Available online at www.sciencedirect.com
ScienceDirect

Journal of Computer and System Sciences 73 (2007) 109–122

JOURNAL OF COMPUTER AND SYSTEM SCIENCES

www.elsevier.com/locate/jcss

On the learnability of vector spaces

Valentina S. Harizanov [a,1], Frank Stephan [b,*,2]

[a] *Department of Mathematics, George Washington University, Washington, DC 20052, USA*
[b] *Department of Mathematics and School of Computing, National University of Singapore, 2 Science Drive 2, Singapore 117543, Singapore*

Received 28 November 2005; received in revised form 6 July 2006

Available online 5 October 2006

Abstract

The central topic of the paper is the learnability of the recursively enumerable subspaces of V_∞/V, where V_∞ is the standard recursive vector space over the rationals with (countably) infinite dimension and V is a given recursively enumerable subspace of V_∞. It is shown that certain types of vector spaces can be characterized in terms of learnability properties: V_∞/V is behaviourally correct learnable from text iff V is finite-dimensional, V_∞/V is behaviourally correct learnable from switching the type of information iff V is finite-dimensional, 0-thin or 1-thin. On the other hand, learnability from an informant does not correspond to similar algebraic properties of a given space. There are 0-thin spaces W_1 and W_2 such that W_1 is not explanatorily learnable from an informant, and the infinite product $(W_1)^\infty$ is not behaviourally correct learnable from an informant, while both W_2 and the infinite product $(W_2)^\infty$ are explanatorily learnable from an informant.

Keywords: Computational learning theory; Inductive inference; Learning algebraic structures; Recursively enumerable vector spaces; 0-thin and 1-thin spaces

Contents

* Corresponding author.
E-mail addresses: harizanv@gwu.edu (V.S. Harizanov), fstephan@comp.nus.edu.sg (F. Stephan).
[1] Valentina Harizanov was partially supported by the NSF grant DMS-0502499 and by the Columbian Research Fellowship of GWU—she thanks William Frawley for strong research support.
[2] Frank Stephan was supported in part by NUS grant No. R252-000-212-112.

0022-0000/$ – see front matter
doi:10.1016/j.jcss.2006.09.001

1. Introduction

In Gold's framework of inductive inference a learner, presented with data about a recursively enumerable language (equivalently, a recursively enumerable set), is allowed to make finitely many incorrect hypotheses before converging to a correct one. A central theme in inductive inference is the relation between learning from all data, that is, learning from an informant, and learning from positive data only, that is, learning from text. Learning from text is much more restrictive than learning from an informant, as shown by Gold [7]. Gold proved that the collection consisting of an infinite set together with all of its finite subsets can be learned from an informant, but not from text. On the other hand, Sharma [23] established that combining learning from an informant with the restrictive convergence requirement that the first hypothesis is already the correct one implies learnability from text.

Hence it is natural to investigate what reasonable notions might exist between these two extremes. Using a non-recursive oracle as an additional tool cannot completely close the gap. Even the most powerful oracles do not allow learning all computably enumerable sets from text [9], while the oracle for the halting set K does for learning from an informant. Restrictions on texts reduce their irregularity and allow them to provide further information implicitly [21,26]. Texts can be strengthened by permitting additional queries to retrieve information not contained in standard texts [13]. Ascending texts allow the learner to reconstruct complete negative information in the case of infinite sets, but might fail to do so in the case of finite sets. Thus, the class consisting of an infinite set together with all of its finite subsets remains unlearnable even for learning from ascending text.

Motoki [16] and later Baliga, Case and Jain [3] added to the positive information on the language L to be learned some, but not all, negative information. They considered two ways of supplying negative data: (a) there is a finite set of negative information $S \subseteq \overline{L}$ such that the learner always succeeds in learning the set L from input S plus a text for L; (b) there is a finite set $S \subseteq \overline{L}$ such that the learner always succeeds in learning the set L from a text for L plus a text for a set H, disjoint from L and containing S, that is, satisfying $S \subseteq H \subseteq \overline{L}$. Since in case (a) one can learn all recursively enumerable sets by a single learner, the notion (b) is more interesting.

Jain and Stephan [10] treated positive and negative data symmetrically and defined notions less powerful than the ones in [3] that we discussed. The most convenient way to introduce these notions is to use the idea of a minimum adequate teacher as, for example, described by Angluin [2]. Among the learning concepts considered by Jain and Stephan [10], the following one turned out to be most important. The learner requests positive or negative data from the teacher who, whenever almost all requests are of the same type, has to eventually reveal all information of that type.

In the present work, this type of information is applied to a natural model theoretic setting: learning recursively enumerable subspaces of a given recursive vector space. Such a subspace is given as the quotient space of the standard recursive infinite-dimensional space over the rationals with the dependence algorithm, V_∞, and its recursively enumerable subspace V. Alternatively, this can be viewed as learning the following class of vector spaces:

$$\mathcal{L}(V) = \{W\colon\ V \subseteq W \subseteq V_\infty \wedge (W \text{ is recursively enumerable})\}.$$

This class forms a filter in the lattice $\mathcal{L}(V_\infty)$ of all recursively enumerable subspaces of V_∞. Stephan and Ventsov [25] have previously shown that, in the case of learning all ideals of a recursive ring, learnability from text has strong connections to the algebraic properties of the ring. Here, it also turns out that the two notions of learnability of the class $\mathcal{L}(V)$, from positive data or from switching type of information, have corresponding algebraic characterizations. On the other hand, we show that supplying complete information, that is, learning from an informant, no longer has such nice algebraic characterizations. One of the reasons is that, while switching type of information provides more learning power than giving positive information only, it is still much weaker than providing information from an informant.

2. Preliminaries

2.1. Notions from recursion theory

Let $\mathbb{N}$ be the set of natural numbers. Sets are often identified with their characteristic functions, so we may write $X(n) = 1$ for $n \in X$ and $X(n) = 0$ for $n \in \overline{X}$. A subset of $\mathbb{N}$ is *recursive* if its characteristic function is recursive. A set of natural numbers is *recursively enumerable* if it is the domain of a partial recursive function or, equivalently, the range of a partial (even total) recursive function. Let $\varphi_0, \varphi_1, \varphi_2, \ldots$ be a fixed effective enumeration of all unary partial recursive functions on $\mathbb{N}$, where φ_e is computed by the Turing program with Gödel index (code) e. We write $\varphi_{e,s}(x) = y$ if $x, y, e < s$ and y is the output of $\varphi_e(x)$ in up to s steps of the Turing program with code e running on input x. For $e, s \in \mathbb{N}$, let W_e be the domain of φ_e and $W_{e,s}$ be the domain of the finite function $\varphi_{e,s}$. Then $W_0, W_1, W_2, \ldots$ is a fixed effective enumeration of all recursively enumerable subsets of $\mathbb{N}$. A Turing degree is recursively enumerable if it contains a recursively enumerable set. Let $\langle \cdot,\cdot \rangle$ be a fixed recursive 1-1 and onto pairing function, i.e., $\langle e, x \rangle$ is the natural number that codes the pair (e, x). We define the set $K = \{\langle e, x \rangle\colon e \in \mathbb{N} \wedge x \in W_e\}$. The set K is a version of the universal halting problem. It is a recursively enumerable and nonrecursive set; its Turing degree is $\mathbf{0}'$. A Turing degree $\mathbf{a} \leqslant \mathbf{0}'$ is *high* if its jump has the highest possible value, that is, $\mathbf{a}' = \mathbf{0}''$. A set $M \subseteq \mathbb{N}$ is called *maximal* if M is recursively enumerable and its complement $\overline{M}$ is cohesive [20, Definition III.4.13]. A set $\overline{M}$ is *cohesive* if it is infinite and there is no recursively enumerable set W such that $W \cap \overline{M}$ and $(\mathbb{N} - W) \cap \overline{M}$ are both infinite. Every maximal set has a high Turing degree. Conversely, every recursively enumerable high Turing degree contains a maximal set. This characterization was established by Martin. For more information, see [12,22,24].

We consider only countable algebraic structures and recursive first-order languages. A countable structure for a recursive language is *recursive* if its domain is recursive and its operations and relations are uniformly recursive. An example of a recursive structure is the field $(Q, +, \cdot)$ of rational numbers.

2.2. Notions from algebra

Let $(F, +, \cdot)$ be a fixed recursive field. Then $(V_\infty, +, \cdot)$ is a recursive $\aleph_0$-dimensional vector space over $(F, +, \cdot)$, consisting of all finitely nonzero infinite sequences of elements of F, under pointwise operations. Metakides and Nerode [14] showed that the study of recursive and other algorithmic vector spaces can be reduced to the study of V_∞ and its subspaces. A standard (default) basis for V_∞ is $\{\epsilon_0, \epsilon_1, \ldots\}$, where ϵ_i is the infinite sequence with the ith term 1 and all other terms 0. For a recursively enumerable vector space, having a recursively enumerable basis is equivalent to the existence of a dependence algorithm. A *dependence algorithm* decides whether any finite set of vectors is linearly dependent. If B is a basis and v a vector, then the *support* of v with respect to B is defined to be the least subset of B whose linear closure (span) contains v.

Every vector in V_∞ can be identified with its Gödel code, so the set V_∞ can be identified with $\mathbb{N}$. A subspace V of V_∞ is *recursive* (*recursively enumerable*, respectively) if its domain is a recursive (recursively enumerable, respectively) subset of the set V_∞. In that case, we also say that the quotient space V_∞/V is recursive (recursively enumerable, respectively). Let $W_0, W_1, \ldots$ be an effective enumeration of all recursively enumerable subsets of V_∞. For every e, let V_e be the vector space generated by W_e, that is, the linear closure of W_e. Then $V_0, V_1, \ldots$ is an effective enumeration of all recursively enumerable subspaces of V_∞. The set of all recursively enumerable vector subspaces of V_∞ is denoted by $\mathcal{L}(V_\infty)$. The class $\mathcal{L}(V_\infty)$, together with the operations of intersection and sum of vector spaces, forms a modular nondistributive lattice. Let V be a fixed recursively enumerable subspace of V_∞. By $\mathcal{L}(V)$ we denote the lattice of all recursively enumerable spaces W such that $V \subseteq W \subseteq V_\infty$. These spaces can be viewed as representatives of the corresponding classes of recursively enumerable subspaces of V_∞/V. For more information, see [19]. In the next two sections we assume that the field $(F, +, \cdot)$ is infinite. Without loss of generality, we can assume that it is $(Q, +, \cdot)$.

Vector spaces are special cases of the so-called closure systems or matroids. A *matroid* consists of a set X equipped with a closure operator Φ, which satisfies certain axioms. In the case of vector spaces, the closure operator is the linear closure of sets. The full axiomatization of matroids will be given in Section 5.

2.3. Notions from learning theory

The main setting in *inductive inference* is that the learner receives more and more data on an object to be learned and outputs a sequence of hypotheses that converges to a description of the object. In general, learning can be viewed as a dialogue between the teacher and the learner, where the learner must succeed in learning, provided the teacher satisfies a certain protocol. The formalization has two aspects: convergence criteria and teacher constraints.

Definition 2.1. [7,10] A class $\mathcal{L}$ of subsets of V_∞ is learnable according to the criteria specified below iff there is a (total) recursive learner M, which alternately requests new data and outputs hypotheses, and which learns every $W \in \mathcal{L}$, whenever the corresponding teacher meets the following requirements.

- All models of learning have in common that the learner makes infinitely many requests that are always either of type 0 (requesting negative data) or of type 1 (requesting positive data). The teacher answers to each request of type y, either by giving a pause symbol or a datum x such that $W(x) = y$.
- Learning from *Text*: The learner requests only information of type 1 and the teacher eventually provides all x with $W(x) = 1$.
- Learning from *Negative Text*: The learner requests only information of type 0 and the teacher provides eventually all x with $W(x) = 0$.
- Learning from *Switching Type of Information*: The learner is allowed to switch the type of information requested. However, whenever the learner almost always requests information of the same type y ($y \in \{0, 1\}$), the teacher eventually gives the learner all x with $W(x) = y$.
- Learning from an *Informant*: The learner alternately requests information of type 0 and type 1, and the teacher eventually provides every $x \in V_\infty$ after some request of type $W(x)$.

The hypotheses output by the learner are indices in the effective enumeration of recursively enumerable subspaces of V_∞. Following the above protocol of the dialogue with the teacher, the learner M has to converge in one of the two models below, where $e_0, e_1, \ldots$ is the infinite sequence of hypotheses output by M during the learning dialogue.

- *Explanatory* Learning: For almost every n, the hypothesis e_n is the same e, which is an index for W (i.e., $W = W_e$). This convergence is also called *syntactic*.
- *Behaviourally Correct* Learning: For almost every n, the hypothesis e_n is an index for W, although these indices e_n are permitted to be different. This convergence is also called *semantic*.

The symbols *Txt*, *Sw* and *Inf* stand for the protocols of learning from text, from switching type of information and from an informant, respectively. The symbols *Ex* and *BC* stand for explanatory learning and for behaviourally correct learning, respectively. The learner is assumed to be algorithmic unless it is explicitly stated otherwise. For example, $\mathcal{L}$ is *SwBC*-learnable iff there is a recursive learner M which for every $W \in \mathcal{L}$ and every teacher for W, respecting the *Sw*-protocol for the dialogue with the learner, outputs almost always a hypothesis for W.

Jain and Stephan [10] introduced three main notions for switching protocols. Among these three notions, the one denoted by *NewSw* in [10] turned out to be most appropriate to model switching types of information. Since the other notions are not considered here, we just write *SwEx* and *SwBC* for *NewSwEx* and *NewSwBC*, respectively.

Theorem 2.2. *Assume that there is $W \in \mathcal{L}$ such that for every finite set D, there are $U, U' \in \mathcal{L}$ such that $U \subset W \subset U'$ and $D \cap U = D \cap W = D \cap U'$. Then $\mathcal{L}$ cannot be SwBC-learned.*

Proof. Let M be a given *SwBC*-learner. Let $\leqslant$ be the ordering induced by some fixed recursive 1-1 enumeration of V_∞, that is, $x \leqslant y$ iff x is enumerated before y or $x = y$. Then there is a teacher (possibly nonalgorithmic) who can present data to M in the following way which confuses M.

1. If the current hypothesis of M is correct for W, and there is a finite sequence of data, that is, a sequence $x_1, \ldots, x_k$ of some length k, corresponding to M's requests $y_1, y_2, \ldots, y_k$, such that after concatenating this sequence the hypothesis of M is incorrect, then the teacher gives x_1 from one of the shortest such sequences.
2. If the current hypothesis of M is incorrect and $y \in \{0, 1\}$ is the data type of M's next request, then output the least x, with respect to the ordering $\leqslant$, which has not yet appeared in the sequence of data given to M and such that $W(x) = y$.
3. In the remaining case, all future hypotheses to requests of M for data consistent with W result in hypotheses for W. We consider the following two subcases.

- If U and U' have already been chosen, then take the least x with respect to the ordering $\leqslant$, which has not yet appeared in the data given by the teacher, and which satisfies $x \in U$ in the case of a request of type 1 and $x \notin U'$ in the case of a request of type 0. If such x does not exist, then give the pause symbol #.
- Otherwise, the learner is given the pause symbol #, while U and U' are chosen as follows. Let D be the set of positive data given to the learner so far. Now, for this D one just chooses U, U' according to the condition in the theorem: $U \subset W \subset U'$ and $D \cap U = D \cap W = D \cap U'$.

For the verification that M does not learn $\mathcal{L}$, assume that M infinitely often conjectures a hypothesis that is incorrect for W. Then the second case applies infinitely often, and the teacher gives either all elements or all nonelements of W to the corresponding requests. Otherwise, from some point on the current hypothesis is correct for W, and the learner ends up in the third case, and U and U' are eventually chosen so that $U \subset W \subset U'$. If infinitely often data of type 1 are requested, then M sees all elements of U and some nonelements of U'. If infinitely often data of type 0 are requested, then M sees all nonelements of U' and some elements of U. In the first case, M is expected to learn U, in the second case, M is expected to learn U'. However, in both cases, M almost always conjectures the set W, and hence does not learn $\mathcal{L}$ from switching.

Note that this proof holds for both convergence criteria, that is, for *SwEx* and *SwBC*. □

The condition of the previous theorem also implies *SwBC*-nonlearnability with respect to general learners, the ones that are not required to be recursive. The reason is that the proof does not require that M is recursive.

3. Learnability and types of quotient spaces

We will show that for learning from text and learning from switching type of information it is possible to characterize learnability of the recursively enumerable subspaces of the quotient space V_∞/V in terms of natural algebraic properties of V_∞/V.

Theorem 3.1. *The following statements are equivalent for any recursively enumerable subspace $V \subseteq V_\infty$.*

(a) *The dimension of V_∞/V is finite.*
(b) *The class $\mathcal{L}(V)$ is TxtEx-learnable.*
(c) *The class $\mathcal{L}(V)$ is TxtBC-learnable.*
(d) *The class $\mathcal{L}(V)$ is SwEx-learnable.*

Proof. (a) ⇒ (b) Assume that the dimension of V_∞/V is finite. Then there is an algorithm that can check for every finite set $D \subset V_\infty$ and every vector $x \in V_\infty$, whether x is in the linear closure of $V \cup D$. As a consequence, the following learner M is recursive.

- Initially set $D = \emptyset$.
- The current hypothesis of M is always the linear closure of $V \cup D$; the hypothesis changes iff a new element is put into D.
- Whenever the teacher provides a datum x, the learner M checks whether x is in the linear closure of $V \cup D$:
  (i) if this is the case, then M does not change D and, therefore, keeps the current hypothesis;
  (ii) otherwise, M updates D to $D \cup \{x\}$ and then also updates its hypothesis.

Since the dimension of V_∞/V is finite, every space in $\mathcal{L}(V)$ is generated by $V \cup D$ for some finite set D. It is easy to verify that the above algorithm finds the set D in the limit. Furthermore, M makes a mind change only at stages at which the current set D properly increases. Thus, M does not make more mind changes than the dimension of V_∞/V, and so the algorithm converges.

(b) ⇒ [(c) and (d)] This follows directly from the definitions.

(c) ⇒ (a) Assume that $\mathcal{L}(V)$ is *TxtBC*-learnable. Furthermore, assume that $V \neq V_\infty$. Let $v_0, v_1, \ldots$ be a recursive enumeration of V_∞, and let U_n be the vector space generated by $V \cup \{v_0, v_1, \ldots, v_n\}$. Clearly, V_∞ is the ascending union of all spaces U_n. It follows from the basic results on learning from text in [7] that such a class can only be learned if the ascending chain is finite. That is, there is m such that $U_n = U_m$ for all $n \geqslant m$. It follows that $V_\infty = U_m$, so the dimension of V_∞ / V is at most $m+1$, hence finite.

(d) ⇒ (a) Assume that $\mathcal{L}(V)$ is *SwEx*-learnable. For a contradiction, assume that V_∞ / V is infinite-dimensional. It is enough to show that V is recursive, since one can find an infinite recursive basis $\{w_0, w_1, \ldots\}$ of a vector space U with $U \cap V = \{0\}$. Let W be the linear closure of $V \cup \{w_x\colon x \in K\}$. The space W is not recursive, and we can now use the following argument below to show that $\mathcal{L}(W)$ is not *SwEx*-learnable. Hence $\mathcal{L}(V)$ is also not *SwEx*-learnable.

Let M be a given recursive learner and let $\leqslant$ be the ordering defined on V_∞ as in Theorem 2.2. There is a teacher for M who does the following.

1. If the current hypothesis of M is old and there is a finite sequence $x_1, x_2, \ldots, x_k$ of data, corresponding to M's requests $y_1, y_2, \ldots, y_k$, such that after concatenating this sequence M changes its hypothesis, then the teacher gives x_1 from a shortest such sequence.
2. If the current hypothesis is new, then, on a request for a datum of type y, the teacher returns the least x (with respect to $\leqslant$) such that x has not yet been given to the learner and $V(x) = y$.

Assume that the protocol continues. Then the learning process goes infinitely often through both cases. It follows that the learner has made infinitely many hypotheses, and that the teacher has either given the learner all elements of V on requests of type 1, or all elements of $\overline{V}$ on requests of type 0. Thus, the learner is given the required information on V, without converging syntactically. Hence, M does not *SwEx*-learn $\mathcal{L}(V)$.

Therefore, there is a stage after which there will be no further mind changes, while data consistent with V continue to be fed to M. We can assume that the current hypothesis is one for V, since otherwise, M would not learn V. Let N be the set of negative data and P the set of positive data seen so far—both sets are, of course, finite.

- Whenever the learner requests positive data, the teacher can provide some elements of V in a way that the learner requests a negative datum after finitely many steps. Let W be a recursively enumerable and proper superspace of V, which is disjoint with N—such a space exists since V_∞ / V has infinite dimension. Thus, there is an infinite sequence of data on which the learner converges to a hypothesis for V, and the teacher gives as positive data only elements of V, and as negative data all nonelements of W. More precisely, the learner requests infinitely often negative data, and after receiving elements from N for some time (up to the stage described above), the learner from that point on receives always the least element $x \notin W$ that the learner has not seen before. Therefore, M does not *SwEx*-learn W.
- Otherwise, the teacher can give M finitely many data consistent with V in such a way that M never later requests data of type 0. Let D be the set of data of type 0 seen so far. Now, one can enumerate $\overline{V}$ as follows: $x \notin V$ iff one can either continue to feed M with data from the linear closure of $V \cup \{x\}$ until a mind change occurs, or a datum of type 0 is requested, or some element of D is enumerated into the linear closure of $V \cup \{x\}$. Hence V is recursive.

This completes the proof that if the dimension of V_∞ / V is infinite, then $\mathcal{L}(V)$ is not *SwEx*-learnable. □

Metakides and Nerode [14] defined a (recursively enumerable) space $V \in \mathcal{L}(V_\infty)$ to be *maximal* if the dimension of V_∞ / V is infinite, and for every recursively enumerable space W such that $V \subseteq W \subseteq V_\infty$, we have that either V_∞ / W is finite-dimensional or W/V is finite-dimensional. Metakides and Nerode used Friedberg's *e*-state method to construct a maximal space. In addition, Shore (see [14]) established that every maximal subset of a recursive basis of V_∞ generates a maximal subspace of V_∞.

Kalantari and Retzlaff [11] defined a space $V \in \mathcal{L}(V_\infty)$ to be *supermaximal* if the dimension of V_∞ / V is infinite, and for every recursively enumerable space $W \supseteq V$, either $W = V_\infty$ or W/V is finite-dimensional. Furthermore, for a natural number $k \geqslant 0$, Kalantari and Retzlaff [11] introduced the concept of a *k-thin* space and showed its existence. A space $V \in \mathcal{L}(V_\infty)$ is *k-thin* if the dimension of V_∞ / V is infinite, and for every recursively enumerable subspace $W \supseteq V$, either the dimension of V_∞ / W is at most k or the dimension of W/V is finite, and there exists $U \in \mathcal{L}(V)$ such that the dimension of V_∞ / U is k. Hence the supermaximal spaces are the same as 0-thin spaces.

Furthermore, Hird [8] introduced the concept of a strongly supermaximal space. A space $V \in \mathcal{L}(V_\infty)$ is *strongly supermaximal* if the dimension of V_∞ / V is infinite, and for every recursively enumerable subset $X \subseteq V_\infty - V$, there exists a finite subset $D \subseteq V_\infty$ such that the set X is contained in the linear closure of $V \cup D$. Hird showed that strongly supermaximal spaces exist. He also established that every strongly supermaximal space is supermaximal, and that not every supermaximal space is strongly supermaximal. Downey and Hird [4] showed that strongly supermaximal spaces exist in every nonzero recursively enumerable Turing degree.

Theorem 3.2. *The class $\mathcal{L}(V)$ is SwBC-learnable iff either V_∞/V is finite-dimensional or V is 0-thin or 1-thin.*

Proof. First assume that V_∞/V has infinite dimension and that V is neither 0-thin nor 1-thin. Then there is a recursively enumerable space W such that $V \subset W \subset V_\infty$, the quotient space W/V has infinite dimension and V_∞/W has dimension at least 2. In particular, there are vectors x_1, x_2 such that $x_1, x_2 \notin W$ and x_1, x_2 are linearly independent over W. Now, for every finite set D of vectors, one can choose a positive integer n such that none of the vectors in $D - W$ is in the linear closure of $W \cup \{x_1 + nx_2\}$. Furthermore, the linear closure of $V \cup (W \cap D)$ has finite dimension over V, and thus is different from W. So the condition in Theorem 2.2 is satisfied, and hence $\mathcal{L}(V)$ is not *SwBC*-learnable.

To prove the converse, we have to consider only the cases of 0-thin and 1-thin spaces, since Theorem 3.1 deals with the case when the dimension of V_∞/V is finite. In these two cases, there is a minimal space W such that $V \subseteq W$ and W/V is infinite-dimensional. Furthermore, if V is 0-thin, we have that $W = V_\infty$. If V is 1-thin, we have that $W \subset V_\infty$ and there is no other such recursively enumerable vector space U with the quotient space U/V having infinite dimension. This property allows us to give the following learning algorithm.

- The learner M requests data of type 0 until one of them is enumerated into W. The learner outputs the hypothesis for V_∞ while no data of type 0 (except pause signs) have shown up, and the hypothesis for W when the first datum of type 0 shows up.
- If some datum of type 0 has shown up in W so far, then M requests data of type 1 and M's current hypothesis is the linear closure of $V \cup E$, where E is the set of all data of type 1 seen so far.

In the cases when the learner has to learn V_∞ or W, the learner M requests only data of type 0. If no datum of type 0 is given, M's hypothesis for V_∞ is correct. If some data of type 0 are given, but they are all in the complement of W, then M's hypothesis for W is correct. In the remaining case, the vector space to be learned is the linear closure of $V \cup D$ for some finite set D. As that space cannot cover W, a datum of type 0 and in W must be given and it causes that from that time on M requests only data of type 1. So the teacher must eventually reveal all elements of the linear closure of $V \cup D$, and after a certain stage, D is contained in the set E of current data used for M's hypothesis. □

4. Learning vector spaces from an informant

The two notions of learning from an informant, *InfEx* and *InfBC*, do not seem to have algebraic characterizations similar to the ones for learning from text in the previous section. In the case of 0-thin vector spaces, the class $\mathcal{L}(V)$ consists just of V_∞ and the vector spaces that are the linear closures of V together with finitely many other vectors. Nevertheless, it depends on the actual choice of V whether the class $\mathcal{L}(V)$ is *InfEx*-learnable. Furthermore, the infinite product of 0-thin spaces (as formalized in Definition 4.2) can be *InfEx*-learnable. On the other hand, the infinite product of 0-thin spaces can be non-*InfBC*-learnable.

For the next theorem, let us recall that K is the halting set, and its complement $\overline{K}$ is the divergence set.

Theorem 4.1. *There is a strongly supermaximal vector space V such that $\overline{K}$ is uniformly enumerable relative to every recursively enumerable vector space W with $V \subseteq W \subset V_\infty$. It will follow that $\mathcal{L}(V)$ is InfEx-learnable.*

Proof. The basic idea is to use the following property, which follows immediately from the definition of a strongly supermaximal vector space.

- If for every recursively enumerable set W with $W \cap V = \emptyset$, we have that W is contained in the linear closure of $V \cup D$ for a finite set D, then either the dimension of V_∞/V is finite or V is strongly supermaximal.

The construction will be in stages. We will enumerate a basis A of V. We will also determine B such that $A \cup B$ is a basis for the whole space V_∞, and B has a recursively enumerable complement. Furthermore, there will be uniformly recursive finite sets $I_{\langle i,j\rangle}$ such that $i \in \overline{K}$ iff there is j such that $I_{\langle i,j\rangle} \cap \overline{V}$ has at least $\langle i, j\rangle + 1$ elements. In the construction, only updated information is stated explicitly, while it is assumed that all other information remains unchanged.

- *Stage* 0: Let $B_0 = \{\epsilon_0, \epsilon_1, \ldots\}$ be the standard basis of V_∞. Let $A_0 = \emptyset$. Let $(I_k)_{k\in\mathbb{N}}$ be a uniformly recursive sequence of pairwise disjoint subsets of B_0, where each I_k is of cardinality $3k$.
- *Stage* $s + 1$ with $s = \langle e, 0, t\rangle$: Assume that $W_{e,s}$ and the linear closure of A_s are disjoint, and that there is an element $x \in W_{e,s}$ such that x is not in the linear closure of $U = A_s \cup I_0 \cup I_1 \cup \cdots \cup I_e$. Assume that x is the least such element. We choose the least basis element $\epsilon_k \in B_s - U$ such that ϵ_k is not in the support of x with respect to the basis $A_s \cup B_s$. We now update A_s and B_s by setting $A_{s+1} = A_s \cup \{x\}$ and $B_{s+1} = B_s - \{\epsilon_k\}$.
- *Stage* $s + 1$ with $s = \langle i, j, t\rangle$, where $j > 0$: If $i \in K_s$ and $I_{\langle i,j\rangle} \cap B_s \neq \emptyset$, then let $A_{s+1} = A_s \cup (I_{\langle i,j\rangle} \cap B_s)$ and $B_{s+1} = B_s - I_{\langle i,j\rangle}$.

We note that if W_e and V are disjoint, then every $x \in W_e$ is in the linear closure of $V \cup I_0 \cup I_1 \cup \cdots \cup I_e$. The set $I_0 \cup I_1 \cup \cdots \cup I_e$ is finite and the linear closure of $V \cup W_e$ has finite dimension over V. Thus, the space V is strongly supermaximal.

Assume that $W \in \mathcal{L}(V_\infty)$ is such that $V \subseteq W \subset V_\infty$. We will show that $\overline{K}$ is uniformly enumerable relative to W. First we conclude that the space W/V has finite dimension, say d.

Given any i, let j be large enough that $\langle i, j\rangle > d$. There are at most $\langle i, j\rangle$ members of $I_{\langle i,j\rangle}$ that might be moved from B, but fail to be enumerated into A at some stage $\langle e, 0, t\rangle + 1$ for a suitable t. There are at least $2 \cdot \langle i, j\rangle$ other vectors in $I_{\langle i,j\rangle}$. These vectors are enumerated into A if $i \in K$, while they all remain in B if $i \notin K$.

In the case when $i \in K$, at most $\langle i, j\rangle$ members of $I_{\langle i,j\rangle}$ can be outside V. It follows that at most $\langle i, j\rangle$ members of $I_{\langle i,j\rangle}$ are in the complement of W.

In the case when $i \notin K$, at least $2 \cdot \langle i, j\rangle$ members of $I_{\langle i,j\rangle}$ permanently remain in B. Since $A \cup B$ is a basis of V_∞, for every $F \subseteq B$, the dimension of the linear closure of $V \cup F$ over V is the cardinality of F. Thus, there can be at most d elements of $I_{\langle i,j\rangle} \cap B$ in W. Since $d < \langle i, j\rangle$, at least $\langle i, j\rangle + 1$ members of $I_{\langle i,j\rangle}$ are in the complement of W.

Thus, it follows that for all recursively enumerable vector spaces W with $V \subseteq W \subset V_\infty$, we have: $i \in \overline{K}$ iff there is j such that $\overline{W} \cap I_{\langle i,j\rangle}$ has at least $\langle i, j\rangle + 1$ elements.

The learning algorithm for $\mathcal{L}(V)$ now uses the fact that the set $\{(e, x)\colon\ x \in V_e\}$ is many-one reducible to K via a recursive function f. Without loss of generality, we may assume that $V_0 = V_\infty$. The *InfEx*-learner now works according to the following protocol. The learner requests alternately positive and negative data, and always outputs the least e such that at stage s, every already seen datum x satisfies the following conditions:

- If x is a negative datum, then $x \notin V_{e,s}$.
- If x is a positive datum, then there is no j such that at least $\langle f(e, x), j\rangle + 1$ negative data from the set $I_{\langle f(e,x),j\rangle}$ have been seen so far.

If the set to be learned is V_∞ itself, then the algorithm always conjectures 0, as long as no negative datum shows up. Otherwise, for every wrong hypothesis e, one will either find an element $x \in W_e$ that appears as a negative datum, or an element $x \notin W_e$ that appears as a positive datum, but it is disproved by the reduction to the set of negative data. It follows that every false intermediate hypothesis is eventually discarded, while a correct hypothesis is never given up. The algorithm converges syntactically to the least index of the space to be learned. Hence the learner *InfEx*-learns $\mathcal{L}(V)$. □

Definition 4.2. For $i \in \mathbb{N}$, let l_i be the linear mapping defined by $l_i(\epsilon_j) = \epsilon_{\langle i,j\rangle}$ for every $j \in \mathbb{N}$. Let V_{lp} be the linear closure of the union $l_0(V) \cup l_1(V) \cup \cdots$ of linear projections. As usual, let $\mathcal{L}(V_{lp})$ be the class of all recursively enumerable superspaces of V_{lp}.

Proposition 4.3. *There is a strongly supermaximal vector space V such that the class $\mathcal{L}(V_{lp})$ is InfEx-learnable.*

Proof. Let V be constructed as in Theorem 4.1. The learning algorithm conjectures the full space V_∞ until an i and an x are found such that $l_i(x) \notin W$ for the space W to be learned. Such an x exists in the case when $W \neq V_\infty$, since the linear closure of the union of all subspaces $l_i(V_\infty)$ is V_∞. The learner now knows that $l_i(V_\infty) \cap W$ is the image of some space U with $V \subseteq U \subset V_\infty$, and so the complement $\overline{K}$ of the halting set can be enumerated by observing which of the basis vectors are in $l_i(V_\infty) - W$. Therefore, one can determine the halting set K, and hence recover the elements that are not in a given set W_e. Thus, the learner always conjectures the least e such that W_e is consistent with the data seen so far, and abandons e whenever either some vector x is enumerated into W_e, although the informant says that $x \notin W_e$, or it is deduced from the enumeration of $\overline{K}$ that some vector x is not in W_e, although the informant said that $x \in W_e$. It is easy to see that this algorithm finds the correct index in the limit. □

Theorem 4.4. *There is a strongly supermaximal vector space V such that the class $\mathcal{L}(V)$ is not InfEx-learnable.*

Proof. The construction is similar to the one in the proof of Theorem 4.1 in that it makes the constructed space V strongly supermaximal, but it does not code any information about K. Instead, it tries to diagonalize infinitely often against every potential recursive learner M_i. We have the following requirements.

- Requirement Set_i: If W_i does not intersect the linear closure of A, then W_i is contained in the linear closure of $A \cup F$ for some finite set F.
- Requirement $Dimension_i$: The dimension of V_∞ / V is at least i.
- Requirement $Hypotheses_{i,j}$: While learning the linear closure of A, either M_i outputs at least j hypotheses, or M_i fails to learn the vector space generated by $A \cup F$ for some finite set F.

At every stage of the construction, there are restraints associated with the requirements. The restraint $R_{\langle i,0\rangle}$ is used for the requirement Set_i, the restraint $R_{\langle i,1\rangle}$ for the requirement $Dimension_i$ and the restraint $R_{\langle i,j\rangle}$, $j \geqslant 2$, for the requirement $Hypotheses_{i,j}$. Each restraint is a set, and $T_{i,s}$ is the union of the restraint sets $R_{j,s}$ with $j < i$.

- *Stage* 0: Recall that $B_0 = \{\epsilon_0, \epsilon_1, \ldots\}$ is the standard basis of V_∞. Let $A_0 = \emptyset$. Furthermore, all restraints are empty, that is, $R_i = \emptyset$ for $i \in \mathbb{N}$.
- *Stage* $s + 1$ with $s = \langle i, 0, t\rangle$ for some t: Let V_s be the linear closure of A_s, and let U_s be the linear closure of $A_s \cup T_{\langle i,0\rangle,s}$. If $W_{e,s} \cap A_s = \emptyset$ and $W_{e,s}$ is not contained in U_s, then let x be the least element in $W_{e,s} - U_s$. Hence x is the linear combination of finitely many elements in $A_s \cup B_s$, where some $y \in B_s - U_s$ is in the support of x. We enumerate x into A and remove the least such y from B_s, that is, we set $A_{s+1} = A_s \cup \{x\}$ and $B_{s+1} = B_s - \{y\}$.
- *Stage* $s + 1$ with $s = \langle i, 1, t\rangle$ for some t: Let $R_{\langle i,1\rangle,s}$ consist of the first s elements in B_s. The sets A_s and B_s remain unchanged.
- *Stage* $s + 1$ with $s = \langle i, j, t\rangle$ for some t and $j \geqslant 2$: Select a teacher who always gives the least datum of requested type, not yet seen by the learner, and who gives the symbol # if there is no such datum. Now search for a finite subset $F \subseteq B_s - T_{i,s}$ and the linear closure U_s of $A_s \cup F$ such that M_i, when given information from U_s by the chosen teacher, outputs at least j hypotheses during the first s rounds of learning. The search is lexicographic over all subsets of the set of first s elements of $B_s - T_{i,s}$, starting with $\emptyset$. In the case when the search succeeds with a parameter set F, let $R_{\langle i,j\rangle,s+1}$ be the set of all negative data seen until the jth hypothesis is output. We enumerate F into A, that is, we set $A_{s+1} = A_s \cup F$ and $B_{s+1} = B_s - F$.

Note that when a requirement $Hypotheses_{i,j}$ with $j \geqslant 2$ is satisfied before the corresponding stage s, then there will be no changes. Furthermore, one can verify, as in a standard finite injury priority construction, that every requirement is either eventually satisfied or eventually not satisfied. Hence, from infinitely many stages of the form $\langle i, j, t\rangle + 1$, only finitely many can make changes. It is easy to verify that the requirements Set_i and $Dimension_i$ are eventually met. Hence, V is strongly supermaximal.

If some requirement $Hypotheses_{i,j}$ is eventually not satisfied, then M_i converges to the same index on V and on the space generated by $V \cup \{v\}$ for some $v \notin V$. Thus, it fails to *InfEx*-learn some set in $\mathcal{L}(V)$. Otherwise, M_i outputs infinitely many hypotheses on the characteristic function of V and, therefore, does not converge syntactically. Therefore, the class $\mathcal{L}(V)$ is not *InfEx*-learnable. □

Theorem 4.5. *There is a strongly supermaximal space V such that $\mathcal{L}(V_{lp})$ is not InfBC-learnable.*

Proof. We can construct a strongly supermaximal space V such that the Turing degree of V (although recursively enumerable) is not high (see [4]). We now consider for every recursive function f, the set W_f generated by the subspaces $l_{\langle i,j\rangle}(V)$ when $f(i) \neq j$, and by the subspaces $l_{\langle i,j\rangle}(V_\infty)$ when $f(i) = j$. The characteristic function of W_f can be computed with oracle V.

If M were an *InfBC*-learner for $\mathcal{L}(V)$, then one could *BC* learn all recursive functions using the oracle V as follows. One translates every informant for f into one for W_f, and every current hypothesis e for W_f into a hypothesis e' for f, which on input i, outputs the first j such that $\epsilon_{\langle i,j\rangle} \in W_e$. However, such a learner with recursively enumerable Turing degree that is not high cannot *BC* learn all recursive functions (see [6]). This contradiction implies the nonlearnability of $\mathcal{L}(V_{lp})$. □

5. Generalizing the results

The previous results hold when the vector space V_∞ is over any infinite recursive field F. If V_∞ is over a finite field, then one of the results changes, namely, the *SwBC*-learnability of classes of superspaces.

Proposition 5.1. *Assume that V_∞ is over a finite field F. Let V be a recursively enumerable subspace of V_∞.*

(a) *The family $\mathcal{L}(V)$ is SwEx-learnable iff V_∞/V is finite-dimensional.*
(b) *If V_∞/V is either finite-dimensional or V is k-thin, then $\mathcal{L}(V)$ is SwBC-learnable.*
(c) *If there is an r.e. subspace W with $V \subset W \subset V_\infty$ such that the quotient spaces V_∞/W and W/V are both infinite-dimensional, then $\mathcal{L}(V)$ is not SwBC-learnable.*

Proof. Part (a) follows from the proof of Theorem 3.1 since the proof does not use the fact that the field is infinite.

Part (b), in the case when V_∞/V is finite-dimensional, is proven using the same learning algorithm as in the case of an infinite field. So, let V be k-thin and let $W \in \mathcal{L}(V)$ be such that V_∞ is the closure of $W \cup \{w_1, w_2, \ldots, w_k\}$ for some $w_1, w_2, \ldots, w_k \in V_\infty$. The set U of linear combinations of $w_1, w_2, \ldots, w_k$ is finite. Now we have the following learning algorithm.

- As long as no element of W shows up as a negative datum, the learner asks for negative data and considers the set $\widetilde{U}$ of those elements of U that have not appeared so far as negative data. Then the current hypothesis is the linear closure of $W \cup \widetilde{U}$.
- Otherwise, that is, when some element of W had been given as a negative datum, the learner starts requesting positive data and always conjectures the linear closure of $V \cup E$, where E is the set of positive data seen so far.

The extension of learning $\mathcal{L}(V)$ from the case of a 1-thin space V to an arbitrary k-thin space V is based on the fact that, due to the finiteness of U, the set $\widetilde{U}$ can be completely determined in the limit. The subspace being learned is generated by $W \cup \widetilde{U}$, unless a negative datum belonging to W is found. Besides this fact, the verification is the same as when V_∞ is over an infinite field.

Part (c) is proven using Theorem 2.2 and the properties of the space W. Recall that $V \subset W \subset V_\infty$, and that the quotient spaces V_∞/W and W/V both have infinite dimension. Let $\{v_0, v_1, \ldots\}$ be a linearly independent set over W, that is, no v_i is in the closure of $W \cup \{v_j\colon j < i\}$. Then W is the lower bound of the linear closures of the sets $W \cup \{v_k\}$ for $k \in \mathbb{N}$. Moreover, W is the upper bound of the linear closures of $V \cup \{u_0, u_1, \ldots, u_i\}$, where $u_0, u_1, \ldots$ is an enumeration of W. Both sequences of spaces converge pointwise to W, from above and from below, and consist of members of $\mathcal{L}(V)$. Thus, $\mathcal{L}(V)$ is not *SwBC*-learnable by Theorem 2.2. □

A more general approach to learning recursively enumerable substructures of a recursive structure with the dependence relations is to consider, instead of V_∞, recursive matroids (see [5,17,18]). A matroid consists of a set X and of a closure operator Φ that maps the power set of X into itself and satisfies certain axioms [15]. The closure operator in matroids corresponds to the linear closure in vector spaces. If we consider infinite recursive matroids, without loss of generality, we can identify X with $\mathbb{N}$.

Definition 5.2. We say that an infinite set X and an operator Φ define a *recursive matroid* (X, Φ) if Φ maps recursively enumerable subsets of X to recursively enumerable subsets of X, and the following axioms are satisfied for $Y, Z \subseteq X$:

(a) If $Y \subseteq Z$ then $\Phi(Y) \subseteq \Phi(Z)$.
(b) We have that $Y \subseteq \Phi(Y)$ and $\Phi(\Phi(Y)) = \Phi(Y)$.
(c) For all $x \in \Phi(Y)$, there is a finite subset $D \subseteq Y$ with $x \in \Phi(D)$.
(d) If $Y = \Phi(Y)$ and $x, y \notin Y$ and $x \in \Phi(Y \cup \{y\})$, then $y \in \Phi(Y \cup \{x\})$.
(e) There is a recursive function f such that $\Phi(W_e) = W_{f(e)}$ for every $e \in \mathbb{N}$.

The axiom (e) can sometimes be strengthened by requiring that we can decide for an element x and a finite set D, whether $x \in \Phi(D)$. However, this condition does not always hold for vector spaces considered here, and we are interested in a generalization of the classes $\mathcal{L}(V)$ for $V \in \mathcal{L}(V_\infty)$. Thus, we adopt only the weaker version (e). The set $\Phi(D)$ generalizes the concept of the linear closure after adding V to D. So $x \in V$ iff $x \in \Phi(\emptyset)$. Thus the axiom (e) corresponds to the case where V is recursively enumerable, while the discussed strengthened version corresponds to V being decidable. Decidability of V is equivalent to the existence of a dependence algorithm over V and implies recursiveness of V as a set (see [14]). Note that the axiom (c) given here is omitted by some authors.

The following remark shows that matroids are sufficiently similar to vector spaces. In particular, one can for submatroids Y, Z of X with $Y \subseteq Z$ introduce a dimension of Z over Y. Here, Y is a submatroid of X if $Y \subseteq X$ and $Y = \Phi(Y)$. As it is understood that the operator on the submatroid Y is the restriction of Φ to Y, the operator is omitted from the submatroid and the submatroid Y is identified with its domain Y.

Remark 5.3. Assume that Y, Z are submatroids of a matroid (X, Φ). Furthermore, assume that $Y \subseteq Z$. One says that Z/Y has dimension n if there is a set D with n elements, but not one with fewer than n elements, such that $Z = \Phi(Y \cup D)$. The space Z/Y has finite dimension iff there is such finite set D.

If $n > 0$, $z \in Z - Y$ and Z/Y has dimension n, then $Z/(\Phi(Y \cup \{z\}))$ has dimension $n - 1$. To see this, first note that there is no set E of cardinality strictly less than $n - 1$ such that $Z = \Phi(Y \cup E \cup \{z\})$, so the dimension of $Z/(\Phi(Y \cup \{z\}))$ is at least $n - 1$. On the other hand, there is a maximal set $F \subset D$ such that $z \notin \Phi(Y \cup F)$. Then $d \in \Phi(Y \cup F \cup \{z\})$ for all $d \in D - F$. It follows that $Z = \Phi(Y \cup F \cup \{z\})$ and the dimension of $Z/(\Phi(Y \cup \{z\}))$ is at most the cardinality of F, in particular, at most $n - 1$.

A consequence is that for every submatroid Y, where X/Y has finite dimension, and for every set $U \subseteq X - Y$, there is a finite set D such that $\Phi(Y \cup D) \cap U = \emptyset$ and $\Phi(Y \cup D \cup U) = X$. Note that D is nonempty iff $\Phi(Y \cup U) \subset X$.

In the following theorem, we consider the class $\mathcal{L}$ of recursively enumerable submatroids of a given infinite recursive matroid. Furthermore, general learners, which do not have to be recursive, are considered. Adleman and Blum [1] proposed to measure the complexity of such learners in terms of their Turing degrees. Together with Theorem 2.2, we have that, whenever a general *SwEx*-learner exists, this learner can be chosen to be recursive relative to the halting problem, i.e., relative to the set K.

Theorem 5.4. *Given a matroid* (X, Φ)*, let* $\mathcal{L}$ *be the class of its recursively enumerable submatroids. Then the following conditions are equivalent*:

(a) *The class* $\mathcal{L}$ *is SwEx-learnable relative to* K.
(b) *The class* $\mathcal{L}$ *is SwBC-learnable relative to* K.
(c) *The condition of Theorem* 2.2 *does not hold.*

Proof. The part (a) ⇒ (b) follows directly from the definition, and (b) ⇒ (c) follows from the proof of Theorem 2.2 since the proof does not use the property that the learner M is recursive. So it also works with general learners that are recursive in K or even in a more powerful oracle. The part (c) ⇒ (a) is shown using the following learning algorithm. Here Φ is defined as in the statement of the theorem, $L_i = \Phi(W_i)$ is the ith recursively enumerable submatroid of X, and D_j is the jth finite subset of X. Furthermore, there are recursive functions f and g such that $\Phi(W_i) = W_{f(i)}$ and $D_j = W_{g(j)}$. Let X_s consist of the first s elements of X with respect to some recursive default enumeration.

Algorithm. After having seen s input data, let P consist of the data of type 1, and N be the data of type 0 seen so far. Find the first triple (i, j, pos) or (i, j, neg) in an enumeration of all such triples satisfying the corresponding condition below:

(pos) $i = g(j)$ and $\Phi(D_j) = \Phi(P)$;
(neg) $N \subseteq \overline{L_i}$ and $P \subseteq L_i$ and $X_s \subseteq \Phi(L_i \cup D_j)$ and $D_j \subseteq \Phi(L_i \cup N)$.

Output the index $f(i)$. In the case of (pos), request positive data, otherwise, request negative data.

Verification. Note that $\Phi(D_j) = \Phi(P)$ iff $D_j \subseteq \Phi(P)$ and $P \subseteq \Phi(D_j)$. So the algorithm only checks whether explicitly given finite sets are contained in some recursively enumerable sets or their complements. Thus the learner is K-recursive. Since there is a triple (i, j, pos) such that $D_j = P$ and $i = g(j)$, the search always terminates and the learner is total.

Assume that Y is the submatroid to be learned. If the dimensions of X/Y and $Y/\Phi(\emptyset)$ are both infinite, then the condition of Theorem 2.2 is satisfied, since for every finite set D, there is a finite set E such that $\Phi(Y \cap D) \subset Y \subset \Phi(Y \cup E)$ and $D \cap \Phi(Y \cup E) = D \cap Y$. So Y could be approximated from below and from above by pointwise convergent series of sets different from Y. Since this cannot happen, at least one of the dimensions X/Y and $Y/\Phi(\emptyset)$ is finite.

Assume that at some stage s, the algorithm takes the triple (i, j, pos). Then $L_i = \Phi(D_j) = \Phi(P)$ and the output is consistent with the input. Since $P \subseteq Y$, we have that $L_i \subseteq Y$. Furthermore, the algorithm will then continue to request positive data until some element outside L_i is seen. This happens eventually iff $Y \neq L_i$.

Assume that the algorithm takes at some stage s the triple (i, j, neg). Then the algorithm will keep this index until it either becomes clear that $X \not\subseteq \Phi(Y \cup W_j)$ or a negative datum in $Y - L_i$ shows up. In the first case, D_j does not witness that the dimension of X/Y is finite, and in the second case, $Y \neq L_i$.

To see that the algorithm converges, let N' and P' be the sets of all negative and positive data seen by the learner throughout the overall running time of the algorithm. If the dimension of $\Phi(P')/\Phi(\emptyset)$ is finite, then there is a triple (i, j, pos) such that $i = g(j)$ and $\Phi(D_j) = \Phi(P')$. From some time on, enough data have been seen so that this triple will be used, unless one of the finitely many triples before it is used almost always. Otherwise, the dimension of $\Phi(P')/\Phi(\emptyset)$ is infinite. Then the dimension of $Y/\Phi(\emptyset)$ is infinite since $P' \subseteq Y$. It follows that the dimension of X/Y is finite. By Remark 5.3, there is a finite set E such that $N' \cap \Phi(Y \cup E) = \emptyset$ and $X = \Phi(Y \cup E \cup N')$. There is a triple (i, j, neg) such that $L_i = \Phi(Y \cup E)$ and $D_j \cap L_i = \emptyset$ and $X = \Phi(L_i \cup D_j)$. Again, this triple will be used almost always unless one of the finitely many triples before it is used almost always. So it follows that the algorithm converges to one triple that is used almost always. In the paragraphs of the verification preceding this one, it has been shown that whenever a triple is used almost always, then the learner converges to the correct hypothesis. □

Each of the *SwBC*-learnable classes of substructures studied so far required only one switch. First, the learner observed negative data. If these data ruled out a fixed subspace W, then the learner switched to positive data and never again abandoned this type of data requests. The following examples give some matroids where unboundedly many switches are required. This situation is more analogous to the general case, for which Jain and Stephan [10] showed that there is a real hierarchy of learnability, depending on the number of switches allowed.

Furthermore, the examples below show that there are classes witnessing both extremes. The class in Example 5.5 has a recursive *SwEx*-learner, while every *SwBC*-learner, and thus also every *SwEx*-learner of the class in Example 5.6 needs oracle K.

Example 5.5. There is a recursive matroid such that every recursively enumerable submatroid is either finite or cofinite. The class of these submatroids can be *SwEx*-learned, but not with any bound on the number of switches.

Proof. Let $X = \mathbb{N}$. Let A be a maximal subset of X. For any given set Y, let

$$\Phi(Y) = \bigl\{x\colon\ (\exists y \in Y)\ \bigl[y = x \vee \bigl(y < x \wedge \{y, y+1, \ldots, x-1\} \subseteq A\bigr) \vee \bigl(y > x \wedge \{x, x+1, \ldots, y-1\} \subseteq A\bigr)\bigr]\bigr\}.$$

Clearly, $\Phi(Y)$ is recursively enumerable whenever Y is.

Let $a_0, a_1, \ldots$ be the ascending (and nonrecursive) enumeration of all elements of $\overline{A}$. Let $A_0 = \{0, 1, \ldots, a_0\}$ and $A_n = \{a_{n-1}+1, a_{n-1}+2, \ldots, a_n\}$ for any $n > 0$. Then $\Phi(Y)$ is the union of all sets A_n that meet Y. If Y is finite, so is $\Phi(Y)$. If Y is infinite, then Y meets infinitely many sets A_n; hence, $\Phi(Y) - A$ is infinite. If Y is in addition recursively enumerable, then the maximality of A implies that $\Phi(Y)$ contains almost all numbers a_n. For these n, the corresponding sets A_n are subsets of $\Phi(Y)$. In particular, $\Phi(Y)$ is cofinite. So whenever Y is recursively enumerable, $\Phi(Y)$ is either finite or cofinite.

Thus, one can *SwEx*-learn all recursively enumerable submatroids. The algorithm is simply to *SwEx*-learn all finite and cofinite sets, as it is done in [10]. □

Example 5.6. There is a recursive matroid whose class of all recursively enumerable submatroids can be *SwBC*-learned with the help of oracle K, but not with any oracle A such that $K \not\leqslant_T A$.

Proof. Let $X = \mathbb{N}$. There is a maximal set A such that the ascending sequence of nonelements $a_0, a_1, \ldots$ of A has the property that for all $n \in K$, the element a_n is larger than the least stage at which n is enumerated into K (see [27]). Let $\Phi(Y) = Y \cup A$. Since A is maximal, the class $\mathcal{L}$ of the recursively enumerable submatroids of (X, Φ) consists of the sets Y such that $A \subseteq Y \subseteq X$, and Y is either a finite variant of A or a finite variant of X.

It is easy to adapt the *SwEx*-learner for finite and cofinite sets from [10] to a *SwEx*-learner using oracle A for $\mathcal{L}$, which, roughly speaking, ignores the elements of A and works as the original learner with respect to the nonelements of A.

Assume now that M is a general *SwBC*-learner for $\mathcal{L}$. Similarly as in Theorem 2.2, one can show the existence of a string σ and a sequence $\tau_0, \tau_1, \ldots$ of strings of answers to requests of M such that:

- the answers in σ are consistent with A, and after receiving them, M will request only positive data unless a nonelement of A is given after some request for a positive datum;
- the mapping $n \to \tau_n$ is recursive in M and for every n, the elements in τ_n contain only elements of $A \cup \{n, n+1, \ldots\}$, and after receiving the answers to its queries in the order occurring in the string $\sigma\tau_n$, the learner M requests a negative datum.

The reason for the existence of these strings is that the dimension of $X/\Phi(\emptyset)$ is infinite, the dimension of $A/\Phi(\emptyset)$ is 0, and the dimension of $X/\Phi(A \cup \{n, n+1, \ldots\})$ is finite. Now consider the following function f, which is recursive relative to M: $f(0)$ is the maximum of a_0 and all elements in $\sigma\tau_0$; $f(n+1)$ is the maximum of all elements in the string $\sigma\tau_{f(n)+1}$. By the choice of σ and $\tau_{f(n)+1}$, some element in $\{f(n)+1, f(n)+2, \ldots\} - A$ occurs in $\tau_{f(n)+1}$. So one can verify inductively that $a_0, a_1, \ldots, a_n \leqslant f(n)$: $a_0 \leqslant f(0)$; between $f(n)+1$ and $f(n+1)$ there is at least another element of $\overline{A}$. It follows that $n \in K$ iff n is enumerated into K within $f(n)$ computational steps, and so K is recursive relative to M. That is, the Turing degree of M is greater or equal than the Turing degree of K. □

References

[1] Leonard M. Adleman, Manuel Blum, Inductive inference and unsolvability, J. Symbolic Logic 56 (1991) 891–900.

[2] Dana Angluin, Learning regular sets from queries and counterexamples, Inform. and Comput. 75 (1987) 87–106.

[3] Ganesh Baliga, John Case, Sanjay Jain, Language learning with some negative information, J. Comput. System Sci. 51 (1995) 273–285.

[4] Rod G. Downey, Geoffrey R. Hird, Automorphisms of supermaximal spaces, J. Symbolic Logic 50 (1985) 1–9.

[5] Rod G. Downey, Jeffrey B. Remmel, Computable algebras and closure systems: Coding properties, in: Yu.L. Ershov, S.S. Goncharov, A. Nerode, J.B. Remmel (Eds.), V.W. Marek (Assoc. Ed.), Handbook of Recursive Mathematics, vol. 2, Elsevier, Amsterdam, 1998, pp. 977–1039.

[6] Lance Fortnow, William Gasarch, Sanjay Jain, Efim Kinber, Martin Kummer, Stuart Kurtz, Mark Pleszkoch, Theodore Slaman, Robert Solovay, Frank Stephan, Extremes in the degrees of inferability, Ann. Pure Appl. Logic 66 (1994) 231–276.

[7] E. Mark Gold, Language identification in the limit, Inform. Control 10 (1967) 447–474.

[8] Geoffrey R. Hird, Recursive properties of relations on models, Ann. Pure Appl. Logic 63 (1993) 241–269.

[9] Sanjay Jain, Arun Sharma, On the nonexistence of maximal inference degrees for language identification, Inform. Process. Lett. 47 (1993) 81–88.

[10] Sanjay Jain, Frank Stephan, Learning by switching type of information, Inform. and Comput. 185 (2003) 89–104.

[11] Iraj Kalantari, Allen Retzlaff, Maximal vector spaces under automorphisms of the lattice of recursively enumerable vector spaces, J. Symbolic Logic 42 (1977) 481–491.

[12] Michael Machtey, Paul Young, An Introduction to the General Theory of Algorithms, North-Holland, New York, 1978.
[13] Wolfgang Merkle, Frank Stephan, Refuting learning revisited, Theoret. Comput. Sci. 298 (2003) 145–177.
[14] George Metakides, Anil Nerode, Recursively enumerable vector spaces, Ann. Math. Logic 11 (1977) 147–171.
[15] George Metakides, Anil Nerode, Recursion theory on fields and abstract dependence, J. Algebra 65 (1980) 36–59.
[16] Tatsuya Motoki, Inductive inference from all positive and some negative data, Inform. Process. Lett. 39 (1991) 177–182.
[17] Anil Nerode, Jeffrey Remmel, Recursion theory on matroids, in: G. Metakides (Ed.), Patras Logic Symposium, in: Stud. Logic Found. Math., vol. 109, North-Holland, 1982, pp. 41–65.
[18] Anil Nerode, Jeffrey B. Remmel, Recursion theory on matroids II, in: C.T. Chong, M.J. Wicks (Eds.), Southeast Asian Conference on Logic, North-Holland, New York, 1983, pp. 133–184.
[19] Anil Nerode, Jeffrey Remmel, A survey of lattices of r.e. substructures, in: A. Nerode, R.A. Shore (Eds.), Recursion Theory, in: Proc. Sympos. Pure Math., vol. 42, Amer. Math. Soc., Providence, RI, 1985, pp. 323–375.
[20] Piergiorgio Odifreddi, Classical Recursion Theory, North-Holland, Amsterdam, 1989.
[21] Daniel N. Osherson, Michael Stob, Scott Weinstein, Systems That Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists, MIT Press, 1986.
[22] Hartley Rogers, Theory of Recursive Functions and Effective Computability, McGraw–Hill, 1967, reprinted by MIT Press, 1987.
[23] Arun Sharma, A note on batch and incremental learnability, J. Comput. System Sci. 56 (1998) 272–276.
[24] Robert I. Soare, Recursively Enumerable Sets and Degrees. A Study of Computable Functions and Computably Generated Sets, Springer-Verlag, Berlin, 1987.
[25] Frank Stephan, Yuri Ventsov, Learning algebraic structures from text, Theoret. Comput. Sci. 268 (2001) 221–273.
[26] Rolf Wiehagen, Identification of formal languages, in: Mathematical Foundations of Computer Science, in: Lecture Notes in Comput. Sci., vol. 53, Springer-Verlag, Berlin, 1977, pp. 571–579.
[27] C.E. Mike Yates, Three theorems on the degree of recursively enumerable sets, Duke Math. J. 32 (1965) 461–468.