LEARNING AND COMPUTING IN THE LIMIT

SEBASTIAAN A. TERWIJN

Abstract. We explore two analogies between computability theory and a basic model of learning, namely Osherson and Weinsteins model theoretic learning paradigm. First, we build up the theory of model theoretic learning in a way analogous to the way computability theory is built up. We then discuss Δ_2-definability of predicates on classes and prove a limit lemma for continuous functionals.

§1. Introduction. The notion of limit computability crops up in a natural way in the study of the arithmetical hierarchy and the notion of relative computability, and has been extensively studied over the last decades, see e.g. the monographs [9, 16]. The idea of learning as a limit process is also central to a large number of models of learning, in particular those introduced by Gold [3], whose paradigm became known under the phrase "learning in the limit". Osherson and Weinstein [12] introduced a model of learning in the limit of first order sentences over models of a first order theory. Below we describe Osherson and Weinsteins model, and we state two results linking Δ_2-definability to limit computability. Before we do so we discuss some related work.

The relation between computation and definability by logical formulas is one of the cornerstones of computability theory, cf. Odifreddi [9], Rogers [14]. Interestingly, prior to his fundamental contribution to the theory of induction [3], Gold already wrote about limit computability [2], apparently unaware of earlier work of Shoenfield [15]. The book by Kelly [6] and the paper Gasarch et al. [1] contain topological characterizations of classes of functions that have a classifier, which is a method for deciding whether a function is in the given class or not. Although technically different, some of these results are similar in form to results such as Theorem 1.2 and Proposition 2.6 below. The role of limit computability in mathematics in general is investigated in the project by Hayashi [4].

The author was supported by the Austrian Research Fund FWF (Lise Meitner grant M699-N05).

Logic Colloquium 2002
Edited by Z. Chatzidakis, P. Koepke, W. Pohlers

A Δ_2-formula is a formula that is equivalent (over some theory and in a language given by the context) to both a Σ_2-formula (i.e. a formula in existential-universal form) and a Π_2-formula (i.e. a formula in universal-existential form). The following lemma is a basic result from computability theory. Let ω be the set of natural numbers. A set $A \subseteq \omega$ (which we identify with its characteristic function) is *limit computable* if there is a computable function f such that for all n, $A(n) = \lim_{s\to\infty} f(n,s)$.

LEMMA 1.1. (Shoenfields Limit Lemma [15]) *A set $A \subseteq \omega$ is definable by an arithmetical Δ_2-formula if and only if A is limit computable.*

The Δ_2-definable arithmetical sets can alternatively be characterized as the class of sets computable relative to the halting set K (Post).

The second result of the type above we discuss is the result of Osherson, Stob, and Weinstein [11] proving that a sentence is learnable over the models of a theory T if and only if the sentence is equivalent to a Δ_2-sentence over T. (See Theorem 1.2 below.) We first describe Osherson and Weinsteins model theoretic learning paradigm [12]. For a survey of this and related models of learning see e.g. [13, 10]. An interesting link between this paradigm and the theory of belief revision was provided in Martin and Osherson [7, 8].

For a first order theory T, $\mathrm{mod}(T)$ denotes the set of *countable* models of T. Since only countable models are considered, we may assume they all have the set of natural numbers ω as their universe. Fix a language $\mathcal{L}$ of finite signature that includes identity, and a countable set $\{v_i : i \in \omega\}$ of variables. A convenient but inessential assumption we make is that $\mathcal{L}$ has no function symbols. The *basic formulas* of $\mathcal{L}$ are the atomic formulas and the negations thereof. An *assignment* is a function $d : \{v_i : i \in \omega\} \to \omega$ that is onto, so that every element in the model is assigned at least one variable. Given a model $M \in \mathrm{mod}(T)$ and an assignment d, the *environment* for M and d is the ω-sequence with e whose i-th element is $e_i = \{\beta \text{ basic} : \mathrm{var}(\beta) \subseteq \{v_0, \dots, v_{i-1}\} \wedge M \models \beta(d{\restriction}i)\}$. That is, the environment e lists all basic facts of that are true in M, in the order determined by the assignment d. The environment determined by M and d is denoted by $[M, d]$. If $\sigma \sqsubset d$ is a finite initial segment of d, the finite initial segment of $[M, d]$ induced by it is denoted by $[M, \sigma]$. Clearly, if e is an environment for both M and N, then M and N are isomorphic. The set of finite initial segments of environments, i.e. segments of the form $e{\restriction}n$, is denoted by SEQ.

A *learner* is defined to be any function from SEQ to the set of $\mathcal{L}$-sentences. So in this setting a learner is conceived of as something that conjectures an $\mathcal{L}$-sentence after having seen a finite basic part of a model of a given theory.

Given a sentence φ and a theory T, we say that a learner Φ *learns* φ on $\mathrm{mod}(T)$ if for every $\tau \in \mathrm{SEQ}$, $\Phi(\tau) \in \{\varphi, \neg\varphi\}$, and for every $M \in \mathrm{mod}(T)$ and every environment e for M

- $\theta = \lim_{n\to\infty} \Phi(e{\upharpoonright}n)$ exists,
- $M \models \theta$.

That is, Φ learns φ on $\mathrm{mod}(T)$ if for any model M of T, Φ can determine in the limit the truth value of φ in M, given any enumeration of the basic truths of M.[1] A sentence φ is *learnable* over $\mathrm{mod}(T)$ if some learner Φ learns φ over $\mathrm{mod}(T)$.

THEOREM 1.2. (Osherson, Stob, and Weinstein [11]) *For any theory T, a sentence φ is learnable over* $\mathrm{mod}(T)$ *if and only if it is equivalent over T to both a Σ_2-sentence and a Π_2-sentence.*

Theorem 1.2 has as a consequence that there exists a computable universal learner, that is able to learn any first-order sentence over a set of models $\mathrm{mod}(T)$, given T as an oracle. Universal learners also exist for the basic setting of Golds model, but they are in general not computable: Stephan and Terwijn [18] proved that a Turing degree contains a universal text learner if and only if the degree is greater than or equal to $\mathbf{0}''$.

In section 2 below we also use the notion of oracle (or relative) computability to introduce some new learning-theoretic notions that parallel basic notions in computability theory. In particular we define analogues for (relative) computability and computable enumerability in the model theoretic setting. We also point out where the analogy breaks down. In section 3 we consider a higher order analogy between the two theories. We prove a limit lemma for functionals and discuss definability on classes.

§2. A first analogy.

We start by defining some learning theoretic concepts that parallel basic notions in computability theory. In learning theory, a change in hypothesis of the learner is called a *mind change*. When counting mind changes, it is customary to allow an initial empty hypothesis "?", such that the change from "?" to the formulation of a first hypothesis does not count as a mind change.

DEFINITION 2.1. We have the following notions of learnability of sentences:

one-shot learnable $\equiv$ learnable with 0 mind changes, starting with '?' Example: any basic formula, any sentence in T.

Σ_1*-learnable* $\equiv$ learnable with at most 1 mind change, starting with 'φ is false'.

[1] This definition of learning the truth of a given sentence φ has obvious variations that can be defined by using sets different from $\{\varphi, \neg\varphi\}$, see e.g. [10].

Π_1-*learnable* $\equiv$ learnable with at most 1 mind change, starting with 'φ is true'.

one-shot learnable relative to a set of sentences S $\equiv$ one-shot learnable when validity of every $\varphi \in S$ in M is known.

The last notion serves as an analogue of the notion of relative computability. A special case is when S consists of all $\forall$-sentences, giving the analogous notion of learnability relative to the halting set K, which we can also think of as a $\forall$-oracle.[2] The analogy between the several notions is summarized in the following table:

Computability theory	**Model theoretic learning**
$A \subseteq \omega$	model M
$n \in A$	$M \models \varphi$
approximating function f	learner Φ
computable	one-shot learnable
limit computable	learnable in the limit
computably enumerable, Σ^0_1-definable	Σ_1-learnable See also Proposition 2.3
co-c.e., Π^0_1-definable	Π_1-learnable See also Proposition 2.3
A-computable	one-shot φ-learnable
Limit Lemma: K-computable = limit computable	Proposition 2.4: one-shot learnable relative $\forall$-oracle = learnable in the limit
Posts theorem: computable = $\Sigma^0_1 \cap \Pi^0_1$	Proposition 2.5
K-computable = $\Sigma^0_2 \cap \Pi^0_2$	Proposition 2.6
limit computable = $\Sigma^0_2 \cap \Pi^0_2$	Theorem 1.2

Below we will use the following notion, which is a specification of the notion of confirmability in [11] to Σ_1-sentences:

Definition 2.2. A sentence φ is Σ_1-*confirmable* in T if for every model M in mod$(T \cup \{\varphi\})$ there is a Σ_1-sentence ψ_M such that

[2] Frank Stephan pointed out that the oracle for a set of sentences still depends on the unknown model, and that in this sense oracles for sets of sentences are less absolute than oracles in the setting of relativized computation.

- $M \models \psi_M$,
- $T \cup \{\psi_M\} \models \varphi$.

PROPOSITION 2.3. *A sentence φ is Σ_1-learnable if and only if it is equivalent (over the background theory T) to a Σ_1-sentence. A sentence φ is Π_1-learnable if and only if it is equivalent (over T) to a Π_1-sentence.*

PROOF. We only prove the difficult direction of the first part. The second part of the proposition can be proved in a similar way. Let Φ be a learner that Σ_1-learns φ. The proof that φ is equivalent to a Σ_1-sentence follows the scheme of the proof of Theorem 1.2, with some simplifications. First we prove that φ is Σ_1-confirmable in T. So let $M \in \mathrm{mod}(T \cup \{\varphi\})$. Let σ be a locking sequence for Φ and M, i.e. a sequence σ such that for all extensions $\tau \sqsupseteq \sigma$, $\Phi([M,\tau]) = \Phi([M,\sigma]) = \varphi$. Such a sequence is easily seen to exist. Let χ be the conjunction of all formulas in $[M,\sigma]$, and let ψ_M be the Σ_1-sentence obtained by quantifying out the free variables in χ with an existential quantifier. Clearly $M \models \psi_M$, and if $N \models T \cup \{\psi_M\}$ then there is a sequence σ' such that $[N,\sigma'] = [M,\sigma]$, so that $\Phi([N,\sigma']) = \Phi([M,\sigma]) = \varphi$, and hence $N \models \varphi$ since Φ Σ_1-learns φ on $\mathrm{mod}(T)$.

Now to conclude the proof, we use a compactness argument to show that φ is Σ_1-learnable if and only if φ is equivalent over T to a Σ_1-sentence. (cf. [11, p669]). The 'if' part is immediate. For the 'only if' part, consider the set

$$\Sigma = \big\{\psi_M : M \in \mathrm{mod}(T \cup \{\varphi\})\big\}.$$

We claim that φ is equivalent to a finite disjunction Δ of sentences in Σ. By definition of ψ_M, every such finite disjunction entails φ. Suppose conversely that for every finite $\Delta \subseteq \Sigma$ the theory $T \cup \{\varphi\} \cup \{\neg\theta : \theta \in \Delta\}$ is consistent. Then by the compactness and Löwenheim-Skolem theorems there is a countable model N of $T \cup \{\varphi\} \cup \{\neg\theta : \theta \in \Sigma\}$, which contradicts $N \models \psi_N$. ⊣

PROPOSITION 2.4. *A sentence φ is one-shot learnable with an oracle for $\forall$-sentences if and only if it is learnable in the limit.*

PROOF. This follows from Theorem 1.2. The proof is similar to that of Proposition 2.3. ⊣

PROPOSITION 2.5. *For any sentence φ the following are equivalent:*

(i) *φ is one-shot learnable,*
(ii) *φ is both Σ_1-learnable and Π_1-learnable*
(iii) *φ is both equivalent (over the background theory T) to a Σ_1-sentence and a Π_1-sentence.*

Proof. (i)⇒(ii) is immediate. (ii)⇔(iii) follows from Proposition 2.3. For (ii)⇒(i), when φ is both Σ_1 and Π_1-learnable then it is one-shot learnable in the following way: Start with hypothesis '?' and wait until the Σ_1 and the Π_1-algorithm have the same output. ⊣

In order to "relativize" Proposition 2.5 to a ∀-oracle, it will be useful to have the following notion of relativized structure. Given the first order language $\mathcal{L}$ and an $\mathcal{L}$-structure M, we want to define a new language $\mathcal{L}^\forall$ and structure $M^\forall$ such that the ∀-formulas from the first setting become atomic in the second. This is easily achieved as follows. For every n-ary predicate $R(x_1,\dots,x_n)$ in $\mathcal{L}$ and every partition of the variables $x_1,\dots,x_n$ into $y_1,\dots,y_k$ and $z_1,\dots,z_l$ ($l>0$), we have a new l-ary predicate $R^{\forall y_1,\dots,y_k}(z_1,\dots,z_l)$ in $\mathcal{L}^\forall$, with the intended meaning

$$R^{\forall y_1,\dots,y_k}(z_1,\dots,z_l) \Longleftrightarrow \forall y_1,\dots,y_k R(x_1,\dots,x_n).$$

(R is assumed to be at least binary here. There is no loss in not considering unary predicates.) The structure $M^\forall$ is the same as M, except that there are now *more* atomic formulas since the language has changed. Since environments enumerate in the limit all basic facts, an environment for $M^\forall$ acts as an oracle for all ∀-formulas for M.

Proposition 2.6. *A sentence φ is one-shot learnable with an oracle for ∀-sentences if and only if it is both equivalent (over T) to a Σ_2-sentence and a Π_2-sentence.*

Proof. We obtain this by "relativizing" Proposition 2.5 to a ∀-oracle:

$$\begin{array}{ll}
\varphi \text{ is one-shot learnable with an } \forall\text{-oracle} & \Longleftrightarrow \\
\varphi \text{ is one-shot learnable over } M^\forall & \Longleftrightarrow \\
\varphi \text{ equivalent to both a } \Sigma_1 \text{ and a } \Pi_1\text{-sentence in } \mathcal{L}^\forall & \Longleftrightarrow \\
\varphi \text{ equivalent to both a } \Sigma_2 \text{ and a } \Pi_2\text{-sentence in } \mathcal{L} &
\end{array}$$

The second equivalence holds by Proposition 2.5. ⊣

Say that φ is Σ_2-*learnable* if it is Σ_1-learnable with a ∀-oracle, and that φ is Π_2-*learnable* if it is Π_1-learnable with a ∀-oracle. From Propositions 2.4, 2.5, and 2.6 we see that *φ is learnable if and only if it is both Σ_2-learnable and Π_2-learnable.*

Example 2.7. Let T be the theory of linear orders R. Then $\varphi = \exists x \forall y\, Rxy$ is not learnable. It is not even Π_2-learnable. However, φ is Σ_2-learnable.

Example 2.8. Let $\mathcal{L} = \langle \leq, R, S\rangle$ be a signature with three binary predicates, and let T be the theory saying that $\leq$ is a linear order. Let $\varphi = \exists x \forall z \exists y \geq z\, Rxy$ be the sentence saying that there is an infinite R-section, and let T contain the axiom $\neg\varphi \leftrightarrow \exists x \forall z \exists y \geq z\, Sxy$, so that every model of T either has an infinite R-section or an infinite S-section. Then φ is not learnable over mod(T), but since φ is Δ_3 over T, it is learnable with an $\forall$-oracle.

EXAMPLE 2.9. One can ask for which theories T the above analogy is closest. Suppose that the language $\mathcal{L}$ contains the language of arithmetic, and that T is a recursive axiomatization of basic computability theory. Then the questions about one-shot learnability, Σ_1-learnability, etc. become identical to the questions of computability theory. So in this sense model theoretic learning is more general than computability theory. One can prove e.g. the following. In analogy to the existence of Turing incomparable c.e. sets (Friedberg-Muchnik) there exist Σ_1-learnable formula's φ and ψ (i.e. sets of Σ_1-sentences parameterized by one formula) such that neither φ is one-shot learnable relative to ψ, nor the other way round.

Despite the analogy between the two theories, there are also differences. One is in the analogy between the Limit Lemma and Proposition 2.4. In the first case we have a set A given by a computable approximation: $A(x) = \lim_s f(x,s)$. Seen as a learning problem, the task is to "learn" (compute) in the limit the value $A(n)$ from computable "data" only. Because the "learner" f is uniformly computable, it occurs in the language $\mathcal{L}$ and hence it can be quantified over, which gives an easy proof that A is K-computable. This possibility does not exist in the learning model, which makes the proof of Proposition 2.4 more difficult.

In the next section we look at a closer, more faithful analogy than the one given in this section, by jumping one level higher at the computability theory side, namely from sets of natural numbers to subsets of 2^ω.

§3. A second analogy. In this section we work with the first-order language of arithmetic with one extra unary predicate A, to be interpreted by a subset of ω. We denote this language by $\mathcal{L}$. A formula $\varphi \in \mathcal{L}$ need not be arithmetical if A is interpreted by a nonarithmetical set. If the interpretation of A is fixed, $\mathcal{L}$ is the language of all formulas that are arithmetical in A. We also write $A \models \varphi$ if $\varphi(A)$ is true. (Note that it would be more correct to write $\langle \omega, A\rangle \models \varphi$, since the universe will always be interpreted by ω, and only the interpretation of A varies.)

In the following, a functional will be a mapping $\Phi : 2^\omega \times \omega \to \omega$. The notion of continuity of functionals is related to the notion of relative computability in a natural way: A mapping Φ from 2^ω to 2^ω or ω is continuous if and only if if there exists an oracle X such that Φ is computable relative to X. (The set X is obtained by a suitable coding of the modulus of continuity.)

DEFINITION 3.1. For a continuous functional Φ we write $\lim \Phi = \varphi(\mathcal{A})$ if for every $A \in \mathcal{A}$, $\lim_{s\to\infty} \Phi(A, s)$ exists and

$$\begin{aligned} \lim_{s\to\infty} \Phi(A, s) = 1 &\iff \varphi(A) \\ \lim_{s\to\infty} \Phi(A, s) = 0 &\iff \neg\varphi(A) \end{aligned}$$

We mostly write $\Phi^A(s)$ instead of $\Phi(A, s)$, to stress that we think of A as an oracle.

We now have the following analogy at the level of classes of sets.[3]

Computability theory	**Model theoretic learning**
a class $\mathcal{A} \subseteq 2^\omega$	mod(T), the countable models of a first-order theory T
$A \in \mathcal{A}$	$M \in \text{mod}(T)$
oracle for A	environment $e =$ oracle for basic truths in M
$\varphi \in \mathcal{L}$, defines the class $\varphi(\mathcal{A}) = \{A \in \mathcal{A} : A \models \varphi\}$	φ sentence in the language of M
continuous functional	learner
Φ such that $\lim \Phi = \varphi(\mathcal{A})$ "Φ learns φ on $\mathcal{A}$"	Φ learns $\varphi \equiv$ $\forall M \forall e$ for M (Φ learns φ from e)

We have the following limit lemma for functionals:

PROPOSITION 3.2. *For every $\mathcal{A} \subseteq 2^\omega$ and every formula $\varphi \in \mathcal{L}$ the following are equivalent:*

(i) *There exists a partial recursive functional Φ such that $\lim \Phi = \varphi(\mathcal{A})$,*
(ii) *There exists a total recursive functional Φ such that $\lim \Phi = \varphi(\mathcal{A})$,*
(iii) *There exists a Δ_2-formula $\psi \in \mathcal{L}$ such that $\{A \in \mathcal{A} : \varphi(A)\} = \{A \in \mathcal{A} : \psi(A)\}$.*

PROOF. (i)$\Rightarrow$(ii) We can easily turn a partial recursive functional Φ into a total one $\hat{\Phi}$ by a looking back technique:

$$\hat{\Phi}^A(s) = \begin{cases} \Phi^A(t) & \text{for the largest } t \text{ such that } \Phi^A(t) \text{ converges in } s \text{ steps,} \\ 0 & \text{if such } t \text{ does not exist.} \end{cases}$$

Then $\hat{\Phi}$ is total and $\lim_s \hat{\Phi}^A(s)$ equals $\lim_s \Phi^A(s)$ whenever this last limit exists. In particular the limits are equal for every $A \in \mathcal{A}$.

---

[3]It was pointed out to us that the left hand side of this table is essentially the paradigm of classification from [1, 17]. It is possible to explicitly translate all these paradigms into each other, using a sufficiently expressive language.

(ii)⇒(iii) Suppose that $\lim \Phi = \varphi(A)$. Then for every A in $\mathcal{A}$,

$$\begin{array}{rcl} \varphi(A) & \Longleftrightarrow & \lim_{s\to\infty} \Phi^A(s) = 1 \\ & \Longleftrightarrow & \exists t \forall s \geq t \ \ \Phi^A(s) = 1 \\ & \Longleftrightarrow & \forall t \exists s \geq t \ \ \Phi^A(s) = 1 \end{array}$$

The last equivalence holds because $\lim_s \Phi^A(s)$ exists for every $A \in \mathcal{A}$. Since Φ is total recursive we see that φ is equivalent on $\mathcal{A}$ to both a Σ_2 and a Π_2-formula.

(iii)⇒(i) Suppose that ψ is a formula such that for every $A \in \mathcal{A}$

$$\begin{array}{rcl} \psi(A) & \Longleftrightarrow & \exists u \forall v \ \ S^A(u,v) \\ \neg\psi(A) & \Longleftrightarrow & \exists x \forall y \ \ R^A(u,v) \end{array}$$

where S and R are computable predicates. Say that $\psi(A)$ is *falsified* up to n at stage s if $\forall u \leq n \, \exists v \leq s \ \neg S^A(u,v)$, and likewise for $\neg\psi(A)$ (with S replaced by R). Define the partial recursive functional Φ as follows: To define $\Phi^A(n)$, search for a stage s such that either $\psi(A)$ or $\neg\psi(A)$ are falsified up to n at stage s. If $\psi(A)$ is falsified up to n let $\Phi^A(n) = 0$, and $\Phi^A(n) = 1$ otherwise. It is easy to see that Φ is total on $\mathcal{A}$. ⊣

The proof of Proposition 3.2 relativizes to an arbitrary oracle X: there is a partial X-recursive functional Φ such that $\lim \Phi = \varphi(\mathcal{A})$ if and only if here exists a Δ_2^X-formula ψ (i.e. a Δ_2-formula in the language $\mathcal{L}$ with an extra predicate for X) such that $\{A \in \mathcal{A} : \varphi(A)\} = \{A \in \mathcal{A} : \psi(A)\}$. Osherson et al. [11] showed that for special classes $\mathcal{A}$, namely for $\mathcal{A}$ the set of models of a first order theory, it is possible to obtain part (iii) from Proposition 3.2 even if the learner Φ from part (i) is not recursive. We formulate a recursion theoretic version of this result in the next theorem, using the Stone topology (see Keisler [5, p59]). The basic closed sets of this topology on 2^ω are the sets

$$\big\{A \in 2^\omega : A \models \varphi\big\}$$

where φ is an $\mathcal{L}$-sentence.[4]

THEOREM 3.3. *Let $\mathcal{A} \subseteq 2^\omega$ be compact in the Stone topology on 2^ω, and let $\varphi \in \mathcal{L}$. Then the following are equivalent:*

(i) *There exists a partial continuous functional Φ such that* $\lim \Phi = \varphi(\mathcal{A})$,
(ii) *There exists a Δ_2-formula $\psi \in \mathcal{L}$ such that $\{A \in \mathcal{A} : \varphi(A)\} = \{A \in \mathcal{A} : \psi(A)\}$.*

PROOF. The implication (ii)⇒(i) holds for every $\mathcal{A} \subseteq 2^\omega$ by relativizing the proof of (iii)⇒(i) in Proposition 3.2. The proof of (i)⇒(ii) is along the

[4] This is called the Stone topology because it is obtained by considering the Stone space of the Lindenbaum algebra of the language $\mathcal{L}$.

same lines as the proof of Theorem 1.2. The compactness of $\mathcal{A}$ is exactly what is needed (twice). ⊣

We end by showing that the condition of compactness in Theorem 3.3 in general cannot be deleted. We first prove a simple lemma.

LEMMA 3.4. *Let φ be the $\mathcal{L}$-sentence $\forall x \exists y(y \geq x \wedge y \in A)$ expressing that A is infinite. Then φ is not equivalent to a Σ_2-sentence, and for every finite initial segment σ we have*

$$(\forall \psi \in \Sigma_2)(\exists A \sqsupset \sigma)\,[\,\varphi(A) \not\leftrightarrow \psi(A)\,]. \tag{1}$$

PROOF. This is a straightforward diagonalization argument. It can also be shown that φ is complete for the Π_2 sentences of $\mathcal{L}$ in a natural sense,[5] but the property (1) will be sufficient here. Fix a Σ_2-sentence ψ, say $\psi = \exists x \forall y(R^A(x,y))$ for some recursive predicate R that uses A as an oracle. If there is a finite set $A \sqsupset \sigma$ such that $\psi(A)$ holds then we are done. Otherwise, we have

$$(\forall A \sqsupset \sigma)\,\big[\,A \text{ finite } \rightarrow \forall x \exists y(\neg R^A(x,y))\,\big]. \tag{2}$$

We use the property (2) to build an infinite set $C \sqsupset \sigma$ with $\neg\psi(C)$. We build $C = \bigcup_s \sigma_s$ using a finite extension construction such that

$$(\forall s)(\exists y \leq |\sigma_{s+1}|)\,[\,\neg R^{\sigma_{s+1}}(s,y)\,]. \tag{3}$$

Stage $s = 0$: Set $\sigma_0 = \sigma$. Stage $s+1$: Denote by $\sigma_s\widehat{\ }\emptyset$ the infinite sequence obtained by extending σ_s with infinitely many 0's. Since this is a finite set, by (2) we have $\exists y(\neg R^{\sigma_s\widehat{\ }\emptyset}(s,y))$. Let y be the smallest such y, and let u be the use of the computation of $\neg R^{\sigma_s\widehat{\ }\emptyset}(s,y)$, i.e. u is the largest number used in that computation. Define $\sigma_{s+1} = \sigma_s\widehat{\ }0^u\widehat{\ }1$, that is the sequence obtained by concatenating u 0's to σ_s followed by a 1. Then we also have $\neg R^{\sigma_{s+1}\widehat{\ }\emptyset}(s,y)$ since the 1 that is added in σ_{s+1} is above the use u.

Now $C = \bigcup_s \sigma_s$ is infinite because every σ_s adds a new 1, and by (3) it satisfies $\neg\psi(C)$. ⊣

PROPOSITION 3.5. *There exist $\mathcal{A} \subseteq 2^\omega$, a continuous functional Φ, and sentence φ such that (i) from Theorem 3.3 holds but (ii) does not.*

PROOF. Note that by Proposition 3.2 Φ will have to be nonrecursive. Let φ be the sentence $\forall x \exists y(y \geq x \wedge y \in A)$. By the property (1) we can choose for every $\psi \in \Sigma_2$ an A_ψ such that $\varphi(A_\psi) \not\leftrightarrow \psi(A_\psi)$. Now if we pick the A_ψ in some canonical way, and we define

$$\mathcal{A} = \big\{A_\psi : \psi \in \Sigma^0_2\big\},$$

[5] Namely, for every Π_2-sentence ψ in $\mathcal{L}$ there is a recursive functional Φ (Φ can even be chosen to be uniform in a code of ψ) such that for every set A, $\psi(A) \Leftrightarrow \varphi(\Phi(A))$.

then φ will be learnable over $\mathcal{A}$ (i.e. $\lim \Phi = \varphi(\mathcal{A})$ for some continuous functional Φ), *provided* that we can read off ψ from A_ψ. To obtain this, we make $\mathcal{A}$ *self-describing*: Under a suitable coding of all formulas we want that if n is the least number in A_ψ, then n is a code of ψ.

So, given ψ with code $n = \ulcorner\psi\urcorner$, we define A_ψ as follows: See if there is a finite set $A \sqsupset 0^{(n-1)}{}^\frown 1$ such that $\psi(A)$. If so, let A_ψ be such an A. If not, pick for A_ψ some infinite $A \sqsupset 0^{(n-1)}{}^\frown 1$ with $\neg\psi(A)$. Such A exists by (1).

Since the collection $\mathcal{A}$ of all A_ψ is self-describing, and given ψ we can decide (noneffectively) which decision has been made in the definition of A_ψ, and hence whether A_ψ is infinite or not, there is a learner Φ with $\lim \Phi = \varphi(\mathcal{A})$. Furthermore, since $\varphi(A_\psi)$ if and only if $\neg\psi(A_\psi)$, item (ii) from Theorem 3.3 does not hold. $\dashv$

Acknowledgement We thank Frank Stephan and the referee for many comments and pointers to the literature.

REFERENCES

[1] W. Gasarch, M. Pleszkoch, F. Stephan, and M. Velauthapillai, *Classification using information*, ***Annals of Mathematics and Artificial Intelligence***, vol. 23 (1998), pp. 147–168.

[2] E. M. Gold, *Limiting recursion*, ***The Journal of Symbolic Logic***, vol. 30 (1965), pp. 28–48.

[3] ———, *Language identification in the limit*, ***Information and Control***, vol. 10 (1967), pp. 447–474.

[4] S. Hayashi, *Limit computable mathematics*, www.shayashi.jp/PALCM/index-eng.html.

[5] H. J. Keisler, *Fundamentals of model theory*, ***Handbook of mathematical logic*** (J. Barwise, editor), North-Holland, 1978, pp. 47–103.

[6] K. Kelly, ***The logic of reliable inquiry***, Oxford University Press, 1995.

[7] E. Martin and D. Osherson, *Scientific discovery based on belief revision*, ***The Journal of Symbolic Logic***, vol. 62(4) (1997), pp. 1352–1370.

[8] ———, ***Elements of scientific inquiry***, MIT Press, 1998.

[9] P. G. Odifreddi, ***Classical recursion theory***, Studies in logic and the foundations of mathematics Vol. 125 and Vol. 143, North-Holland, 1989, 1999.

[10] D. Osherson, D. de Jongh, E. Martin, and S. Weinstein, *Formal learning theory*, ***Handbook of logic and language*** (J. van Benthem and A. ter Meulen, editors), Elsevier, 1997, pp. 737–775.

[11] D. Osherson, M. Stob, and S. Weinstein, *A universal inductive inference machine*, ***The Journal of Symbolic Logic***, vol. 56(2) (1991), pp. 661–672.

[12] D. Osherson and S. Weinstein, *Identification in the limit of first order structures*, ***Journal of Philosophical Logic***, vol. 15 (1986), pp. 55–81.

[13] ———, *Paradigms of truth detection*, ***Journal of Philosophical Logic***, vol. 18 (1989), pp. 1–42.

[14] H. Rogers, ***Theory of recursive functions and effective computability***, McGraw-Hill, 1967.

[15] J. R. Shoenfield, *On degrees of unsolvability*, ***Annals of Mathematics***, vol. 69 (1959), pp. 644–653.

[16] R. I. Soare, ***Recursively enumerable sets and degrees***, Springer, 1987.

[17] F. C. Stephan, *One-sided versus two-sided classification*, ***Archive for Mathematical Logic***, vol. 40 (2001), pp. 489–513.

[18] F. C. Stephan and S. A. Terwijn, *The complexity of universal text-learners*, ***Information and Computation***, vol. 154(2) (1999), pp. 149–166.

TECHNICAL UNIVERSITY OF VIENNA
INSTITUTE OF DISCRETE MATHEMATICS AND GEOMETRY
WIEDNER HAUPTSTRASSE 8–10 / E104
A–1040 VIENNA, AUSTRIA
E-mail: terwijn@logic.at
URL: www.logic.at/people/terwijn/