HAL
archives-ouvertes.fr

Admissible Generalizations of Examples as Rules

Philippe Besnard, Thomas Guyet, Véronique Masson

To cite this version:

Philippe Besnard, Thomas Guyet, Véronique Masson. Admissible Generalizations of Examples as Rules. One-day Workshop on Machine Learning and Explainability 2018, Oct 2018, Orléans, France. pp.1-29. hal-01940129

HAL Id: hal-01940129
https://hal.inria.fr/hal-01940129

Submitted on 30 Nov 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Admissible generalizations of examples as rules

Philippe Besnard[1] — *Thomas Guyet*[3] — *Véronique Masson*[2,3]

[1] *CNRS-IRIT* [2] *Université Rennes-1* [3] *IRISA/LACODAM*

Machine Learning and Explainability – 8 October 2018

And now . . .

Introduction

Formalizing rule learning
- General formalism
- Notations

Admissibility for generalization
- Formalizing admissibility
- Classes of choice functions

Application to an analysis of CN2

Conclusion

Attribute-value rule learning

	(C) Price	(A_1) Area	(A_2) Rooms	(A_3) Energy	(A_4) Town	(A_5) District	(A_6) Exposure
1	low-priced	70	2	D	Toulouse	Minimes	
2	low-priced	75	4	D	Toulouse	Rangueil	
3	expensive	65	3		Toulouse	Downtown	
4	low-priced	32	2	D	Toulouse		SE
5	mid-priced	65	2	D	Rennes		SO
6	expensive	100	5	C	Rennes	Downtown	
7	low-priced	40	2	D	Betton		S

- Task: induce rules to predict the value of the class attribute (C)
- Rules extracted by Algorithm CN2

$$\pi_1^{CN2} : A_5 = \text{Downtown} \Rightarrow C = \text{expensive}$$
$$\pi_2^{CN2} : A_2 < 2.50 \wedge A_4 = \text{Toulouse} \Rightarrow C = \text{low-priced}$$
$$\pi_3^{CN2} : A_1 > 36.00 \wedge A_3 = D \Rightarrow C = \text{low-priced}$$

Interpretability of rules and rulesets

- The logical structure of a rule can be easily interpreted by users

  IF *conditions* THEN *class-label*

- Rule learning algorithms generate rules according to implicit or explicit principles[1]
  - are the generated rules the *interpretable* ones?
  - would it be possible to have different rulesets?
  - why a ruleset would be better than another one from the interpretability point of view?

⇒ We need ways to analyze the interpretability of the outputs of rule learning algorithms

[1] principles mainly based on statistical properties!

Analyzing the interpretability of rules

Analyzing the interpretativeness of ruleset

- Objective criteria on ruleset syntax [CZV13, BS15]
  - size of the rule (number of attributes)
  - size of the ruleset
- Intuitiveness of rules through the effects of cognitive biases [KBF18]

$\Rightarrow$ Our approach formalizes rule learning and some expected properties on rules to shed light on properties of some extracted ruleset

Analyzing the interpretability of rules

Analyzing the interpretativeness of ruleset

- Objective criteria on ruleset syntax [CZV13, BS15]
  - size of the rule (number of attributes)
  - size of the ruleset
- Intuitiveness of rules through the effects of cognitive biases [KBF18]

⇒ Our approach formalizes rule learning and some expected properties on rules to shed light on properties of some extracted ruleset

In this talk

- We present the formalisation of rule learning, and we focus on the generalization of examples as a rule
- We introduce the notion of admissible rule that attempts to capture an intuitive generalization of the examples
- We develop the example of numerical attributes

Impact of examples generalization on rule interpretability

	(C) *Price*	(A_1) *Area*	(A_2) *Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
1	low-priced	70	2	D	Toulouse	Minimes	
2	low-priced	75	4	D	Toulouse	Rangueil	
3	expensive	65	3		Toulouse	Downtown	
4	low-priced	32	2	D	Toulouse		SE
5	mid-priced	65	2	D	Rennes		SO
6	expensive	100	5	C	Rennes	Downtown	
7	low-priced	40	2	D	Betton		S

- Rules extracted by Algorithm CN2

$$\pi_1^{CN2} : A_5 = \text{Downtown} \Rightarrow C = \text{expensive}$$
$$\pi_2^{CN2} : A_2 < 2.50 \wedge A_4 = \text{Toulouse} \Rightarrow C = \text{low-priced}$$
$$\pi_3^{CN2} : A_1 > 36.00 \wedge A_3 = D \Rightarrow C = \text{low-priced}$$

Impact of examples generalization on rule interpretability

	(C) *Price*	(A_1) *Area*	(A_2) *Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
1	low-priced	70	2	D	Toulouse	Minimes	
2	low-priced	75	4	D	Toulouse	Rangueil	
3	expensive	65	3		Toulouse	Downtown	
4	low-priced	32	2	D	Toulouse		SE
5	mid-priced	65	2	D	Rennes		SO
6	expensive	100	5	C	Rennes	Downtown	
7	low-priced	40	2	D	Betton		S

- Rules extracted by Algorithm CN2

$$\pi_1^{CN2} : A_5 = \text{Downtown} \Rightarrow C = \text{expensive}$$
$$\pi_2^{CN2} : A_2 < 2.50 \wedge A_4 = \text{Toulouse} \Rightarrow C = \text{low-priced}$$
$$\pi_3^{CN2} : A_1 > 36.00 \wedge A_3 = D \Rightarrow C = \text{low-priced}$$

Toward the notion of admissibility

	(C) *Price*	(A_1) *Area*
1	low-priced	70
2	low-priced	75
4	low-priced	32
7	low-priced	40

- Rote learning of a rule

$$A_1 = \{75\} \Rightarrow C = \text{low-priced}$$

- Most generalizing rule

$$A_1 = [32 : 75] \Rightarrow C = \text{low-priced}$$

- Would the following rule be better?

$$A_1 = [32 : 40] \cup [70 : 75] \Rightarrow C = \text{low-priced}$$

Toward the notion of admissibility

	(C) *Price*	(A_1) *Area*
1	low-priced	70
2	low-priced	75
4	low-priced	32
7	low-priced	40

- Rote learning of a rule
  $$A_1 = \{75\} \Rightarrow C = \text{low-priced}$$
- Most generalizing rule
  $$A_1 = [32 : 75] \Rightarrow C = \text{low-priced}$$
- Would the following rule be better?

$$A_1 = [32 : 40] \cup [70 : 75] \Rightarrow C = \text{low-priced}$$

$\Rightarrow$ this is the question of admissibility!

Toward the notion of admissibility

	(C) *Price*	(A_1) *Area*
1	low-priced	70
2	low-priced	75
4	low-priced	32
7	low-priced	40

- Rote learning of a rule
  $$A_1 = \{75\} \Rightarrow C = \text{low-priced}$$
- Most generalizing rule
  $$A_1 = [32 : 75] \Rightarrow C = \text{low-priced}$$
- Would the following rule be better?
  $$A_1 = [32 : 40] \cup [70 : 75] \Rightarrow C = \text{low-priced}$$

$\Rightarrow$ this is the question of admissibility!

The notion of admissibility has to capture an intuitive notion of generalization . . .

And now . . .

Introduction

Formalizing rule learning

- General formalism
- Notations

Admissibility for generalization

- Formalizing admissibility
- Classes of choice functions

Application to an analysis of CN2

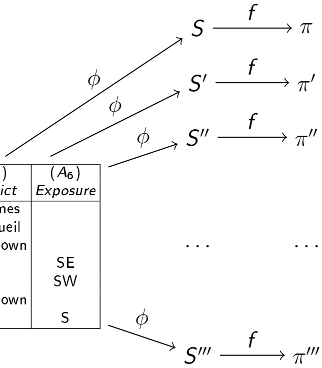
Conclusion

At a glance

Rule learning is formalized by two main functions

- ϕ: selects possible subsets of data
- f: generalizes examples as a rule (LearnOneRule process [Mit82])

(C) *Price*	(A_1) *Area*	(A_2) *#Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
low-priced	70	2	D	Toulouse	Minimes	
low-priced	75	4	D	Toulouse	Rangueil	
expensive	65	3		Toulouse	Downtown	
low-priced	32	2	D	Toulouse		SE
mid-priced	65	2	D	Rennes		SW
expensive	100	5	C	Rennes	Downtown	
low-priced	40	2	D	Betton		S



At a glance

Rule learning is formalized by two main functions

- ϕ: selects possible subsets of data
- f: generalizes examples as a rule (LearnOneRule process [Mit82])

(C) *Price*	(A_1) *Area*	(A_2) *#Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
low-priced	70	2	D	Toulouse	Minimes	
low-priced	75	4	D	Toulouse	Rangueil	
expensive	65	3		Toulouse	Downtown	
low-priced	32	2	D	Toulouse		SE
mid-priced	65	2	D	Rennes		SW
expensive	100	5	C	Rennes	Downtown	
low-priced	40	2	D	Betton		S

$\xrightarrow{\phi} S \xrightarrow{f} \pi$

$\xrightarrow{\phi} S' \xrightarrow{f} \pi'$

$\xrightarrow{\phi} S'' \xrightarrow{f} \pi''$

$\dots \qquad \dots$

$\xrightarrow{\phi} S''' \xrightarrow{f} \pi'''$

At a glance

Rule learning is formalized by two main functions

- ϕ: selects possible subsets of data
- f: generalizes examples as a rule (LearnOneRule process [Mit82])

(C) *Price*	(A_1) *Area*	(A_2) *#Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
low-priced	70	2	D	Toulouse	Minimes	
low-priced	75	4	D	Toulouse	Rangueil	
expensive	65	3		Toulouse	Downtown	
low-priced	32	2	D	Toulouse		SE
mid-priced	65	2	D	Rennes		SW
expensive	100	5	C	Rennes	Downtown	
low-priced	40	2	D	Betton		S

$\xrightarrow{\phi} S \xrightarrow{f} \pi$

$\xrightarrow{\phi} S' \xrightarrow{f} \pi'$

$\xrightarrow{\phi} S'' \xrightarrow{f} \pi''$

$\dots \quad \dots$

$\xrightarrow{\phi} S''' \xrightarrow{f} \pi'''$

The attribute-value model

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	$\cdots$	A_n
item 1	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	$a_{1,6}$	$a_{1,7}$	$\cdots$	$a_{1,n}$
item 2	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{2,6}$	$a_{2,7}$	$\cdots$	$a_{2,n}$
item 3	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	$a_{3,6}$	$a_{3,7}$	$\cdots$	$a_{3,n}$
item 4	$a_{4,1}$	$a_{4,2}$	$a_{4,3}$	$a_{4,4}$	$a_{4,5}$	$a_{4,6}$	$a_{4,7}$	$\cdots$	$a_{4,n}$
item 5	$a_{5,1}$	$a_{5,2}$	$a_{5,3}$	$a_{5,4}$	$a_{5,5}$	$a_{5,6}$	$a_{5,7}$	$\cdots$	$a_{5,n}$
item 6	$a_{6,1}$	$a_{6,2}$	$a_{6,3}$	$a_{6,4}$	$a_{6,5}$	$a_{6,6}$	$a_{6,7}$	$\cdots$	$a_{6,n}$
item 7	$a_{7,1}$	$a_{7,2}$	$a_{7,3}$	$a_{7,4}$	$a_{7,5}$	$a_{7,6}$	$a_{7,7}$	$\cdots$	$a_{7,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
item m	$a_{m,1}$	$a_{m,2}$	$a_{m,3}$	$a_{m,4}$	$a_{m,5}$	$a_{m,6}$	$a_{m,7}$	$\cdots$	$a_{m,n}$

$$\left.\begin{array}{ll} \text{Rows:} & \text{items } x_1, x_2, \ldots, x_m \\ \text{Columns:} & \text{attributes } A_1, A_2, \ldots, A_n \end{array}\right] \forall i, j \;\; a_{j,i} \in \mathrm{Rng}\, A_i$$

$\mathrm{Rng}\, A_i$ *denotes the set of possible values for attribute* A_i

Subsets of data to generalize

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9
item 1	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	$a_{1,6}$	$a_{1,7}$	$a_{1,8}$	$a_{1,9}$
item 2	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{2,6}$	$a_{2,7}$	$a_{2,8}$	$a_{2,9}$
item 3	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	$a_{3,6}$	$a_{3,7}$	$a_{3,8}$	$a_{3,9}$
item 4	$a_{4,1}$	$a_{4,2}$	$a_{4,3}$	$a_{4,4}$	$a_{4,5}$	$a_{4,6}$	$a_{4,7}$	$a_{4,8}$	$a_{4,9}$
item 5	$a_{5,1}$	$a_{5,2}$	$a_{5,3}$	$a_{5,4}$	$a_{5,5}$	$a_{5,6}$	$a_{5,7}$	$a_{5,8}$	$a_{5,9}$
item 6	$a_{6,1}$	$a_{6,2}$	$a_{6,3}$	$a_{6,4}$	$a_{6,5}$	$a_{6,6}$	$a_{6,7}$	$a_{6,8}$	$a_{6,9}$
item 7	$a_{7,1}$	$a_{7,2}$	$a_{7,3}$	$a_{7,4}$	$a_{7,5}$	$a_{7,6}$	$a_{7,7}$	$a_{7,8}$	$a_{7,9}$
item 8	$a_{8,1}$	$a_{8,2}$	$a_{8,3}$	$a_{8,4}$	$a_{8,5}$	$a_{8,6}$	$a_{8,7}$	$a_{8,8}$	$a_{8,9}$
item 9	$a_{9,1}$	$a_{9,2}$	$a_{9,3}$	$a_{9,4}$	$a_{9,5}$	$a_{9,6}$	$a_{9,7}$	$a_{9,8}$	$a_{9,9}$

$\Rightarrow$ "Square" $=$ selection of rows and columns in the data

Rules

A rule π expresses constraints (for a generic item x) which lead to conclusion $C(x)$ (class which the item belongs to)

$$\pi : A_1(x) \in v_1^\pi \wedge \cdots \wedge A_n(x) \in v_n^\pi \to C(x) \in v_0^\pi \qquad (*)$$

where $\begin{cases} v_i^\pi \subseteq \mathrm{Rng}\, A_i \text{ pour } i = 1, \ldots, n, \\ v_0^\pi \subseteq \mathrm{Rng}\, C. \end{cases}$

attributes $i = 1, \ldots, n$ without constraints are such that $v_i^\pi = \mathrm{Rng}\, A_i$.

And now . . .

Introduction

Formalizing rule learning
- General formalism
- Notations

Admissibility for generalization
- Formalizing admissibility
- Classes of choice functions

Application to an analysis of CN2

Conclusion

Eliciting a rule

- S being a square is supposed to capture a rule π requires that every item of S satisfies π

  $\rightarrow$ generalisation does not capture the statistical representativeness of dataset, but only elicits a rule generalizing all its items

(C) *Price*	(A_1) *Area*	(A_2) *#Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
low-priced	70	2	D	Toulouse	Minimes	
low-priced	75	4	D	Toulouse	Rangueil	
expensive	65	3		Toulouse	Downtown	
low-priced	32	2	D	Toulouse		SE
mid-priced	65	2	D	Rennes		SW
expensive	100	5	C	Rennes	Downtown	
low-priced	40	2	D	Betton		S

$$\downarrow f$$

$$A_0 = 2 \wedge A_4 = \text{Toulouse} \Rightarrow C = \text{low-priced}$$

(C) *Price*	(A_1) *Area*	(A_2) *#Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
low-priced	70	2	D	Toulouse	Minimes	
low-priced	75	4	D	Toulouse	Rangueil	
expensive	65	3		Toulouse	Downtown	
low-priced	32	2	D	Toulouse		SE
mid-priced	65	2	D	Rennes		SW
expensive	100	5	C	Rennes	Downtown	
low-priced	40	2	D	Betton		S

$$\downarrow f$$

$$A_0 \in [2, 4] \Rightarrow C \in \{\text{low-priced}, \text{expensive}\}$$

Eliciting a rule (f function)

	(A_0) *Price*	(A_1) *Area*	(A_2) *Rooms*
1	low-priced	70	2
2	low-priced	75	4
4	low-priced	32	2
7	low-priced	40	2

$S_0 = \{\text{low-priced}\}$

$S_1 = \{32, 40, 70, 75\}$

$S_2 = \{2, 4\}$

- For every attribute A_i, S_i is the set of values of A_i in items of S

Eliciting a rule (f function)

	(A_0) *Price*	(A_1) *Area*	(A_2) *Rooms*
1	low-priced	70	2
2	low-priced	75	4
4	low-priced	32	2
7	low-priced	40	2

$S_0 = \{\text{low-priced}\}$ $\quad S_2 = \{2, 4\}$

$S_1 = \{32, 40, 70, 75\}$

$\widehat{S}_0 \quad \widehat{S}_1 \quad \widehat{S}_2$

- For every attribute A_i, S_i is the set of values of A_i in items of S
- Each superset of S_i is, theoretically speaking, a generalization of S_i

Eliciting a rule (f function)

	(A_0) *Price*	(A_1) *Area*	(A_2) *Rooms*
1	low-priced	70	2
2	low-priced	75	4
4	low-priced	32	2
7	low-priced	40	2

$$S_0 = \{\text{low-priced}\} \qquad S_1 = \{32, 40, 70, 75\} \qquad S_2 = \{2, 4\}$$

$$S_0 \xrightarrow{f} \widehat{S}_0 \qquad S_1 \rightarrow \widehat{S}_1 \qquad S_2 \xrightarrow{f} \widehat{S}_2$$

- For every attribute A_i, S_i is the set of values of A_i in items of S
- Each superset of S_i is, theoretically speaking, a generalization of S_i
- The generalisation process thus consists in selecting one of these supersets:
  - f choice function that is given as input a collection of supersets of S_i and picks one

We are looking for an appropriate $\widehat{\cdot}$ for (§) i.e.

$$A_1(x) \in \widehat{S}_1 \wedge \cdots \wedge A_n(x) \in \widehat{S}_n \rightarrow C(x) \in \widehat{S}_0 \qquad (\S)$$

Eliciting a rule (f function)

	(A_0) *Price*	(A_1) *Area*	(A_2) *Rooms*
1	low-priced	70	2
2	low-priced	75	4
4	low-priced	32	2
7	low-priced	40	2

$S_0 = \{\text{low-priced}\}$ $\quad$ $S_2 = \{2, 4\}$

$S_1 = \{32, 40, 70, 75\}$

$S_0 \xrightarrow{f} \widehat{S}_0 \qquad S_1 \xrightarrow{f} \widehat{S}_1 \qquad S_2 \xrightarrow{f} \widehat{S}_2$

- For every attribute A_i, S_i is the set of values of A_i in items of S
- Each superset of S_i is, theoretically speaking, a generalization of S_i
- The generalisation process thus consists in selecting one of these supersets:
  - f choice function that is given as input a collection of supersets of S_i and picks one

We are looking for an appropriate $\widehat{\cdot}$ for (§) i.e.

$$A_1(x) \in \widehat{S}_1 \wedge \cdots \wedge A_n(x) \in \widehat{S}_n \rightarrow C(x) \in \widehat{S}_0 \qquad (\S)$$

Generalization of S_i : $\widehat{S}_i = f(\{Y \mid S_i \subseteq Y \subseteq \mathrm{Rng}\, A_i\})$

Notion of admissibility: propositions

Generalization of S_i: $\widehat{S}_i = f(\{Y \mid S_i \subseteq Y \subseteq \mathrm{Rng}\, A_i\})$

What collection $\mathcal{X} = \{\widehat{S}_i \mid S_i \subseteq \mathrm{Rng}\, A_i\}$ would do?

What choice function(s) can in practice capture these expected algebraic properties?

Notion of admissibility: propositions

Generalization of S_i : $\widehat{S_i} = f(\{Y \mid S_i \subseteq Y \subseteq \mathrm{Rng}\, A_i\})$

What collection $\mathcal{X} = \{\widehat{S_i} \mid S_i \subseteq \mathrm{Rng}\, A_i\}$ would do?

(i) $\mathrm{Rng}\, A_i \in \mathcal{X}$

(ii) if X and Y are in $\mathcal{X}$ then so $X \cap Y$.

- $\mathcal{X}$ is a closure system upon $\mathrm{Rng}\, A_i$.
- $\widehat{\cdot}$ is an operation enjoying weaker properties than closure operators; alternatives looked at:
  - pre-closure operator
  - capping operator

What choice function(s) can in practice capture these expected algebraic properties?

Notion of admissibility: propositions

Generalization of S_i: $\widehat{S_i} = f(\{Y \mid S_i \subseteq Y \subseteq \mathrm{Rng}\, A_i\})$

What collection $\mathcal{X} = \{\widehat{S_i} \mid S_i \subseteq \mathrm{Rng}\, A_i\}$ would do?

(i) $\mathrm{Rng}\, A_i \in \mathcal{X}$

(ii) if X and Y are in $\mathcal{X}$ then so $X \cap Y$.

- $\mathcal{X}$ is a closure system upon $\mathrm{Rng}\, A_i$.
- $\widehat{\cdot}$ is an operation enjoying weaker properties than closure operators; alternatives looked at:
  - pre-closure operator
  - capping operator

What choice function(s) can in practice capture these expected algebraic properties?

- Proposal for some classes of choice functions generating specific types of operators
- Concrete examples of such functions for numerical rules

Weakening closure operators

- List of Kuratowski's axioms [Kur14] (closure system):

$$\begin{aligned}
&\widehat{\emptyset} = \emptyset \\
&S \subseteq \widehat{S} \subseteq \mathrm{Rng}\, A_i \\
&\widehat{\widehat{S}} = \widehat{S} \\
&\widehat{S \cup S'} = \widehat{S} \cup \widehat{S'} \qquad \text{(pre-closure)}
\end{aligned}$$

- Actually, we downgrade Kuratowski's axioms as follows

$$\begin{aligned}
&\widehat{S} \subseteq \widehat{S'} \text{ whenever } S \subseteq S' && \text{(closure)} \\
&\widehat{S} = \widehat{S'} \text{ whenever } S \subseteq S' \subseteq \widehat{S} && \text{(cumulation)} \\
&\widehat{S \cup S'} \subseteq \widehat{S} \text{ whenever } S' \subseteq \widehat{S} && \text{(capping)}
\end{aligned}$$

Lemma: Kuratowksi $\Rightarrow$ closure $\Rightarrow$ cumulation $\Rightarrow$ capping

Class of choice functions satisfying pre-closure

Theorem. *Given a set Z, let $f : 2^{2^Z} \to 2^Z$ be a function st for every upward closed $\mathcal{X} \subseteq 2^Z$ and every $\mathcal{Y} \subseteq 2^Z$:*

1. $f(2^Z) = \emptyset$
2. $f(\mathcal{X}) \in \mathcal{X}$
3. $f(\mathcal{X} \cap \mathcal{Y}) = f(\mathcal{X}) \cup f(\mathcal{Y})$
   whenever $\bigcup \min(\mathcal{X} \cap \mathcal{Y}) = \bigcup \min \mathcal{X} \cup \bigcup \min \mathcal{Y}$

Then, $\widehat{\cdot} : 2^Z \to 2^Z$ *as defined by*

$$\widehat{X} \stackrel{\text{def}}{=} f(\{Y \mid X \subseteq Y \subseteq Z\})$$

is a pre-closure operator upon Z.

Intuition: Z is $\mathrm{Rng}\, A_i$
$\mathcal{X}$ (and $\mathcal{Y}$, too) is a collection of intervals over $\mathrm{Rng}\, A_i$
moreover, $\mathcal{X}$ is a collection containing all super-intervals of an interval belonging to the collection

Class of choice functions satisfying pre-closure

Theorem. *Given a set Z, let $f : 2^{2^Z} \to 2^Z$ be a function st for every upward closed $\mathcal{X} \subseteq 2^Z$ and every $\mathcal{Y} \subseteq 2^Z$:*

1. $f(2^Z) = \emptyset$
2. $f(\mathcal{X}) \in \mathcal{X}$
3. $f(\mathcal{X} \cap \mathcal{Y}) = f(\mathcal{X}) \cup f(\mathcal{Y})$

   whenever $\bigcup \min(\mathcal{X} \cap \mathcal{Y}) = \bigcup \min \mathcal{X} \cup \bigcup \min \mathcal{Y}$

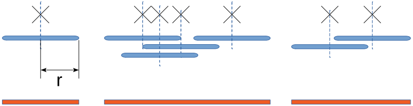
Then, $\widehat{\cdot} : 2^Z \to 2^Z$ *as defined by*

$$\widehat{X} \stackrel{\text{def}}{=} f(\{Y \mid X \subseteq Y \subseteq Z\})$$

is a pre-closure operator upon Z.

<u>Numerical attributes:</u> principle of single point (u) interpolation

$$A_i(x) \in [u - r : u + r] \to C(x) = c.$$



Class of choice functions satisfying capping

Theorem. *Given a set Z, let $f : 2^{2^Z} \to 2^Z$ be a function st for every $\mathcal{X} \subseteq 2^Z$ such that $\bigcap \mathcal{X} \in \mathcal{X}$ and for every $\mathcal{Y} \subseteq 2^Z$*

1. $f(\mathcal{X}) \in \mathcal{X}$
2. *if* $\mathcal{Y} \subseteq \mathcal{X}$ *and* $\exists W \in \mathcal{Y}$, $W \subseteq f(\mathcal{X})$ *then* $f(\mathcal{Y}) \subseteq f(\mathcal{X})$

Then, $\widehat{\cdot} : 2^Z \to 2^Z$ *as defined by*

$$\widehat{X} \stackrel{\text{def}}{=} f(\{Y \mid X \subseteq Y \subseteq Z\})$$

is a capping operator upon Z.

Intuition: Z is $\mathrm{Rng}\, A_i$
$\mathcal{X}$ (and $\mathcal{Y}$, too) is a collection of intervals over $\mathrm{Rng}\, A_i$
moreover, $\mathcal{X}$ is a collection whose intersection
is itself a member of the collection

Class of choice functions satisfying capping

Theorem. *Given a set Z, let $f : 2^{2^Z} \to 2^Z$ be a function st for every $\mathcal{X} \subseteq 2^Z$ such that $\bigcap \mathcal{X} \in \mathcal{X}$ and for every $\mathcal{Y} \subseteq 2^Z$*

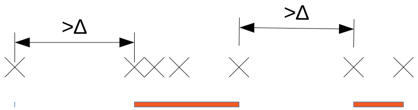
1. $f(\mathcal{X}) \in \mathcal{X}$
2. *if* $\mathcal{Y} \subseteq \mathcal{X}$ *and* $\exists W \in \mathcal{Y}$, $W \subseteq f(\mathcal{X})$ *then* $f(\mathcal{Y}) \subseteq f(\mathcal{X})$

Then, $\widehat{\cdot} : 2^Z \to 2^Z$ *as defined by*

$$\widehat{X} \stackrel{\text{def}}{=} f(\{Y \mid X \subseteq Y \subseteq Z\})$$

is a capping operator upon Z.

Numerical attributes: principle of pairwise point interpolation



And now . . .

Introduction

Formalizing rule learning
- General formalism
- Notations

Admissibility for generalization
- Formalizing admissibility
- Classes of choice functions

Application to an analysis of CN2

Conclusion

Illustrations of the behaviour of CN2

Generation of synthetic data:

- Data with 2 dimensions: a numerical attribute and a symbolic class attribute
- Data with two classes (green and blue)

Objective:

- Illustrate the behaviour of the rule learning algorithm in terms of characteristics of generalisation of examples
  - ▷ based on a pre-closure (*interpolation over single points*)
  - ▷ based on a capping (*interpolation over pairs of points*)

Using Algorithm CN2 [CN89]

Separating interesting intervals

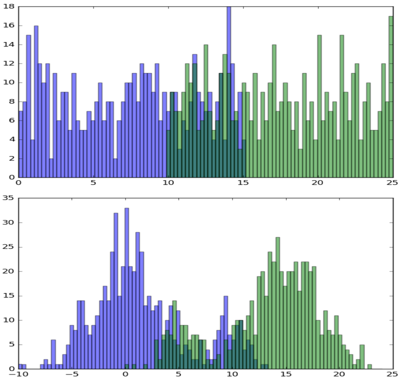


Figure: Distributions of the data for the classes `blue` and `green`.

Comparison of rules obtained out of uniform distributions vs. normal distributions

- distance between two successive values are small wrt the range of the attribute
- mixing normal distributions causes disparate average distances (pairwise distance between examples)
- the second dataset can be viewed as a super set of the first dataset (add of examples in between examples)

Separating interesting intervals

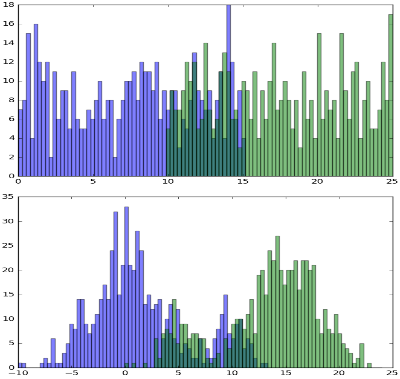


Figure: Distributions of the data for the classes `blue` and `green`.

Expected rules assuming capping or pre-closure for each class:

- topmost dataset:
  - $v \in [-\infty : 15] \Rightarrow A_0 = \texttt{blue}$
  - $v \in [10 : +\infty] \Rightarrow A_0 = \texttt{green}$
- bottom dataset:
  - $v \in [-\infty : 15] \Rightarrow A_0 = \texttt{blue}$
  - $v \in [0 : +\infty] \Rightarrow A_0 = \texttt{green}$

Separating interesting intervals

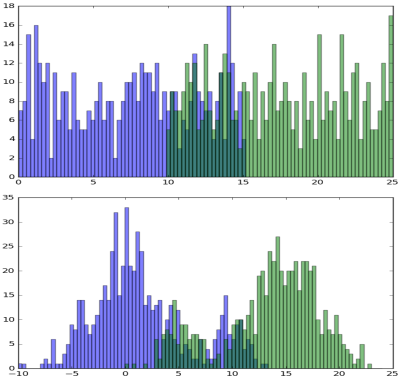


Figure: Distributions of the data for the classes `blue` and `green`.

Rules learned by CN2, topmost dataset:

- $v \in [-\infty : 10.03] \Rightarrow A_0 = \texttt{blue}$
- $v \in [12.73 : 14.83] \Rightarrow A_0 = \texttt{blue}$
- $v \in [10.65 : 12.81] \Rightarrow A_0 = \texttt{green}$
- $v \in [15.01 : +\infty] \Rightarrow A_0 = \texttt{green}$

Rules learned by CN2, bottom dataset:

- $v \in [-\infty : 0.96] \Rightarrow A_0 = \texttt{blue}$
- $v \in [0.97 : 2.57] \Rightarrow A_0 = \texttt{blue}$
- $v \in [3.09 : 10.04] \Rightarrow A_0 = \texttt{blue}$
- $v \in [3.50 : 7.18] \Rightarrow A_0 = \texttt{green}$
- $v \in [11.55 : 13.14] \Rightarrow A_0 = \texttt{green}$
- $v \in [13.15 : +\infty] \Rightarrow A_0 = \texttt{green}$

Does density influence the choice of boundaries?

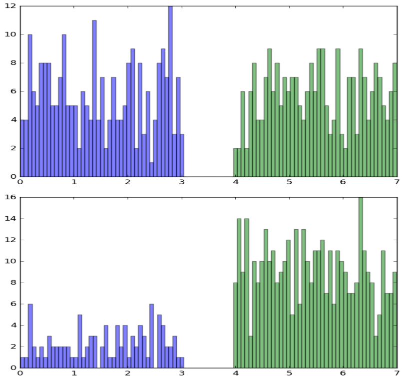


Figure: Distributions of the data for the classes `blue` and `green`.

Comparison of rules obtained out of well-separated uniform distributions, for two similar situations

- topmost dataset: same number of examples in both classes
- bottom dataset: the `blue` class is under-represented as compared to the `green` class

Does density influence the choice of boundaries?

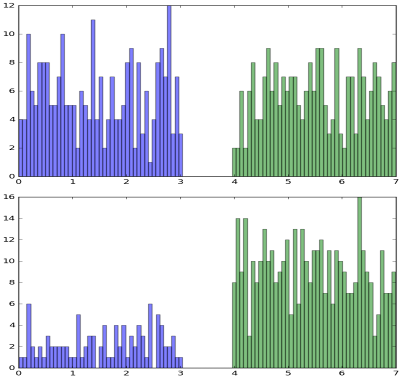


Figure: Distributions of the data for the classes `blue` and `green`.

Observed behaviour:

- no difference between the extracted rules for either dataset
- CN2 systematically chooses the boundary to be the middle of the limits in between the two classes

Does density influence the choice of boundaries?

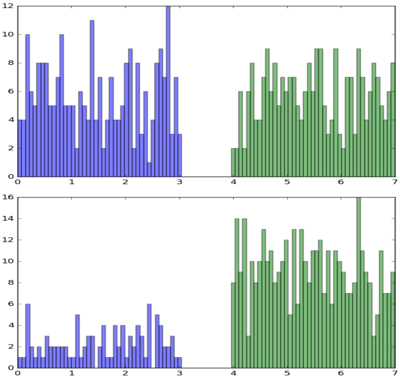
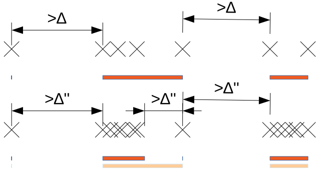


Figure: Distributions of the data for the classes `blue` and `green`.

Behaviour from *capping* :

- adding extra examples can alter boundaries



⇒ to be insensitive to density of examples corresponds to a *cumulation* operator

And now . . .

Introduction

Formalizing rule learning

- General formalism
- Notations

Admissibility for generalization

- Formalizing admissibility
- Classes of choice functions

Application to an analysis of CN2

Conclusion

Conclusion (1)

- The logical structure of rules makes them easy to read but ...
- The interpretability of rules learned from examples requires, in particular, to take care of the way examples are generalized
  - Example of numerical attributes, but also symbolic attributes with structures (e.g. orders)
- Qualifying the interpretable nature of rule learning outputs is challenging

What can our approach do for rule interpretability

- Our work contributes by giving a way to do such analysis
  - A proposal of a general framework for rule learning
  - A topological study of *admissible generalisations* of examples

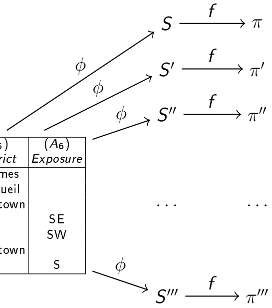
Conclusion (2)

Formalisation of rule learning

- ϕ: selects possible subsets of data
- $\widehat{\cdot}$: elicits the rule

$\rightarrow$ offer a framework for analysing rule learning algorithms

(C) *Price*	(A_1) *Area*	(A_2) *#Rooms*	(A_3) *Energy*	(A_4) *Town*	(A_5) *District*	(A_6) *Exposure*
low-priced	70	2	D	Toulouse	Minimes	
low-priced	75	4	D	Toulouse	Rangueil	
expensive	65	3		Toulouse	Downtown	
low-priced	32	2	D	Toulouse		SE
mid-priced	65	2	D	Rennes		SW
expensive	100	5	C	Rennes	Downtown	
low-priced	40	2	D	Betton		S



Conclusion (3)

Admissible generalisation of examples

- Admissible generalisations resulting of a choice among the supersets of the examples
- Proposed topological property of the choice: closure-like operators (pre-closure, capping)
- Definition of classes of choice functions
  - Proposal of concrete choice functions upon numerical attributes
  - Can be generalized to symbolic attributes, including attributes with structure (e.g. total order)

Perspectives

- Long term objective: study the characteristics of extracted rulesets
  - Comparing the *set* of rules extracted by machine learning
- Need a formalism to represent a set of rules
  - A formalism that enables to represent
    - → rules actually extracted by machine learning algorithms (e.g., Ripper, CN2, etc)
    - → rules selected using a selection criteria (interestingness measures, etc)
  - Formalize essential notions of rule learning
  - The formalism will be a way to reason about the machine learning algorithms

Bibliography I

Fernando Benites and Elena Sapozhnikova, *Hierarchical interestingness measures for association rules with generalization on both antecedent and consequent sides*, Pattern Recognition Letters **65** (2015), 197–203.

Peter Clark and Tim Niblett, *The CN2 induction algorithm*, Machine Learning **3** (1989), no. 4, 261–283.

Alberto Cano, Amelia Zafra, and Sebastián Ventura, *An interpretable classification rule mining algorithm*, Information Sciences **240** (2013), 1–20.

Tomás Kliegr, Stepán Bahník, and Johannes Fürnkranz, *A review of possible effects of cognitive biases on interpretation of rule-based machine learning models*, CoRR **abs/1804.02969** (2018).

Kazimierz Kuratowski, *Topology*, vol. 1, Elsevier, 2014.

Tom M Mitchell, *Generalization as search*, Artificial Intelligence **18** (1982), 203–226.