On the Data Consumption Benefits of Accepting Increased Uncertainty

Eric Martin [a,b] Arun Sharma [c] and Frank Stephan [d,e]

[a] *School of Computer Science and Engineering, UNSW Sydney, NSW 2052, Australia*

[b] *National ICT Australia, UNSW Sydney NSW 2052, Australia*

[c] *Division of Research and Commercialisation, Queensland University of Technology, 2 George street, GPO Box 2434, Brisbane QLD 4001, Australia*

[d] *School of Computing, National University of Singapore, 3 Science Drive 2, Singapore 117543, Republic of Singapore*

[e] *Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543, Republic of Singapore*

Abstract

In the context of learning paradigms of identification in the limit, we address the question: why is uncertainty sometimes desirable? We use mind change bounds on the output hypotheses as a measure of uncertainty, and interpret 'desirable' as reduction in data memorization, also defined in terms of mind change bounds. The resulting model is closely related to iterative learning with bounded mind change complexity, but the dual use of mind change bounds — for hypotheses and for data — is a key distinctive feature of our approach. We show that situations exists where the more mind changes the learner is willing to accept, the lesser the amount of data it needs to remember in order to converge to the correct hypothesis. We also investigate relationships between our model and learning from good examples, set-driven, monotonic and strong-monotonic learners, as well as class-comprising versus class-preserving learnability.

[1] National ICT Australia is funded by the Australian Government's Department of Communications, Information Technology and the Arts and the Australian Research Council through Backing Australia's Ability and the ICT Centre of Excellence Program.

[2] Frank Stephan is supported by the NUS research grant R252–000–212–112.

1 Introduction

Human beings excel at making decisions based on incomplete information. Even in situations where access to larger sets of data would result in a definitely correct hypothesis, they might not wait to receive the complete information which would allow them to conclude with certainty that their decision is definitely the best. They often propose a hypothesis having received little or no evidence, knowing that this hypothesis might not be correct and could be revised; they then gather a few more data and propose another and hopefully better hypothesis. Making imperfect decisions, but making them now and in the near future, is better than having to wait an unknown and possibly very long time to make the perfect decision. From a logical point of view, human beings draw uncertain and nonmonotonic inferences on the basis of few assumptions when they could make deductions on the basis of much larger sets of premises. From the point of view of Inductive inference, they accept to change their hypotheses (execute a mind change) when they could learn the underlying concept with certainty (without any mind change) if they waited to receive a larger set of data.

Let us illustrate these considerations with an example. Take a class $\mathcal{L}$ of languages (nonempty r.e. subsets of the set $\mathbb{N}$ of natural numbers) and a text t for a member L of $\mathcal{L}$ (so every positive example—member of L—occurs in the enumeration t while no nonmember of L—negative example—occurs in t). A learner M (algorithmic device that maps finite sequences of data to hypotheses, represented as r.e. indexes) that successfully Ex-identifies L in the limit from t (outputs as hypothesis an r.e. index for L in response to all but finitely many initial segments of t) will often find that many elements in t are useless at the time they are presented to it. For example, assume that $\mathcal{L}$ is the set of all nonempty final segments of $\mathbb{N}$. If 7 is the first natural number that occurs in t then M will likely conjecture that $L = \{7, 8, 9, \ldots\}$. If 11 is the next natural number that occurs in L then 11 will be seen as useless and can be forgotten. Only a natural number smaller than 7 will, in case it occurs in t, be used by M. We follow the line of research based on limiting the long term memory [3,4,6,13,14]. We propose a model where the learner can decide to either remember or forget any incoming datum, based on previously remembered data. The learner can output conjectures using only the data it has recorded. It could for instance remember at most three data, in which case it could output at most four hypotheses—one before the first datum is presented, and one after the presentation of each of the three data. More generality is achieved by assuming that the learner has an ordinal data counter, that can store values smaller than a given ordinal α. When it sees a new datum, the learner can either memorize it and decrease the value of its data counter, or decide not to use that datum and leave the value of the data counter unchanged. The basic idea is that learning incurs a cost whenever the

learner considers that a piece of information is relevant to the situation and should be memorized; we propose to capture this cost by down-counting an ordinal counter. This forces learners to make a restrictive or frugal use of data.

In this paper we start investigating why a learner would accept to give up certainty for uncertainty, or a low degree of uncertainty for a higher degree of uncertainty. As a measure of uncertainty, we take mind change bounds on the output hypotheses (in a model where outputting a new hypothesis requires to decrease the value of an ordinal counter), rather than probabilistic accuracy on the output hypotheses. The main reason for thinking of uncertainty in terms of mind change bounds is that we measure the benefit of additional uncertainty—frugal use of data—in terms of ordinals, and it is natural to use a common 'unit of measure.' Hence we can think of learners as being equipped with two kinds of ordinal counters: a hypothesis counter that measures uncertainty, and a data counter that measures the cost of keeping track of information.

Section 2 describes the framework, which is basically a model of iterative learning with at most α mind changes, but where α is interpreted as an upper bound on the number of data that can be remembered and used rather than an upper bound on the number of mind changes as to what could be the target concept. Section 3 contains the conceptually most interesting result in the paper, that determines the exact tradeoff between frugal use of data consumption and degree of uncertainty for a particular class of languages to be learnt. Intuitively, the more mind changes the learner is willing to accept, the lesser the amount of data it needs to remember in order to converge to the correct hypothesis. This section also exhibits a class of languages that can be learnt by learners making frugal use of data if data come from fat texts (where positive examples occur infinitely often), but not if data come from arbitrary texts. Section 4 investigates some of the relationships between our framework and learning paradigms with more restrictive learners. In particular, it is shown that frugal use of data is a weaker concept than learnability by consistent and strong-monotonic learners, but not weaker than learnability by consistent and monotonic learners. Section 5 investigates some of the relationships between our framework and learning paradigms with more restrictive learning criteria. In particular, it is shown that class-comprising learnability from good examples, but not class-preserving learnability from good examples, is a weaker concept than frugal use of data. In Section 6, we generalize the notion of frugality by allowing learners to forget data. More precisely, if a learner exchanges a new datum for one that has been remembered so far then no cost will be incurred w.r.t. this new notion: costs will only be associated with situations when new data are memorized while all of the data still remembered are kept in memory. We conclude in Section 7.

2 Model description

The set of rational numbers is denoted $\mathbb{Q}$. We follow the basic model of Inductive inference as introduced by Gold [8] while applying some notation from our previous work [17,18]. $\mathcal{D}$ denotes a recursive set of data. This set can be assumed to be equal to $\mathbb{N}$ and code, depending on the context, natural numbers, finite sequences of natural numbers, finite sequences of ordinals, etc. We usually do not make the coding explicit. We denote by $\sharp$ an extra symbol that represents an empty piece of information. Given a set X, the set of finite sequences of members of X is denoted $X^\star$; the concatenation of two members σ, τ of $X^\star$ is represented by $\sigma \star \tau$, just written $\sigma \star x$ in case τ is of the form (x). Given a member σ of $(\mathcal{D} \cup \{\sharp\})^\star$, we denote by $\mathrm{cnt}(\sigma)$ the set of members of $\mathcal{D}$ that occur in σ. Given a sequence t whose length is at least equal to an integer n, and given $i \leq n$, $t[i]$ represents the initial segment of e of length i and if $i < n$, $t(i)$ represents its $(i+1)$st element. The cardinality of a finite set D is denoted $|D|$.

We fix an acceptable enumeration $(\varphi_e)_{e \in \mathbb{N}}$ of the unary partial recursive functions over $\mathcal{D}$. For all $e \in \mathbb{N}$, W_e denotes the domain of φ_e, and D_e denotes the finite set whose canonical index is e (*i.e.*, $D_e = \emptyset$ if $e = 0$ and $D_e = \{x_0, \ldots, x_n\}$ if $e = 2^{x_0} + \ldots + 2^{x_n}$ for some $n, x_0, \ldots, x_n \in \mathbb{N}$ with $x_0 < \ldots < x_n$). The symbol $\mathcal{L}$ refers to a class of r.e. subsets of $\mathcal{D}$, representing the class of languages to be learnt; hence $\mathcal{L}$ is a class of sets of the form W_e for e ranging over a particular subset of $\mathbb{N}$. A member σ of $(\mathcal{D} \cup \{\sharp\})^\star$ is said to be *consistent in* $\mathcal{L}$ iff there exists $L \in \mathcal{L}$ with $\mathrm{cnt}(\sigma) \subseteq L$. A *text* for a member L of $\mathcal{L}$ is an enumeration t of $L \cup \{\sharp\}$ where each member of L occurs at least once; t is said to be *fat* if every member of L has infinitely many occurrences in t. A *learner* is a partial computable function from $(\mathcal{D} \cup \{\sharp\})^\star$ into $\mathbb{N}$. Learnability means Ex-identification (in the limit): for all $L \in \mathcal{L}$ and for all texts t for L, the learner has to output on cofinitely many strict initial segments of t a fixed $e \in \mathbb{N}$ such that $W_e = L$.

We first define the key notion of memory function.

Definition 1 *A* memory function *is defined as a partial recursive function h from $(\mathcal{D} \cup \{\sharp\})^\star$ into $2^{\mathcal{D}}$ such that $h(()) = \emptyset$ and for all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$:*

- *for all $\tau \in (\mathcal{D} \cup \{\sharp\})^\star$ with $\sigma \subseteq \tau$, if $h(\sigma)$ is undefined then $h(\tau)$ is undefined;*
- *for all $x \in \mathcal{D} \cup \{\sharp\}$, $h(\sigma \star x)$ is undefined, or equal to $h(\sigma)$, or equal to $h(\sigma) \cup \{x\}$.*

Let Z be the set of all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ such that $h(\sigma)$ is defined. Given $\sigma, \tau \in Z$ let $R(\sigma, \tau)$ hold iff $\tau \subseteq \sigma$ and $h(\sigma) \neq h(\tau)$. The ordinal complexity of h *is undefined if (Z, R) is not well founded and is equal to the length of (Z, R)*

otherwise.[3]

Definition 2 *Let a learner M be given.*

Given a memory function h, we say that M has h-memory *iff M can be represented as $g \circ h$ for some partial recursive function g.*

The data consumption complexity of M *is defined and equal to ordinal α iff α is the least ordinal such that M has h-memory, for some memory function h whose ordinal complexity is equal to α.*

When it faces an incoming datum d, a learner that has h-memory for some memory function h will either decide not to memorize d, or memorize d for ever—the learner cannot memorize d for some time and then forget it.

Definition 3 *Let a learner M be given. Let Z be the set of all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ such that $M(\sigma)$ is defined. Given $\sigma, \tau \in Z$ let $R(\sigma, \tau)$ hold iff there is $\eta \in Z$ with $\tau \subseteq \eta \subseteq \sigma$ and $M(\sigma) \neq M(\eta)$. The* mind change complexity of M *is undefined if (Z, R) is not well founded, and equal to the length of (Z, R) otherwise.*

We will often write 'DC complexity' for 'data consumption complexity,' and 'MC complexity' for 'mind change complexity.' If M Ex-identifies $\mathfrak{L}$ and if the mind change complexity of M is equal to ordinal α, then M Ex-identifies $\mathfrak{L}$ with less than α mind changes, that is, with at most β mind changes if α is of the form $\beta + 1$ and with no mind change if $\alpha = 1$. Note that if α is a limit ordinal, then Ex-identifying $\mathfrak{L}$ with less than α mind changes is not expressible as Ex-identifying $\mathfrak{L}$ with at most β mind changes for some β.

In the definitions above, and in case (Z, R) is well founded, one can furthermore define the *effective* length of (Z, R) as the least ordinal α such that there exists a recursive well-ordering $\sqsubseteq$ of Z that is isomorphic to $\{\beta : \beta < \alpha\}$, with $\sigma \sqsubset \tau$ whenever $R(\sigma, \tau)$; note that σ, τ might be equivalent with respect to $\sqsubseteq$ if they are not strictly ordered by R. Since Z is a subset of a recursively enumerable tree and R is compatible with $\{(\sigma, \tau) : \sigma, \tau \in (\mathcal{D} \cup \{\sharp\})^\star, \sigma \supset \tau\}$, the restriction of the Kleene-Brouwer ordering of a tree is always such an ordering, and so the effective length of (Z, R) is defined whenever the length of (Z, R) is defined. The effective length of (Z, R) yields the notions of effective data consumption complexity and mind change complexity, in the respective definitions. The effective mind change complexity is equivalent to the traditional one [1,7,12,10,20] defined by ordinal counters; the noneffective version is used

[3] Assuming that (Z, R) is well founded, and denoting by ρ_R the unique function from Z into the class of ordinals such that $\rho_R(x) = \sup\{\rho_R(y) + 1 : y \in Z,\ R(y, x)\}$ for all $x \in Z$, the length of (Z, R) is defined as the least ordinal not in the range of ρ_R.

in our previous studies of the noneffective setting [17,18]. In [1] it is observed that any learner which makes on all texts only finitely many mind changes has an effective mind change complexity, that is, the effective mind change complexity is defined whenever the noneffective one is defined. The results in this paper hold for both the effective and the noneffective versions of mind change and data consumption complexities. Note that both the notions of data consumption complexity and of mind change complexity are defined 'externally,' without requiring the learner to explicit update an ordinal counter, hence avoiding the issues related to programs for the ordinals. More precisely, these externally defined notions provide a lower bound on the ordinal given by a learner that explicitly updates an ordinal counter, using a particular program for the ordinals.

Definition 4 *A learner whose MC complexity is defined is said to be* confident.

A learner whose DC complexity is defined is said to be frugal.

In the literature, a confident learner is usually defined as a learner that converges on any text, whether this text is or is not for a member of the class $\mathcal{L}$ of languages to identify in the limit. We use the term 'confident' in the definition above relying on the fact (see [1]) that Ex-identification with a bounded number of mind changes if equivalent to Ex-identification by a confident learner.

3 DC complexity versus MC complexity

We first state a few obvious relationships.

Property 5 *If the DC complexity of a leaner M is defined and equal to ordinal α, then the MC complexity of M is defined and at most equal to $\alpha + 1$.*

Remember the definition of an iterative learner:

Definition 6 *A learner M is said to be* iterative *iff it is total and there exists a total recursive function $g : \mathbb{N} \times (\mathcal{D} \cup \{\sharp\}) \rightarrow \mathbb{N}$ such that for all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ and $x \in \mathcal{D} \cup \{\sharp\}$, $M(\sigma \star x) = g(M(\sigma), x)$.*

Coding data into hypotheses immediately yields the following properties.

Property 7 *Let a nonnull ordinal α be given. If some iterative learner whose MC complexity is equal to α Ex-identifies $\mathcal{L}$, then some learner whose DC complexity is at most equal to α Ex-identifies $\mathcal{L}$.*

Property 8 *Let an ordinal α be given. If some learner whose DC complexity*

is equal to α Ex-identifies $\mathcal{L}$, then some iterative learner whose MC complexity is at most equal to $\alpha + 1$ Ex-identifies $\mathcal{L}$.

The notion of data consumption complexity does not collapse to any ordinal smaller than ω_1^{ck}, the first nonrecursive ordinal:

Proposition 9 *For all any recursive ordinal α, $\mathcal{L}$ can be chosen such that:*

- *α is the least ordinal such that some learner whose DC complexity is equal to α Ex-identifies $\mathcal{L}$;*
- *α is the least ordinal such that some learner whose MC complexity is equal to α Ex-identifies $\mathcal{L}$;*
- *some learner whose DC and MC complexities are equal to α Ex-identifies $\mathcal{L}$.*

PROOF. It suffices to define $\mathcal{L}$ as the class of all sets of the form $\{\gamma < \alpha : \gamma \geq \beta\}$ for β ranging over the set of ordinals smaller than a given nonnull ordinal α. (What is really considered here is a class of sets of codes of ordinals, for an appropriate coding of the ordinals smaller than α. We do not explicit the coding in order not to clutter the argument.) The optimal learner memorizes a datum iff it is an ordinal β smaller than all ordinals seen so far; the conjecture is $\{\gamma < \alpha : \gamma \geq \beta\}$.

The next proposition captures the fundamental idea that in some sense, a learner can be better and better off in terms of frugal use of data if it is keen to accept higher and higher degrees of uncertainty:

Proposition 10 *For all nonnull $k \in \mathbb{N}$, $\mathcal{L}$ can be chosen such that:*

- *for all $h < k$, some learner whose DC is $\omega \times (k - h)$ Ex-identifies $\mathcal{L}$ with at most h mind changes, and no learner whose DC is smaller than $\omega \times (k - h)$ Ex-identifies $\mathcal{L}$ with at most h mind changes—in particular, some learner whose DC is $\omega \times k$ Ex-identifies $\mathcal{L}$ with no mind change, and no learner whose DC is smaller than $\omega \times k$ Ex-identifies $\mathcal{L}$ with no mind change;*
- *some learner whose DC is $2k$ Ex-identifies $\mathcal{L}$ with at most k mind changes, and no learner whose DC is smaller than $2k$ Ex-identifies $\mathcal{L}$.*

PROOF. For all $n \in \mathbb{N}$, let I_n and J_n be the two subsets of $\mathbb{N}$ of cardinality $n + 3$ such that $\{I_n : n \in \mathbb{N}\} \cup \{J_n : n \in \mathbb{N}\}$ is a partition of $\mathbb{N}$ and for all $n \in \mathbb{N}$, the members of I_n are smaller than the members of J_n, and the members of J_n are smaller than the members of I_{n+1}. Hence $I_0 = \{0, 1, 2\}$, $J_0 = \{3, 4, 5\}$, $I_1 = \{6, 7, 8, 9\}$, $J_1 = \{10, 11, 12, 13\}$, etc. Let a nonnull $k \in \mathbb{N}$ be given. Define $\mathcal{L}$ to be the class of all unions of k disjoint sets of the form

I_n or $(I_n \setminus \{\min(I_n)+m\}) \cup \{\min(J_n)+m\}$ with $n \in \mathbb{N}$ and $m < n+3$. For instance, if $k=1$ then $\mathfrak{L}$ contains $\{0,1,2\}$, $\{3,1,2\}$, $\{0,4,2\}$ and $\{0,1,5\}$.

Note that if a learner M gets a datum x from a member L of $\mathfrak{L}$, then M can determine the unique $n \in \mathbb{N}$ such that $x \in I_n \cup J_n$, and M knows that precisely $n+3$ members of $L \cap (I_n \cup J_n)$ will eventually occur in the text t for L it is presented with. Since k is fixed, it follows that M can Ex-identify $\mathfrak{L}$ with no mind change. Let $L \in \mathfrak{L}$ be such that for all $n \in \mathbb{N}$, $J_n \cap L = \emptyset$. Let X be the set of all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ that are consistent in $\mathfrak{L}$. Let Y be the set of all $\sigma \in X$ such that for all $n \in \mathbb{N}$ and $m < n+3$, if $\min(J_n)+m$ occurs in $\mathrm{cnt}(\sigma)$ then all members of $I_n \setminus \{\min(I_n)+m\}$ occur in $\mathrm{cnt}(\sigma)$ before the first occurrence of $\min(J_n)+m$ in $\mathrm{cnt}(\sigma)$. Given $Z \in \{X,Y\}$, let h_Z be the (unique) memory function such that for all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$, $h_Z(\sigma) = \mathrm{cnt}(\sigma)$ if $\sigma \in Z$, and $h_Z(\sigma) = \uparrow$ otherwise. It is immediately verified that both h_X and h_Y have ordinal complexity equal to $\omega \times k$. Finally, if a learner M having h-memory Ex-identifies $\mathfrak{L}$ with no mind change, then h necessarily extends h_Y, because for all $\sigma \in Y$, M has to remember all members of $\mathrm{cnt}(\sigma)$. We conclude that some learner whose DC is $\omega \times k$ Ex-identifies $\mathfrak{L}$ with no mind change, but no learner whose DC is smaller than $\omega \times k$ Ex-identifies $\mathfrak{L}$ with no mind change.

Let a learner M have the following property when processing a text t for a member L of $\mathfrak{L}$. For all $n \in \mathbb{N}$, if $I_n \cap L$ is nonempty then M remembers (1) the first member of $I_n \cup J_n$ that occurs in t and (2) the (unique) member of J_n that occurs in t, if such is the case. Obviously, M can have a data complexity of $2k$. Moreover, M can Ex-identify $\mathfrak{L}$ with at most k mind changes: it suffices to make the first guess when data from k sets of the form $I_n \cup J_n$ have been observed, conjecturing that L contains I_n if some member of I_n but no member of J_n has been observed, and conjecturing that $(I_n \setminus \{\min(I_n)+m\}) \cup \{\min(J_n)+m\}$ is included in L if $\min(J_n)+m$ has been observed. Clearly, M does not Ex-identify $\mathfrak{L}$ with less than k mind changes. Finally, it is easy to verify that a learner whose data complexity is smaller than $2k$ cannot Ex-identify $\mathfrak{L}$.

The remaining part of the proposition is proved using a combination of the previous arguments.

The notion of data consumption complexity is very restrictive, but becomes more general when learning from fat texts:

Proposition 11 *It is possible to choose $\mathfrak{L}$ such that:*

- *some learner whose DC complexity is ω Ex-identifies $\mathfrak{L}$ from fat texts;*
- *no frugal learner Ex-identifies $\mathfrak{L}$ from texts.*

PROOF. Suppose that $\mathfrak{L}$ consists of the set $2\mathbb{N}$ of even numbers and every set of the form $D \cup \{2x+1\}$ where $|D| = x$ and $D \subseteq 2\mathbb{N}$.

Assume that the learner sees $\sigma \in (2\mathbb{N} \cup \{\sharp\})^\star$. Let $y = 2|\operatorname{cnt}(\sigma)| + 1$. The learner has to memorize $\operatorname{cnt}(\sigma)$ since it cannot exclude that the set to be learned is $\operatorname{cnt}(\sigma) \cup \{y\}$ and the current text is $\sigma \star y \star \sharp^\infty$. Thus the learner cannot be frugal.

A learner that learns from fat text can have a memory function h that extracts from any σ the first odd number $2x+1$ – if there is any – and then the first occurrences of up to x many even numbers after the first occurrence of $2x+1$. The hypothesis of the learner on input σ is $2\mathbb{N}$ if $h(\sigma) = \emptyset$ and $\operatorname{cnt}(h(\sigma))$ otherwise. The learner succeeds because fat texts enable to delay the reading of the members of D after it has seen $2x+1$, whenever the set to be learnt is of the form $D \cup \{2x+1\}$.

The next proposition gives other relationships between data consumption complexity, mind change complexity, and learning from fat texts.

Proposition 12 *For all recursive ordinals α, $\mathfrak{L}$ can be chosen such that:*

- *some learner whose DC and MC complexities are equal to α Ex-identifies $\mathfrak{L}$;*
- *some nonfrugal learner whose MC complexity is equal to 1 Ex-identifies $\mathfrak{L}$;*
- *some learner whose DC complexity is equal to $\alpha+1$ and whose MC complexity is equal to 1 Ex-identifies $\mathfrak{L}$ on fat texts;*
- *no frugal learner whose MC complexity is smaller than α Ex-identifies $\mathfrak{L}$.*

PROOF. Let a recursive ordinal α be given. Suppose that $\mathfrak{L}$ consists of the following, for any $n > 0$, nonempty decreasing sequence $(\alpha_1, \ldots, \alpha_n)$ of ordinals smaller than α, and increasing sequence $(q_1, \ldots, q_n)$ of rational numbers:

- all nonempty initial segments of $((\alpha_1, q_1), \ldots, (\alpha_n, q_n))$;
- all rational numbers at least equal to q_n.

A learner that only remembers sequences of the form $((\alpha_1, q_1), \ldots, (\alpha_n, q_n))$ and outputs an index for $\{((\alpha_1, q_1), \ldots, (\alpha_n, q_n))\} \cup \{q \in \mathbb{Q} : q \geq q_n\}$ for the longest such sequence, will clearly have data consumption complexity equal to α, and can Ex-identify $\mathfrak{L}$ with less than α mind changes, but cannot do any better.

A learner that only remembers sequences of the form $((\alpha_1, q_1), \ldots, (\alpha_n, q_n))$ and rational numbers smaller than any rational number remembered before will wait till both (α', q) and q are observed for some α' and q, thanks to which

it can Ex-identify $\mathcal{L}$ with no mind change. On the other hand, considering texts that start with decreasing sequences of rational numbers that are long enough, it is easily verified that this learner has undefined data consumption complexity.

On fat texts, a learner can Ex-identify $\mathcal{L}$ with no mind change by only remembering sequences of the form $((\alpha_1, q_1), \ldots, (\alpha_n, q_n))$, followed by a rational number that also occurs last in the (necessarily) longest remembered sequence of the previous form. Such a learner will clearly have DC complexity equal to $\alpha + 1$.

The last claim of the proposition is easily verified.

When $\mathcal{L}$ is an indexed family, it is sometimes essential to have a partial memory function as opposed to a total one:

Proposition 13 *It is possible to choose $\mathcal{L}$ such that:*

- *$\mathcal{L}$ is an indexed family;*
- *there exists a frugal learner that Ex-identifies $\mathcal{L}$;*
- *for all total computable memory functions h and frugal learners M, if M has h-memory then M does not Ex-identify $\mathcal{L}$.*

PROOF. Define $\mathcal{L}$ to be the set of all sets of the form $\{(x,0), \ldots, (x,y)\}$ where $x, y \in \mathbb{N}$ and there exists $e \leq x$ with $\varphi_e(x)$ being defined and at least equal to y. Clearly, $\mathcal{L}$ is an indexed family. Define a learner M as follows. Supposed that M has memorized $((x,y_1), \ldots, (x,y_k))$ for some $k, y_1, \ldots, y_k \in \mathbb{N}$ with $y_1 < \ldots < y_k$ and is presented with (x,y). If $y \leq y_k$ then M discards (x,y). If $y > y_k$ then M tries to compute $\varphi_0(x), \ldots, \varphi_x(x)$. In case one computation eventually converges and its output is at least equal to y then M remembers (x,y); otherwise M discards (x,y). The hypothesis output by M is $\{(x,0), \ldots, (x,y)\}$ with (x,y) being the pair remembered last. It is easily verified that M has data consumption complexity bounded by $\omega \times \omega$, and that M Ex-identifies $\mathcal{L}$.

Let a total memory function h and a frugal learner M be such that M has h-memory. For all $x \in \mathbb{N}$, let $\widehat{h}(x)$ be the least $y > 0$ such that

$$h((x,0), \ldots, (x,y)) = \{(x,0), \ldots, (x,y-1)\}.$$

Let e be an index for $\widehat{h}$. Put:

$$L_1 = \{(e,0), \ldots, (e,\widehat{h}(e))\} \text{ and } L_2 = \{(e,0), \ldots, (e,\widehat{h}(e)-1)\}.$$

Then L_1 and L_2 both belong to $\mathfrak{L}$. But it follows immediately from the definition of $\widehat{h}(x)$ that M will not correctly converge on both $(e,0),\ldots,(e,\widehat{h}(e)),\sharp,\ldots$ (text for L_1) and $(e,0),\ldots,(e,\widehat{h}(e)-1),\sharp,\ldots$ (text for L_2).

4 Relationships with some restricted learners

In this section, we will investigate how some of the usual restrictions on learners [19] affect the notion of data consumption complexity. Whether it can be enforced that every hypothesis is consistent with the data seen so far (consistency), whether it can be enforced that every hypothesis extends the previous hypothesis (monotonicity), whether it can be enforced that every hypothesis equals the previous hypothesis H in case all data seen so far are consistent with H (conservativeness), are among the questions that have received a lot of attention, see for example [2,11,13,16,19,21].

Definition 14 *A learner M is said to be* consistent *iff for all $L \in \mathfrak{L}$ and $\sigma \in (L \cup \{\sharp\})^\star$, $M(\sigma)$ is the index of a set that contains all data that occur in σ.*

Definition 15 *A learner M is said to be* conservative *iff for all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ and members x of $\mathcal{D} \cup \{\sharp\}$, if $M(\sigma)$ is defined and is the index of a set that contains x then $M(\sigma \star x)$ is defined and equal to $M(\sigma)$.*

Definition 16 *Let a learner M be given.*

(1) M is said to be strong-monotonic *iff for all $L \in \mathfrak{L}$, texts t for L and $i,j \in \mathbb{N}$ with $i<j$, if $M(t[i])$ and $M(t[j])$ are both defined then $W_{M(t[i])} \subseteq W_{M(t[j])}$.*

(2) M is said to be monotonic *iff for all $L \in \mathfrak{L}$, texts t for L and $i,j \in \mathbb{N}$ with $i<j$, if $M(t[i])$ and $M(t[j])$ are both defined then $W_{M(t[i])} \cap L \subseteq W_{M(t[j])} \cap L$.*

Definition 17 *A learner M is said to be* set-driven *iff for all $\sigma,\tau \in (\mathcal{D} \cup \{\sharp\})^\star$, if $\mathrm{cnt}(\sigma)$ is equal to $\mathrm{cnt}(\tau)$ then $M(\sigma)=M(\tau)$. We write $M(D)$ for $M(\sigma)$, for any member σ of $(\mathcal{D} \cup \{\sharp\})^\star$ with $\mathrm{cnt}(\sigma)=D$.*

Property 18 *A set-driven learner M Ex-identifies $\mathfrak{L}$ iff there is a recursive function f mapping (canonical indices of) finite sets to r.e. indices such that for all $L \in \mathfrak{L}$, there exists a finite subset D of L such that for all finite subsets E of L, if $D \subseteq E$ then $f(E)=f(D)$ and $f(D)$ is an r.e. index of L.*

Monotonicity and strong monotonicity cannot be enforced by bounds on the data consumption complexity beyond the trivial bounds 0 for strong monotonicity and 1 for monotonicity (the trivial bounds are obtained from the fact

that mind change complexity 1 enforces strong monotonicity and 2 enforces monotonicity):

Example 19 *If* $\mathcal{L}$ *consists of* $2\mathbb{N}$ *and, for all* $x \in \mathbb{N}$*, the finite sets*

$$\{0, 2, \ldots, 2x\} \cup \{2x+1\} \text{ and } \{0, 2, \ldots, 2x\} \cup \{2x+1, 2x+2, 2x+3\},$$

then some learner whose DC complexity is 2 *Ex-identified* $\mathcal{L}$*, but no monotonic learner Ex-identifies* $\mathcal{L}$*.*

Consistency, strong-monotonicity, and frugality guarantee that DC complexity is defined:

Proposition 20 *If some consistent, set-driven, strong-monotonic and confident learner Ex-identifies* $\mathcal{L}$ *then some frugal learner Ex-identifies* $\mathcal{L}$*.*

PROOF. Let N be a consistent, strong-monotonic and confident learner that Ex-identifies $\mathcal{L}$. We define a memory function h and a learner M having h-memory as follows. Put $M(()) = N(())$. Let $\sigma \in \mathcal{D} \cup \{\sharp\})^\star$ and $x \in \mathcal{D} \setminus \mathrm{cnt}(\sigma)$ be given. The definition of $h(\sigma \star x)$ and $M(\sigma \star x)$ is by cases.

Case 1: $N(\tau)$ is defined for all initial segments τ of $h(\sigma) \star x$ and $N(h(\sigma) \star x)$ is equal to $N(h(\sigma))$. Then $h(\sigma \star x) = h(\sigma)$.

Case 2: $N(\tau)$ is defined for all initial segments τ of $h(\sigma) \star x$ but $N(h(\sigma) \star x)$ is distinct from $N(h(\sigma))$. Then $h(\sigma \star x) = h(\sigma) \cup \{x\}$ and $M(\sigma \star x) = N(h(\sigma \star x))$.

Case 3: Otherwise, both $h(\sigma \star x)$ and $M(\sigma \star x)$ are undefined.

Note that Case 3 happens if and only if $\sigma \star x$ is not consistent in $\mathcal{L}$. Let t be an infinite enumeration of members of $\mathcal{D} \cup \{\sharp\}$ such that for all $i \in \mathbb{N}$, $t[i]$ is consistent in $\mathcal{L}$ and assume that $N(t[i])$ is defined. Then $M(t[i])$ is defined for all $i \in \mathbb{N}$, and since the mind change complexity of N is defined, $\{h(t[i]) : i \in \mathbb{N}\}$ is finite. Hence the mind change complexity of M is also defined.

As N Ex-identifies $\mathcal{L}$, and using the facts that N is consistent, set-driven and strong-monotonic, it is easily verified that:

- M itself is consistent, set-driven and strong-monotonic;
- for all $L \in \mathcal{L}$, texts t for L and $i \in \mathbb{N}$, $M(t[i]) \subseteq L$.

We infer that M converges on any text for L, for any member L of $\mathcal{L}$.

If the strong-monotonicity requirement is lifted, then there is no longer any guarantee that the data consumption complexity is defined:

Proposition 21 *It is possible to choose* $\mathfrak{L}$ *such that:*

- *some learner whose MC complexity is equal to* 1 *Ex-identifies* $\mathfrak{L}$*;*
- *some confident, consistent, set-driven, monotonic learner Ex-identifies* $\mathfrak{L}$*;*
- *some confident, consistent, set-driven, conservative learner Ex-identifies* $\mathfrak{L}$*;*
- *no frugal learner Ex-identifies* $\mathfrak{L}$*.*

PROOF. Let $\mathfrak{L}$ consist of the set $2\mathbb{N}$ of even numbers and all sets of the form $D \cup \{2x+1\}$ where $D \subseteq \{2, 4, \ldots\}$ and $|D| = x$. A learner that waits until either 0 or a number of the form $2x+1$ together with x members of $\{2, 4, \ldots\}$ have appeared can obviously Ex-identify $\mathfrak{L}$ with no mind change.

Define a learner M as follows. Let $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ be given and choose the appropriate case.

Case 1: If all members of $\mathrm{cnt}(\sigma)$ are even then $M(\sigma) = \{0, 2, 4, \ldots\}$.

Case 2: If $\mathrm{cnt}(\sigma)$ contains a unique number of the form $2x+1$ and less than x even numbers then $M(\sigma) = \{2x+1, 0, 2, 4, \ldots\}$.

Case 3: If $\mathrm{cnt}(\sigma)$ contains a unique number of the form $2x+1$ and precisely x even numbers then $M(\sigma) = \mathrm{cnt}(\sigma)$.

Case 4: Otherwise $M(\sigma)$ is undefined.

Note that:

- for all texts t for $\{0, 2, 4, \ldots\}$ and $i \in \mathbb{N}$, $M(t[i]) = \{0, 2, 4, \ldots\}$;
- for all members L of $\mathfrak{L}$ of the form $L = \{2x+1, y_1, y_2, \ldots, y_x\}$ and for all texts t for L, M will output in response the longer and longer initial segments of t only the hypotheses $\{0, 2, 4, \ldots\}$, $\{2x+1, 0, 2, 4, \ldots\}$ and L (maybe not all of them), in that order, and $L \cap \{0, 2, 4, \ldots\} \subseteq L \cap \{2x+1, 0, 2, 4, \ldots\} \subseteq L$.

Hence M is monotonic, and it is easily verified that M is consistent and set-driven, has a defined mind change complexity, and Ex-identifies $\mathfrak{L}$.

Define a learner N as follows. Let $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ be given. The definition of $N(\sigma)$ is by cases.

Case 1: If all members of $\mathrm{cnt}(\sigma)$ are even then $N(\sigma) = \{0, 2, 4, \ldots\}$.

Case 2: If $\mathrm{cnt}(\sigma)$ contains a unique number of the form $2x+1$ and at most x even numbers then $N(\sigma) = \mathrm{cnt}(\sigma)$.

Case 3: Otherwise $N(\sigma)$ is undefined.

It is immediately verified that N is a consistent, set-driven and conservative learner that Ex-identifies $\mathfrak{L}$, and that the mind change complexity of N is defined and equal to $\omega + 1$.

Let P be a learner that has h-memory, for some memory function h whose ordinal complexity is defined. Let a sequence $(\sigma)_{n\in\mathbb{N}}$ of finite sequences be defined as follows. Put $\sigma_0 = ()$. Let a nonnull $n \in \mathbb{N}$ be given, and assume that σ_m has been defined for all $m < n$, but σ_n has not been defined yet. If there exists a sequence τ consisting of two members of $\{6n, 6n+2, 6n+4\}$ such that $h(\sigma_{n-1} \star \tau)$ is distinct from $h(\sigma_{n-1})$ then $\sigma_n = \sigma_{n-1} \star \tau$ for such a sequence τ. Otherwise $\sigma_m = \sigma_n$ for all $m \geq n$. Since the ordinal complexity of h is defined, there exists a least $p \in \mathbb{N}$ with $\sigma_{p-1} = \sigma_p$. Note that σ_p contains precisely $2p$ even numbers. Then $e_1 = \sigma_p \star (6p, 4p+1, \sharp, \sharp, \ldots)$ and $e_2 = \sigma_p \star (6p+2, 4p+1, \sharp, \sharp, \ldots)$ are two texts for two distinct members of $\mathfrak{L}$ but for all $m \geq 2p$, $P(e_1[m]) = P(e_2[m])$, hence P fails to Ex-identify $\mathfrak{L}$.

5 Relationships with some learning criteria

Frugality turns out to be related to learnability from good examples. There are two main notions of learnability from good examples [5,9,15]:

Definition 22 *$\mathfrak{L}$ is said to be* class-comprisingly learnable from good examples *iff there is a numbering $(H_i)_{i\in\mathbb{N}}$ of uniformly recursively enumerable sets containing all members of $\mathfrak{L}$, a recursive function g mapping r.e. indices to canonical indices of finite sets and a set-driven learner M mapping canonical indices of finite sets to r.e. indices such that for all $e \in \mathbb{N}$:*

- *if $H_e \in \mathfrak{L}$ then $D_{g(e)} \subseteq H_e$;*
- *if $H_e \in \mathfrak{L}$, $D_{g(e)} \subseteq E \subseteq H_e$ and E is finite then $H_{M(E)} = H_e$.*

Moreover, if $\mathfrak{L} = \{H_0, H_1, \ldots\}$ then $\mathfrak{L}$ is said to be class-preservingly learnable from good examples.

Example 23 [9, Theorem 1(b)] *It is possible to choose $\mathfrak{L}$ such that $\mathfrak{L}$ is confidently learnable, but not learnable from good examples. An example of such a class consists of the following subsets of $\mathbb{N} \times \mathbb{N}$: $\{(i,0),(i,1),\ldots\}$ for all i and, for a given recursive repetition-free enumeration $j_0, j_1, \ldots$ of the halting-problem, each set $\{(j_i,0),(j_i,1),\ldots,(j_i,k)\}$ with $k \leq i$.*

So confidently learnable classes may fail to be learnable from good examples, but this result does not carry over to frugally learnable classes:

Proposition 24 *Suppose that some frugal learner Ex-identifies $\mathfrak{L}$. Then $\mathfrak{L}$ is class-comprisingly learnable from good examples. More precisely, there is a*

superclass of $\mathfrak{L}$ which is class-preservingly learnable from good examples with respect to a repetition-free enumeration of this class.

PROOF. Let a memory function h and a frugal learner M be such that M has h-memory and M Ex-identifies $\mathfrak{L}$. First one introduces some useful concepts. Let t be any recursive sequence over $\mathbb{N} \cup \{\sharp\}$ that contains infinitely many occurrences of each member of $\mathbb{N} \cup \{\sharp\}$. Given $F \subseteq \mathbb{N}$, let t_F be such that for all $i \in \mathbb{N}$, $t_F(i) = t(i)$ if $t(i) \in F$ and $t_F(i) = \sharp$ otherwise. Given $F \subseteq \mathbb{N}$, define σ_F to be the shortest initial segment of t_F such that $h(\sigma_F)$ is defined and for all initial segments τ of t_F that extend $t_F(i)$, $h(\tau)$ is defined and equal to $h(\sigma_F)$; if such an initial segment does not exist then σ_F is undefined. Intuitively, this means that whenever σ_F exists, M does not remember any member of t_F that occurs in t_F beyond σ_F, and therefore there is no $x \in F \cup \{\#\}$ such that M remembers the last occurrence of x in $\sigma_F \star x$. Note that the partial function that maps a finite subset D of $\mathbb{N}$ such that σ_D exists, to σ_D itself, is partial recursive.

We now define the enumeration of the superclass which is class-preservingly learnable from good examples. We let $(\eta_i)_{i \in \mathbb{N}}$ be a recursive repetition-free enumeration of all defined sequences σ_D for finite sets D; Since M is a frugal learner, σ_D is defined at least for all those D which are consistent with $\mathfrak{L}$. Having this, we define H_i to be the union of $\mathrm{cnt}(\eta_i)$ with

$$\{x \in W_{M(\eta_i)} - \mathrm{cnt}(\eta_i) : \sigma_D \text{ exists and } \sigma_D = \eta_i \text{ where } D = \mathrm{cnt}(\eta_i) \cup \{x\}\}$$

and we verify the following: (a) $\mathfrak{L} \subseteq \{H_i : i \in \mathbb{N}\}$; (b) the enumeration $(H_i)_{i \in \mathbb{N}}$ is repetition-free; (c) $\{H_i : i \in \mathbb{N}\}$ is class-preservingly learnable from good examples via the function G mapping the set i to the set $G(i) = \mathrm{cnt}(\eta_i)$ of good examples of H_i and the learner N mapping each finite set E to the index i of the string η_i equal to σ_E whenever σ_E exists; N is undefined if σ_E does not exist.

For (a), consider any $L \in \mathfrak{L}$ and the sequence σ_L. Note that $\sigma_L = \sigma_D$ for the finite set $D = \mathrm{cnt}(\sigma_L)$. Therefore there is an index i with $\sigma_L = \eta_i$. Since M converges on t_L to an index of L it holds that $W_{M(\sigma_L)} = L$. Every $x \in L$ satisfies that $E = D \cup \{x\}$ is consistent with $\mathfrak{L}$ and thus there σ_E exists. Furthermore, $\sigma_E = \sigma_D$ since $D = \mathrm{cnt}(\sigma_L) \subseteq E \subseteq L$. Thus $H_i = L$.

For (b), consider any i, j with $H_i = H_j$. Then η_i and η_j are both prefixes of t_{H_i}, say $\eta_i \preceq \eta_j$. It follows that $h(\eta_j) = h(\eta_i)$ since otherwise there is a $x \in \mathrm{cnt}(\eta_j) - \mathrm{cnt}(\eta_i)$ which would then be in $H_j - H_i$ in contradiction to $H_i = H_j$. Since the enumeration $(\eta_k)_{k \in \mathbb{N}}$ is repetition-free, $i = j$ and the enumeration $(H_k)_{k \in \mathbb{N}}$ is also repetition-free.

For (c), consider any $i \in \mathbb{N}$ and H_i. It follows directly from the definition of H_i that $G(i) \subseteq H_i$. Furthermore, every finite $D \subseteq H_i$ with $G(i) \subseteq D \subseteq H_i$ satisfies that η_i is a prefix of t_D and that $h(\eta_i x) = h(\eta_i)$ for all $x \in D \cup \{\sharp\}$. Thus $\sigma_D = \eta_i$ and $N(D)$ is defined and equal to i. So N is a learner which infers $\{H_i : i \in \mathbb{N}\}$ class-comprisingly from good examples with respect to the repetition-free enumeration $(H_i)_{i \in \mathbb{N}}$.

Note that one can replace every iterative learner for $\mathfrak{L}$ by an equivalent learner M that converges on every text for a finite set. Then M satisfies the property that σ_D exists for every σ consistent in $\mathfrak{L}$. This property was the only essential condition used in the proof which follows from the definition of a frugal learner but not from the definition of an iterative learner. Thus one obtains:

Corollary 25 *If $\mathfrak{L}$ is iteratively learnable then $\mathfrak{L}$ is class-comprisingly learnable from good examples.*

By [9, Theorem 2] that there are classes which are iteratively learnable as they consist of finite sets only, but which are not class-preservingly learnable from good examples. For an example of such a class, consider the class consisting of all $\{x\}$ with $x \in K'$ and $\{x, y\}$ with $x < y$; K' is the halting problem for computations relative to K. Still, this class does not have a class-preserving iterative learner. The next example shows that there is also a class-comprisingly frugally learnable class which is not class-preservingly learnable from good examples.

Proposition 26 *There is a class which can be learned with data consumption complexity 3, but which is not class-preservingly learnable from good examples.*

PROOF. If a class $\mathfrak{L}$ can be learned class-preservingly from good examples then there is a total function φ_i such that $W_{\varphi_i(0)}, W_{\varphi_i(1)}, \ldots$ is an enumeration of $\mathfrak{L}$ (not containing any nonmember of $\mathfrak{L}$) and there is a recursive function φ_j mapping every e to the finite subset $D_{\varphi_j(e)}$ of $W_{\varphi_i(e)}$ such that no $L \in \mathfrak{L}$ satisfies $D_{\varphi_j(e)} \subseteq L \subset W_{\varphi_i(e)}$.

The goal is to construct a class $\mathfrak{L}$ which violates this constraint. The class $\mathfrak{L}$ is given by an indexed family which consists of $\emptyset$ and the following sets:

- $L_{i,j} = \{(i,j,0), (i,j,1), \ldots\}$ for all $i, j \in \mathbb{N}$;
- $F_{i,j} = D_{\varphi_j(e_{i,j})} \cup \{(i,j,s_{i,j}+1)\}$ whenever there is a least pair $(e_{i,j}, s_{i,j}) \in \mathbb{N}^2$ such that $\varphi_j(e_{i,j})$ is defined,

  $$\emptyset \subset D_{\varphi_j(e_{i,j})} \subset \{(i,j,0), (i,j,1), \ldots, (i,j,s_{i,j})\} \cap W_{\varphi_i(e_{i,j})},$$

  $e_{i,j} \leq s_{i,j}$, and all these facts can be verified in time $s_{i,j}$. If such a pair $(e_{i,j}, s_{i,j})$ does not exist then no set $F_{i,j}$ is added to $\mathfrak{L}$.

Note that whenever $W_{\varphi_i(0)}, W_{\varphi_i(1)}, \ldots$ is a class-preserving enumeration of $\mathfrak{L}$ and φ_j maps every $e \in \mathbb{N}$ to the canonical index of a strict finite subset of $W_{\varphi_i(e)}$, then $e_{i,j}$ must exist and $W_{\varphi_i(e_{i,j})} = L_{i,j}$. Indeed, $W_{\varphi_i(e_{i,j})}$, belonging to $\mathfrak{L}$, is equal either to $L_{i,j}$ or to $F_{i,j}$. But if $W_{\varphi_i(e_{i,j})}$ were equal to $F_{i,j}$ then one would derive

$$W_{\varphi_i(e_{i,j})} = D_{\varphi_j(e_{i,j})} \cup \{(i,j,s_{i,j}+1\} \subset W_{\varphi_i(e_{i,j})} \cup \{(i,j,s_{i,j}+1)\} = W_{\varphi_i(e_{i,j})},$$

which is impossible. Hence $W_{\varphi_i(e_{i,j})} = L_{i,j}$. Furthermore,

$$D_{\varphi_j(e_{i,j})} \subseteq F_{i,j} \subset L_{i,j}.$$

Then $\mathfrak{L}$ has no class-preserving learner from good examples which uses the numbering $(W_{\varphi_i(k)})_{k\in\mathbb{N}}$ and the function φ_j to compute the set of good examples from the index since such a learner would have to map $F_{i,j}$ to $F_{i,j}$ and to $L_{i,j}$ at the same time. So $\mathfrak{L}$ is not class-preservingly learnable from good examples.

A class-preserving learner for $\mathfrak{L}$ has data consumption complexity 3. It is initialized with $\emptyset$ and it memorizes a new datum (i,j,k) iff:

- all previously memorized data (i',j',k') satisfy $i' = i$, $j' = j$ and $k' < k$;
- either (i,j,k) is the first datum to be memorized,
  or the set $F_{i,j}$ exists and $k = s_{i,j} + 1$,
  or the set $F_{i,j}$ exists and the previous hypothesis is $F_{i,j}$ and $k > s_{i,j} + 1$.

Although it is impossible to check whether $F_{i,j}$ exists one can check whether $F_{i,j}$ exists and $s_{i,j} < k$.

When the leaner memorizes a new datum (i,j,k), either $F_{i,j}$ exists and $s_{i,j}+1$ is equal to k, in which case the learner outputs $F_{i,j}$, or the learner outputs $L_{i,j}$.

It is easy to see that the learner is correct and memorizes a maximum amount of data when the hypotheses $L_{i,j}, F_{i,j}, L_{i,j}$ are output; a new datum is then memorized for each hypothesis.

6 Generalized data consumption complexity

Frugality is a very restrictive notion. It is often natural to allow learners to discard not only newly received data, but also data that have been memorized at some stage of the learning process, but become redundant or useless in the face of a new datum. For instance, when learning the class of all nonempty

final segments of $\mathbb{N}$, an element smaller than the smallest element previously received can replace the latter, rather than being memorized together with the latter. This suggests generalizing the notions of memory function and frugal use of data as follows.

Definition 27 *A* generalized memory function *is defined as a partial recursive function h from $(\mathcal{D} \cup \{\sharp\})^\star$ into $2^{\mathcal{D}}$ such that $h(()) = \emptyset$ and for all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$:*

- *for all $\tau \in (\mathcal{D} \cup \{\sharp\})^\star$ with $\sigma \subseteq \tau$, if $h(\sigma)$ is undefined then $h(\tau)$ is undefined;*
- *for all $x \in \mathcal{D} \cup \{\sharp\}$, $h(\sigma \star x)$ is undefined or included in $h(\sigma) \cup \{x\}$.*

Let Z be the set of all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ such that $h(\sigma)$ is defined. Given $\sigma, \tau \in Z$ let $R(\sigma, \tau)$ hold iff $\tau \subseteq \sigma$ and the cardinality of $h(\sigma)$ is greater than the cardinality of $h(\tau)$. The ordinal complexity of h *is undefined if (Z, R) is not well founded and is equal to the length of (Z, R) otherwise.*

Definition 28 *Let a learner M be given.*

*Given a generalized memory function h, we say that M has h-*memory *iff M can be represented as $g \circ h$ for some partial recursive function g.*

The generalized data consumption complexity of M *is defined and equal to ordinal α iff α is the least ordinal such that M has h-memory, for some generalized memory function h whose ordinal complexity is equal to α.*

We will usually write 'GDC complexity' for 'generalized data consumption complexity.'

Being Ex-identifiable by a learner who GDC is defined is incomparable to being Ex-identifiable by an iterative learner. The class of finite sets clearly witnesses one direction of this claim:

Proposition 29 *$\mathcal{L}$ can be chosen such that some iterative learner Ex-identifies $\mathcal{L}$, but no learner whose GDC is defined Ex-identifies $\mathcal{L}$.*

For the other direction:

Proposition 30 *$\mathcal{L}$ can be chosen such that some learner whose GDC is defined Ex-identifies $\mathcal{L}$, but no iterative learner Ex-identifies $\mathcal{L}$.*

PROOF. Put $\mathcal{L} = \{\mathbb{N} \setminus \{0\}\} \cup \{\{0, \ldots, n\} : n \in \mathbb{N}\}$. It is immediately verified that no iterative learner Ex-identifies $\mathcal{L}$. A learner that only memorizes the largest number seen so far, together with 0 if 0 appears, Ex-identifies $\mathcal{L}$ and has a GDC equal to 2.

Trivially:

Property 31 *If the DC complexity of a learner M is defined and equal to ordinal α then the GDC of M is defined and as most equal to α.*

Generalized data consumption complexity and data consumption complexity are sometimes equivalent:

Proposition 32 *For all recursive ordinals α, $\mathfrak{L}$ can be chosen such that:*

- *some learner whose DC complexity is equal to α Ex-identifies $\mathfrak{L}$;*
- *no learner whose GDC complexity is smaller than α Ex-identifies $\mathfrak{L}$.*

PROOF. Let a recursive ordinal α and a recursive tree T of height α be given. Let $\mathfrak{L}$ be (identified with) the set of the graphs of the recursive functions from the branches of T into $\{0,1\}$. Let X be the set of all $\sigma \in (\mathcal{D} \cup \{\sharp\})^\star$ that are consistent in $\mathfrak{L}$. Clearly, the function that maps $\sigma \in X$ to $\mathrm{cnt}(\sigma)$ is a memory function of ordinal complexity α. Hence $\mathfrak{L}$ is Ex-identifiable by a learner whose DC complexity is equal to α. Moreover, it is easily verified that for all learners M, generalized memory functions h and branches B of T, if M learns $\mathfrak{L}$, M has h-memory, and $(x_1, \ldots, x_n)$ is the (unique) enumeration of all initial members of B with $x_1 \subset \ldots \subset x_n$, then for all $y_1, \ldots, y_n \in \{0,1\}$ and $i \leq n$ $h((x_1,y_1),\ldots,(x_i,y_i)) = \{(x_1,y_1),\ldots,(x_i,y_i)\}$. We conclude that M has GDC complexity at least equal to α.

Still the generalized notion of data consumption complexity is more powerful, as witnessed by the class of finite initial segments of all members of $\mathbb{N}$:

Proposition 33 *$\mathfrak{L}$ can be chosen such that some learner whose GDC is equal to 1 Ex-identifies $\mathfrak{L}$, but no frugal learner Ex-identifies $\mathfrak{L}$.*

Though memory functions and generalized memory functions return sets of data, making a frugal use of data and being set-driven can turn out to be incompatible requirements. The next proposition states an even stronger fact.

Proposition 34 *$\mathfrak{L}$ can be chosen such that:*

- *some frugal learner Ex-identifies $\mathfrak{L}$;*
- *no set-driven learner whose GDC complexity is defined Ex-identifies $\mathfrak{L}$.*

PROOF. Define $\mathfrak{L}$ as the class consisting of the set of even numbers together with all sets of the form $\{2n+1\} \cup L$ where L is a set of even numbers such that for all $m \in \mathbb{N}$, $2m \in L$ iff $2m + 2n \in L$. Clearly, $\mathfrak{L}$ is Ex-identifiable by a learner that has a memory function of ordinal complexity $\omega + 1$, discarding

all data not of the form $2n+1$ until (possibly) a datum of the form $2n+1$ appears, at which point n extra data need to be memorized. Let a learner M and a generalized memory function h be such that the ordinal complexity of h is defined, M has h-memory, and M Ex-identifies $\mathcal{L}$. Choose $\sigma \in (2\mathbb{N})^\star$ such that for all $\tau \in (2\mathbb{N})^\star$ and $x \in 2\mathbb{N}$, if τ extends σ then the cardinality of $h(\tau \star x)$ is at most equal to the cardinality of $h(\sigma)$. Let k denote the cardinality of $h(\sigma)$, and let $n \in \mathbb{N}$ be greater than $2k+2$. Choose $L_1 \in \mathcal{L}$ such that $2n+1 \in L$, $h(\sigma) \subseteq L_1$, and $\{0, 2, \ldots, 2(n-1)\} \cap L_1$ has cardinality $k+1$. Let $\tau \in (2\mathbb{N})^\star$ be such that $\sigma \star (2n+1) \star \tau$ is a locking sequence for M and L_1. Note that the cardinality of $h(\sigma \star \tau)$ is at most equal to k. Let L_2 be the union of $h(\sigma \star \tau)$ with $\{2n+1\}$. Obviously, L_2 is a strict subset of L_1. Let $\eta \in L_2^\star$ be such that $M(\sigma \star \tau \star (2+1) \star \eta) = L_2$. Since $\sigma \star (2n+1) \star \tau$ is a locking sequence for M and L_1, $M(\sigma \star (2+1) \star \tau \star \eta) = L_2$. Hence M is not set-driven.

7 Conclusion

Traditional models of computation and learning have looked at intrinsic uncertainty (for example, conditions under which a class of languages is learnable with no less than a given number of mind changes), but do not conceive of 'self-imposed' uncertainty as a desirable feature. These models do not capture the behaviour of human beings in most real-life decisions: even when he knows that the degree of uncertainty will eventually decrease, a decision maker accepts the extra uncertainty of the moment and takes action. In this paper we have proposed a simple approach to formally justify why uncertainty might be desirable. The model is crude, in particular because the notion of data consumption complexity is very restrictive. The generalized notion of data consumption complexity is more powerful by allowing learners to forget data. We intend to apply it to the logical setting, whose expressive power can shed more insights on the issue of 'self-imposed' uncertainty.

Acknowledgements

The authors would like to thank Steffen Lange for very helpful comments on a first draft of this paper.

References

[1] Ambainis, A., Jain, S., Sharma, A.: *Ordinal Mind Change Complexity of Language Identification.* In Ben-David, S.: Proc. of the 3rd Europ. Conf. on

Comput. Learning Theory. Springer-Verlag, LNCS 1208 pp. 301-315 (1997)

[2] Dana Angluin. *Inductive inference of formal languages from positive data.* Information and Control, 45(2): 117–135 (1980)

[3] Case, J., Jain, S., Sharma, A.: *Vacillatory Learning of Nearly Minimal Size Grammars.* Journal of Computer and System Sciences 49: 189-207 (1994)

[4] Case, J., Jain, S., Lange, S., Zeugmann, T.: *Incremental Concept Learning for Bounded Data Mining.* Information and Computation 152(1): 74-110 (1999)

[5] Freivalds, R. Kinber, E. B., Wiehagen, R.: *On the Power of Inductive Inference from Good Examples.* Theoretical Computer Science 110(1): 131-144 (1993)

[6] Freivalds, R., Kinber, E. B., Smith, C. H.: *On the Impact of Forgetting on Learning Machines.* In Pitt, L.: Proc. of the 6th Annual Conf. of Comput. Learning Theory. ACM Press pp. 165-174 (1993)

[7] Freivalds, R., Smith, C. H.: *On the role of procrastination for machine learning.* Information and Computation 107(2): 237-271 (1993)

[8] Gold, E. M.: *Language identification in the limit.* Inform. and Control 10 (1967)

[9] Jain, S., Lange, S., Nessel, J.: *On the learnability of recursively enumerable languages from good examples.* Theoretical Computer Science 261(1): 3-29 (2001)

[10] S. Jain, D. N. Osherson. J. S. Royer and A. Sharma. *Systems that learn: An Introduction to Learning Theory, Second Edition.* The MIT Press (1999)

[11] Jain, S., Sharma, A.: *On monotonic strategies for learning r.e. languages.* In Jantke, K., Arikawa, S.: Proc. of the 5th Intern. Workshop on Alg. Learning Theory. Springer-Verlag, LNAI 872 pp. 349-364 (1994)

[12] Jain, S., Sharma, A.: *Mind change complexity of learning logic programs.* Theoretical Computer Science 284(1): 143-160 (2002)

[13] Kinber, E. B., Stephan, F.: *Language learning from texts: mind changes, limited memory and monotonicity.* Information and Computation 123(2): 224-241 (1995)

[14] Lange, S., Grieser, G.: *On the power of incremental learning.* Theoretical Computer Science 288(2): 277-307 (2002)

[15] Lange, S., Nessel, J., Wiehagen, R.: *Learning Recursive Languages from Good Examples.* Annals of Mathematics and Artificial Intelligence 23(1-2): 27-52 (1998)

[16] Lange, S., Zeugmann, T., Kapur, S: *Monotonic and Dual Monotonic Language Learning.* Theoretical Computer Science 155(2): 365-410 (1996)

[17] Martin, E., Sharma, A., Stephan, F.: *Logic, Learning, and Topology in a Common Framework.* In Cesa-Bianchi, N., Numao, M., Reischuk, R.: Proc. of the 13th Intern. Conf. on Alg. Learning Theory. Springer-Verlag, LNAI 2533 pp. 248-262 (2002)

[18] Martin, E., Sharma, A, Stephan, F.: *On Ordinal VC-Dimension and Some Notions of Complexity.* In Gavalda, R., Jantke, P., Takimoto, E.: Proc. of the 14th Intern. Conf. on Alg. Learning Theory. Springer-Verlag, LNAI 2842 pp. 54-68 (2003)

[19] Osherson, D., Stob, M., Weinstein, S. *Systems that learn.* The MIT Press (1986)

[20] Sharma, A., Stephan, F., Ventsov, Y.: *Generalized Notions of Mind Change Complexity.* In Freund, Y., Shapire, R.: Proc. of the 10th Annual Conf. of Comput. Learning Theory. ACM Press pp. 96-108 (1997)

[21] Zeugmann, T., Lange, S., Kapur, S.: *Characterizations of Monotonic and Dual Monotonic Language Learning.* Inform, and Comput. 120(2): 155-173 (1995)