Symbolic Knowledge Extraction via PSyKE

A tutorial

Giovanni Ciatto[1] Matteo Magnini[1] Federico Sabbatini[2]

giovanni.ciatto@unibo.it matteo.magnini@unibo.it

f.sabbatini1@campus.uniurb.it

[1] Dipartimento di Informatica – Scienza e Ingegneria (DISI)
Alma Mater Studiorum—Università di Bologna, Cesena, Italy

[2] Dipartimento di Scienze Pure e Applicate (DiSPeA)
Università di Urbino, Urbino, Italy

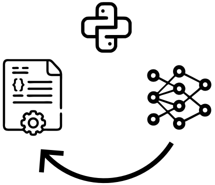
24th International Conference on
Principles and Practice of Multi-Agent Systems
November 16, 2022

Next in Line...

1. What and Why
2. Background
3. PSyKE
4. Tutorial
5. Discussion

What

PSyKE: a (Python) platform for symbolic knowledge extraction



GitHub Repository

https://github.com/psykei/psyke-python
(please star us :)

Main papers

- [Sabbatini et al., 2021a]
- [Sabbatini et al., 2022b]
- [Sabbatini et al., 2022a]

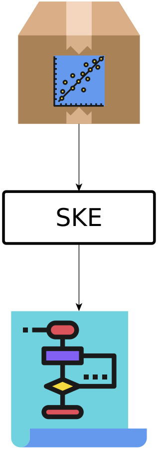
Why

- Pervasive adoption of sub-symbolic, ML-based predictors in AI
- Their opacity[Lipton, 2018] brings drawbacks[Guidotti et al., 2018]:
  - difficulty in understanding what a black-box has learned from data
    - e.g. "snowy background" problem[Ribeiro et al., 2016]
  - difficulty in spotting "bugs" in what a numeric predictor has learned
    - because such knowledge is not explicitly represented
  - several blatant failures of ML-based systems reported so far
    - e.g. black people classified as gorillas [Crawford, 2016]
    - e.g. wolves classified because of snowy background [Ribeiro et al., 2016]
    - e.g. unfair decisions in automated legal systems [Wexler, 2017]
  - recognised citizens' right to meaningful explanations[Selbst and Powles, 2017]
    - about the logic behind automated decision making
    - e.g. in General Data Protection Regulation (GDPR) [EU Parliament and Council, 2016]

→ Need to inspect and understand how ML predictors operate

Next in Line...

Symbolic Knowledge Extraction I



Key insights:

- Explaining supervised ML predictors. . .
- . . . by search of a surrogate interpretable model. . .
- . . . consisting of symbolic knowledge

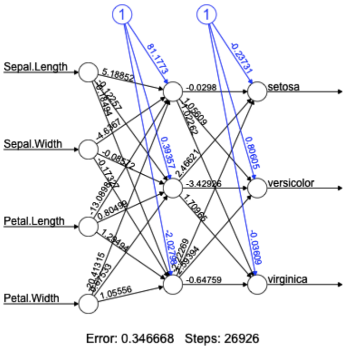
Symbolic Knowledge Extraction II

Definition

Any algorithmic procedure accepting trained sub-symbolic predictors as input and producing symbolic knowledge as output, in such a way that the extracted knowledge reflects the behaviour of the predictor with high fidelity.

Symbolic Knowledge Extraction III

Example:



$\rightarrow$

$$Class = \texttt{setosa} \leftarrow PetalWidth \leq 1.0.$$

$$Class = \texttt{versicolor} \leftarrow PetalLength > 4.9 \wedge SepalWidth \in [2.9,\ 3.2].$$

$$Class = \texttt{versicolor} \leftarrow PetalWidth > 1.6.$$

$$Class = \texttt{virginica} \leftarrow SepalWidth \leq 2.9.$$

$$Class = \texttt{virginica} \leftarrow SepalLength \in [5.4,\ 6.3].$$

$$Class = \texttt{virginica} \leftarrow PetalWidth \in [1.0,\ 1.6].$$

What does 'symbolic' actually mean? I

According to [van Gelder, 1990], symbolic representations of knowledge

- involves a set of symbols,
- which can be combined (e.g., concatenated) in (possibly) infinitely many ways,
- following precise syntactical rules, and
- where both elementary symbols and any admissible combination of them can be assigned with meaning

  ie each symbol can be mapped into some entity from the domain at hand.

Notable example

- formal logic

What does ‘symbolic’ actually mean? II

Opposite notion: **distributed** representations

- where symbols alone have no meaning
- unless it is considered along with its neighbourhood

    ie any other symbol which is close (according to some notion of closeness)

Plenty of SKE methods from the literature I

Table: Summary of the knowledge-extraction algorithms. Symbol ∗ means that the related dimension of the algorithm is not bounded. Symbol † means that the output is a power law.

#	Method	Translucency	Task	Input	Expressiveness	Shape
1	[Breiman et al., 1984]	P	C+R	C+D	P	DT
2	[Quinlan, 1986]	P	C	D	P	DT
3	[Saito and Nakano, 1988]	P	C	D	P	L
4	[Clark and Niblett, 1989]	P	C	C+D	P	L
5	[Masuoka et al., 1990]	D (NN)	C	C	F	L
6	[Hayashi, 1990]	D (NN)	C	B	F	L
7	[Towell and Shavlik, 1991]	D (NN)	C	D	MN	L
8	[Berenji, 1991]	D (NN)	C	C	F	L
9	[Brunk and Pazzani, 1991]	P	C	C+D	P	L
10	[Murphy and Pazzani, 1991]	P	C	D	MN	DT
11	[Horikawa et al., 1992]	D (NN)	C	C	F	L
12	[Tresp et al., 1992]	D (NN)	R	C	P	L
13	[Towell and Shavlik, 1993]	D (NN)	C	D	P	L
14	[Thrun, 1993]	D (NN)	C	C	P+MN	L
15	[Cohen, 1993]	P	C	C+D	P	L

Plenty of SKE methods from the literature II

16	[Quinlan, 1993]	P	C	C+D	P	DT
17	[Fu, 1994]	D (NN)	C	D	P	L
18	[Halgamuge and Glesner, 1994]	D (NN)	C	C	F	L
19	[Mitra, 1994]	D (NN)	C	C+D	F	L
20	[Craven and Shavlik, 1994]	P	C	B	P+MN	L
21	[Fürnkranz and Widmer, 1994]	P	C	D	P	L
22	[Sestito and Dillon, 1994]	P	C	C	P	L
23	[Andrews and Geva, 1995]	D (NN)	C	C+D	P	L
24	[Matthews and Jagielska, 1995]	D (NN)	C	B	F	L
25	[Cohen, 1995]	P	C	C+D	P	L
26	[Pop et al., 1994]	P	C	B	P	L
27	[Setiono and Liu, 1996]	D (NN)	C	B	P	L
28	[Tickle et al., 1996]	P	C	B	P	L
29	[Yuan and Zhuang, 1996]	P	C	D	F	L
30	[Craven and Shavlik, 1996]	P	C	B	P+MN	DT
31	[Hong and Lee, 1996]	P	C	C	F	L
32	[Setiono and Liu, 1997]	D (NN3)	C	C+D	O	L
33	[Setiono, 1997]	D (NN)	C	D	P	L
34	[Nauck and Kruse, 1997]	D (NN)	C	D	F	L

Plenty of SKE methods from the literature III

35	[Saito and Nakano, 1997]	D (NN)	R	C	†	†
36	[Benítez et al., 1997]	D (NN)	C+R	C	F	L
37	[Ishibuchi et al., 1997]	P	C	C	F	L
38	[Taha and Ghosh, 1999]	D (NN)	C	C	P	L
39	[Taha and Ghosh, 1999]	D (NN)	C	C	P	L
40	[Krishnan et al., 1999b]	D (NN)	C	B	P	L
41	[Nauck and Kruse, 1999]	D (NN)	R	D	F	L
42	[Taha and Ghosh, 1999]	P	C	B	P	L
43	[Krishnan et al., 1999a]	P	C	C	P	DT
44	[?]	P	C+R	C+D	P	DT
45	[Hong and Chen, 1999]	P	C	C	F	L
46	[Setiono, 2000]	D (NN)	C	B	MN	L
47	[Tsukimoto, 2000]	D (NN)	C	C+D	P	L
48	[Kim and Lee, 2000]	D (NN4)	C	C+D	P	DT
49	[Setiono and Leow, 2000]	D (NN)	R	C+D	P+MN+O	DT
50	[Zhou et al., 2000]	P	C	C+D	P	L
51	[Hong and Chen, 2000]	P	C	C	F	L
52	[Sato and Tsukimoto, 2001]	D (NN3)	R	C+D	P	DT
53	[Parpinelli et al., 2001]	P	C	C+D	P	L

Plenty of SKE methods from the literature IV

54	[Castillo et al., 2001]	P	C+R	C+D	F	L
55	[Saito and Nakano, 2002]	D (NN)	R	C+D	P	L
56	[Setiono et al., 2002]	D (NN3)	R	C+D	P	L
57	[Liu et al., 2002]	P	C	C+D	P	L
58	[Boz, 2002]	P	C	C+D	P	DT
59	[Markowska-Kaczmar and Trelak, 2003]	P	C	C+D	F	L
60	[Zhou et al., 2003]	P	C	C+D	P	L
61	[Setiono and Thong, 2004]	D (NN3)	R	C+D	P	L
62	[Fu et al., 2004]	D (SVM)	C	C+D	P	L
63	[Markowska-Kaczmar and Chumieja, 2004]	P	C	C+D	P	L
64	[Rabuñal et al., 2004]	P	C	C+D	P	L
65	[Chen, 2004]	P	C	C	P	L
66	[Liu et al., 2004]	P	C	C+D	P	L
67	[Browne et al., 2004]	P	C	C+D	P+MN	DT
68	[Zhang et al., 2005]	D (SVM)	C	C	P	L
69	[Barakat and Diederich, 2008]	D (SVM)	C+R	*	*	*
70	[Fung et al., 2005]	D (SVM+LC)	C	C	P	L
71	[Chaves et al., 2005]	D (SVM)	C	C	F	L
72	[Torres and Rocco, 2005]	P	C	C+D	P+MN	DT

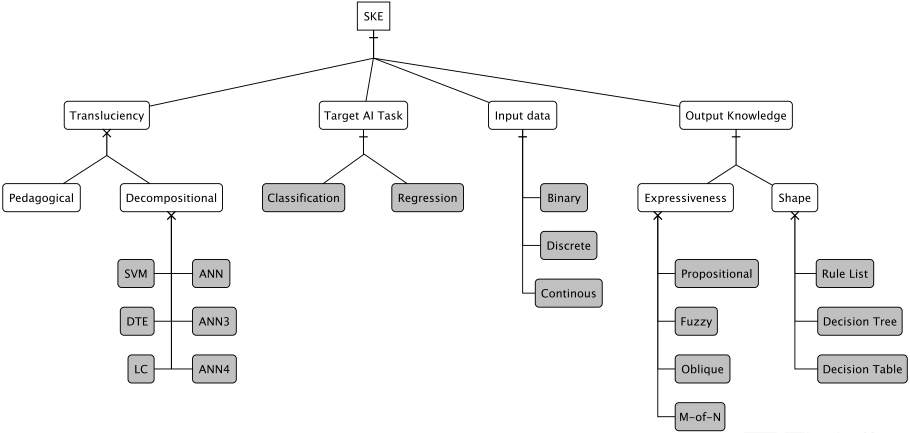
Plenty of SKE methods from the literature V

73	[Etchells and G., 2006]	P	C	C+D	P	L
74	[He et al., 2006]	P	C	C+D	P	DT
75	[Huysmans et al., 2006]	P	R	C	P	L
76	[Bader et al., 2007]	D (NN)	C	B	P	L
77	[Schetinin et al., 2007]	D (DTE)	R	C	P	DT
78	[Chen et al., 2007]	D (SVM)	C	C	P	L
79	[Barakat and Bradley, 2007]	D (SVM)	C	C+D	P	L
80	[Saad and Wunsch II, 2007]	P	C	C+D	O	L
81	[Martens et al., 2007]	P	C	C+D	P	L
82	[Núñez et al., 2008]	D (SVM)	C	C	P+O	L
83	[Setiono et al., 2008]	P	C	C+D	P+O	L
84	[Odajima et al., 2008]	P	C	D	P	L
85	[Konig et al., 2008]	P	C+R	C+D	F	DT
86	[Bader, 2009]	D (NN)	C	B	P	L
87	[Martens et al., 2009]	D (SVM)	C	*	*	*
88	[Lehmann et al., 2010]	P	C	B	P	L
89	[Augasta and Kathirvalavakumar, 2012]	P	C	C+D	P	L
90	[Sethi et al., 2012]	P	C	C+D	P	TA
91	[Zilke et al., 2016]	D (NN)	R	C+D	P	DT

Plenty of SKE methods from the literature VI

92	[Chan and Chan, 2017]	D (NN)	R	C	P	L
93	[Yedjour and Benyettou, 2018]	P	C	B	P	L
94	[Chan and Chan, 2020]	D (NN)	R	C	P	L
95	[Wang et al., 2020]	D (DTE)	C	C	P	L
96	[Sabbatini et al., 2021b]	P	R	C	P	L

Taxonomy of SKE methods I



Taxonomy of SKE methods II

target AI task for the predictor undergoing extraction

- **classification** i.e., finite amount of possible predictions
- **regression** i.e., continuous predictions

translucency what kind of ML predictor does the SKE method support?

- **pedagogical:** any supervised predictor
- **decompositional:** a particular sort of ML predictor (e.g. NN, SVM, DT)

input data supported by the predictor undergoing extraction

- **binary:** $\mathcal{X} \equiv \{0, 1\}^n$
- **discrete:** $\mathcal{X} \in \{x_1, \ldots, x_n\}^n$
- **continuous:** $\mathcal{X} \subseteq \mathbb{R}^n$

Taxonomy of SKE methods III

shape of the extracted knowledge

- rule list: i.e. ordered sequences of if-then-else rules
- decision tree: hierarchical set of if-then-else rules involving a comparison among a variable and a constant
- decision table: 2D tables summarising decisions for each possible assignment of variables

Taxonomy of SKE methods IV

expressiveness of the extracted knowledge

propositional: boolean statements + logic connectives

- there including arithmetic comparisons among variables and constants

fuzzy: hierarchical set of if-then-else rules involving a comparison among a variable and a constant

oblique: boolean statements + logic connectives + arithmetic comparisons

M-of-N: any of the above + statements like $m - \text{of} - \{\phi_1, \ldots, \phi_n\}$

Examples of methods and their classification – CART I

CART:[Breiman et al., 1984] classification and regression trees

- **translucency:** pedagogical
- **target AI task:** classification OR regression
- **input data:** binary OR discrete OR continuous
- **shape:** decision tree
- **expressiveness:** propositional

Examples of methods and their classification – CART II

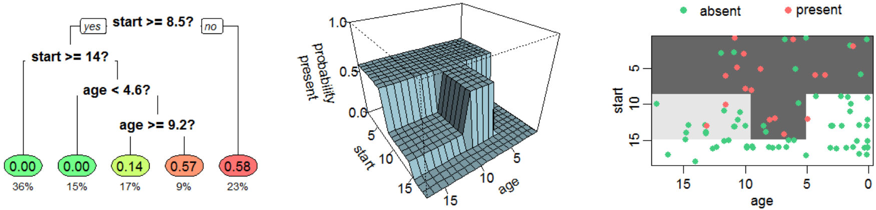


Figure: An example decision tree estimating the probability of kyphosis after spinal surgery, given the *age* of the patient and the vertebra at which surgery was *start*ed [Wikipedia contributors, 2021]. Notice that all decision trees subtend a partition of the input space, and that those trees themselves provide intelligible representations of *how* predictions are attained.

Examples of methods and their classification – CART III

Using CART for SKE

1. generate a 'fake' dataset by feeding the predictor undergoing SKE
2. train a decision tree on the 'fake' dataset
3. compute fidelity and repeat step 2 until satisfied
4. [opt.] rewrite the tree as a list of rules

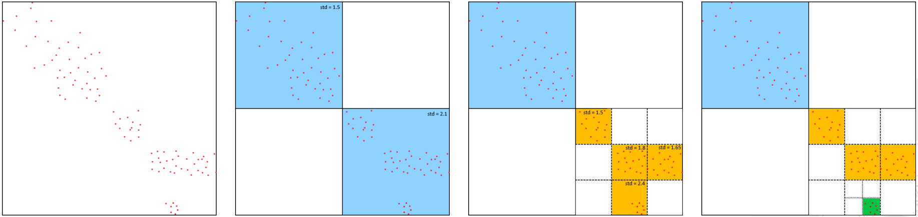
Examples of methods and their classification – GridEx I

GridEx:[Sabbatini et al., 2021b] grid extractor

- **translucency:** pedagogical
- **target AI task:** regression
- **input data:** continuous
- **shape:** rule list
- **expressiveness:** propositional

Examples of methods and their classification – GridEx II

Figure: Example of GridEx's hyper-cube partitioning (merging step not reported)



(a) Surrounding cube

(b) Iteration 1 ($p_1 = 2$)

(c) Iteration 2 ($p_2 = 3$).

(d) Iteration 3 ($p_3 = 2$).

Examples of methods and their classification – GridEx III

Using GridEx for SKE

1. partition the input space into p_1^n hypercubes
   - evenly splitting the n dimensions into p_1 bins
2. partition each non empty-region into p_2^n hypercubes
   - evenly splitting the n dimensions into p_2 bins
3. repeat the splitting arbitrarily
4. assign a prediction with each non-empty partition (e.g. average value)
5. write an if-then rule for each non-empty partition:
   - *if*: expressions delimiting the partition
   - *then*: prediction of that partition

Examples of methods and their classification – REFANN I

REFANN:[Setiono et al., 2002] rule extraction from function approximating NN

- **translucency:** decompositional (3-layered NN)
- **target AI task:** regression
- **input data:** continuous OR discrete
- **shape:** rule list
- **expressiveness:** propositional

Examples of methods and their classification – REFANN II

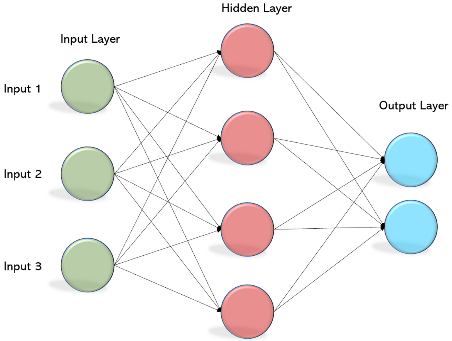


Figure: An example 3-layered multi-layer perceptron (MLP)

Examples of methods and their classification – REFANN III

Using REFANN for SKE

1. prune the network's hidden units and input neurons
2. approximate the hidden units' activation function with a 2-steps-wise linear function
3. approximate the output units' activation function with a 3- or 5-step-wise linear function
4. rewrite each output neuron as a linear combination of the input neuron
5. rewrite the linear combinations as rules
    - hence attaining a list of rules

Examples of methods and their classification – REFANN IV

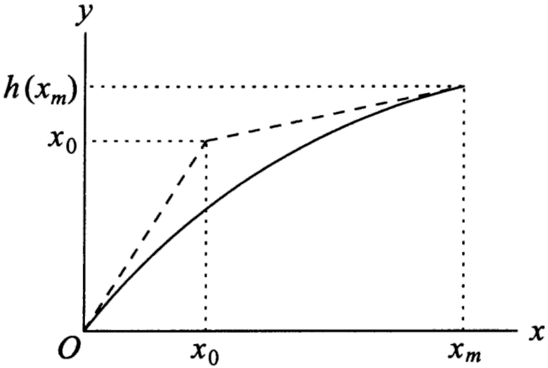


Figure: (from [Setiono et al., 2002]) The *tanh*(x) function (solid curve) for $x \in [0, x_m]$ is approximated by a 2-piece linear function (dashed lines)

Examples of methods and their classification – REFANN V

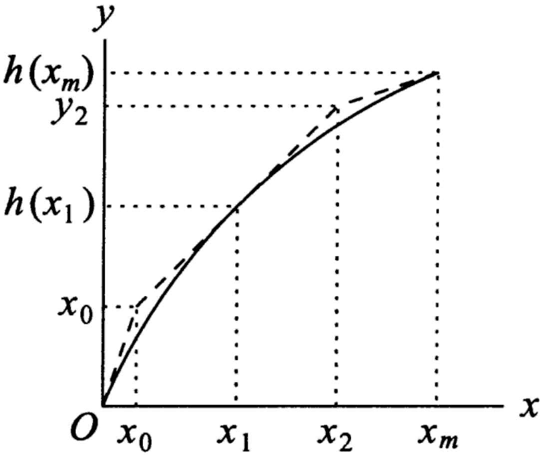
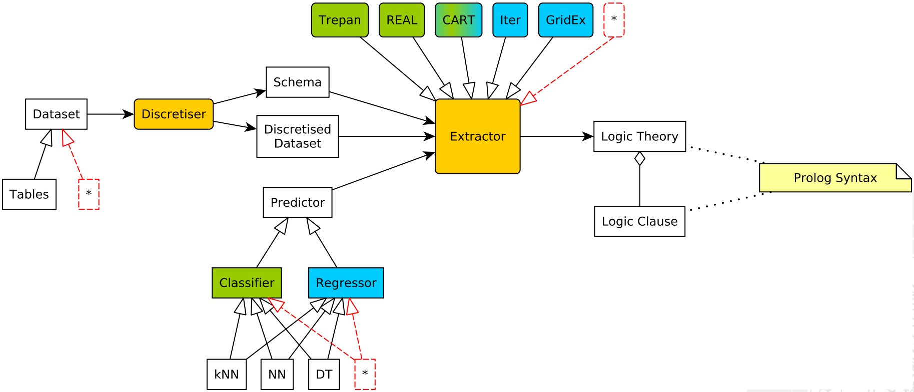


Figure: (from [Setiono et al., 2002]) The *tanh*(x) function (solid curve) for $x \in [0, x_m]$ is approximated by a 3-piece linear function (dashed lines)

Next in Line. . .

Overall Design I



Overall Design II

Key components:

extractor: any entity capable of extracting symbolic knowledge out of sub-symbolic predictors

- possibly, in the form of logic knowledge bases
- possibly, leveraging upon the dataset the predictor was trained upon ...
  - possibly, after a discretization step
- ... and its schema

predictor: some trained classifier/regressor from which knowledge should be extracted

discretiser: any component capable to turn continuous datasets into discrete form, following some strategy

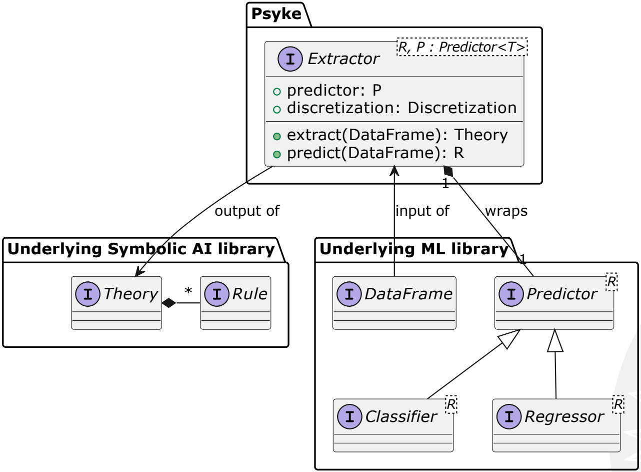
logic theory: outcome of the extraction process

Overall Design III

Unified API for SKE

- 1 interface for `Extractor`, several implementations
  eg CART, REAL, GridEx
- 1 interface for `Discretiser`, several implementations
- 1 interface for `Predictor`, several implementations
  eg NN, kNN, DT

API Design I


Psyke
Extractor
R, P : Predictor<T>
predictor: P
discretization: Discretization
extract(DataFrame): Theory
predict(DataFrame): R
output of
input of
wraps
1
1
Underlying Symbolic AI library
Theory
*
Rule
Underlying ML library
DataFrame
Predictor
R
Classifier
R
Regressor
R

API Design II

General assumptions:

- underlying ML library (e.g. Scikit-Learn [Pedregosa et al., 2011]), providing:

  `DataFrame` a container of tabular data

  `Predictor<R>` a computational entity which can be trained (a.k.a. fitted) against a `DataFrame` and used to draw predictions of type `R`;

  `Classifier<R>` a particular case of predictor where `R` represents a type having a finite amount of admissible values;

  `Regressor<R>` a particular case of predictor where `R` represents a type having a potentially infinite (possibly continuous) amount of admissible values.

API Design III

- underlying symbolic AI library (e.g. 2P-Kt[Ciatto et al., 2021]), providing:
  - `Rule` a semantic, intelligible representation of the function mapping `Predictor`'s inputs into the corresponding outputs, for a particular portion of the input space;
  - `Theory` an ordered collection of rules.

About the Extracted Knowledge I

Knowledge extracted from **classifiers**

$$
\begin{array}{lcl}
\langle task \rangle(X_1, \ldots, X_n, \mathtt{y_1}) & \texttt{:-} & p_{1,1}(\bar{X}), \; \ldots, \; p_{n,1}(\bar{X}). \\
\langle task \rangle(X_1, \ldots, X_n, \mathtt{y_2}) & \texttt{:-} & p_{1,2}(\bar{X}), \; \ldots, \; p_{n,2}(\bar{X}). \\
 & \vdots & \\
\langle task \rangle(X_1, \ldots, X_n, \mathtt{y}_m) & \texttt{:-} & p_{1,m}(\bar{X}), \; \ldots, \; p_{n,m}(\bar{X}).
\end{array}
$$

About the Extracted Knowledge II

Knowledge extracted from **regressors**

$$
\begin{array}{rcl}
\langle task \rangle(X_1, \ldots, X_n, Y) & :- & p_{1,1}(\bar{X}),\ \ldots,\ p_{n,1}(\bar{X}), \\
 & & Y \texttt{ is } f_1(\bar{X}). \\
\langle task \rangle(X_1, \ldots, X_n, Y) & :- & p_{1,2}(\bar{X}),\ \ldots,\ p_{n,2}(\bar{X}), \\
 & & Y \texttt{ is } f_2(\bar{X}). \\
 & \vdots & \\
\langle task \rangle(X_1, \ldots, X_n, Y) & :- & p_{1,m}(\bar{X}),\ \ldots,\ p_{n,m}(\bar{X}), \\
 & & Y \texttt{ is } f_m(\bar{X}).
\end{array}
$$

About the Extracted Knowledge III

...where:

- *task* is the $(n+1)$-ary relation representing the classification or regression task at hand,
- each X_i is a logic variable named after the i^{th} input attribute of the currently available data set,
- $\bar{X}$ is the n-nuple $X_1, \ldots, X_n$,
- each $p_{i,j}$ is either a n-ary predicate expressing some constraint about one, two or more variables, or the `true` literal—which can be omitted,
- y_i is the output of the i^{th} prediction rule,
- f_j is an n-ary function computing the output value for the regression task in the particular portion of the input space handled by the j^{th} rule, and
- *is*/2 is the well-known Prolog predicate aimed at evaluating functions.

About the Extracted Knowledge IV

Underlying assumptions

1. the input space is partitioned into a finite set of regions
2. each region is assigned with a particular outcome, namely:
   - a class, for classification problems
   - a constant, or a simpler function, for regression problems
3. one rule generated describing for each region and its corresponding outcome

Next in Line...

Tutorial

Two ways to reproduce the tutorial:

GitHub Repository (long way)

https://github.com/pikalab-unibo/prima-tutorial-2022

DockerHub Images (quick way)

https://hub.docker.com/r/pikalab/prima-tutorial-2022/tags

Next in Line...

How to set the tutorial up from GitHub I

Enviromental pre-requisites

- Python `3.9.x`
- JDK $\geq$ `11`
- Git

1. `git clone https://github.com/pikalab-unibo/prima-tutorial-2022`
2. `cd prima-tutorial-2022`
3. `pip install -r requirements.txt`
4. `jupyter notebook`

How to set the tutorial up from GitHub II

5. Your browser should automatically open showing the following page:

6. open the `psyke-tutorial.ipynb` notebook
7. listen to the speaker presenting the tutorial =)

Next in Line...

How to set the tutorial up via Docker I

Enviromental pre-requisites

- Docker

1. $$\texttt{DOCKER\_IMAGE=}\begin{cases}\texttt{pikalab/prima-tutorial-2022:latest}\\ \texttt{pikalab/prima-tutorial-2022:latest-apple-m1}\end{cases}$$
2. `docker pull $DOCKER_IMAGE`
   - in case of lacking Internet access:

     `docker image load -i /path/to/local/image/file.tar`
3. `docker run -it -rm -name prima-tutorial-ske-ski -p 8888:8888 $DOCKER_IMAGE`
4. Some textual output such as the following one should appear:

How to set the tutorial up via Docker II

```
[I 09:51:46.940 NotebookApp] Writing notebook server cookie secret to /root/.local/
    share/jupyter/runtime/notebook_cookie_secret
[I 09:51:47.159 NotebookApp] Serving notebooks from local directory: /notebook
[I 09:51:47.159 NotebookApp] Jupyter Notebook 6.5.2 is running at:
[I 09:51:47.159 NotebookApp] http://cb0a3641caf0:8888/?token=2
    b02d31671c6ad9e9cf8e036eb6962d3592af9cfdd5e60bd
[I 09:51:47.159 NotebookApp]  or http://127.0.0.1:8888/?token=2
    b02d31671c6ad9e9cf8e036eb6962d3592af9cfdd5e60bd
[I 09:51:47.160 NotebookApp] Use Control-C to stop this server and shut down all
    kernels (twice to skip confirmation).
[C 09:51:47.162 NotebookApp]

To access the notebook, open this file in a browser:
    file:///root/.local/share/jupyter/runtime/nbserver-7-open.html
Or copy and paste one of these URLs:
    http://cb0a3641caf0:8888/?token=2
        b02d31671c6ad9e9cf8e036eb6962d3592af9cfdd5e60bd
or http://127.0.0.1:8888/?token=2b02d31671c6ad9e9cf8e036eb6962d3592af9cfdd5e60bd
```

How to set the tutorial up via Docker III

5. Copy-paste into your browser any link of the form:

   `http://cb0a3641caf0:8888/?token=`*`TOKEN`*

6. Your browser should now be showing the following page:

7. open the `psyke-tutorial.ipynb` notebook
8. listen to the speaker presenting the tutorial =)

Next in Line. . .

Notable Remarks

- commitment to a particular output shape / expressiveness
  - to preserve both human- and machine-interpretability
  - other syntaxes may exist
- discretization of the input space
- discretization of the output space
- features should have semantics per se
- further refinements may be applied to rules
- rules constitute global explanations

Current Limitations

- tabular data as input $\rightarrow$ doesn't really work with images
- high dimensional datasets $\rightarrow$ very large, poorly readable rules
- highly variable input spaces $\rightarrow$ many rules $\rightarrow$ poor readability

Future research activities

- target images or highly dimensional data in general
- target reinforcement learning (when based on NN)
- target unsupervised learning
- design and prototype your own extraction algorithm

Symbolic Knowledge Extraction via PSyKE

A tutorial

Giovanni Ciatto[1] Matteo Magnini[1] Federico Sabbatini[2]
giovanni.ciatto@unibo.it matteo.magnini@unibo.it
f.sabbatini1@campus.uniurb.it

[1] Dipartimento di Informatica – Scienza e Ingegneria (DISI)
Alma Mater Studiorum—Università di Bologna, Cesena, Italy

[2] Dipartimento di Scienze Pure e Applicate (DiSPeA)
Università di Urbino, Urbino, Italy

24^{th} International Conference on
Principles and Practice of Multi-Agent Systems
November 16, 2022

References I

[Andrews and Geva, 1995] Andrews, R. and Geva, S. (1995).
Rulex & cebp networks as the basis for a rule refinement system.
In Hallam, J., editor, *Hybrid Problems, Hybrid Solutions*, pages 1–12. IOS Press.

[Augasta and Kathirvalavakumar, 2012] Augasta, M. G. and Kathirvalavakumar, T. (2012).
Reverse engineering the neural networks for rule extraction in classification problems.
Neural Process. Lett., 35(2):131–150
DOI:10.1007/s11063-011-9207-8.

[Bader, 2009] Bader, S. (2009).
Extracting propositional rules from feedforward neural networks by means of binary decision diagrams.
In d'Avila Garcez, A. S. and Hitzler, P., editors, *Proceedings of the Fifth International Workshop on Neural-Symbolic Learning and Reasoning, NeSy 2009, Pasadena, CA, USA, July 11, 2009*, volume 481 of *CEUR Workshop Proceedings*. CEUR-WS.org
http://ceur-ws.org/Vol-481/paper-5.pdf.

[Bader et al., 2007] Bader, S., Hölldobler, S., and Mayer-Eichberger, V. (2007).
Extracting propositional rules from feed-forward neural networks – A new decompositional approach.
In d'Avila Garcez, A. S., Hitzler, P., and Tamburrini, G., editors, *Proceedings of the 3rd International Workshop on Neural-Symbolic Learning and Reasoning, NeSy'07, held at IJCAI-07, Hyderabad, India, January 8, 2007*, volume 230 of *CEUR Workshop Proceedings*. CEUR-WS.org
http://ceur-ws.org/Vol-230/04-bader.pdf.

[Barakat and Bradley, 2007] Barakat, N. H. and Bradley, A. P. (2007).
Rule extraction from support vector machines: A sequential covering approach.
IEEE Trans. Knowl. Data Eng., 19(6):729–741
DOI:10.1109/TKDE.2007.190610.

References II

[Barakat and Diederich, 2008] Barakat, N. H. and Diederich, J. (2008).
Eclectic rule-extraction from support vector machines.
International Journal of Computer and Information Engineering, 2(5):1672–1675
DOI:10.5281/zenodo.1055511.

[Benítez et al., 1997] Benítez, J. M., Castro, J. L., and Requena, I. (1997).
Are artificial neural networks black boxes?
IEEE Trans. Neural Networks, 8(5):1156–1164
DOI:10.1109/72.623216.

[Berenji, 1991] Berenji, H. R. (1991).
Refinement of approximate reasoning-based controllers by reinforcement learning.
In Birnbaum, L. and Collins, G., editors, *Proceedings of the Eighth International Workshop (ML91), Northwestern University, Evanston, Illinois, USA*, pages 475–479. Morgan Kaufmann
DOI:10.1016/b978-1-55860-200-7.50097-0.

[Boz, 2002] Boz, O. (2002).
Converting a trained neural network to a decision tree DecText - decision tree extractor.
In Wani, M. A., Arabnia, H. R., Cios, K. J., Hafeez, K., and Kendall, G., editors, *Proceedings of the 2002 International Conference on Machine Learning and Applications - ICMLA 2002, June 24-27, 2002, Las Vegas, Nevada, USA*, pages 110–116. CSREA Press.

[Breiman et al., 1984] Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984).
Classification and Regression Trees.
CRC Press.

References III

[Browne et al., 2004] Browne, A., Hudson, B. D., Whitley, D. C., Ford, M. G., and Picton, P. (2004).
Biological data mining with neural networks: implementation and application of a flexible decision tree extraction algorithm to genomic problem domains.
Neurocomputing, 57:275–293
DOI:10.1016/j.neucom.2003.10.007.

[Brunk and Pazzani, 1991] Brunk, C. and Pazzani, M. J. (1991).
An investigation of noise-tolerant relational concept learning algorithms.
In Birnbaum, L. and Collins, G., editors, *Proceedings of the Eighth International Workshop (ML91), Northwestern University, Evanston, Illinois, USA*, pages 389–393. Morgan Kaufmann
DOI:10.1016/b978-1-55860-200-7.50080-5.

[Castillo et al., 2001] Castillo, L. A., González Muñoz, A., and Pérez, R. (2001).
Including a simplicity criterion in the selection of the best rule in a genetic fuzzy learning algorithm.
Fuzzy Sets Syst., 120(2):309–321
DOI:10.1016/S0165-0114(99)00095-0.

[Chan and Chan, 2017] Chan, V. and Chan, C. W. (2017).
Towards developing the piece-wise linear neural network algorithm for rule extraction.
Int. J. Cogn. Informatics Nat. Intell., 11(2):57–73
DOI:10.4018/IJCINI.2017040104.

[Chan and Chan, 2020] Chan, V. K. and Chan, C. W. (2020).
Towards explicit representation of an artificial neural network model: Comparison of two artificial neural network rule extraction approaches.
Petroleum, 6(4):329–339.
SI: Artificial Intelligence (AI), Knowledge-based Systems (KBS), and Machine Learning (ML)
DOI:https://doi.org/10.1016/j.petlm.2019.11.005.

References IV

[Chaves et al., 2005] Chaves, A. d. C. F., Vellasco, M. M. B. R., and Tanscheit, R. (2005).
Fuzzy rule extraction from support vector machines.
In Nedjah, N., de Macedo Mourelle, L., Abraham, A., and Köppen, M., editors, *5th International Conference on Hybrid Intelligent Systems (HIS 2005), 6-9 November 2005, Rio de Janeiro, Brazil*, pages 335–340. IEEE Computer Society
DOI:10.1109/ICHIS.2005.51.

[Chen, 2004] Chen, F. (2004).
Learning accurate and understandable rules from svm classifiers.
Master's thesis.

[Chen et al., 2007] Chen, Z., Li, J., and Wei, L. (2007).
A multiple kernel support vector machine scheme for feature selection and rule extraction from gene expression data of cancer tissue.
Artif. Intell. Medicine, 41(2):161–175
DOI:10.1016/j.artmed.2007.07.008.

[Ciatto et al., 2021] Ciatto, G., Calegari, R., and Omicini, A. (2021).
2P-Kt: A logic-based ecosystem for symbolic AI.
SoftwareX, 16:100817:1–7
DOI:10.1016/j.softx.2021.100817.

[Clark and Niblett, 1989] Clark, P. and Niblett, T. (1989).
The CN2 induction algorithm.
Mach. Learn., 3:261–283
DOI:10.1007/BF00116835.

References V

[Cohen, 1993] Cohen, W. W. (1993).
Efficient pruning methods for separate-and-conquer rule learning systems.
In Bajcsy, R., editor, *Proceedings of the 13th International Joint Conference on Artificial Intelligence. Chambéry, France, August 28 - September 3, 1993*, pages 988–994. Morgan Kaufmann.

[Cohen, 1995] Cohen, W. W. (1995).
Fast effective rule induction.
In Prieditis, A. and Russell, S. J., editors, *Machine Learning, Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, California, USA, July 9-12, 1995*, pages 115–123. Morgan Kaufmann
DOI:10.1016/b978-1-55860-377-6.50023-2.

[Craven and Shavlik, 1994] Craven, M. W. and Shavlik, J. W. (1994).
Using sampling and queries to extract rules from trained neural networks.
In *Machine Learning Proceedings 1994*, pages 37–45. Elsevier
DOI:10.1016/B978-1-55860-335-6.50013-1.

[Craven and Shavlik, 1996] Craven, M. W. and Shavlik, J. W. (1996).
Extracting tree-structured representations of trained networks.
In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E., editors, *Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference*, pages 24–30. The MIT Press
http://papers.nips.cc/paper/1152-extracting-tree-structured-representations-of-trained-networks.pdf.

[Crawford, 2016] Crawford, K. (2016).
Artificial intelligence's white guy problem.
The New York Times, 25.

References VI

[Etchells and G., 2006] Etchells, T. A. and G., L. P. J. (2006).
Orthogonal search-based rule extraction (OSRE) for trained neural networks: a practical and efficient approach.
IEEE Trans. Neural Networks, 17(2):374–384
DOI:10.1109/TNN.2005.863472.

[EU Parliament and Council, 2016] EU Parliament and Council (2016).
Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec.
http://data.europa.eu/eli/reg/2016/679/oj.
Online; accessed on October 11, 2019.

[Fu, 1994] Fu, L. (1994).
Rule generation from neural networks.
IEEE Trans. Syst. Man Cybern. Syst., 24(8):1114–1124
DOI:10.1109/21.299696.

[Fu et al., 2004] Fu, X., Ong, C., Keerthi, S., Hung, G. G., and Goh, L. (2004).
Extracting the knowledge embedded in support vector machines.
In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, volume 1, pages 291–296
DOI:10.1109/IJCNN.2004.1379916.

References VII

[Fung et al., 2005] Fung, G., Sandilya, S., and Rao, R. B. (2005).
Rule extraction from linear support vector machines.
In Grossman, R., Bayardo, R. J., and Bennett, K. P., editors, *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005*, pages 32–40. ACM
DOI:10.1145/1081870.1081878.

[Fürnkranz and Widmer, 1994] Fürnkranz, J. and Widmer, G. (1994).
Incremental reduced error pruning.
In Cohen, W. W. and Hirsh, H., editors, *Machine Learning, Proceedings of the Eleventh International Conference, Rutgers University, New Brunswick, NJ, USA, July 10-13, 1994*, pages 70–77. Morgan Kaufmann
DOI:10.1016/b978-1-55860-335-6.50017-9.

[Guidotti et al., 2018] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2018).
A survey of methods for explaining black box models.
ACM Computing Surveys, 51(5):1–42
DOI:10.1145/3236009.

[Halgamuge and Glesner, 1994] Halgamuge, S. K. and Glesner, M. (1994).
Neural networks in designing fuzzy systems for real world applications.
Fuzzy Sets and Systems, 65(1):1–12
DOI:https://doi.org/10.1016/0165-0114(94)90242-9.

References VIII

[Hayashi, 1990] Hayashi, Y. (1990).
A neural expert system with automated extraction of fuzzy if-then rules.
In Lippmann, R., Moody, J. E., and Touretzky, D. S., editors, *Advances in Neural Information Processing Systems 3, [NIPS Conference, Denver, Colorado, USA, November 26-29, 1990]*, pages 578–584. Morgan Kaufmann
http://papers.nips.cc/paper/355-a-neural-expert-system-with-automated-extraction-of-fuzzy-if-then-rules.

[He et al., 2006] He, J., Hu, H.-J., Harrison, R., Tai, P., and Pan, Y. (2006).
Rule generation for protein secondary structure prediction with support vector machines and decision tree.
IEEE Transactions on NanoBioscience, 5(1):46–53
DOI:10.1109/TNB.2005.864021.

[Hong and Chen, 1999] Hong, T. and Chen, J. (1999).
Finding relevant attributes and membership functions.
Fuzzy Sets Syst., 103(3):389–404
DOI:10.1016/S0165-0114(97)00187-5.

[Hong and Chen, 2000] Hong, T. and Chen, J. (2000).
Processing individual fuzzy attributes for fuzzy rule induction.
Fuzzy Sets Syst., 112(1):127–140
DOI:10.1016/S0165-0114(98)00179-1.

[Hong and Lee, 1996] Hong, T. and Lee, C. (1996).
Induction of fuzzy rules and membership functions from training examples.
Fuzzy Sets Syst., 84(1):33–47
DOI:10.1016/0165-0114(95)00305-3.

References IX

[Horikawa et al., 1992] Horikawa, S., Furuhashi, T., and Uchikawa, Y. (1992).
On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm.
IEEE Trans. Neural Networks, 3(5):801–806
DOI:10.1109/72.159069.

[Huysmans et al., 2006] Huysmans, J., Baesens, B., and Vanthienen, J. (2006).
ITER: An algorithm for predictive regression rule extraction.
In *Data Warehousing and Knowledge Discovery (DaWaK 2006)*, pages 270–279. Springer
DOI:10.1007/11823728_26.

[Ishibuchi et al., 1997] Ishibuchi, H., Nii, M., and Murata, T. (1997).
Linguistic rule extraction from neural networks and genetic-algorithm-based rule selection.
In *Proceedings of International Conference on Neural Networks (ICNN'97), Houston, TX, USA, June 9-12, 1997*, pages 2390–2395. IEEE
DOI:10.1109/ICNN.1997.614441.

[Kim and Lee, 2000] Kim, D. and Lee, J. (2000).
Handling continuous-valued attributes in decision tree with neural network modeling.
In López de Mántaras, R. and Plaza, E., editors, *Machine Learning: ECML 2000*, pages 211–219, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Konig et al., 2008] Konig, R., Johansson, U., and Niklasson, L. (2008).
G-REX: A versatile framework for evolutionary data mining.
In *2008 IEEE International Conference on Data Mining Workshops (ICDM 2008 Workshops)*, pages 971–974
DOI:10.1109/ICDMW.2008.117.

References X

[Krishnan et al., 1999a] Krishnan, R., Sivakumar, G., and Bhattacharya, P. (1999a).
Extracting decision trees from trained neural networks.
Pattern Recognit., 32(12):1999–2009
DOI:10.1016/S0031-3203(98)00181-2.

[Krishnan et al., 1999b] Krishnan, R., Sivakumar, G., and Bhattacharya, P. (1999b).
A search technique for rule extraction from trained neural networks.
Pattern Recognit. Lett., 20(3):273–280
DOI:10.1016/S0167-8655(98)00145-7.

[Lehmann et al., 2010] Lehmann, J., Bader, S., and Hitzler, P. (2010).
Extracting reduced logic programs from artificial neural networks.
Appl. Intell., 32(3):249–266
DOI:10.1007/s10489-008-0142-y.

[Lipton, 2018] Lipton, Z. C. (2018).
The mythos of model interpretability.
Queue, 16(3):31–57
DOI:10.1145/3236386.3241340.

[Liu et al., 2002] Liu, B., Abbass, H. A., and McKay, R. I. (2002).
Density-based heuristic for rule discovery with ant-miner.
In *The 6th Australia-Japan joint workshop on intelligent and evolutionary system*, volume 184.

[Liu et al., 2004] Liu, B., Abbass, H. A., and McKay, R. I. (2004).
Classification rule discovery with ant colony optimization.
IEEE Intell. Informatics Bull., 3(1):31–35
http://www.comp.hkbu.edu.hk/%7Ecib/2004/Feb/2004/Feb/cib_vol3no1_article4.pdf.

References XI

[Markowska-Kaczmar and Chumieja, 2004] Markowska-Kaczmar, U. and Chumieja, M. (2004).
Discovering the mysteries of neural networks.
Int. J. Hybrid Intell. Syst., 1(3-4):153–163
http://content.iospress.com/articles/international-journal-of-hybrid-intelligent-systems/his016.

[Markowska-Kaczmar and Trelak, 2003] Markowska-Kaczmar, U. and Trelak, W. (2003).
Extraction of fuzzy rules from trained neural network using evolutionary algorithm.
In *ESANN 2003, 11th European Symposium on Artificial Neural Networks, Bruges, Belgium, April 23-25, 2003, Proceedings*, pages 149–154
https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2003-9.pdf.

[Martens et al., 2009] Martens, D., Baesens, B., and Van Gestel, T. (2009).
Decompositional rule extraction from support vector machines by active learning.
IEEE Trans. Knowl. Data Eng., 21(2):178–191
DOI:10.1109/TKDE.2008.131.

[Martens et al., 2007] Martens, D., De Backer, M., Haesen, R., Vanthienen, J., Snoeck, M., and Baesens, B. (2007).
Classification with ant colony optimization.
IEEE Trans. Evol. Comput., 11(5):651–665
DOI:10.1109/TEVC.2006.890229.

[Masuoka et al., 1990] Masuoka, R., Watanabe, N., Kawamura, A., Owada, Y., and Asakawa, K. (1990).
Neurofuzzy systems – Fuzzy inference using a structured neural network.
In *Proceedings of International Conference on Fuzzy Logic and Neural Networks, Iizuka Japan, July, 1990*, pages 173–177.

References XII

[Matthews and Jagielska, 1995] Matthews, C. and Jagielska, I. (1995).
Fuzzy rule extraction from a trained multilayer neural network.
In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 2, pages 744–748 vol.2
DOI:10.1109/ICNN.1995.487510.

[Mitra, 1994] Mitra, S. (1994).
Fuzzy mlp based expert system for medical diagnosis.
Fuzzy Sets and Systems, 65(2):285–296.
Fuzzy Methods for Computer Vision and Pattern Recognition
DOI:https://doi.org/10.1016/0165-0114(94)90025-6.

[Murphy and Pazzani, 1991] Murphy, P. M. and Pazzani, M. J. (1991).
Id2-of-3: Constructive induction of m-of-n concepts for discriminators in decision trees.
In *Machine Learning Proceedings 1991*, pages 183–187. Elsevier.

[Nauck and Kruse, 1997] Nauck, D. D. and Kruse, R. (1997).
A neuro-fuzzy method to learn fuzzy classification rules from data.
Fuzzy Sets Syst., 89(3):277–288
DOI:10.1016/S0165-0114(97)00009-2.

[Nauck and Kruse, 1999] Nauck, D. D. and Kruse, R. (1999).
Neuro-fuzzy systems for function approximation.
Fuzzy Sets Syst., 101(2):261–271
DOI:10.1016/S0165-0114(98)00169-9.

References XIII

[Núñez et al., 2008] Núñez, H., Angulo, C., and Català, A. (2008).
Rule extraction based on support and prototype vectors.
In Diederich, J., editor, *Rule Extraction from Support Vector Machines*, volume 80 of *Studies in Computational Intelligence*, pages 109–134. Springer
DOI:10.1007/978-3-540-75390-2_5.

[Odajima et al., 2008] Odajima, K., Hayashi, Y., Tianxia, G., and Setiono, R. (2008).
Greedy rule generation from discrete data and its use in neural network rule extraction.
Neural Networks, 21(7):1020–1028
DOI:10.1016/j.neunet.2008.01.003.

[Parpinelli et al., 2001] Parpinelli, R. S., Lopes, H. S., and Freitas, A. A. (2001).
An ant colony based system for data mining: applications to medical data.
In *Proceedings of the genetic and evolutionary computation conference (GECCO-2001)*, pages 791–797. Citeseer.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011).
Scikit-learn: Machine learning in Python.
Journal of Machine Learning Research (JMLR), 12:2825–2830
https://dl.acm.org/doi/10.5555/1953048.2078195.

[Pop et al., 1994] Pop, E., Hayward, R., and Diederich, J. (1994).
RULENEG: Extracting rules from a trained ANN by stepwise negation.
Technical report, Neurocomputing Research Centre, Queensland University of Technology.

References XIV

[Quinlan, 1986] Quinlan, J. R. (1986).
Induction of decision trees.
Mach. Learn., 1(1):81–106
DOI:10.1023/A:1022643204877.

[Quinlan, 1993] Quinlan, J. R. (1993).
C4.5: Programming for machine learning.
Morgan Kauffmann
https://dl.acm.org/doi/10.5555/152181.

[Rabuñal et al., 2004] Rabuñal, J. R., Dorado, J., Pazos, A., Pereira, J., and Rivero, D. (2004).
A new approach to the extraction of ANN rules and to their generalization capacity through GP.
Neural Comput., 16(7):1483–1523
DOI:10.1162/089976604323057461.

[Ribeiro et al., 2016] Ribeiro, M. T., Singh, S., and Guestrin, C. (2016).
"why should I trust you?": Explaining the predictions of any classifier.
In Krishnapuram, B., Shah, M., Smola, A. J., Aggarwal, C. C., Shen, D., and Rastogi, R., editors, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144. ACM
DOI:10.1145/2939672.2939778.

[Saad and Wunsch II, 2007] Saad, E. W. and Wunsch II, D. C. (2007).
Neural network explanation using inversion.
Neural Networks, 20(1):78–93
DOI:10.1016/j.neunet.2006.07.005.

References XV

[Sabbatini et al., 2021a] Sabbatini, F., Ciatto, G., Calegari, R., and Omicini, A. (2021a).
On the design of PSyKE: A platform for symbolic knowledge extraction.
In Calegari, R., Ciatto, G., Denti, E., Omicini, A., and Sartor, G., editors, *WOA 2021 – 22nd Workshop "From Objects to Agents"*, volume 2963 of *CEUR Workshop Proceedings*, pages 29–48. Sun SITE Central Europe, RWTH Aachen University.
22nd Workshop "From Objects to Agents" (WOA 2021), Bologna, Italy, 1–3 September 2021. Proceedings
http://ceur-ws.org/Vol-2963/paper14.pdf.

[Sabbatini et al., 2022a] Sabbatini, F., Ciatto, G., Calegari, R., and Omicini, A. (2022a).
Symbolic knowledge extraction from opaque ML predictors in PSyKE: Platform design & experiments.
Intelligenza Artificiale, 16(1):27–48
DOI:10.3233/IA-210120.

[Sabbatini et al., 2021b] Sabbatini, F., Ciatto, G., and Omicini, A. (2021b).
GridEx: An algorithm for knowledge extraction from black-box regressors.
In Calvaresi, D., Najjar, A., Winikoff, M., and Främling, K., editors, *Explainable and Transparent AI and Multi-Agent Systems. Third International Workshop, EXTRAAMAS 2021, Virtual Event, May 3–7, 2021, Revised Selected Papers*, volume 12688 of *LNCS*, pages 18–38. Springer Nature, Basel, Switzerland
DOI:10.1007/978-3-030-82017-6_2.

[Sabbatini et al., 2022b] Sabbatini, F., Ciatto, G., and Omicini, A. (2022b).
Semantic web-based interoperability for intelligent agents with PSyKE.
In Calvaresi, D., Najjar, A., Winikoff, M., and Främling, K., editors, *Explainable and Transparent AI and Multi-Agent Systems*, volume 13283 of *Lecture Notes in Computer Science*, chapter 8, pages 124–142. Springer
DOI:10.1007/978-3-031-15565-9_8.

References XVI

[Saito and Nakano, 1988] Saito, K. and Nakano, R. (1988).
Medical diagnostic expert system based on PDP model.
In *Proceedings of International Conference on Neural Networks (ICNN'88), San Diego, CA, USA, July 24-27, 1988*, pages 255–262. IEEE
DOI:10.1109/ICNN.1988.23855.

[Saito and Nakano, 1997] Saito, K. and Nakano, R. (1997).
Law discovery using neural networks.
In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence, IJCAI 97, Nagoya, Japan, August 23-29, 1997, 2 Volumes*, pages 1078–1083. Morgan Kaufmann
http://ijcai.org/Proceedings/97-2/Papers/042.pdf.

[Saito and Nakano, 2002] Saito, K. and Nakano, R. (2002).
Extracting regression rules from neural networks.
Neural Networks, 15(10):1279–1288
DOI:10.1016/S0893-6080(02)00089-8.

[Sato and Tsukimoto, 2001] Sato, M. and Tsukimoto, H. (2001).
Rule extraction from neural networks via decision tree induction.
In *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*, volume 3, pages 1870–1875. IEEE
DOI:10.1109/IJCNN.2001.938448.

[Schetinin et al., 2007] Schetinin, V., Fieldsend, J. E., Partridge, D., Coats, T. J., Krzanowski, W. J., Everson, R. M., Bailey, T. C., and Hernandez, A. (2007).
Confident interpretation of bayesian decision tree ensembles for clinical applications.
IEEE Trans. Inf. Technol. Biomed., 11(3):312–319
DOI:10.1109/TITB.2006.880553.

References XVII

[Selbst and Powles, 2017] Selbst, A. D. and Powles, J. (2017).
Meaningful information and the right to explanation.
International Data Privacy Law, 7(4):233–242
DOI:10.1093/idpl/ipx022.

[Sestito and Dillon, 1994] Sestito, S. and Dillon, T. S. (1994).
Automated knowledge acquisition.
Prentice Hall International series in computer science and engineering. Prentice Hall.

[Sethi et al., 2012] Sethi, K. K., Mishra, D. K., and Mishra, B. (2012).
KDRuleEx: A novel approach for enhancing user comprehensibility using rule extraction.
In *2012 Third International Conference on Intelligent Systems Modelling and Simulation*, pages 55–60
DOI:10.1109/ISMS.2012.116.

[Setiono, 1997] Setiono, R. (1997).
Extracting rules from neural networks by pruning and hidden-unit splitting.
Neural Comput., 9(1):205–225
DOI:10.1162/neco.1997.9.1.205.

[Setiono, 2000] Setiono, R. (2000).
Extracting M-of-N rules from trained neural networks.
IEEE Trans. Neural Networks Learn. Syst., 11(2):512–519
DOI:10.1109/72.839020.

[Setiono et al., 2008] Setiono, R., Baesens, B., and Mues, C. (2008).
Recursive neural network rule extraction for data with mixed attributes.
IEEE Trans. Neural Networks, 19(2):299–307
DOI:10.1109/TNN.2007.908641.

References XVIII

[Setiono and Leow, 2000] Setiono, R. and Leow, W. K. (2000).
FERNN: An algorithm for fast extraction of rules from neural networks.
Appl. Intell., 12(1-2):15–25
DOI:10.1023/A:1008307919726.

[Setiono et al., 2002] Setiono, R., Leow, W. K., and Zurada, J. M. (2002).
Extraction of rules from artificial neural networks for nonlinear regression.
IEEE Transactions on Neural Networks, 13(3):564–577
DOI:10.1109/TNN.2002.1000125.

[Setiono and Liu, 1996] Setiono, R. and Liu, H. (1996).
Symbolic representation of neural networks.
Computer, 29(3):71–77
DOI:10.1109/2.485895.

[Setiono and Liu, 1997] Setiono, R. and Liu, H. (1997).
Neurolinear: A system for extracting oblique decision rules from neural networks.
In van Someren, M. and Widmer, G., editors, *Machine Learning: ECML-97, 9th European Conference on Machine Learning, Prague, Czech Republic, April 23-25, 1997, Proceedings*, volume 1224 of *Lecture Notes in Computer Science*, pages 221–233. Springer
DOI:10.1007/3-540-62858-4_87.

[Setiono and Thong, 2004] Setiono, R. and Thong, J. Y. L. (2004).
An approach to generate rules from neural networks for regression problems.
Eur. J. Oper. Res., 155(1):239–250
DOI:10.1016/S0377-2217(02)00792-0.

References XIX

[Taha and Ghosh, 1999] Taha, I. A. and Ghosh, J. (1999).
Symbolic interpretation of artificial neural networks.
IEEE Trans. Knowl. Data Eng., 11(3):448–463
DOI:10.1109/69.774103.

[Thrun, 1993] Thrun, S. B. (1993).
Extracting provably correct rules from artificial neural networks.
Technical report, University of Bonn.

[Tickle et al., 1996] Tickle, A. B., Orlowski, M., and Diederich, J. (1996).
DEDEC: A methodology for extracting rules from trained artificial neural networks.
In Andrews, R. and Diederich, J., editors, *Rules and Networks: Proceedings of the Rule Extraction from Trained Artificial Neural Networks Workshop*, pages 90–102. Neurocomputing Research Centre, Queensland University of Technology.

[Torres and Rocco, 2005] Torres, D. E. D. and Rocco, C. M. S. (2005).
Extracting trees from trained SVM models using a TREPAN based approach.
In Nedjah, N., de Macedo Mourelle, L., Abraham, A., and Köppen, M., editors, *5th International Conference on Hybrid Intelligent Systems (HIS 2005), 6-9 November 2005, Rio de Janeiro, Brazil*, pages 353–360. IEEE Computer Society
DOI:10.1109/ICHIS.2005.41.

[Towell and Shavlik, 1991] Towell, G. G. and Shavlik, J. W. (1991).
Interpretation of artificial neural networks: Mapping knowledge-based neural networks into rules.
In Moody, J. E., Hanson, S. J., and Lippmann, R., editors, *Advances in Neural Information Processing Systems 4, [NIPS Conference, Denver, Colorado, USA, December 2-5, 1991]*, pages 977–984. Morgan Kaufmann
http://papers.nips.cc/paper/546-interpretation-of-artificial-neural-networks-mapping-knowledge-based-neural-networks-into-rules.

References XX

[Towell and Shavlik, 1993] Towell, G. G. and Shavlik, J. W. (1993).
Extracting refined rules from knowledge-based neural networks.
Machine Learning, 13(1):71–101
DOI:10.1007/BF00993103.

[Tresp et al., 1992] Tresp, V., Hollatz, J., and Ahmad, S. (1992).
Network structuring and training using rule-based knowledge.
In Hanson, S. J., Cowan, J. D., and Giles, C. L., editors, *Advances in Neural Information Processing Systems 5, [NIPS Conference, Denver, Colorado, USA, November 30 - December 3, 1992]*, pages 871–878. Morgan Kaufmann
http://papers.nips.cc/paper/638-network-structuring-and-training-using-rule-based-knowledge.

[Tsukimoto, 2000] Tsukimoto, H. (2000).
Extracting rules from trained neural networks.
IEEE Trans. Neural Networks Learn. Syst., 11(2):377–389
DOI:10.1109/72.839008.

[van Gelder, 1990] van Gelder, T. (1990).
Why distributed representation is inherently non-symbolic.
In Dorffner, G., editor, *Konnektionismus in Artificial Intelligence und Kognitionsforschung. Proceedings 6. Österreichische Artificial Intelligence-Tagung (KONNAI), Salzburg, Österreich, 18. bis 21. September 1990*, volume 252 of *Informatik-Fachberichte*, pages 58–66. Springer
DOI:10.1007/978-3-642-76070-9_6.

[Wang et al., 2020] Wang, S., Wang, Y., Wang, D., Yin, Y., Wang, Y., and Jin, Y. (2020).
An improved random forest-based rule extraction method for breast cancer diagnosis.
Appl. Soft Comput., 86
DOI:10.1016/j.asoc.2019.105941.

References XXI

[Wexler, 2017] Wexler, R. (2017).
When a computer program keeps you in jail: How computers are harming criminal justice.
New York Times.

[Wikipedia contributors, 2021] Wikipedia contributors (2021).
Decision tree learning — Wikipedia, the free encyclopedia.
`https://en.wikipedia.org/w/index.php?title=Decision_tree_learning`.
[Online; accessed 17-September-2021].

[Yedjour and Benyettou, 2018] Yedjour, D. and Benyettou, A. (2018).
Symbolic interpretation of artificial neural networks based on multiobjective genetic algorithms and association rules mining.
Appl. Soft Comput., 72:177–188
DOI:10.1016/j.asoc.2018.08.007.

[Yuan and Zhuang, 1996] Yuan, Y. and Zhuang, H. (1996).
A genetic algorithm for generating fuzzy classification rules.
Fuzzy Sets Syst., 84(1):1–19
DOI:10.1016/0165-0114(95)00302-9.

[Zhang et al., 2005] Zhang, Y., Su, H., Jia, T., and Chu, J. (2005).
Rule extraction from trained support vector machines.
In Ho, T. B., Cheung, D. W., and Liu, H., editors, *Advances in Knowledge Discovery and Data Mining, 9th Pacific-Asia Conference, PAKDD 2005, Hanoi, Vietnam, May 18-20, 2005, Proceedings*, volume 3518 of *Lecture Notes in Computer Science*, pages 61–70. Springer
DOI:10.1007/11430919_9.

References XXII

[Zhou et al., 2000] Zhou, Z., Chen, S., and Chen, Z. (2000).
A statistics based approach for extracting priority rules from trained neural networks.
In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, IJCNN 2000, Neural Computing: New Challenges and Perspectives for the New Millennium, Como, Italy, July 24-27, 2000, Volume 3*, pages 401–406. IEEE Computer Society
DOI:10.1109/IJCNN.2000.861337.

[Zhou et al., 2003] Zhou, Z., Jiang, Y., and Chen, S. (2003).
Extracting symbolic rules from trained neural network ensembles.
AI Commun., 16(1):3–15
http://content.iospress.com/articles/ai-communications/aic272.

[Zilke et al., 2016] Zilke, J. R., Mencía, E. L., and Janssen, F. (2016).
DeepRED – Rule extraction from deep neural networks.
In Calders, T., Ceci, M., and Malerba, D., editors, *Discovery Science - 19th International Conference, DS 2016, Bari, Italy, October 19-21, 2016, Proceedings*, volume 9956 of *Lecture Notes in Computer Science*, pages 457–473
DOI:10.1007/978-3-319-46307-0_29.

EXPECTATION

EXPECTATION

PERSONALIZED EXPLAINABLE ARTIFICIAL INTELLIGENCE FOR DECENTRALIZED AGENTS WITH HETEROGENEOUS KNOWLEDGE

This presentation was partially supported by the CHIST-ERA IV project "EXPECTATION" – CHIST-ERA-19-XAI-005 –, co-funded by EU and the Italian MUR (Ministry for University and Research).