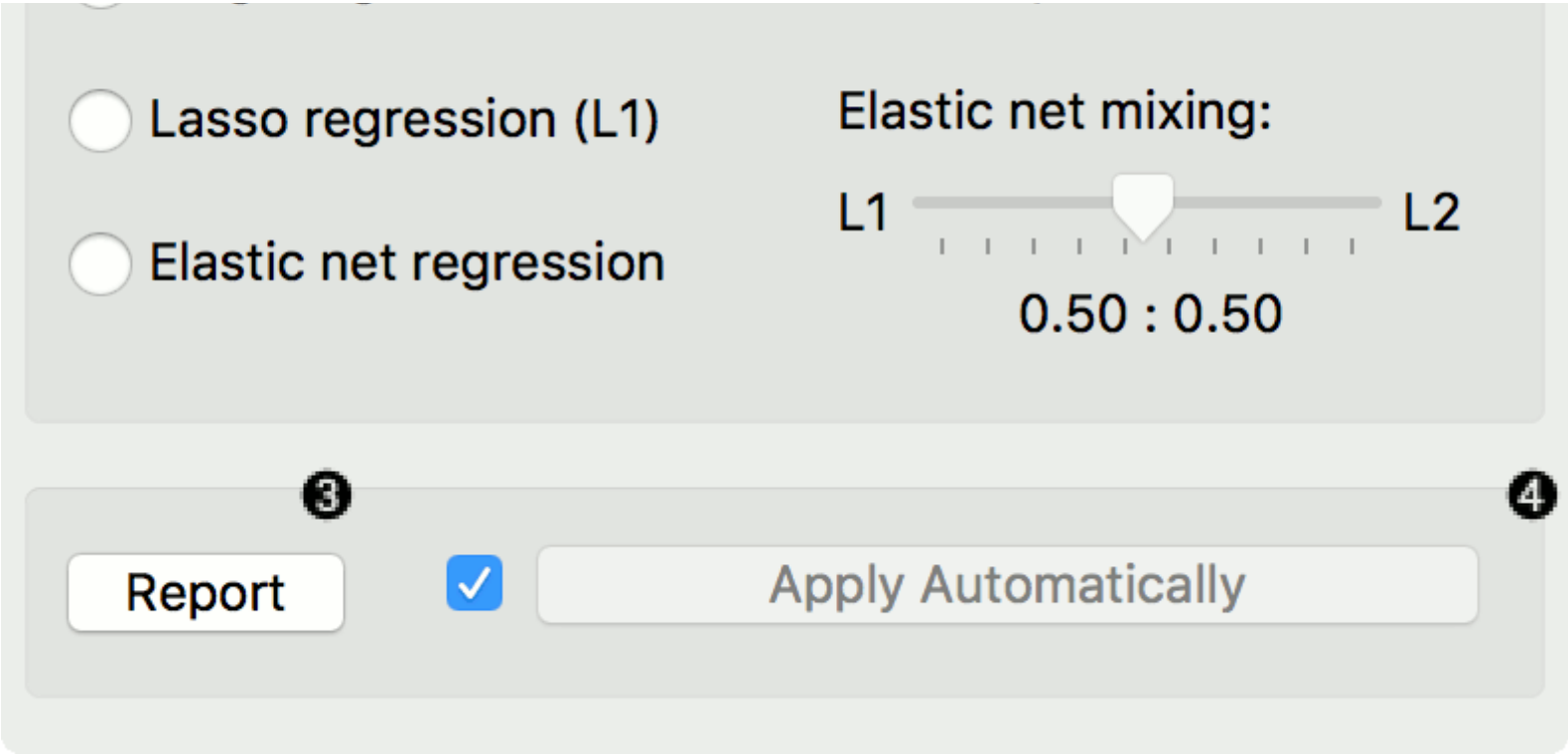
Outputs

- Learner: linear regression learning algorithm
- Model: trained model
- Coefficients: linear regression coefficients

The **Linear Regression** widget constructs a learner/predictor that learns a linear function from its input data. The model can identify the relationship between a predictor xi and the response variable y. Additionally, Lasso and Ridge regularization parameters can be specified. Lasso regression minimizes a penalized version of the least squares loss function with L1-norm penalty and Ridge regularization with L2-norm penalty.

Linear regression works only on regression tasks.

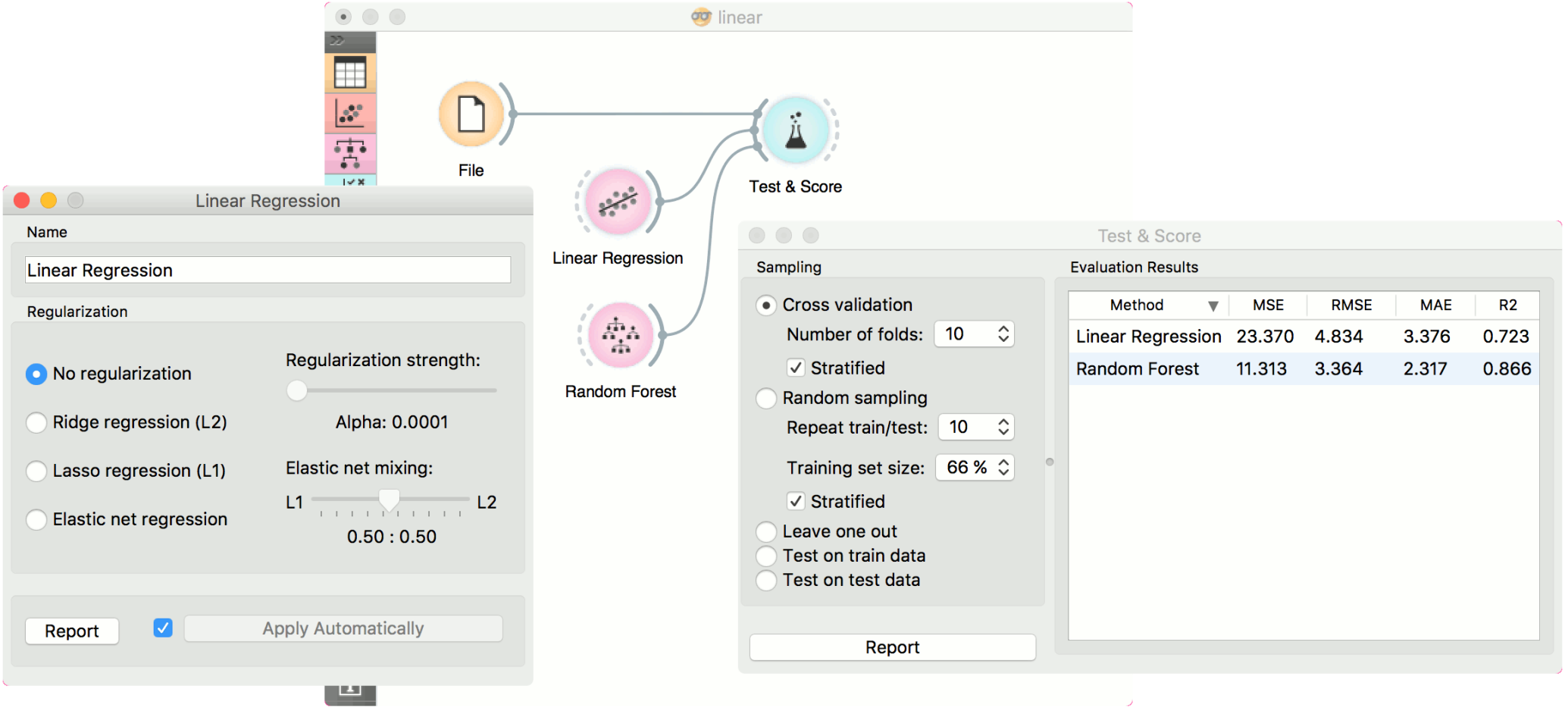




1. The learner/predictor name
2. Choose a model to train:
    - no regularization
    - a Ridge regularization (L2-norm penalty)
    - a Lasso bound (L1-norm penalty)
    - an Elastic net regularization
3. Produce a report.
4. Press *Apply* to commit changes. If *Apply Automatically* is ticked, changes are committed automatically.

Example

Below, is a simple workflow with *housing* dataset. We trained **Linear Regression** and Random Forest and evaluated their performance in Test & Score.


linear
File
Linear Regression
Random Forest
Test & Score
Linear Regression
Name
Linear Regression
Regularization
No regularization
Ridge regression (L2)
Lasso regression (L1)
Elastic net regression
Regularization strength:
Alpha: 0.0001
Elastic net mixing:
L1
L2
0.50 : 0.50
Report
Apply Automatically
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
Test on test data
Report
Evaluation Results
Method MSE RMSE MAE R2
Linear Regression 23.370 4.834 3.376 0.723
Random Forest 11.313 3.364 2.317 0.866