

Horn Minimization

Simon Vilmin

Supervisor: Sergei. A. Obiedkov

HSE - ISIMA

June 18, 2018

Introduction

- ▶ Correlation relations (*implications*).
e.g: movies and genres, *cyber-punk* \longrightarrow *sci-fi*
- ▶ Minimization without loss of knowledge.
e.g: *sci-fi* \longrightarrow *sci-fi* is useless
- ▶ *Study* of existing algorithms.

Outline

I - Horn theories

Closure and implications

Minimization task

II - Some Algorithms

Minimizing the input

Building the result

III - Experiments

Experiments

Closure operator and systems

1.1 - Closure and implications

Set Σ of attributes. A map $\varphi : 2^\Sigma \longrightarrow 2^\Sigma$ is a *closure operator* if, $\forall X, Y \subseteq \Sigma$:

- ▶ $X \subseteq \varphi(X)$ (*extensive*)
- ▶ $X \subseteq Y \longrightarrow \varphi(X) \subseteq \varphi(Y)$ (*monotone*)
- ▶ $\varphi(\varphi(X)) = \varphi(X)$ (*idempotent*)

Associated terminology:

- ▶ X is *closed* if $X = \varphi(X)$,
- ▶ Σ^φ set of closed sets: *closure system*,
- ▶ Σ^φ is closed under *intersection*, contains Σ .

Implications

1.1 - Closure and implications

$A, B \subseteq \Sigma$. An *implication* is:

- ▶ $A \longrightarrow B$, A *premise*, B *conclusion*,
- ▶ $M \subseteq \Sigma$ *model* of $A \longrightarrow B$:
 - ▷ $B \subseteq M \vee A \not\subseteq M$, ($\simeq B \vee \neg A$)
 - ▷ $M \models A \longrightarrow B$.

Set of implications \mathcal{L} : *implication system*.

- ▶ $\mathcal{L} \models A \longrightarrow B$: all models of \mathcal{L} are models of $A \longrightarrow B$,

Implications and closure

1.1 - Closure and implications

\mathcal{L} an implication system:

- ▶ models of \mathcal{L} form a *closure system*:

$$\Sigma^{\mathcal{L}} = \{M \subseteq \Sigma \mid M = \mathcal{L}(M)\}$$

- ▶ *closure operator* $\mathcal{L}(X)$: *smallest model* (inclusion wise) of \mathcal{L} containing X , $X \subseteq \Sigma$:

$$\mathcal{L}(X) = \bigcap \{M \in \Sigma^{\mathcal{L}} \mid X \subseteq M\}$$

- ▶ $\mathcal{L} \models A \longrightarrow B$ iff $B \subseteq \mathcal{L}(A)$.

Redundancy, equivalence, minimality

1.2 - Minimization task

\mathcal{L} and \mathcal{L}' implication systems:

- ▶ $A \rightarrow B \in \mathcal{L}$ *redundant* if $\mathcal{L} - \{A \rightarrow B\} \models A \rightarrow B$,
- ▶ $\mathcal{L}' \models \mathcal{L}$: all implications of \mathcal{L} follow from \mathcal{L}' ,
- ▶ $\mathcal{L}' \models \mathcal{L}$ and $\mathcal{L} \models \mathcal{L}'$: *equivalent* systems.
- ▶ \mathcal{L} *minimum* if no possible \mathcal{L}' such that:
 - ▷ \mathcal{L}' equivalent to \mathcal{L} ,
 - ▷ \mathcal{L}' has fewer implications than \mathcal{L} .

Recall property: $(\mathcal{L}^- := \mathcal{L} - \{A \rightarrow B\})$

- ▶ $\mathcal{L}^- \models A \rightarrow B$ iff $B \subseteq \mathcal{L}^-(A)$.

Canonical basis

1.2 - Minimization task

Particular sets:

- ▶ $P \subseteq \Sigma$ *pseudo-closed* in \mathcal{L} if:
 - ▷ $P \neq \mathcal{L}(P)$,
 - ▷ if $Q \subset P$ and Q pseudo-closed, then $\mathcal{L}(Q) \subseteq P$.
- ▶ $Q \subseteq \Sigma$ *quasi-closed* in \mathcal{L} if:
 - ▷ $\forall P \subseteq Q, \mathcal{L}(P) \subseteq Q$ or $\mathcal{L}(P) = \mathcal{L}(Q)$,
- ▶ pseudo-closed \longrightarrow quasi-closed.

Duquenne-Guigues (*canonical*) base:

$$\{P \longrightarrow \mathcal{L}(P) \mid P \text{ pseudo-closed} \}$$

Small Example

1.2 - Minimization task

Let \mathcal{L} be an implication system:

- ▶ $\Sigma = \{a, b, c, d, e, f\},$
- ▶ $\mathcal{L} = \{ab \longrightarrow cde, cd \longrightarrow f, c \longrightarrow a, d \longrightarrow b, abcd \longrightarrow ef\}.$

We have:

- ▶ $\mathcal{L}(b) = b$, b is *closed*, hence a *model* of \mathcal{L} ,
- ▶ $\mathcal{L}(ab) = abcdef$, ab is *not* a model, ($abcdef$ is)
- ▶ $abcd \longrightarrow ef$ is *redundant*,

Two main ideas:

- ▶ *minimizing* input system
 - ▷ algorithms from FCA and databases,
- ▶ *building* a minimum system against the input
 - ▷ query learning interpretation,

Some notations, given \mathcal{L} :

- ▶ $|\mathcal{B}|$ number of implications, $|\Sigma|$ number of attributes,
- ▶ $|\mathcal{L}| = O(|\mathcal{B}| |\Sigma|)$, size of \mathcal{L} .

First algorithm

2.1 - Minimizing the input

MINCOVER:

- ▶ from Day, Wild, (80's),
- ▶ two steps:
 1. *maximize* the conclusion
 $A \longrightarrow B$ becomes $A \longrightarrow \mathcal{L}(A)$
 2. *remove* redundant information
 if $\mathcal{L}(A) = \mathcal{L}^-(A)$, $A \longrightarrow \mathcal{L}(A)$ is *redundant*
- ▶ output: *canonical* base, complexity: $O(|\mathcal{B}| |\mathcal{L}|)$.

Recall: $\mathcal{L}^- = \mathcal{L} - \{A \longrightarrow \mathcal{L}(A)\}$

A variation

2.1 - Minimizing the input

DUQUENNE MIN:

- ▶ variation of MINCOVER by Duquenne (2007),
- ▶ three steps:
 1. *quasi-closure* and *redundancy* elimination
if $B \subseteq \mathcal{L}^-(A)$, $A \rightarrow B$ is useless
else $A \rightarrow B$ becomes $\mathcal{L}^-(A) \rightarrow (\mathcal{L}^-(A) \cup B)$
 2. sort implications in \subseteq -compatible order (premises)
 3. *iteratively build* Duquenne-Guigues base
if $\mathcal{L}^-(A)$ *pseudo-closed*, add $\mathcal{L}^-(A) \rightarrow \mathcal{L}(A)$ to the result
- ▶ output: *canonical* base, complexity: $O(|\mathcal{B}| |\mathcal{L}|)$.

Database approach

2.1 - Minimizing the input

MAIERMIN:

- ▶ functional dependency based algorithm, Maier (80's),
- ▶ steps:
 1. *redundancy* elimination
if $B \subseteq \mathcal{L}^-(A)$, $A \longrightarrow B$ is useless
 3. *equivalence classes* reduction
group implications by premises *closure*
reduce those classes
- ▶ output: minimum basis, complexity: $O(|\mathcal{B}| |\mathcal{L}|)$.

General principle

2.2 - Building the result

Query learning and Angluin algorithm (90's):

- ▶ aim: learn a theory by constructing an *hypothesis*
- ▶ *oracle* answering *queries* (questions)
- ▶ queries:
 - ▷ *equivalence*: is our hypothesis equivalent to the target?
 - ▷ *membership*: is a set a model of the target?
- ▶ improve by *counterexample*
 - ▷ *positive*: model of the target, not of the hypothesis
 - ▷ *negative*: model of the hypothesis, not of the target

AFP-Based Algorithm

2.2 - Building the result

AFP-BASED:

- ▶ derived from Angluin, *no proof yet*,
- ▶ principle:
 1. take implications one by one,
 2. refine the hypothesis:
 - use *premises* to generate possible counter-examples
 - add right-closed implications or refine old ones
- ▶ expected output: *canonical* base,
- ▶ *idea* of complexity: $O(|\mathcal{B}|^3 |\mathcal{L}|)$.

Using minimality constraint

2.2 - Building the result

BERCZIMIN:

- ▶ logic based, Berczi, 2017
- ▶ principle:
 1. *build* a basis \mathcal{L}_c against the input \mathcal{L}
 2. repeat minimality selection up to *equivalence*
 3. *minimality selection*:
 - select the next minimal negative counter-example A ,
 - add $A \longrightarrow \mathcal{L}(A)$ to \mathcal{L}_c
- ▶ output: *canonical* base, complexity: $O(|\mathcal{B}|^2 |\mathcal{L}|)$.

Context 1

3.1 - Experiments

Practical aspect:

- ▶ context of FCA, previous study of closure operators,
- ▶ use datasets from UCI repository (*scaling*):
 - ▷ solar flare: 49 attributes,
 - ▷ SPECT: 23 attributes,
- ▶ use of CLOSURE (\simeq *forward chaining*)
- ▶ C++, boost, python.

Context 2

3.1 - Experiments

1 dataset give rise to 5 systems:

- ▶ Duquenne-Guigues basis (*DG*),
- ▶ minimal generators (*mingen*, right-closed),
- ▶ proper implications (*proper*)
- ▶ Maier minimum on mingen (*min 1*, right-closed),
- ▶ Maier minimum on proper (*min 2*)

\mathcal{L}		$ \Sigma $	$ \mathcal{B} $
Flare	minimum	49	3382
	mingen		39787
	proper		10692
SPECT	minimum	23	2169
	mingen		44341
	proper		8358

Table: Summary of real datasets characteristics

Overhaul results

3.1 - Experiments

\mathcal{L}		MINCov	DUQ	MAIER	BERCZI	AFP
Flare	DG	0.097	0.117	0.211	27.922	96.178
	min 1	0.134	0.194	0.288	27.750	98.145
	min 2	0.200	0.190	0.308	30.063	111.944
	proper	1.684	0.933	0.917	88.375	402.453
	mingen	16.047	7.981	7.576	160.328	2514.610
SPECT	DG	0.045	0.066	0.108	10.328	22.454
	min 1	0.061	0.080	0.134	8.156	19.438
	min 2	0.078	0.070	0.150	8.250	26.980
	proper	0.930	0.394	0.451	51.063	114.564
	mingen	24.077	10.206	10.858	194.875	863.903

Table: Comparison of the algorithms on real datasets (execution in s)

Observations *on these data*:

- ▶ cost of AFP and BERCZIMIN,
- ▶ DUQUENNEMIN, MAIERMIN efficient on *non-minimum cases*,
- ▶ MINCOVER slightly better on right-closed minimum cases.

Generating random implication (given $|\Sigma|$, $|\mathcal{B}|$):

- ▶ discrete uniform distribution on *size*, *elements*,
- ▶ premise A , conclusion B , yield $A - B \longrightarrow B$.

Minimum tests 1

3.1 - Experiments



NATIONAL RESEARCH
UNIVERSITY

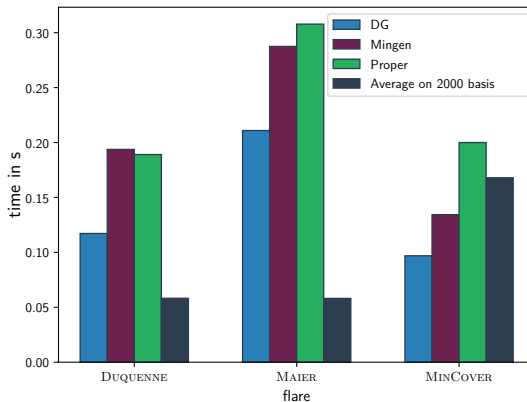


Figure: Flare - Random against and minimum basis: 49 attr, 3382 imp

Minimum tests 2

3.1 - Experiments



NATIONAL RESEARCH
UNIVERSITY

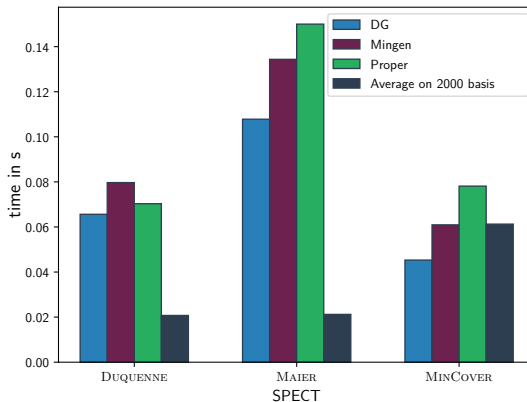


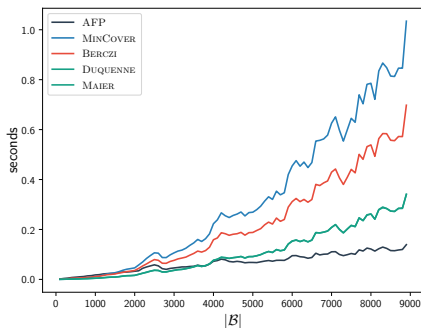
Figure: SPECT - Random against minimum basis: 23 attr, 2169 imp

Insight on random tests

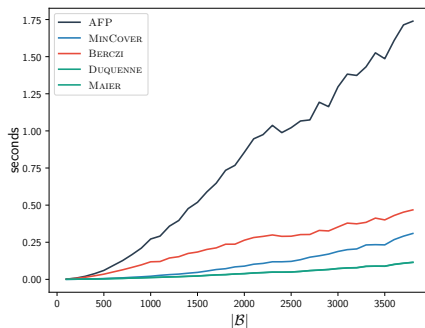
3.1 - Experiments



NATIONAL RESEARCH
UNIVERSITY



(a) Average time, $|\Sigma| = 100$



(b) Average time, $|\Sigma| = 500$

Figure: Random generated tests for fixed $|\Sigma|$ (over 500 ex)

Observations:

- ▶ times of MAIERMIN, DUQUENNEMIN on redundant cases,
- ▶ first glance at random: efficiency of AFP.

Explanations:

- ▶ *redundancy elimination* as first step,
- ▶ suggests a study of underlying *structure* (AFP).

Boundaries:

- ▶ valid in *our context*,
- ▶ random generation,
- ▶ suggests *extension* of tests.

Conclusion

Purpose:

- ▶ *study* of minimization algorithms.

Results:

- ▶ algorithms from various communities,
- ▶ in practice: *redundancy elimination*

Perspectives:

- ▶ *theoretical* study of systems structure, AFP proof and complexity,
- ▶ *experimental* aspect, extend tests.