CN2 Rule Induction

Induce rules from data using CN2 algorithm.

Inputs

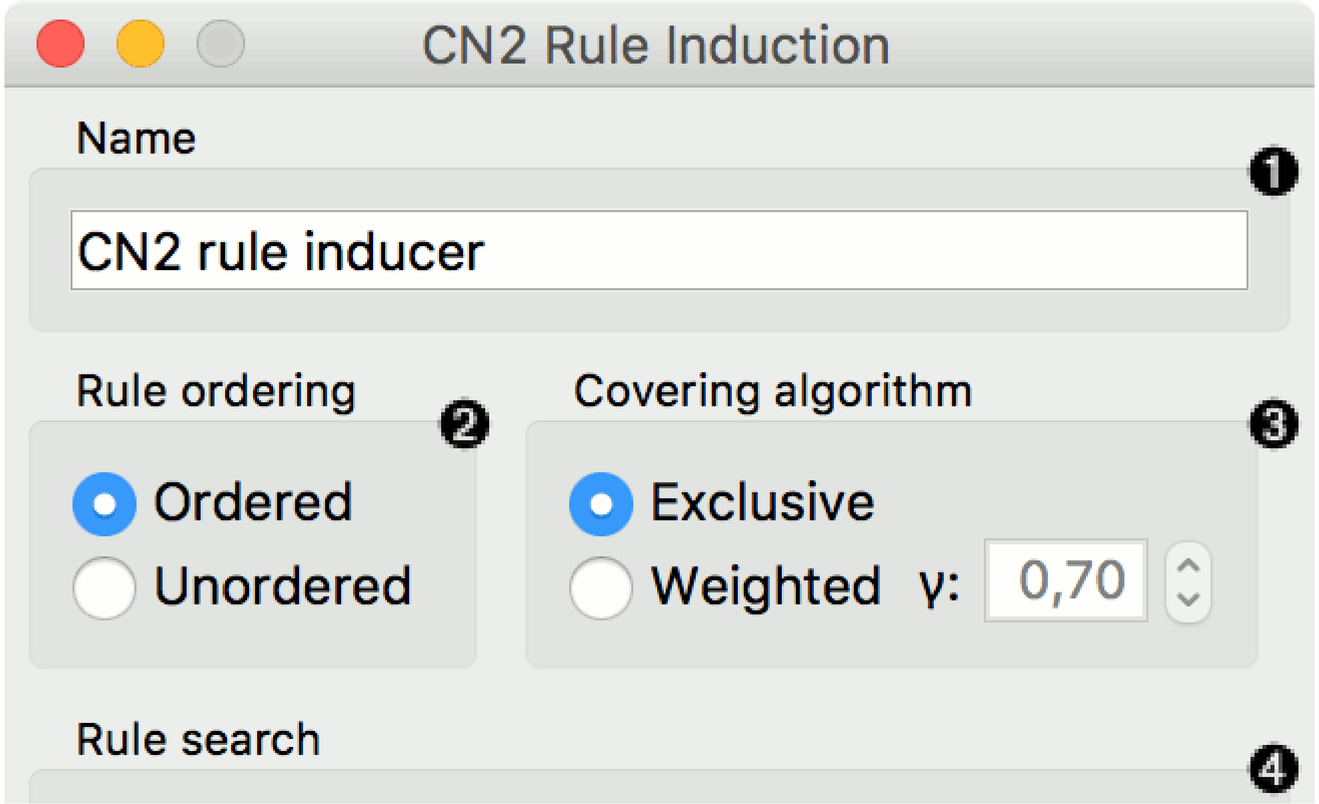
- Data: input dataset
- Preprocessor: preprocessing method(s)

Outputs

- Learner: CN2 learning algorithm
- CN2 Rule Classifier: trained model

The CN2 algorithm is a classification technique designed for the efficient induction of simple, comprehensible rules of form "if *cond*then predict *class*", even in domains where noise may be present.

CN2 Rule Induction works only for classification.



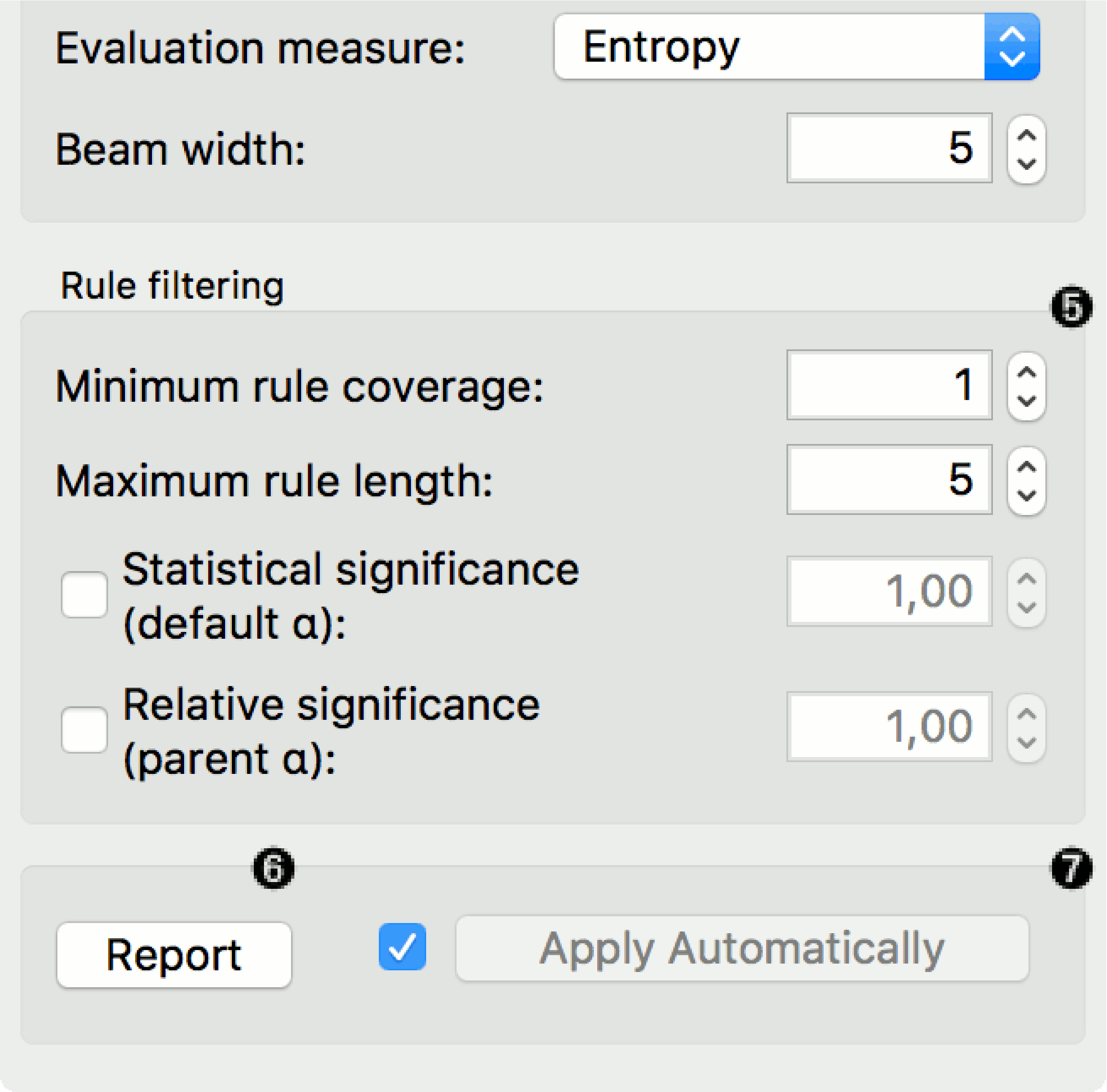
CN2 Rule Induction

Name: CN2 rule inducer

Rule ordering
● Ordered
○ Unordered

Covering algorithm
● Exclusive
○ Weighted γ: 0,70

Rule search



Evaluation measure: Entropy

Beam width: 5

Rule filtering

Minimum rule coverage: 1

Maximum rule length: 5

□ Statistical significance (default α): 1,00

□ Relative significance (parent α): 1,00

Report

☑ Apply Automatically

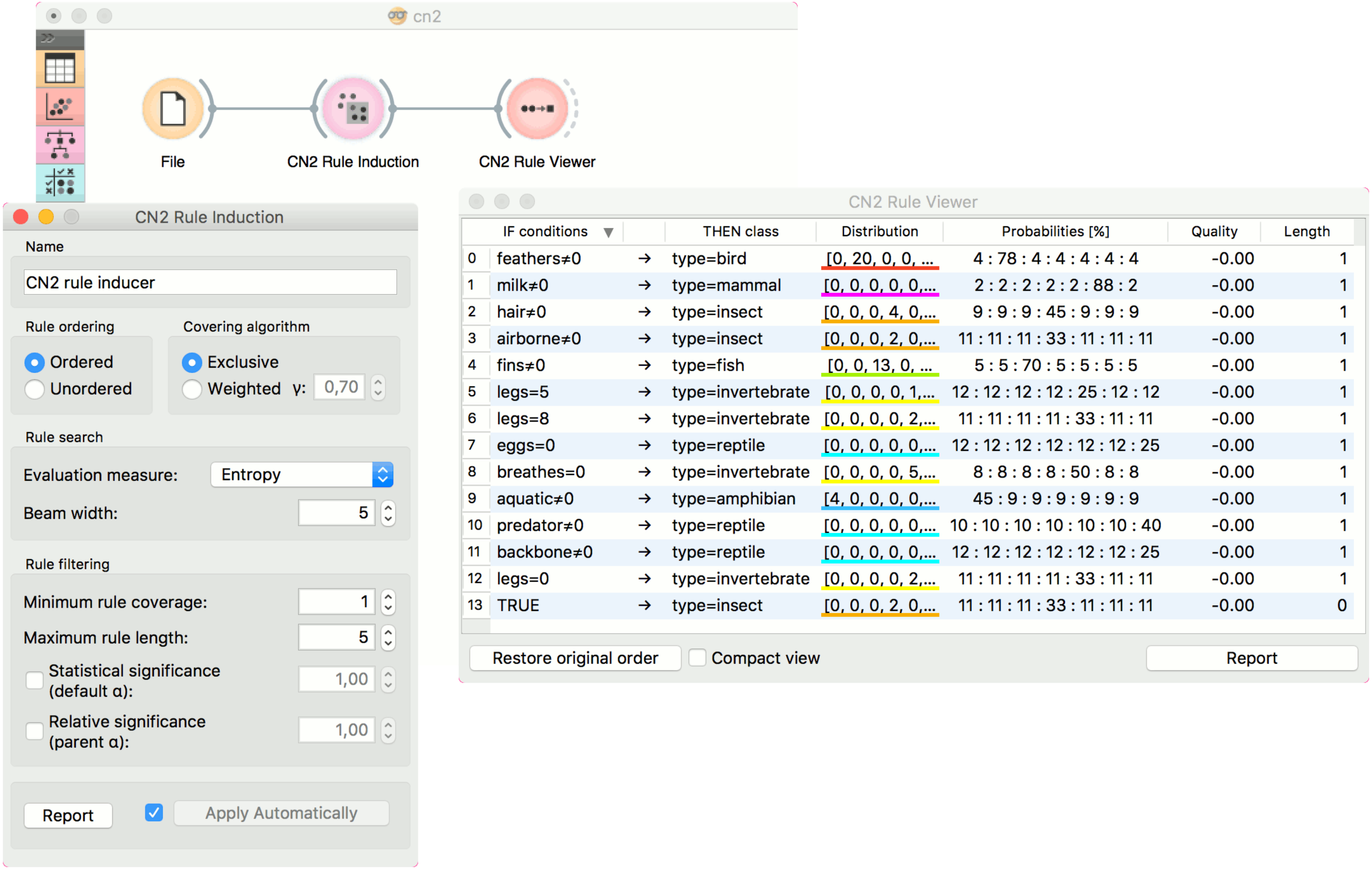
1. Name under which the learner appears in other widgets. The default name is *CN2 Rule Induction*.
2. *Rule ordering*:
   - **Ordered**: induce ordered rules (decision list). Rule conditions are found and the majority class is assigned in the rule head.

- **Unordered**: induce unordered rules (rule set). Learn rules for each class individually, in regard to the original learning data.

3. *Covering algorithm*:
    - **Exclusive**: after covering a learning instance, remove it from further consideration.
    - **Weighted**: after covering a learning instance, decrease its weight (multiplication by *gamma*) and in-turn decrease its impact on further iterations of the algorithm.
4. *Rule search*:
    - **Evaluation measure**: select a heuristic to evaluate found hypotheses:
        - Entropy (measure of unpredictability of content)
        - Laplace Accuracy
        - Weighted Relative Accuracy
    - **Beam width**; remember the best rule found thus far and monitor a fixed number of alternatives (the beam).
5. *Rule filtering*:
    - **Minimum rule coverage**: found rules must cover at least the minimum required number of covered examples. Unordered rules must cover this many target class examples.
    - **Maximum rule length**: found rules may combine at most the maximum allowed number of selectors (conditions).
    - **Default alpha**: significance testing to prune out most specialised (less frequently applicable) rules in regard to the initial distribution of classes.
    - **Parent alpha**: significance testing to prune out most specialised (less frequently applicable) rules in regard to the parent class distribution.
6. Tick ‘Apply Automatically’ to auto-communicate changes to other widgets and to immediately train the classifier if learning data is connected. Alternatively, press ‘Apply‘ after configuration.

Examples

For the example below, we have used *zoo* dataset and passed it to **CN2 Rule Induction**. We can review and interpret the built model with CN2 Rule Viewer widget.



CN2 Rule Induction

Name

CN2 rule inducer

Rule ordering

● Ordered
○ Unordered

Covering algorithm

● Exclusive
○ Weighted γ: 0,70

Rule search

Evaluation measure: Entropy

Beam width: 5

Rule filtering

Minimum rule coverage: 1

Maximum rule length: 5

□ Statistical significance (default α): 1,00

□ Relative significance (parent α): 1,00

Report ☑ Apply Automatically

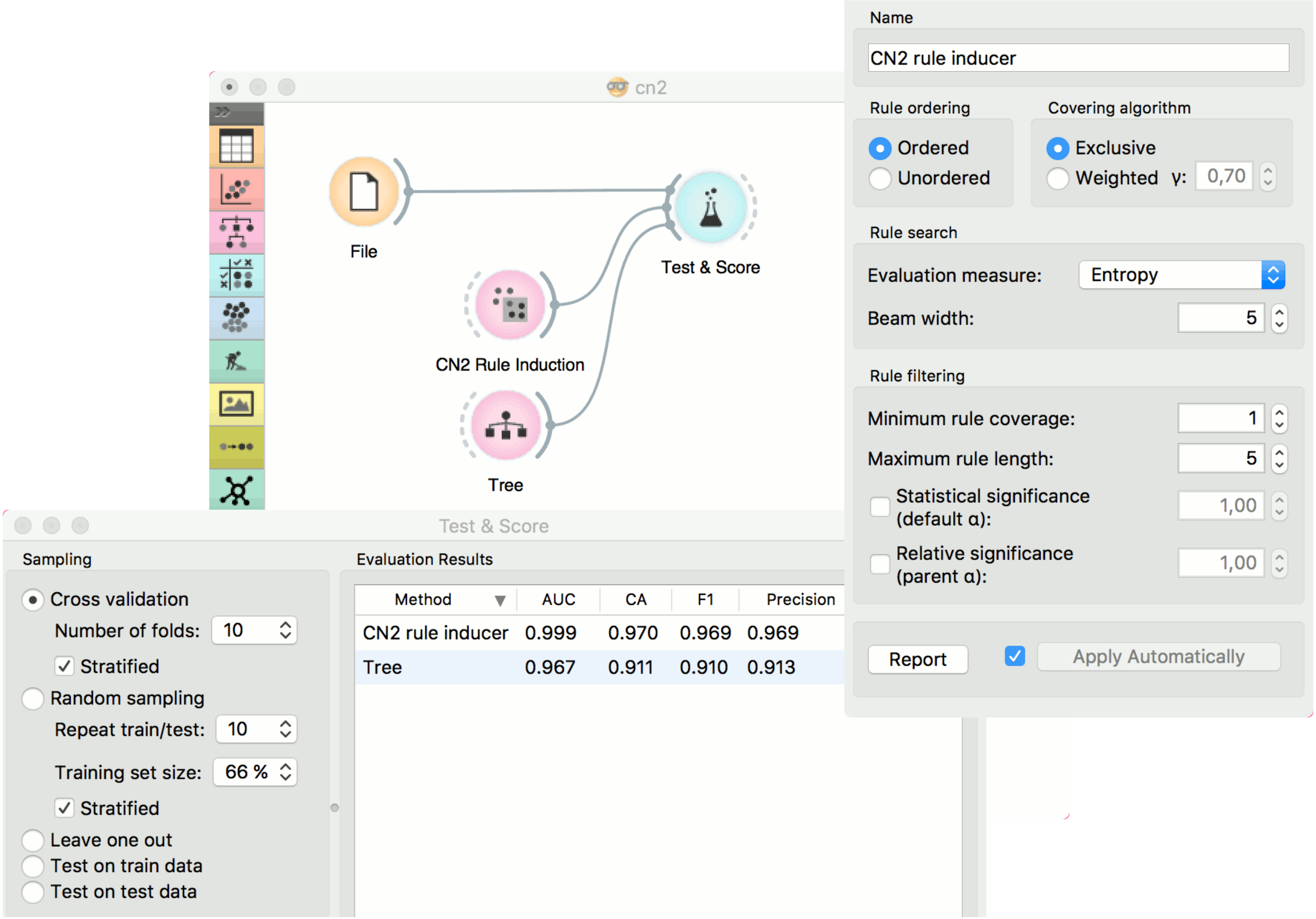
CN2 Rule Viewer

	IF conditions ▼		THEN class	Distribution	Probabilities [%]	Quality	Length
0	feathers≠0	→	type=bird	[0, 20, 0, 0, ...	4 : 78 : 4 : 4 : 4 : 4 : 4	-0.00	1
1	milk≠0	→	type=mammal	[0, 0, 0, 0, 0,...	2 : 2 : 2 : 2 : 2 : 88 : 2	-0.00	1
2	hair≠0	→	type=insect	[0, 0, 0, 4, 0,...	9 : 9 : 9 : 45 : 9 : 9 : 9	-0.00	1
3	airborne≠0	→	type=insect	[0, 0, 0, 2, 0,...	11 : 11 : 11 : 33 : 11 : 11 : 11	-0.00	1
4	fins≠0	→	type=fish	[0, 0, 13, 0, ...	5 : 5 : 70 : 5 : 5 : 5 : 5	-0.00	1
5	legs=5	→	type=invertebrate	[0, 0, 0, 0, 1,...	12 : 12 : 12 : 12 : 25 : 12 : 12	-0.00	1
6	legs=8	→	type=invertebrate	[0, 0, 0, 0, 2,...	11 : 11 : 11 : 11 : 33 : 11 : 11	-0.00	1
7	eggs=0	→	type=reptile	[0, 0, 0, 0, 0,...	12 : 12 : 12 : 12 : 12 : 12 : 25	-0.00	1
8	breathes=0	→	type=invertebrate	[0, 0, 0, 0, 5,...	8 : 8 : 8 : 8 : 50 : 8 : 8	-0.00	1
9	aquatic≠0	→	type=amphibian	[4, 0, 0, 0, 0,...	45 : 9 : 9 : 9 : 9 : 9 : 9	-0.00	1
10	predator≠0	→	type=reptile	[0, 0, 0, 0, 0,...	10 : 10 : 10 : 10 : 10 : 10 : 40	-0.00	1
11	backbone≠0	→	type=reptile	[0, 0, 0, 0, 0,...	12 : 12 : 12 : 12 : 12 : 12 : 25	-0.00	1
12	legs=0	→	type=invertebrate	[0, 0, 0, 0, 2,...	11 : 11 : 11 : 11 : 33 : 11 : 11	-0.00	1
13	TRUE	→	type=insect	[0, 0, 0, 2, 0,...	11 : 11 : 11 : 33 : 11 : 11 : 11	-0.00	0

Restore original order □ Compact view Report

The second workflow tests evaluates **CN2 Rule Induction** and Tree in Test & Score.

CN2 Rule Induction



Name

CN2 rule inducer

Rule ordering

● Ordered
○ Unordered

Covering algorithm

● Exclusive
○ Weighted γ: 0,70

Rule search

Evaluation measure: Entropy

Beam width: 5

Rule filtering

Minimum rule coverage: 1

Maximum rule length: 5

□ Statistical significance (default α): 1,00

□ Relative significance (parent α): 1,00

Report ☑ Apply Automatically

Test & Score

Sampling

● Cross validation

Number of folds: 10

☑ Stratified

○ Random sampling

Repeat train/test: 10

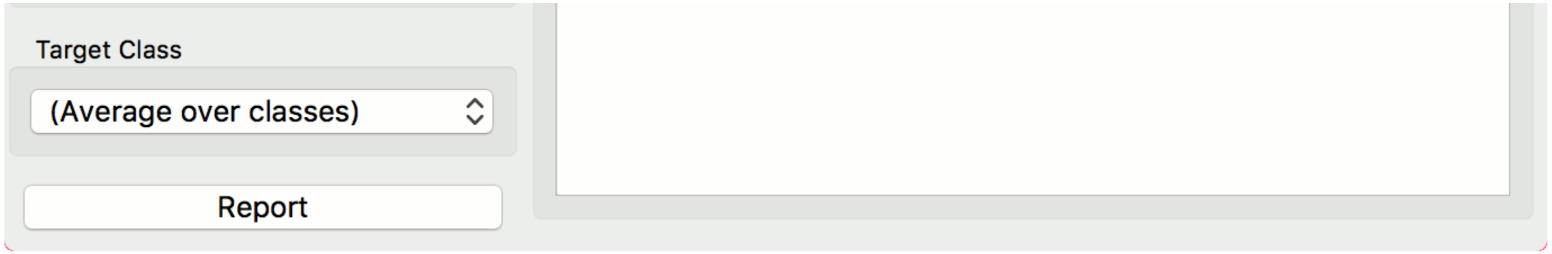
Training set size: 66 %

☑ Stratified

○ Leave one out
○ Test on train data
○ Test on test data

Evaluation Results

Method	AUC	CA	F1	Precision
CN2 rule inducer	0.999	0.970	0.969	0.969
Tree	0.967	0.911	0.910	0.913



Target Class

(Average over classes)

References

1. Fürnkranz, Johannes. “Separate-and-Conquer Rule Learning”, Artificial Intelligence Review 13, 3-54, 1999.
2. Clark, Peter and Tim Niblett. “The CN2 Induction Algorithm”, Machine Learning Journal, 3 (4), 261-283, 1989.
3. Clark, Peter and Robin Boswell. “Rule Induction with CN2: Some Recent Improvements”, Machine Learning - Proceedings of the 5th European Conference (EWSL-91),151-163, 1991.
4. Lavrač, Nada et al. “Subgroup Discovery with CN2-SD”,Journal of Machine Learning Research 5, 153-188, 2004