Conclusive Update and Computability

Nina Gierasimczuk
(results from a joint work with Dick de Jongh)

Institute for Logic, Language and Computation
University of Amsterdam

LogiCIC Kick-Off Workshop
December 14^{th}, 2012

LEARNING

DEL intuitive, determinate manners of updating models

FLT no prescribed ways of learning but restricted by computability

Compare the two aspects: determinateness and computability.

Outline

Epistemic spaces and learning

Conclusive learnability

Computational assumption on agency

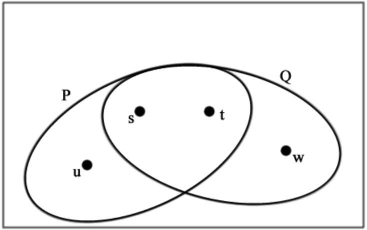
Preset learning

Fastest learning

An agent's uncertainty is represented by an epistemic space (S, Φ), where:

- $S = \{s_0, s_1, \ldots\}$ of epistemic possibilities, or possible worlds, and
- $\Phi \subseteq \mathcal{P}(S)$ a family of propositions.

Φ represent facts or observables being true or false in possible worlds.



Epistemic spaces and learning

- Learner L receives information about a possible world (the actual one).
- The information is an open-ended (infinite) sequence of propositions.
- Data stream $\varepsilon = (\varepsilon_1, \varepsilon_2 \dots)$ is a data stream for $s \in S$ just in case

$$\{\varepsilon_n : n \in \mathbb{N}\} = \{p \in \Phi : s \in p\}.$$

- Learner L is a function that on input of an epistemic space (S, Φ) and a finite sequence of observations $\sigma = (\sigma_0, \dots, \sigma_n)$ outputs a hypothesis, i.e.,

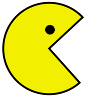
$$L((S, \Phi), \sigma) \subseteq S.$$

- Learner L receives information about a possible world (the actual one).
- The information is an open-ended (infinite) sequence of propositions.
- Data stream $\varepsilon = (\varepsilon_1, \varepsilon_2 \dots)$ is a data stream for $s \in S$ just in case

$$\{\varepsilon_n : n \in \mathbb{N}\} = \{p \in \Phi : s \in p\}.$$

- Learner L is a function that on input of an epistemic space (S, Φ) and a finite sequence of observations $\sigma = (\sigma_0, \dots, \sigma_n)$ outputs a hypothesis, i.e.,

$$L((S, \Phi), \sigma) \subseteq S.$$



Basic types of learnability

(S, Φ)
$s_1 : p_1, p_3, p_4$
$s_2 : p_2, p_4, p_5$
$s_3 : p_1, p_3, p_5$
$s_4 : p_4, p_6$

(S', Φ')
$t_1 : p_1, p_3, p_4$
$t_2 : p_2, p_4, p_5$
$t_3 : p_1, p_3, p_5$
$t_4 : p_1, p_3, p_4, p_6$

Basic types of learnability

(S, Φ)
$s_1 : 1, 3, 4$
$s_2 : 2, 4, 5$
$s_3 : 1, 3, 5$
$s_4 : 4, 6$

(S', Φ')
$t_1 : 1, 3, 4$
$t_2 : 2, 4, 5$
$t_3 : 1, 3, 5$
$t_4 : 1, 3, 4, 6$

Basic types of learnability

(S, Φ)
$s_1 : 1, 3, 4$
$s_2 : 2, 4, 5$
$s_3 : 1, 3, 5$
$s_4 : 4, 6$

(S', Φ')
$t_1 : 1, 3, 4$
$t_2 : 2, 4, 5$
$t_3 : 1, 3, 5$
$t_4 : 1, 3, 4, 6$

Basic types of learnability

(S, Φ)
s_1 : 1, 3, 4
s_2 : 2, 4, 5
s_3 : 1, 3, 5
s_4 : 4, 6

(S', Φ')
t_1 : 1, 3, 4
t_2 : 2, 4, 5
t_3 : 1, 3, 5
t_4 : 1, 3, 4, 6

(S, Φ)
$s_1 : 1, 3, 4$
$s_2 : 2, 4, 5$
$s_3 : 1, 3, 5$
$s_4 : 4, 6$

Conclusive Learnability

- Certainty in finite time.
- Only one answer,
- based on certainty.
- No chance to change later.

Conclusive learnability: once-definedness

The range of learning function L is extended by $\uparrow$ ("I do not know").

Definition

Learning function L is once defined on (S, Φ) iff for any stream ε for any world in S there is exactly one $n \in \mathbb{N}$ such that $L(\varepsilon \upharpoonright n) \in \mathbb{N}$ (i.e., is not an $\uparrow$-answer).

DEFINITION

Take an epistemic space (S, Φ).

- A world $s_m \in S$ is conclusively learnable in a computable way by a function L if L is computable, once-defined, and for every data stream ε for s, there exists a finite stage n such that $L((S, \Phi), \varepsilon_0, \ldots, \varepsilon_k) = \{m\}$.

Definition

Take an epistemic space (S, Φ).

- A world $s_m \in S$ is conclusively learnable in a computable way by a function L if L is computable, once-defined, and for every data stream ε for s, there exists a finite stage n such that $L((S, \Phi), \varepsilon_0, \ldots, \varepsilon_k) = \{m\}$.
- The epistemic space (S, Φ) is said to be conclusively learnable in an computable way by L if L is computable and all its worlds in S are conclusively learnable in an computable way by L.
- Finally, the epistemic space (S, Φ) is conclusively learnable in an computable way just in case there is a computable learning function that can conclusively learn it in an computable way.

Definition

Take (S, Φ). A set $D_i \subseteq \Phi$ is a definite finite tell-tale set (DFTT) for s_i in S if:

1. D_i is finite,
2. $s_i \in \bigcap D_i$, and
3. for any $s_j \in S$, if $s_j \in \bigcap D_i$ then $s_i = s_j$.

Theorem

(S, Φ) is conclusively learnable in an computable way just in case there is a computable function $f : S \to \mathcal{P}^{<\omega}(\Phi)$ s.t. $f(s)$ is a DFTT for s.

Definition

Take (S, Φ). A set $D_i \subseteq \Phi$ is a definite finite tell-tale set (DFTT) for s_i in S if:

1. D_i is finite,
2. $s_i \in \bigcap D_i$, and
3. for any $s_j \in S$, if $s_j \in \bigcap D_i$ then $s_i = s_j$.

Theorem

(S, Φ) is conclusively learnable in an computable way just in case there is a computable function $f : S \to \mathcal{P}^{<\omega}(\Phi)$ s.t. $f(s)$ is a DFTT for s.

a world is conclusively learnable

if it makes true a finite conjunction of propositions

that together is false everywhere else

DEFINITION

Take (S, Φ) and $x \in \Phi$. The eliminative power of x with respect to (S, Φ) is determined by a function $El_{(S,\Phi)} : \Phi \to \mathcal{P}(\mathbb{N})$, such that:

$$El_{(S,\Phi)}(x) = \{i \,|\, s_i \notin x \;\&\; s_i \text{ in } S\}.$$

Additionally, for $X \subseteq \Phi$ we write $El_{(S,\Phi)}(X)$ for $\bigcup_{x \in X} El_{(S,\Phi)}(x)$.

eliminative power of a proposition is the complement of its extension

Definition (Fin-Id Problem)

Instance: A finite epistemic space (S, Φ), a world s_i in S.
Question: Is s_i conclusively learnable within (S, Φ)?

Theorem

Fin-Id *Problem is in P.*

Minimality of DFTT's: two kinds

set	a minimal DFTT	minimal-size DFTTs
$\{5, 7, 8\}$	$\{7, 8\}$	$\{5, 8\}$ or $\{7, 8\}$
$\{6, 8, 9\}$	$\{8, 9\}$	$\{6\}$
$\{5, 7, 9\}$	$\{7, 9\}$	$\{5, 9\}$ or $\{7, 9\}$
$\{8, 10\}$	$\{10\}$	$\{10\}$

Minimality of DFTT's: complexity

finding a minimal DFTT is easy

finding a minimal DFTT is easy

Proposition

Let (S, Φ) be a conclusively learnable finite epistemic space. Finding a minimal DFTT of s_i in (S, Φ) can be done in polynomial time w.r.t. $card(\{x | s_i \in x\})$.

Minimality of DFTT's: complexity

finding a minimal-size DFTT is (most probably) harder

finding a minimal-size DFTT is (most probably) harder

Definition (Min-size DFTT Problem)

Instance: (S, Φ), $s_i \in S$, and $k \leq card(\{p|s_i \in p\})$.
Question: Is there a DFTT X_i of s_i of size $\leq k$?

Theorem

The Min-size DFTT *Problem is NP-complete.*

finding a minimal-size DFTT is (most probably) harder

Definition (Min-size DFTT Problem)

Instance: (S, Φ), $s_i \in S$, and $k \leq card(\{p | s_i \in p\})$.
Question: Is there a DFTT X_i of s_i of size $\leq k$?

Theorem

The Min-size DFTT *Problem is NP-complete.*

teaching efficiently might be hard

Computational assumption about epistemic spaces

Definition

An epistemic space (S, Φ) is uniformly decidable just in case there is a computable function $f : S \times \Phi \to \{0, 1\}$ such that:

$$f(s, p) = \begin{cases} 1 & \text{if } s \in p, \\ 0 & \text{if } s \notin p. \end{cases}$$

Uniform decidability and agency

- In epistemic logic uniform decidability is primitive.
- However the problem is non-trivial, e.g., in scientific scenarios.
- Epistemic space represents an uncertainty of a TM-representable mind.
- Simple and appealing condition vs properties of convergence to knowledge.

Learners taking a more prescribed course of action by basing their conjectures on symptoms (DFTTs).

Preset learning

Learners taking a more prescribed course of action
by basing their conjectures on symptoms (DFTTs).

Learners taking a more prescribed course of action
by basing their conjectures on symptoms (DFTTs).

Objection: infinite collections of DFTTs.
Solution: f_{dftt}, which for a finite X and s_i says if X is a DFTT of s_i.

If (S, Φ) is conclusively learnable
then there is f_{dftt} that for each world recognizes at least one DFTT.

Preset learning: some results

1. conclusive learnability = preset conclusive learnability
2. preset learners are exactly those that react solely to the content

Fastest learner:

conclusively learns a world s_i as soon as objective 'ambiguity' disappears;

settles on the right world as soon as any DFTT for it has been given.

Definition

(S, Φ) is *conclusively learnable in the fastest way* if and only if there is a learning function L such that, for each ε and for each $i \in \mathbb{N}$,

$$L(\varepsilon{\restriction}n) = i \quad \text{iff} \quad \exists D_i^j \in \mathbb{D}_i \; (D_i^j \subseteq \mathsf{set}(\varepsilon{\restriction}n)) \; \& \\ \neg\exists D_i^k \in \mathbb{D}_i \; (D_i^k \subseteq \mathsf{set}(\varepsilon{\restriction}n-1)).$$

Such L is a *fastest learning function*.

THEOREM

There is a uniformly decidable epistemic space that is conclusively learnable, but is not conclusively learnable in the fastest way.

fastest conclusive learnability is properly included in conclusive learnability

Conclusions

- Complexity of learning/teaching strategies in conclusive learning.
- Complexity of min-DFTT and min-size DFTT related concepts.
- The notion of preset learner in fconclusive learning.
- Not all conclusively learnable classes are learnable in the fastest way.
- We have established a new, more restrictive kind of learning.

even if computable convergence to certainty is possible
it may not be computably reachable just when objective ambiguity disappears

Thank you!

Gierasimczuk, N. and De Jongh, D. (2012). On the Complexity of Conclusive Update. The Computer Journal 2012.

Gierasimczuk, N. (2009). Learning by Erasing in Dynamic Epistemic Logic. LATA 2009, LNCS 5457, pp. 362-373.

Gierasimczuk, N. (2010). Knowing Ones Limits. Logical Analysis of Inductive Inference. PhD thesis, Universiteit van Amsterdam.

Degremont, C. and Gierasimczuk, N. (2011). Finite identification from the viewpoint of epistemic update. Information and Computation, 209(3):383-396.

Proof I

Definition (Smullyan 1958)

Let $A, B \subset \mathbb{N}$. A separating set is $C \subset \mathbb{N}$ such that $A \subset C$ and $B \cap C = \emptyset$. In particular, if A and B are disjoint then A itself is a separating set for the pair, as is B. If a pair of disjoint sets A and B has no computable separating set, then the two sets are *computablely inseparable*.

Let A and B be two disjoint r.e. computablely inseparable sets, such that:

- $x \in A$ iff $\exists y\, Rxy$ with R computable, and
- $x \in B$ iff $\exists y\, Sxy$ with S computable.

For each x there is at most one y, s.t. Rxy and at most one y, s.t. Sxy.
We define $(S_i)_{i \in \mathbb{N}}$:

$$S_i = \{2i, 2i+1\} \cup \{2j \mid Rji\} \cup \{2j+1 \mid Sji\}.$$

Proof II

The idea is that $S_i = \{2i, 2i+1\}$ except that, for some m, Rim or Sim may be true, and then $2i \in S_m$ or $2i+1 \in S_m$, respectively.

Note that:

- There can be at most one such m, and for that m only one of Rim or Sim can be true.
- Since A and B are computably inseparable there is no computable f that makes the choice for each i.
- Except for such intruders the languages are disjoint.

Proof III

The argument:

- $\{2i, 2i+1\}$ is a DFTT for S_i.
- But, $\{2i+1\}$ is a DFTT for S_i if $i \notin B$, and $\{2i\}$ is a DFTT for S_i if $i \notin A$.
- However, a computable function that would give the minimal DFTTs of S_i gives a computable separating set of A and B.
- And this is impossible, since A and B are computably inseparable.

So there cannot be a computable fastest learner!