

Learning in the Presence of Inaccurate Information

Mark Fulk *

Sanjay Jain †

Abstract

The present paper considers the effects of introducing inaccuracies in a learner's environment in Gold's learning model of identification in the limit. Three kinds of inaccuracies are considered: presence of spurious data is modeled as learning from a noisy environment, missing data is modeled as learning from incomplete environment, and the presence of a mixture of both spurious and missing data is modeled as learning from imperfect environment.

Two learning domains are considered, namely, identification of programs from graphs of computable functions and identification of grammars from positive data about recursively enumerable languages. Many hierarchies and tradeoffs resulting from the interplay between the number of errors allowed in the final hypotheses, the number of inaccuracies in the data, the types of inaccuracies, and the type of success criteria are derived.

An interesting result is that in the context of function learning, incomplete data is strictly worse for learning than noisy data.

1 Introduction

Consider the scenario in which a subject is attempting to learn its environment. At any given time, the subject receives a finite piece of data about its environment, and based on this finite information, conjectures an explanation about the environment. The subject is said to *learn* its environment just in case the explanations conjectured by the subject become fixed over time, and this fixed explanation is a correct representation of the subject's environment. Computational learning theory provides a framework for the study of the above scenario when the subject is an algorithmic device. The above model of learning is based on the work initiated by Gold [Gol67] and has been used in inductive inference of both functions and languages. We refer the reader to [AS83, BB75, CS83, OSW86, KW80] for background material in this field.

*Genesee Valley Software, 92 Monteroy Road, Brighton, NY 14618, USA

†Department of Information Systems and Computer Science, National University of Singapore, Singapore 0511, Republic of Singapore. Email: sanjay@iscs.nus.sg

In this paper we consider the effect of inaccuracies in the data presented to the learner. In the context of linguistic development, children likely receive ungrammatical sentences and may not receive some sentences. However, these inaccuracies do not seem to influence the outcome of linguistic development. Similarly, in the context of scientific discovery, the business of science progresses despite experimental errors and unfeasibility of performing certain experiments.

In this paper, both for identification of functions and languages, we consider the effects of having inaccuracies in the input data. We show a hierarchy based on the number of inaccuracies present in the input data. We also show tradeoff results based on the amount and type of inaccuracies present. Osherson, Stob and Weinstein [OSW86] and Schäfer-Richter [SR86] have also considered inaccurate information in the above model of learning.

We consider three kinds of inaccuracies that could creep into natural environments of learners.

- **Noisy data:** Ungrammatical intrusions into the language presented to a child is a very reasonable assumption about a child’s environment. Similarly, experimental error caused by a faulty equipment could result in spurious data that is not representative of the reality under investigation.
- **Incomplete data:** Natural linguistic environments may omit sentences from the ambient language, and it is possible that a child’s learning function can identify a natural language despite the systematic omission of sentences from its environment. Similarly, some experiments cannot be performed either due to technological limitations or due to ethical considerations.
- **Imperfect data:** Most natural linguistic environments are likely to be victims of both ungrammatical intrusions and omission of sentences from the ambient language. Such environments that contain a mixture of noisy and incomplete inaccuracies are referred to as environments with *imperfect* data. Similarly, in most experimental investigations, the inaccuracies are a mixture of both noisy and incomplete data.

The three kinds of inaccuracies discussed above yield three kind of information sequences—noisy, incomplete, and imperfect. However, a further distinction is made based on whether the number of inaccuracies in the input is finite or infinite. In this paper we only examine the case where inaccuracies are finite in number.

It should be noted that the inaccuracies discussed here model spurious data and unavailability of data; they don’t say anything about situations like “data is correct within 10% of actual value.”

2 Preliminaries

2.1 Notation

Recursion-theoretic concepts not explained below are treated in [Rog67]. N denotes the set of natural numbers, $\{0, 1, 2, 3, \dots\}$, and N^+ denotes the set of positive integers, $\{1, 2, 3, \dots\}$. The symbols $\emptyset, \in, \subseteq, \subset, \supseteq, \supset$ respectively denote, emptyset, member of, subset, proper subset, superset and proper superset.

Unless otherwise specified, $e, i, j, k, l, m, n, r, s, t, w, x, y, z$, with or without decorations¹, range over N . $*$ is a non-member of N satisfying $(\forall n \in N)[n < * < \infty]$. a and b , with or without decorations, range over $N \cup \{*\}$.

A, S, P, X , with or without decorations, range over subsets of N . D ranges over finite subsets of N . $\text{card}(P)$ denotes the cardinality of P . So then, ' $\text{card}(P) \leq *$ ' means that $\text{card}(P)$ is finite. $\min(P)$ and $\max(P)$ respectively denote the minimum and maximum element in P . We take $\min(\emptyset)$ to be ∞ and $\max(\emptyset)$ to be 0.

Let $\lambda x, y. \langle x, y \rangle$ denote a fixed pairing function (a recursive, bijective mapping: $N \times N \rightarrow N$) [Rog67]. $\lambda x, y. \langle x, y \rangle$ and its inverses are useful to simulate the effect of having multiple argument functions. π_1 and π_2 are corresponding projection functions, i.e., $(\forall x, y)[\pi_1(\langle x, y \rangle) = x \wedge \pi_2(\langle x, y \rangle) = y]$. $\langle \cdot, \cdot \rangle$ can be extended to n -tuple $\langle \cdot, \dots, \cdot \rangle$ in a natural way.

η and ξ range over partial functions. For $a \in N \cup \{*\}$, $\eta_1 =^a \eta_2$ means that $\text{card}(\{x \mid \eta_1(x) \neq \eta_2(x)\}) \leq a$. In this case we also say that η_1 is an a -variant of η_2 . $\text{domain}(\eta)$ and $\text{range}(\eta)$ respectively denote the domain and range of partial function η . $\text{graph}(\eta)$ denotes the set $\{(x, \eta(x)) \mid x \in \text{domain}(\eta)\}$.

\mathcal{R} denotes the class of all *recursive* functions, i.e., total computable functions with arguments and values from N . f, h and F , with or without decorations, range over \mathcal{R} . \mathcal{S}, \mathcal{C} , with or without decorations, range over subsets of \mathcal{R} .

We fix φ to be an *acceptable programming system* [Rog58, Rog67, MY78] for the partial recursive functions: $N \rightarrow N$. By φ_i we denote the partial recursive function computed by φ -program i . In some contexts p ranges over programs. In other contexts, p ranges over total functions, with its range being construed as programs. $\text{MinProg}(f) = \min(\{j \mid \varphi_j = f\})$, the minimum program for f in the φ system. We let W_i denote $\text{domain}(\varphi_i)$. W_i is, then, the r.e. set/language ($\subseteq N$) accepted (or equivalently, generated) by the φ -program i . We let Φ be an arbitrary Blum complexity measure [Blu67] associated with acceptable programming system φ ; such measures exist for any acceptable programming system [Blu67]. Then, $W_{i,s}$ denotes the

¹Decorations are subscripts, superscripts and the like.

set $\{x \mid x < s \wedge \Phi_i(x) \leq s\}$.

L , with or without decorations, ranges over *recursively enumerable* (r.e.) subsets of N , which subsets are usually construed as codings of formal languages. \mathcal{E} denotes the class of all recursively enumerable languages $\subseteq N$. We let \mathcal{L} , with or without decorations, range over subsets of \mathcal{E} . $L_1 \Delta L_2$ denotes $(L_1 - L_2) \cup (L_2 - L_1)$, the symmetric difference of L_1 and L_2 . For $a \in (N \cup \{*\})$, $L_1 =^a L_2$ means that $\text{card}(L_1 \Delta L_2) \leq a$. In this case we also say that L_1 is an a -variant of L_2 . Let $\mathbf{FIN} = \{L \mid \text{card}(L) < \infty\}$.

The quantifiers ‘ \forall^∞ ’ and ‘ \exists^∞ ’ mean ‘for all but finitely many’ and ‘there exist infinitely many,’ respectively. The quantifier ‘ $\exists!$ ’ means ‘there exists a unique.’

We concern ourselves with formally investigating learning of two kinds of objects: computable functions and recursively enumerable languages.

2.2 Learning Machines

An *information sequence* for functions is a mapping from N (or an initial segment of N) to $N \times N \cup \{\#\}$. An *information sequence* for languages is a mapping from N (or an initial segment of N) to $N \cup \{\#\}$. Intuitively, $\#$ denotes the pauses in the presentation of data. We often drop “for functions” and “for languages” from “information sequences for functions” and “information sequences for languages” respectively; context determines which information sequence we are dealing with. We often refer to infinite information sequences for languages as texts. SEG denotes the set of all finite function information sequences, and SEQ denotes the set of all finite language information sequences.

We let G and T , with or without decorations, range over infinite information sequences (i.e. information sequences with domain N). We let σ and τ , with or without decorations, range over finite information sequences (i.e. information sequences with domain an initial segment of N). We let $\text{content}(G)$ denote the set $\text{range}(G) - \{\#\}$. Similarly, $\text{content}(\sigma)$ denotes the set $\text{range}(\sigma) - \{\#\}$.

Suppose G is an infinite information sequence. Then $G[n]$ denotes the finite initial segment of G with length n . For finite information sequence σ of length $\geq n$, $\sigma[n]$, denotes the initial segment of σ with length n . $|\sigma|$ denotes the length of the finite information sequence σ . $\sigma \subseteq G$ means that σ is an initial sequence of G . Similarly, $\sigma \subseteq \tau$, means that σ is an initial sequence of τ . We often identify a total function f , with its canonical information sequence G , defined as

$$G(k) = (k, f(k))$$

Thus $f[n]$, is same as the finite information sequence, σ defined as follows:

$$\sigma(k) = \begin{cases} (k, f(k)), & \text{if } k < n; \\ \uparrow, & \text{otherwise.} \end{cases}$$

For a finite information sequence σ for languages, $\sigma \diamond y$ denotes the *concatenation* of y at the end of the sequence σ . Formally, $\sigma' = \sigma \diamond y$ is defined as follows:

$$\sigma'(k) = \begin{cases} \sigma(k), & \text{if } k < |\sigma|; \\ y, & \text{if } k = |\sigma|; \\ \uparrow, & \text{otherwise.} \end{cases}$$

Similarly, for a finite information sequence σ for functions, $\sigma \diamond (y, z)$ denotes the concatenation of (y, z) at the end of the sequence σ .

Definition 1 A *learning machine* is an algorithmic mapping from finite information sequences to N .

We let \mathbf{M} , with or without decorations, range over learning machines.

A machine, \mathbf{M} , converges on an information sequence G (denoted by $\mathbf{M}(G)\downarrow$) iff there exists an i , such that $(\forall n)[\mathbf{M}(G[n]) = i]$. Otherwise we say that \mathbf{M} diverges on G (denoted by $\mathbf{M}(G)\uparrow$). If \mathbf{M} converges on G , then we define $\mathbf{M}(G)$ as the unique i such that $(\forall n)[\mathbf{M}(G[n]) = i]$. Otherwise $\mathbf{M}(G)$ is undefined.

Definition 2 A machine \mathbf{M} is *rearrangement independent* if $(\forall \sigma, \tau \mid \text{content}(\sigma) = \text{content}(\tau) \wedge |\sigma| = |\tau|)[\mathbf{M}(\sigma) = \mathbf{M}(\tau)]$.

Thus the output of a rearrangement independent machine depends only on the content and length of its input. For all the explanatory identification criteria considered in this paper we can assume without loss of generality that the machine is rearrangement independent (see [Ful90, Ful85]). In some of our proofs, when we assume that the machine is rearrangement independent, we often consider the learning machine as taking two arguments, a set representing the content of its input and a number representing the length of its input.

2.3 Fundamental Function Identification Paradigms

We now introduce two different criteria for a learning machine to successfully infer a function (from perfect input data).

2.3.1 Explanatory Function Identification

Definition 3 [Gol67, BB75, CS83] Let $a \in N \cup \{*\}$.

- (a) $\mathbf{M} \mathbf{Ex}^a$ -identifies f (written: $f \in \mathbf{Ex}^a(\mathbf{M})$) $\iff (\exists i \mid \varphi_i =^a f)[\mathbf{M}(f)\downarrow = i]$.
- (b) $\mathbf{Ex}^a = \{\mathcal{S} \mid (\exists \mathbf{M})[\mathcal{S} \subseteq \mathbf{Ex}^a(\mathbf{M})]\}$.

Case and Smith [CS83] motivate anomalies (or, mistakes) in the final programs in Definition 3 from the fact that physicists sometimes do employ explanations with anomalies. The $a = *$ case was introduced by L. Blum and M. Blum [BB75] and the other $a > 0$ cases were first considered by Case and Smith [CS83].

2.3.2 Behaviorally Correct Function Identification

Case and Smith [CS83] introduced another infinite hierarchy of identification criteria which we describe below. “**Bc**” stands for *behaviorally correct*. Barzdin [Bar74] essentially introduced \mathbf{Bc}^0 .

Definition 4 [CS83] Let $a \in N \cup \{*\}$.

- (i) $\mathbf{M} \mathbf{Bc}^a$ -identifies f (written: $f \in \mathbf{Bc}^a(\mathbf{M})$) $\iff (\forall n)[\varphi_{\mathbf{M}(f[n])} =^a f]$.
- (ii) $\mathbf{Bc}^a = \{\mathcal{S} \mid (\exists \mathbf{M})[\mathcal{S} \subseteq \mathbf{Bc}^a(\mathbf{M})]\}$.

We usually write \mathbf{Ex} for \mathbf{Ex}^0 and \mathbf{Bc} for \mathbf{Bc}^0 . Theorem 5 below describes some of the basic results about the two kinds of function identification criteria described above.

Theorem 5 For all $a \in N$,

- (a) $\mathbf{Ex}^a \subset \mathbf{Ex}^{a+1}$.
- (b) $\bigcup_{a \in N} \mathbf{Ex}^a \subset \mathbf{Ex}^*$.
- (c) $\mathbf{Ex}^* \subset \mathbf{Bc}$.
- (d) $\mathbf{Bc}^a \subset \mathbf{Bc}^{a+1}$.
- (e) $\bigcup_{a \in N} \mathbf{Bc}^a \subset \mathbf{Bc}^*$.
- (f) $\mathcal{R} \in \mathbf{Bc}^*$.

Parts (a), (b), (d), and (e) are due to Case and Smith [CS83]. John Steel first observed that $\mathbf{Ex}^* \subseteq \mathbf{Bc}$ and the diagonalization in part (c) is due to Harrington and Case [CS83]. Part (f) is due to Harrington [CS83]. Blum and Blum [BB75] first showed that $\mathbf{Ex} \subset \mathbf{Ex}^*$. Barzdin [Bar74] independently showed $\mathbf{Ex} \subset \mathbf{Bc}$.

Let $\mathbf{AEZ} = \{f \in \mathcal{R} \mid \text{card}(\{x \mid f(x) \neq 0\}) < \infty\}$. Let $\mathbf{SELFREF} = \{f \in \mathcal{R} \mid \varphi_{f(0)} = f \wedge \text{card}(\{x \mid f(x) \neq 0\}) = \infty\}$. We will be using the above classes and their variants in several

of our proofs. The following Theorem from [BB75] shows that **Ex** is not closed under union as witnessed by **AEZ** and **SELFREF**.

Theorem 6 [BB75] $\mathbf{AEZ} \in \mathbf{Ex}$, $\mathbf{SELFREF} \in \mathbf{Ex}$, but $\mathbf{AEZ} \cup \mathbf{SELFREF} \notin \mathbf{Ex}^* \cup \bigcup_i \mathbf{Bc}^i$.

2.4 Fundamental Language Identification Paradigms

Definition 7 A text T is for a language L iff $\text{content}(T) = L$.

Intuitively, a text for a language is an enumeration or sequential presentation of all the objects in the language with the $\#$'s representing pauses in the listing or presentation of such objects. For example, the only text for the empty language is just an infinite sequence of $\#$'s.

2.4.1 Explanatory Language Identification

We now introduce criteria for a learning machine to be successful on a language.

Definition 8 [Gol67, CL82, OW82a] Let $a \in N \cup \{*\}$.

- (a) \mathbf{M} **TxtEx** ^{a} -identifies L (written: $L \in \mathbf{TxtEx}^a(\mathbf{M})$) $\iff (\forall \text{ texts } T \text{ for } L)(\exists i \mid W_i =^a L)[\mathbf{M}(T) \downarrow = i]$.
- (b) $\mathbf{TxtEx}^a = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxtEx}^a(\mathbf{M})]\}$.

We usually write **TxtEx** for **TxtEx**⁰. The generalization of Gold's paradigm to the $a > 0$ case above was motivated by the observation that humans rarely learn a language perfectly. The $a > 0$ case in Definition 8 is due to Case and Lynes [CL82]. Osherson and Weinstein [OW82a], independently of Case and Lynes, introduced the $a = *$ case. The influence of Gold's paradigm to human language learning is discussed by Pinker [Pin79], Wexler and Culicover [WC80], Wexler [Wex82], and Osherson, Stob, and Weinstein [OSW82, OSW84, OSW86].

Definition 9 [Ful85, Ful90] σ is a **TxtEx**-stabilizing sequence for \mathbf{M} on $L \iff \text{content}(\sigma) \subseteq L$ and $(\forall \sigma' \mid \text{content}(\sigma') \subseteq L \wedge \sigma \subseteq \sigma')[\mathbf{M}(\sigma') = \mathbf{M}(\sigma)]$.

Definition 10 [BB75, OW82b] Let $a \in N \cup \{*\}$. σ is a **TxtEx** ^{a} -locking sequence for \mathbf{M} on $L \iff \sigma$ is a **TxtEx**-stabilizing sequence for \mathbf{M} on L and $W_{\mathbf{M}(\sigma)} =^a L$.

For rearrangement independent machines, we often refer to a stabilizing (locking) sequence σ by $\langle \text{content}(\sigma), |\sigma| \rangle$.

We now present a very important lemma in learning theory due to L. Blum and M. Blum [BB75].

Lemma 11 [BB75, OW82b] *If \mathbf{M} **TxtEx** ^{a} -identifies L , then there is a **TxtEx** ^{a} -locking sequence for \mathbf{M} on L .*

2.4.2 Vacillatory Language Identification

Case [Cas88], as a refinement of a result by Osherson and Weinstein [OW82a], considered the question whether humans converge to more than one distinct, but equivalent, correct grammar. He captured this notion through a new criterion of language learning, viz., **TxtFex-identification**—a more general criteria than Gold’s **TxtEx-identification**.

Before we describe **TxtFex-identification**, we first consider in Definition 12 just below what it means for a learning machine to converge on a text to a finite set of grammars.

Definition 12 [Cas88] Suppose \mathbf{M} is a learning machine and T is a text. Then, $\mathbf{M}(T)$ *finitely-converges* (written: $\mathbf{M}(T)\Downarrow$) $\iff \{\mathbf{M}(\sigma) \mid \sigma \subset T\}$ is finite. If $\mathbf{M}(T)\Downarrow$, then we say that $\mathbf{M}(T)\Downarrow = \{i \mid (\exists^\infty \sigma \subset T)[\mathbf{M}(\sigma) = i]\}$, otherwise we say that $\mathbf{M}(T)$ *finitely-diverges* (written: $\mathbf{M}(T)\Uparrow$).

Definition 13 [Cas88] Let $a \in N \cup \{*\}$ and $b \in N^+ \cup \{*\}$.

- (i) \mathbf{M} **TxtFex** $_b^a$ -*identifies* L (written: $L \in \mathbf{TxFex}_b^a(\mathbf{M})$) $\iff (\forall \text{ texts } T \text{ for } L)(\exists D \mid \text{card}(D) \leq b \wedge (\forall i \in D)[W_i =^a L])[\mathbf{M}(T)\Downarrow = D]$.
- (ii) $\mathbf{TxFex}_b^a = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxFex}_b^a(\mathbf{M})]\}$.

In **TxtFex** $_b^a$ -identification, the b is a “bound” on the number of final grammars and the a is a “bound” on the number of anomalies allowed in these final grammars. A “bound” of $*$ just means unbounded, but finite. We sometimes refer to **TxtFex** $_b^0$ as **TxtFex** $_b$. **TxtFex** $_*$ -identification was first studied by Osherson and Weinstein [OW82a].

2.4.3 Behaviorally Correct Language Identification

Definition 14 [CL82] Let $a \in N \cup \{*\}$.

- (i) \mathbf{M} **TxtBc** a -*identifies* L (written: $L \in \mathbf{TxBc}^a(\mathbf{M})$) $\iff (\forall \text{ texts } T \text{ for } L)(\forall n)[W_{\mathbf{M}(T[n])} =^a L]$.
- (ii) $\mathbf{TxBc}^a = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxBc}^a(\mathbf{M})]\}$.

We usually write **TxtBc** for **TxBc** 0 .

The following definition is an analogue of Definition 10 for **TxBc-identification**.

Definition 15 (Based on [BB75, CL82]) Let $a \in N \cup \{*\}$. σ is a **TxBc** a -*locking sequence* for \mathbf{M} on L $\iff \text{content}(\sigma) \subseteq L$ and $(\forall \sigma' \mid [\sigma \subseteq \sigma'] \wedge [\text{content}(\sigma') \subseteq L])[W_{\mathbf{M}(\sigma')} =^a L]$.

There is an analogue of Lemma 11 for **TxBc-identification** [CL82].

Lemma 16 (Based on [BB75, CL82]) *If \mathbf{M} \mathbf{TxtBc}^a -identifies L , then there is a \mathbf{TxtBc}^a -locking sequence for \mathbf{M} on L .*

Theorem 17 below states some of the basic results about the three kinds of language identification criteria just described.

Theorem 17 *For all $a, b \in \mathbb{N}$,*

- (a) $\mathbf{TxtEx}^{a+1} - \mathbf{TxtFex}_*^a \neq \emptyset$.
- (b) $\mathbf{TxtEx}^{2a+1} - \mathbf{TxtBc}^a \neq \emptyset$.
- (c) $\mathbf{TxtFex}_*^{2a} \subset \mathbf{TxtBc}^a$.
- (d) $\mathbf{TxtFex}_{b+1}^0 - \mathbf{TxtFex}_b^* \neq \emptyset$.
- (e) $\mathbf{TxtFex}_* - \bigcup_{b \in \mathbb{N}} \mathbf{TxtFex}_b^* \neq \emptyset$.
- (f) $\mathbf{TxtEx}^* - \bigcup_{a \in \mathbb{N}} \mathbf{TxtBc}^a \neq \emptyset$.
- (g) $\mathbf{TxtBc} - \mathbf{TxtFex}_*^* \neq \emptyset$.
- (h) $\bigcup_{a \in \mathbb{N}} \mathbf{TxtBc}^a \subset \mathbf{TxtBc}^*$.

Parts (a), (d), (e) and (g) are due to Case [Cas88]. Part (b) and (f) are due to Case and Lynes [CL82] and part (c) is based on the techniques of Case and Lynes [CL82]. Part (h) follows from part (e) in Theorem 5. Osherson and Weinstein [OW82a] have independently shown that $\mathbf{TxtEx} \subset \mathbf{TxtFex}_*$.

We will need the following Theorem from [CL82]

Theorem 18 [CL82] $\{N\} \cup \mathbf{FIN} \notin \mathbf{TxtBc}^*$.

Corollary 19 *For L , define $L' = \{\langle j, x \rangle \mid j \geq 1 \wedge x \in L\}$. Let $\mathcal{L}' = \{L' \mid L \in \mathbf{FIN} \cup \{N\}\}$. Then $\mathcal{L}' \notin \mathbf{TxtBc}^*$.*

Theorem 20 (based on the techniques used to prove Theorem 18) *Let $\mathcal{L} = \{L \mid \text{card}(N - L) = 1\} \cup \mathbf{FIN}$. Then $\mathcal{L} \notin \mathbf{TxtBc}^*$.*

3 Inaccurate Data

3.1 Information Sequences with Finite Number of Inaccuracies

Pursuant to the classification of inaccuracies, we define three kinds of inaccurate texts for languages. *-noisy, *-incomplete and *-imperfect texts and the corresponding identification criteria were first considered by [OSW86]. [SR86] considered a slight variant of noisy information sequences.

Definition 21 Let $a \in N \cup \{*\}$.

- (a) A text T is a -noisy for $L \iff L \subseteq \text{content}(T)$ and $\text{card}(\text{content}(T) - L) \leq a$.
- (b) A text T is a -incomplete for $L \iff \text{content}(T) \subseteq L$ and $\text{card}(L - \text{content}(T)) \leq a$.
- (c) A text T is a -imperfect for $L \iff \text{card}(L \Delta \text{content}(T)) \leq a$.

An a -noisy text T for a language L can be pictured as a text for L into which elements from a finite set of cardinality at most a have been inserted. Note that any single such intrusion may occur arbitrarily often in T . An a -incomplete text T for L can be pictured as a text for L from which all occurrences of a given finite set of cardinality at most a have been removed. An a -imperfect text T for L could be viewed as a text for L into which elements from a set S_1 have been inserted and elements from a set S_2 have been deleted, where $\text{card}(S_1) + \text{card}(S_2) \leq a$.

Note that in the above three definitions, $a = *$ case implies that the number of inaccuracies is any finite number. The other $a \in N$ cases model situations when a scientist may be aware, a priori, of an upper bound on the number of inaccuracies infesting its environment; possible sources of such information could be previous experience and nature of instruments used.

We now turn our attention to defining inaccurate information sequences for functions.

Definition 22 Let $a \in N \cup \{*\}$.

- (a) An information sequence G is a -noisy for $f \iff \text{graph}(f) \subseteq \text{content}(G)$ and $\text{card}(\text{content}(G) - \text{graph}(f)) \leq a$.
- (b) An information sequence G is a -incomplete for $f \iff \text{content}(G) \subseteq \text{graph}(f)$ and $\text{card}(\text{graph}(f) - \text{content}(G)) \leq a$.
- (c) An information sequence G is a -imperfect for $f \iff \text{card}(\text{graph}(f) \Delta \text{content}(G)) \leq a$.

Note that in the case of noisy information sequences for functions, two incorrect values for $f(n)$ count as two distinct noise points, i.e., if the correct value of $f(n) = x$ and both (n, y) and (n, z) , where x, y and z are distinct, are present in an inaccurate information sequence for f , then the data points (n, y) and (n, z) contribute towards two distinct noise points. Also, if the actual value of $f(n) = y$, but (n, y) doesn't appear in an information sequence and instead (n, z) , $y \neq z$, appears, then these contribute two to the imperfection count.

We now introduce the learning criteria when there may be inaccuracies in the input data.

Definition 23 Let $a, b \in N \cup \{*\}$.

- (a.1) $\mathbf{M} \ \mathbf{N}^a \mathbf{TxtEx}^b$ -identifies L (written: $L \in \mathbf{N}^a \mathbf{TxtEx}^b(\mathbf{M})$) \iff
 $(\forall a\text{-noisy texts } T \text{ for } L)[\mathbf{M}(T) \downarrow \wedge W_{\mathbf{M}(T)} =^b L]$.
- (a.2) $\mathbf{N}^a \mathbf{TxtEx}^b = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{N}^a \mathbf{TxtEx}^b(\mathbf{M})]\}$.
- (b.1) $\mathbf{M} \ \mathbf{In}^a \mathbf{TxtEx}^b$ -identifies L (written: $L \in \mathbf{In}^a \mathbf{TxtEx}^b(\mathbf{M})$) \iff

($\forall a$ -incomplete texts T for L) $[\mathbf{M}(T)\downarrow \wedge W_{\mathbf{M}(T)} =^b L]$.

(b.2) $\mathbf{In}^a \mathbf{TxtEx}^b = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{In}^a \mathbf{TxtEx}^b(\mathbf{M})]\}$.

(c.1) $\mathbf{M} \mathbf{Im}^a \mathbf{TxtEx}^b$ -identifies L (written: $L \in \mathbf{Im}^a \mathbf{TxtEx}^b(\mathbf{M})$) \iff

($\forall a$ -imperfect texts T for L) $[\mathbf{M}(T)\downarrow \wedge W_{\mathbf{M}(T)} =^b L]$.

(c.2) $\mathbf{Im}^a \mathbf{TxtEx}^b = \{\mathcal{L} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{Im}^a \mathbf{TxtEx}^b(\mathbf{M})]\}$.

Definition 24 Let $a, b \in N \cup \{*\}$.

(a.1) $\mathbf{M} \mathbf{N}^a \mathbf{Ex}^b$ -identifies f (written: $f \in \mathbf{N}^a \mathbf{Ex}^b(\mathbf{M})$) \iff

($\forall a$ -noisy information sequences G for f) $[\mathbf{M}(G)\downarrow \wedge \varphi_{\mathbf{M}(G)} =^b f]$.

(a.2) $\mathbf{N}^a \mathbf{Ex}^b = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{N}^a \mathbf{Ex}^b(\mathbf{M})]\}$.

(b.1) $\mathbf{M} \mathbf{In}^a \mathbf{Ex}^b$ -identifies f (written: $f \in \mathbf{In}^a \mathbf{Ex}^b(\mathbf{M})$) \iff

($\forall a$ -incomplete information sequences G for f) $[\mathbf{M}(G)\downarrow \wedge \varphi_{\mathbf{M}(G)} =^b f]$.

(b.2) $\mathbf{In}^a \mathbf{Ex}^b = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{In}^a \mathbf{Ex}^b(\mathbf{M})]\}$.

(c.1) $\mathbf{M} \mathbf{Im}^a \mathbf{Ex}^b$ -identifies f (written: $f \in \mathbf{Im}^a \mathbf{Ex}^b(\mathbf{M})$) \iff

($\forall a$ -imperfect information sequences G for f) $[\mathbf{M}(G)\downarrow \wedge \varphi_{\mathbf{M}(G)} =^b f]$.

(c.2) $\mathbf{Im}^a \mathbf{Ex}^b = \{\mathcal{C} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{Im}^a \mathbf{Ex}^b(\mathbf{M})]\}$.

Similarly, we can define the language identification paradigms: $\mathbf{N}^a \mathbf{TxtFex}_c^b$, $\mathbf{In}^a \mathbf{TxtFex}_c^b$, $\mathbf{Im}^a \mathbf{TxtFex}_c^b$, $\mathbf{N}^a \mathbf{TxtBc}^b$, $\mathbf{In}^a \mathbf{TxtBc}^b$, $\mathbf{Im}^a \mathbf{TxtBc}^b$, and the function identification paradigms: $\mathbf{N}^a \mathbf{Bc}^b$, $\mathbf{In}^a \mathbf{Bc}^b$, $\mathbf{Im}^a \mathbf{Bc}^b$.

Analogue of Lemma 11 holds for identification from inaccurate information sequences also (for both function learning and language learning). We state as an example the lemma for $\mathbf{N}^a \mathbf{TxtEx}^b$ -identification.

Lemma 25 *If $\mathbf{M} \mathbf{N}^a \mathbf{TxtEx}^b$ -identifies L , then for every L' such that $L \subseteq L'$ and $\text{card}(L' - L) \leq a$, there exists a stabilizing sequence σ for \mathbf{M} on L' such that $W_{\mathbf{M}(\sigma)} =^b L$.*

3.2 Hierarchy Results

Theorem 26 below is about comparing extra anomalies in the final program inferred in the presence of inaccuracies. Theorem 26 strengthens Theorem 5 to consider inaccuracies in the input data. Theorem 26(a), for example, says that there are classes of functions for which an $(i + 1)$ -error program can be identified from information sequences with an arbitrary but finite number of imperfections, but for which an i -error program *cannot* be identified even from accurate information sequences. Theorem 26(d) is a similar result about \mathbf{Bc} -identification. It is also interesting to note that there is a machine that can \mathbf{Bc}^* -identify every computable function even from information sequences with an unbounded finite number of imperfections. Theorem 26 yields a hierarchy exhibited in Corollary 27.

Theorem 26 For all $i \in N$

- (a) $\mathbf{Im}^* \mathbf{Ex}^{i+1} - \mathbf{Ex}^i \neq \emptyset$.
- (b) $\mathbf{Im}^* \mathbf{Ex}^* - \bigcup_i \mathbf{Ex}^i \neq \emptyset$.
- (c) $\mathbf{Im}^* \mathbf{Bc} - \mathbf{Ex}^* \neq \emptyset$.
- (d) $\mathbf{Im}^* \mathbf{Bc}^{i+1} - \mathbf{Bc}^i \neq \emptyset$.
- (e) $\mathcal{R} \in \mathbf{Im}^* \mathbf{Bc}^*$.

PROOF. (a) Let $\mathcal{C} = \{f \mid \varphi_{f(0)} =^{i+1} f \wedge (\forall x)[f(2x) = f(0)]\}$. It is easy to see that $\mathcal{C} \in \mathbf{Im}^* \mathbf{Ex}^{i+1}$. An easy modification of the proof of $\mathbf{Ex}^{i+1} - \mathbf{Ex}^i \neq \emptyset$ in [CS83] can be used to show that $\mathcal{C} \notin \mathbf{Ex}^i$. We omit the details.

(b), (c) and (d) can be shown by similar modifications of the proofs in [CS83]. We leave the details to the reader.

(e) Let $S(\sigma, n) = \{n\} \cup \{j \leq n \mid (\forall (x, y) \in \text{content}(\sigma))[\Phi_j(x) \leq n \wedge (x, \varphi_j(x)) \in \text{content}(\sigma)]\}$. Let $\mathbf{M}(\sigma) = p$ such that $(\forall x)[\varphi_p(x) = \varphi_{\min(S(\sigma, x))}(x)]$. We now claim that $\mathbf{M} \mathbf{Im}^* \mathbf{Bc}^*$ -identifies \mathcal{R} . To see this suppose $f \in \mathcal{R}$. Suppose G is a $*$ -imperfect information sequence for f . Let j be minimum program such that

- (a) $\text{graph}(\varphi_j) \subseteq \text{content}(G)$, and
- (b) $(\forall x \mid (\exists y)[(x, y) \in \text{content}(G)])[\varphi_j(x) \downarrow]$.

(Note that for any j satisfying the above two properties, we have $\varphi_j =^* f$).

It is easy to verify that $(\forall n)(\forall m)[\min(S(T[n], m)) = j]$. It thus follows that $\mathbf{M} \mathbf{Im}^* \mathbf{Bc}^*$ -identifies f . ■

Corollary 27 Let $a \in N \cup \{*\}$. Then

- (a) $\mathbf{N}^a \mathbf{Ex}^0 \subset \mathbf{N}^a \mathbf{Ex}^1 \subset \dots \subset \mathbf{N}^a \mathbf{Ex}^*$.
- (b) $\mathbf{N}^a \mathbf{Bc}^0 \subset \mathbf{N}^a \mathbf{Bc}^1 \subset \dots \subset \mathbf{N}^a \mathbf{Bc}^* = 2^{\mathcal{R}}$.
- (c) $\mathbf{In}^a \mathbf{Ex}^0 \subset \mathbf{In}^a \mathbf{Ex}^1 \subset \dots \subset \mathbf{In}^a \mathbf{Ex}^*$.
- (d) $\mathbf{In}^a \mathbf{Bc}^0 \subset \mathbf{In}^a \mathbf{Bc}^1 \subset \dots \subset \mathbf{In}^a \mathbf{Bc}^* = 2^{\mathcal{R}}$.
- (e) $\mathbf{Im}^a \mathbf{Ex}^0 \subset \mathbf{Im}^a \mathbf{Ex}^1 \subset \dots \subset \mathbf{Im}^a \mathbf{Ex}^*$.
- (f) $\mathbf{Im}^a \mathbf{Bc}^0 \subset \mathbf{Im}^a \mathbf{Bc}^1 \subset \dots \subset \mathbf{Im}^a \mathbf{Bc}^* = 2^{\mathcal{R}}$.

Corollary 27 thus shows that the hierarchies given by Theorem 5 hold even in presence of inaccuracies in the input data.

We now turn our attention to investigating if a price is being paid for learning from information sequences for which the number of inaccuracies is higher. Theorem 28 below answers this question in the affirmative.

Theorem 28 $(\forall j \in N)[\mathbf{Im}^j \mathbf{Ex} - [\mathbf{N}^{j+1} \mathbf{Ex}^* \cup \mathbf{In}^{j+1} \mathbf{Ex}^* \cup (\bigcup_i \mathbf{In}^{j+1} \mathbf{Bc}^i) \cup (\bigcup_i \mathbf{N}^{j+1} \mathbf{Bc}^i)]] \neq \emptyset$.

PROOF. Fix j . For $f \in \mathcal{R}$, define f' and f'' as follows.

For all $i > 0$ and for all x , let $f'(\langle i, x \rangle) = f''(\langle i, x \rangle) = f(x)$.

For all $x \leq j$, let $f'(\langle 0, x \rangle) = 0$; $f''(\langle 0, x \rangle) = 1$.

For all $x > j$, let $f'(\langle 0, x \rangle) = f''(\langle 0, x \rangle) = 0$.

Now consider the following collections of functions:

$\mathcal{S}' = \{f' \mid f \in \mathbf{AEZ}\}$.

$\mathcal{S}'' = \{f'' \mid f \in \mathbf{SELFREF}\}$.

$\mathcal{C} = \mathcal{S}' \cup \mathcal{S}''$.

It is easy to verify that $\mathcal{S}' \in \mathbf{Im}^j \mathbf{Ex}$ and $\mathcal{S}'' \in \mathbf{Im}^j \mathbf{Ex}$.

Note that for a j -imperfect information sequence G for $f \in \mathcal{C}$, $\text{card}(\{x \leq j \mid (\langle 0, x \rangle, 0) \in \text{content}(G)\}) \geq \text{card}(\{x \leq j \mid (\langle 0, x \rangle, 1) \in \text{content}(G)\}) \Leftrightarrow f(\langle 0, 0 \rangle) = 0 \Leftrightarrow f \in \mathcal{S}'$. This along with the fact that $\mathcal{S}' \in \mathbf{Im}^j \mathbf{Ex}$, $\mathcal{S}'' \in \mathbf{Im}^j \mathbf{Ex}$ implies that $\mathcal{C} \in \mathbf{Im}^j \mathbf{Ex}$.

We show that $\mathcal{C} \notin \mathbf{N}^{j+1} \mathbf{Ex}^* \cup [\bigcup_i \mathbf{N}^{j+1} \mathbf{Bc}^i]$. It can similarly be shown that $\mathcal{C} \notin \mathbf{In}^{j+1} \mathbf{Ex}^* \cup [\bigcup_i \mathbf{In}^{j+1} \mathbf{Bc}^i]$.

Suppose \mathbf{I} is either \mathbf{Ex}^* or \mathbf{Bc}^i for some i . Suppose by way of contradiction, machine \mathbf{M} $\mathbf{N}^{j+1} \mathbf{I}$ -identifies \mathcal{C} . Then, using \mathbf{M} , we show how to construct a machine \mathbf{M}' which \mathbf{I} -identifies $\mathbf{AEZ} \cup \mathbf{SELFREF}$ contradicting Theorem 6.

Let twit be a recursive function (by s-m-n theorem such a recursive function exists) such that, for all $\sigma, \tau \in \text{SEG}$

$\sigma \subseteq \tau \Rightarrow \text{twit}(\sigma) \subseteq \text{twit}(\tau)$, and

$\text{content}(\text{twit}(\sigma)) = \{(\langle r, x \rangle, y) \mid (x, y) \in \text{content}(\sigma) \wedge 1 \leq r \leq |\sigma|\} \cup \{(\langle 0, x \rangle, 1) \mid x \leq j\} \cup \{(\langle 0, r \rangle, 0) \mid r \leq |\sigma|\}$.

Let untwit be a recursive function (by s-m-n theorem such a recursive function exists) such that, for all i and x , $\varphi_{\text{untwit}(i)}(x) = \varphi_i(\langle 1, x \rangle)$.

Now, let $\mathbf{M}'(\sigma) = \text{untwit}(\mathbf{M}(\text{twit}(\sigma)))$. It is easy to see that if \mathbf{M} $\mathbf{N}^{j+1} \mathbf{I}$ -identifies \mathcal{C} , then \mathbf{M}' \mathbf{I} -identifies $\mathbf{AEZ} \cup \mathbf{SELFREF}$. But, this is not possible (Theorem 6). Thus, $\mathcal{C} \notin \mathbf{N}^{j+1} \mathbf{I}$. ■

Theorem 28 yields several corollaries which highlight the loss in learning ability as a result of an increase in the bound on inaccuracies allowed in the information sequence.

Corollary 29 *Let $a \in N \cup \{*\}$. Then*

(a) $\mathbf{N}^0 \mathbf{Ex}^a \supset \mathbf{N}^1 \mathbf{Ex}^a \supset \mathbf{N}^2 \mathbf{Ex}^a \supset \dots$.

(b) $\mathbf{In}^0 \mathbf{Ex}^a \supset \mathbf{In}^1 \mathbf{Ex}^a \supset \mathbf{In}^2 \mathbf{Ex}^a \supset \dots$.

(c) $\mathbf{Im}^0 \mathbf{Ex}^a \supset \mathbf{Im}^1 \mathbf{Ex}^a \supset \mathbf{Im}^2 \mathbf{Ex}^a \supset \dots$.

However, the above result does not say anything about relative learning abilities between situations in which a preassigned bound on the finite number of inaccuracies is available and when such a bound is not available. Theorem 30 below answers this question.

Theorem 30 *There exists a class of functions, \mathcal{C} such that (a) $(\forall i)[\mathcal{C} \in \mathbf{Im}^i \mathbf{Ex}]$, and (b) $\mathcal{C} \notin (\mathbf{In}^* \mathbf{Ex}^* \cup \mathbf{N}^* \mathbf{Ex}^* \cup \bigcup_j [\mathbf{N}^* \mathbf{Bc}^j \cup \mathbf{In}^* \mathbf{Bc}^j])$.*

PROOF. For $f \in \mathcal{R}$ define f', f'' as follows.

For $j > 0$, let $f'(\langle j, x \rangle) = f''(\langle j, x \rangle) = f(x)$.

For $x > f(0)$, let $f'(\langle 0, x \rangle) = f''(\langle 0, x \rangle) = 0$.

For $x \leq f(0)$, let $f'(\langle 0, x \rangle) = 0, f''(\langle 0, x \rangle) = 1$.

Now consider the following class of functions:

$\mathcal{S}' = \{f' \mid f \in \mathbf{AEZ}\}$.

$\mathcal{S}'' = \{f'' \mid f \in \mathbf{SELFREF}\}$.

$\mathcal{C} = \mathcal{S}' \cup \mathcal{S}''$.

It is easy to extend the proof of Theorem 28 to prove that $\mathcal{C} \notin \mathbf{N}^* \mathbf{Ex}^* \cup \mathbf{In}^* \mathbf{Ex}^* \cup \bigcup_j \mathbf{N}^* \mathbf{Bc}^j \cup \bigcup_j \mathbf{In}^* \mathbf{Bc}^j$.

Fix i . We now show that $\mathcal{C} \in \mathbf{Im}^i \mathbf{Ex}$. Let $S_i = \{j \leq i \mid \varphi_j(0) = j \wedge \varphi_j \in \mathbf{SELFREF}\}$. We describe a machine \mathbf{M} which $\mathbf{Im}^i \mathbf{Ex}^*$ -identifies \mathcal{C} . Let F' be a recursive mapping from finite sets to N such that the following holds (note that by s-m-n theorem such a function exists). Below A denotes a finite set.

$$\varphi_{F'(A)}(\langle i, x \rangle) = \begin{cases} 0, & \text{if } i = 0; \\ y, & \text{if } i > 0 \wedge \\ & [(\exists z)[(x, z) \in A] \wedge y = \min(\{z \mid (x, z) \in A\})]; \\ 0, & \text{otherwise.} \end{cases}$$

Let F'' be a recursive function such that the following holds (note that by s-m-n theorem such a function exists).

$$\varphi_{F''(e)}(\langle i, x \rangle) = \begin{cases} 0, & \text{if } i = 0, x > e; \\ \varphi_e(x), & \text{if } i > 0; \\ 1, & \text{otherwise.} \end{cases}$$

$\mathbf{M}(\sigma)$

1. Let $m_\sigma(x) = \max(\{j \mid (\exists y)[(\langle j, x \rangle, y) \in \text{content}(\sigma)]\})$.

Let $A_\sigma = \{(x, y) \mid m_\sigma(x) > 0 \wedge (\langle m_\sigma(x), x \rangle, y) \in \text{content}(\sigma)\}$.

Let $A'_\sigma = \{(x, y) \in A_\sigma \mid y \neq 0\}$.

2. Let $e_\sigma = \min(\{|\sigma|\} \cup \{y \mid (0, y) \in A_\sigma\})$.

3. **if** $e_\sigma \leq i$
 then
 3a. **if** $[e_\sigma \in S_i \wedge (\forall(x, y) \in A_\sigma)[\varphi_{e_\sigma}(x) = y]]$
 then output $F''(e_\sigma)$.
 else output $F'(A'_\sigma)$.
 endif
 else
 3b. **if** $[\text{card}(\{x \leq e_\sigma \mid (\langle 0, x \rangle, 0) \in \text{content}(\sigma)\}) \geq \text{card}(\{x \leq e_\sigma \mid (\langle 0, x \rangle, 1) \in \text{content}(\sigma)\})]$
 then output $F'(A'_\sigma)$.
 else output $F''(e_\sigma)$.
 endif
 endif
end

Consider any $g \in \mathcal{C}$ and an i -imperfect information sequence G for g . Let f be such that $g = f'$, if $g \in \mathcal{S}'$ and $g = f''$, if $g \in \mathcal{S}''$. It is easy to verify that, for large enough initial segment σ of G , the following properties are satisfied.

(P1) $A_\sigma \subseteq \text{graph}(f)$.

(P2) $e_\sigma = f(0)$.

(P3) $(\forall x \leq f(0))[(\langle 0, x \rangle, y) \in \text{content}(G) \Leftrightarrow (\langle 0, x \rangle, y) \in \text{content}(\sigma)]$.

(P4) $f \in \mathbf{AEZ} \Rightarrow A'_\sigma = \{(x, y) \mid f(x) \neq 0\}$

(P5) $[f(0) \in S_i \wedge f \in \mathbf{AEZ}] \Rightarrow [(\exists(x, y) \in A_\sigma)[\varphi_{f(0)}(x) \neq y]]$.

For the following suppose that σ is large enough so that the above properties hold.

Case 1: $f(0) \leq i \wedge f \in \mathbf{SELFREF}$.

In this case “if clause of step (3a)” will succeed (by properties (P1) and (P2)) and thus **M** outputs $F''(e_\sigma)$, which is a program for g .

Case 2: $f(0) \leq i \wedge f \notin \mathbf{SELFREF}$

In this case “if clause of step (3a)” will fail (by properties (P1), (P2), and (P5)). Thus **M** outputs $F'(A'_\sigma)$, which is a program for g (by property (P4)).

Case 3: $f(0) > i \wedge f \in \mathbf{AEZ}$.

In this case by property (P3) “if clause in step 3b” will succeed (due to bound on the imperfection of G). Thus **M** outputs $F'(A'_\sigma)$, which by property (P4) is a program for g .

Case 4: $f(0) > i \wedge f \in \mathbf{SELFREF}$.

In this case by property (P3) and the bound on imperfection, “if clause in step 3b” will fail.

Thus \mathbf{M} outputs $F''(e_\sigma)$, which is a program for g .

From the above cases we have that \mathbf{M} $\mathbf{Im}^i\mathbf{Ex}$ -identifies \mathcal{C} . ■

We now briefly turn our attention to identification of languages from inaccurate information sequences. Theorem 31 below is the language identification counterpart of Theorem 26 and can be shown using techniques similar to Theorem 26 and results from [CL82, Cas88]. We omit the details.

Theorem 31 *For all $i \in \mathbb{N}$*

- (a) $\mathbf{Im}^*\mathbf{TxtEx}^{i+1} - \mathbf{TxtFex}_*^i \neq \emptyset$.
- (b) $\mathbf{Im}^*\mathbf{TxtEx}^* - \bigcup_i \mathbf{TxtBc}^i \neq \emptyset$.
- (c) $\mathbf{Im}^*\mathbf{TxtFex}_{i+1} - \mathbf{TxtFex}_*^i \neq \emptyset$.
- (d) $\mathbf{Im}^*\mathbf{TxtFex}_* - \bigcup_{i \in \mathbb{N}} \mathbf{TxtFex}_*^i \neq \emptyset$.
- (e) $\mathbf{Im}^*\mathbf{TxtEx}^{2i+1} - \mathbf{TxtBc}^i \neq \emptyset$.
- (f) $\mathbf{Im}^*\mathbf{TxtBc} - \mathbf{TxtFex}_*^* \neq \emptyset$.
- (g) $\mathbf{Im}^*\mathbf{TxtBc}^{i+1} - \mathbf{TxtBc}^i \neq \emptyset$.

Theorem 32 $(\forall i \in \mathbb{N})[\mathbf{Im}^i\mathbf{TxtEx} - [\mathbf{N}^{i+1}\mathbf{TxtBc}^* \cup \mathbf{In}^{i+1}\mathbf{TxtBc}^*] \neq \emptyset]$.

PROOF. For any finite language L , let $L' = \{\langle j, x \rangle \mid j \geq 1 \wedge x \in L\} \cup \{\langle 0, i + x + 1 \rangle \mid x \leq i\}$. Also, let $N' = \{\langle j, x \rangle \mid j \geq 1, x \in \mathbb{N}\} \cup \{\langle 0, x \rangle \mid x \leq i\}$.

Now, consider the collection of languages $\mathcal{L} = \{L' \mid L \in \mathbf{FIN}\} \cup \{N'\}$. It is easy to see that $\mathcal{L} \in \mathbf{Im}^i\mathbf{TxtEx}$. For any text T , let T', T'' (effectively obtained from T) be such that $\text{content}(T') = \{\langle j, x \rangle \mid j \geq 1 \wedge x \in \text{content}(T)\} \cup \{\langle 0, x \rangle \mid x \leq 2i + 1\}$ and $\text{content}(T'') = \{\langle j, x \rangle \mid j \geq 1 \wedge x \in \text{content}(T)\}$. Note that if T is a text for $L \in \mathbf{FIN} \cup \{N\}$, then T' is an $(i+1)$ -noisy text for L' and T'' is an $(i+1)$ -incomplete text for L' . Moreover, for $L \in \mathbf{FIN} \cup \{N\}$, a grammar for a finite variant of L' can be effectively converted to a grammar for a finite variant of L . Thus $[\mathcal{L} \in \mathbf{N}^{i+1}\mathbf{TxtBc}^* \vee \mathcal{L} \in \mathbf{In}^{i+1}\mathbf{TxtBc}^*] \Rightarrow \mathbf{FIN} \cup \{N\} \in \mathbf{TxtBc}^*$. But, this is not true (Theorem 18). Thus, $\mathcal{L} \notin (\mathbf{N}^{i+1}\mathbf{TxtBc}^* \cup \mathbf{In}^{i+1}\mathbf{TxtBc}^*)$. ■

Theorem 33 *There exists a \mathcal{L} , such that (a) $(\forall i)[\mathcal{L} \in \mathbf{Im}^i\mathbf{TxtEx}]$ and (b) $\mathcal{L} \notin \mathbf{N}^*\mathbf{TxtBc}^* \cup \mathbf{In}^*\mathbf{TxtBc}^*$.*

PROOF. For a set X and $j \in \mathbb{N}$ define $L_{j,X}^0$ and $L_{j,X}^1$ as follows.

$$\text{Let } L_{j,X}^0 = \{\langle 2, x \rangle \mid x \in X\} \cup \{\langle 1, k \cdot j \rangle \mid k \in \mathbb{N}\} \cup \{\langle 0, x \rangle \mid x < j\}.$$

Let $L_{j,X}^1 = \{\langle 2, x \rangle \mid x \in X\} \cup \{\langle 1, k \cdot j \rangle \mid k \in N\} \cup \{\langle 0, x \rangle \mid j \leq x < 2j\}$.

Let S be a set which is not in Σ_3 of the Kleene hierarchy.

Let $\mathcal{L} = \{L_{j,X}^0 \mid X \in \mathbf{FIN} \wedge j \notin S\} \cup \{L_{j,N}^1 \mid j \in S\}$.

It is easy to see using technique of Theorem 30 that for each $i \in N$, $\mathcal{L} \in \mathbf{Im}^i \mathbf{TxtEx}$.

We show that $\mathcal{L} \notin \mathbf{N}^* \mathbf{TxtBc}^*$. Proof is similar for $\mathcal{L} \notin \mathbf{In}^* \mathbf{TxtBc}^*$. Suppose by way of contradiction machine $\mathbf{M} \mathbf{N}^* \mathbf{TxtBc}^*$ -identifies \mathcal{L} . Then

$j \in S \Rightarrow (\exists \sigma \mid \text{content}(\sigma) \subseteq L_{j,N}^1 \cup \{\langle 0, x \rangle \mid x < 2j\}) (\forall \tau \supseteq \sigma \mid \text{content}(\tau) \subseteq L_{j,N}^1 \cup \{\langle 0, x \rangle \mid x < 2j\}) [\text{card}(\{x \mid \langle 2, x \rangle \in W_{\mathbf{M}(\tau)}\}) = \infty]$.

On the other hand

$j \notin S \Rightarrow \neg [(\exists \sigma \mid \text{content}(\sigma) \subseteq L_{j,N}^1 \cup \{\langle 0, x \rangle \mid x < 2j\}) (\forall \tau \supseteq \sigma \mid \text{content}(\tau) \subseteq L_{j,N}^1 \cup \{\langle 0, x \rangle \mid x < 2j\}) [\text{card}(\{x \mid \langle 2, x \rangle \in W_{\mathbf{M}(\tau)}\}) = \infty]]$. Thus $S \in \Sigma_3$. A contradiction.

Thus $\mathcal{L} \notin \mathbf{N}^* \mathbf{TxtBc}^*$. ■

It should be noted that the above theorems give hierarchies for language learning, similar in nature to the hierarchies in the function learning.

3.3 Relative Effects

Results in the preceding section underlined a common theme: But for the highly impractical \mathbf{Bc}^* -identification, inaccurate data is ‘bad’ for learning. It is time now to compare which kinds of inaccuracies do the most damage. First, a comparison between the price paid for learning from noisy information sequences and incomplete information sequences.

The result below exhibits the apparent advantages of noise over missing data, as there are collections of functions for which a program can be identified from information sequences with a finite, but without any preassigned bound, number of spurious data, but for which a program for even a finite variant *cannot* be identified from information sequences which possibly are missing just *one* data. Corollary 35 gives the language identification counterpart of this result and follows from Theorem 34. The following result also appears in [FJO94]. We include the proof for completeness.

Theorem 34 [FJO94] $\mathbf{N}^* \mathbf{Ex} - [\mathbf{In}^1 \mathbf{Ex}^* \cup \bigcup_i \mathbf{In}^1 \mathbf{Bc}^i] \neq \emptyset$.

PROOF. For any $f \in \mathcal{R}$, we describe a function f' as follows:

Let $p_f = \text{MinProg}(f)$.

For $j < p_f$, let $\text{err}_j = \min(\{x \mid \varphi_j(x) \neq f(x)\})$.

$f'(0) = \langle p_f, \langle \text{err}_0, \text{err}_1, \text{err}_2, \dots, \text{err}_{p_f-1} \rangle \rangle$.

For all x, k : $f'(1 + \langle k, x \rangle) = f(x)$.

Let $\mathcal{C} = \{f' \mid f \in \mathcal{R}\}$.

We first show that $\mathcal{C} \notin \mathbf{In}^1\mathbf{Ex}^* \cup \bigcup_i \mathbf{In}^1\mathbf{Bc}^i$. Suppose \mathbf{I} is either \mathbf{Ex}^* or \mathbf{Bc}^i . Suppose by way of contradiction, that $\mathbf{M} \mathbf{In}^1\mathbf{I}$ -identifies \mathcal{C} . We then describe a machine \mathbf{M}' which \mathbf{I} -identifies \mathcal{R} —yielding a contradiction (Theorem 5). For any f , let G_f be an information sequence such that $\text{content}(G_f) = \{(1 + \langle k, x \rangle, f(x)) \mid x, k \in N\}$, and $G_f[n]$ can be found effectively from $f[n]$. Note that G_f is an 1-incomplete information sequence for f' . Let F be a recursive function such that for all x and p , $\varphi_{F(p)}(x) = \varphi_p(\langle 1 + \langle 0, x \rangle \rangle)$. Define \mathbf{M}' such that $\mathbf{M}'(f[n]) = F(\mathbf{M}(G_f[n]))$. Note that such a machine \mathbf{M}' can easily be constructed from \mathbf{M} . Clearly, for $f \in \mathcal{R}$, if $\varphi_p =^a f'$, then $\varphi_{F(p)} =^a f$. Since $\mathbf{M} \mathbf{In}^1\mathbf{I}$ -identifies \mathcal{C} , it follows that $\mathbf{M}' \mathbf{I}$ -identifies \mathcal{R} . A contradiction (Theorem 5). Thus, $\mathcal{C} \notin \mathbf{In}^1\mathbf{I}$.

We now show that $\mathcal{C} \in \mathbf{N}^*\mathbf{Ex}$. For an information sequence G , define G' such that for all n ,

$$G'(n) = \begin{cases} (1 + \langle i, x \rangle, y), & \text{if } G(n) = (1 + \langle i, x \rangle, y) \wedge \\ & [(\forall j, z \mid (1 + \langle j, x \rangle, z) \in \text{content}(G)) \\ & (\exists j' \geq j)[(1 + \langle j', x \rangle, y) \in \text{content}(G)]]; \\ (0, y), & \text{if } G(n) = (0, y); \\ \#, & \text{otherwise.} \end{cases}$$

Let F be a mapping from SEG to SEG such that for $n \leq |\sigma|$:

$$(F(\sigma))(n) = \begin{cases} (1 + \langle i, x \rangle, y), & \text{if } \sigma(n) = (1 + \langle i, x \rangle, y) \wedge \\ & [(\forall j, z \mid (1 + \langle j, x \rangle, z) \in \text{content}(\sigma)) \\ & (\exists j' \geq j)[(1 + \langle j', x \rangle, y) \in \text{content}(\sigma)]]; \\ (0, y), & \text{if } \sigma(n) = (0, y); \\ \#, & \text{otherwise.} \end{cases}$$

Intuitively, for a *-noisy sequence, G , for $f' \in \mathcal{C}$, G' is formed from G by removing all noisy elements from G , except for noise on input 0. F is an algorithmic mechanism for obtaining G' from G .

Suppose G is a *-noisy sequence for $f' \in \mathcal{C}$. It is easy to verify that:

- (a) $(\forall^\infty n)[G'[n] = F(G[n])]$.
- (b) $(\forall i, x, y)[(1 + \langle i, x \rangle, y) \in \text{content}(G') \iff f'(1 + \langle i, x \rangle) = y]$.
- (c) $\pi_1(f'(0)) = \max(\{j \mid (\exists \text{err}_0, \text{err}_1, \dots, \text{err}_{j-1} \mid (0, \langle j, \langle \text{err}_0, \text{err}_1, \dots, \text{err}_{j-1} \rangle \rangle) \in \text{content}(G))[(\forall i < j)[\varphi_i(\text{err}_i) \neq f'(1 + \langle 0, \text{err}_i \rangle)]]\})$.
- (d) Suppose $e = \pi_1(f'(0))$. For $j < e$, let $\text{err}_j = \min(\{x \mid \varphi_j(x) \neq \varphi_e(x)\})$. Then $f'(0) = \langle e, \langle \text{err}_0, \dots, \text{err}_{e-1} \rangle \rangle$.

Thus $f'(0)$ (and therefore a program for f') can be determined from G' (and thus G) in the limit. ■

Corollary 35

(a) $\mathbf{N}^*\mathbf{TxtEx} - \mathbf{In}^1\mathbf{TxtEx}^* \cup \bigcup_i \mathbf{In}^1\mathbf{TxtBc}^i \neq \emptyset$.

(b) $\mathbf{N}^*\mathbf{Ex} - \mathbf{Im}^1\mathbf{Ex}^* \cup \bigcup_i \mathbf{Im}^1\mathbf{Bc}^i \neq \emptyset$.

(c) $\mathbf{N}^*\mathbf{TxtEx} - \mathbf{Im}^1\mathbf{TxtEx}^* \cup \bigcup_i \mathbf{Im}^1\mathbf{TxtBc}^i \neq \emptyset$.

PROOF. Part (b) is immediate corollary of Theorem 34. We show how to obtain part (a) using Theorem 34 (this is a standard trick used to convert diagonalization results from function learning to diagonalization results in language learning). Part (c) can be obtained from part (b) similarly.

Suppose $\mathcal{C} \in \mathbf{N}^*\mathbf{Ex} - (\mathbf{In}^1\mathbf{Ex}^* \cup \bigcup_i \mathbf{In}^1\mathbf{Bc}^i)$. For any total function f , define $L_f = \{\langle x, y \rangle \mid f(x) = y\}$. Let $\mathcal{L} = \{L_f \mid f \in \mathcal{C}\}$. We claim that $\mathcal{L} \in \mathbf{N}^*\mathbf{TxtEx} - \mathbf{In}^1\mathbf{TxtEx}^* \cup \bigcup_i \mathbf{In}^1\mathbf{TxtBc}^i$.

We first show that $\mathcal{L} \in \mathbf{N}^*\mathbf{TxtEx}$. To show this we show:

- (i) how to effectively convert an a -noisy text for L_f to an a -noisy information sequence for f , and
- (ii) how to effectively convert a program for f to a grammar for L_f .

To show (i), define G_T as follows:

$$G_T(n) = \begin{cases} \#, & \text{if } T(n) = \#; \\ (x, y), & \text{if } T(n) = \langle x, y \rangle. \end{cases}$$

It is easy to see that G_T satisfies (i).

To show (ii), define g_p such that $W_{g_p} = \{\langle x, y \rangle \mid \varphi_p(x) = y\}$. It is easy to see (using s-m-n theorem) that such a g_p can be effectively obtained from p and satisfies (ii).

Now suppose $\mathcal{C} \subseteq \mathbf{N}^*\mathbf{Ex}(\mathbf{M})$. Let \mathbf{M}' be defined as follows: $\mathbf{M}'(T[n]) = g_{\mathbf{M}(G_T[n])}$. It follows using (i) and (ii) that $\mathcal{L} \subseteq \mathbf{N}^*\mathbf{TxtEx}(\mathbf{M}')$.

We now show that $\mathcal{L} \notin \mathbf{In}^1\mathbf{TxtEx}^* \cup \bigcup_i \mathbf{In}^1\mathbf{TxtBc}^i$. To show this we show:

- (iii) how to effectively convert an 1-incomplete information sequence for f to an 1-incomplete text for L_f , and
- (iv) how to effectively convert an a -error grammar for L_f to an a -error program for f .

To show (iii), define T_G as follows:

$$T_G(n) = \begin{cases} \#, & \text{if } G(n) = \#; \\ \langle x, y \rangle, & \text{if } G(n) = (x, y). \end{cases}$$

It is easy to see that T_G satisfies (iii).

To show (iv), define p_g such that $\varphi_{p_g}(x) = y$, such that $\langle x, y \rangle \in W_g$ (if there are several y such that $\langle x, y \rangle \in W_g$, then choose the first one in some arbitrary enumeration of W_g).

Now suppose by way of contradiction that $\mathcal{L} \subseteq \mathbf{In}^1\mathbf{TxtEx}^*(\mathbf{M})$ (respectively $\mathcal{L} \subseteq \mathbf{In}^1\mathbf{TxtBc}^i(\mathbf{M})$). Let \mathbf{M}' be defined as follows: $\mathbf{M}'(G[n]) = p_{\mathbf{M}(T_G[n])}$. It follows using (iii) and (iv) that $\mathcal{C} \subseteq \mathbf{In}^1\mathbf{Ex}^*(\mathbf{M}')$ (respectively $\mathcal{C} \subseteq \mathbf{In}^1\mathbf{Bc}^i(\mathbf{M}')$) contradicting Theorem 34. Thus $\mathcal{L} \notin \mathbf{In}^1\mathbf{TxtEx}^* \cup \bigcup_i \mathbf{In}^1\mathbf{TxtBc}^i$. \blacksquare

Since $\mathcal{E} \notin \mathbf{TxtBc}^*$, there is scope for further fine tuning of Corollary 35(a). An attempt at such a refinement is the subject of Theorem 36 below.

Theorem 36 $(\forall j \in N)[\mathbf{N}^j\mathbf{TxtEx} - \mathbf{In}^1\mathbf{TxtBc}^* \neq \emptyset]$.

PROOF. For $n \in N^+$, let $L_n = \{\langle i, x \rangle \mid i \geq 1 \wedge x \leq n\} \cup \{\langle 0, n \rangle\}$. Let $L_0 = \{\langle i, x \rangle \mid i \geq 1 \wedge x \in N\} \cup \{\langle 0, 0 \rangle\}$. Let $\mathcal{L} = \{L_n \mid n \in N\}$.

We claim that $\mathcal{L} \in \mathbf{N}^j\mathbf{TxtEx}$. Suppose T is a j -noisy text for $L \in \mathcal{L}$. Then, $L = L_0 \iff (\exists x)[x > \max(\{w \mid \langle 0, w \rangle \in \text{content}(T)\}) \wedge (\forall k \mid 1 \leq k \leq j+1)[\langle k, x \rangle \in \text{content}(T)]]$. Thus, it can be determined, in the limit, if T is a j -noisy text for L_0 . Also, it is easy to see that $\{L_n \mid n \in N^+\} \in \mathbf{N}^j\mathbf{TxtEx}$. Thus, $\mathcal{L} \in \mathbf{N}^j\mathbf{TxtEx}$.

Let $L'_n = L_n - \{\langle 0, n \rangle\}$. Now, $[\mathcal{L} \in \mathbf{In}^1\mathbf{TxtBc}^*] \Rightarrow \{L'_n \mid n \in N\} \in \mathbf{TxtBc}^*$. But, this is not true (Corollary 19). Thus, $\mathcal{L} \notin \mathbf{In}^1\mathbf{TxtBc}^*$. \blacksquare

Corollary 37 $(\forall j \in N)[\mathbf{N}^j\mathbf{TxtEx} - \mathbf{Im}^1\mathbf{TxtBc}^* \neq \emptyset]$.

It is open at present if Theorem 36 above can be extended to the assertion “ $\mathbf{N}^*\mathbf{TxtEx} - \mathbf{In}^1\mathbf{TxtBc}^* \neq \emptyset$.”

The reader should note that Theorem 34 implies that there are collections of functions that can be learned from the ‘most offensive noisy information sequences,’ but cannot be learned from the ‘least offensive incomplete information sequences.’ A natural question that arises is if there are collections of functions that can be learned from the ‘most offensive incomplete information sequences,’ but cannot be learned from the ‘least offensive noisy information sequences.’ Theorem 38 below answers this question negatively thereby implying that, in the context of function identification, missing data are strictly more harmful than noisy data.

Theorem 38 *Let $a, b \in N \cup \{*\}$.*

(a) $\mathbf{In}^a\mathbf{Ex}^b \subseteq \mathbf{N}^a\mathbf{Ex}^b$.

(b) $\mathbf{In}^a\mathbf{Bc}^b \subseteq \mathbf{N}^a\mathbf{Bc}^b$.

PROOF. Let machine \mathbf{M} be given. We construct a machine \mathbf{M}' such that $\mathbf{In}^a\mathbf{Ex}^b(\mathbf{M}) \subseteq \mathbf{N}^a\mathbf{Ex}^b(\mathbf{M}')$ and $\mathbf{In}^a\mathbf{Bc}^b(\mathbf{M}) \subseteq \mathbf{N}^a\mathbf{Bc}^b(\mathbf{M}')$.

For any information sequence G let G' be a information sequence such that for all n, x, y :

$$G'(n) = \begin{cases} (x, y), & \text{if } G(n) = (x, y) \wedge \\ & (\forall z \neq y)[(x, z) \notin \text{content}(G)]; \\ \#, & \text{otherwise.} \end{cases}$$

Let F be a mapping from SEG to SEG such that, for all σ , for all $n < |\sigma|$, and for all $x, y \in N$:

$$(F(\sigma))(n) = \begin{cases} (x, y), & \text{if } \sigma(n) = (x, y) \wedge \\ & (\forall z \neq y)[(x, z) \notin \text{content}(\sigma)]; \\ \#, & \text{otherwise.} \end{cases}$$

Now, if G is an a -noisy information sequence for $f \in \mathcal{R}$, then G' is an a -incomplete information sequence for f . Moreover $(\bigvee^\infty n)[G'[n] = F(G[n])]$.

Let $\mathbf{M}'(\sigma) = \mathbf{M}(F(\sigma))$. It is easy to see that $\mathbf{In}^a\mathbf{Ex}^b(\mathbf{M}) \subseteq \mathbf{N}^a\mathbf{Ex}^b(\mathbf{M}')$ and $\mathbf{In}^a\mathbf{Bc}^b(\mathbf{M}) \subseteq \mathbf{N}^a\mathbf{Bc}^b(\mathbf{M}')$. ■

The story, however, is different for language identification as implied by Theorem 39 below.

Theorem 39 $(\forall i \in N)[\mathbf{In}^{2i-1}\mathbf{TxtEx}^i - \mathbf{N}^i\mathbf{TxtEx}^* \neq \emptyset]$.

PROOF. For any $L \in \mathcal{E}$, let $S_{x,L} = \{z \mid \langle x, z \rangle \in L\}$. Let $\mathcal{L} = \{L \mid \text{card}(\{x \leq 2 \mid 1 \leq \text{card}(S_{x,L}) < \infty \wedge W_{\max(S_{x,L})} = L\}) \geq 2\}$.

In Lemma 40 below, it will be shown that $\mathcal{L} \notin \mathbf{TxtEx}^*$.

For $L \in \mathcal{L}$, let

$$L'_i = \{\langle j, x \rangle \mid j \geq 1 \wedge x \in L\} \cup \{\langle 0, 3j + x \rangle \mid j < i \wedge x \leq 2 \wedge 1 \leq \text{card}(S_{x,L}) < \infty \wedge W_{\max(S_{x,L})} = L\}.$$

Let $\mathcal{L}_i = \{L'_i \mid L \in \mathcal{L}\}$. It is easy to see that $\mathcal{L}_i \in \mathbf{In}^{2i-1}\mathbf{TxtEx}^i$.

Let $\text{Noisy}_i(L) = L \cup \{\langle 0, k \rangle \mid k < 3i\}$. Clearly, for any $L \in \mathcal{L}_i$, any text T for $\text{Noisy}_i(L)$, is also an i -noisy text for L . Corollary 41 thus implies that $\mathcal{L}_i \notin \mathbf{N}^i\mathbf{TxtEx}^*$. ■

Lemma 40 *Let \mathcal{L} be as defined in the proof of Theorem 39. Then $\mathcal{L} \notin \mathbf{TxtEx}^*$.*

PROOF. Suppose by way of contradiction that $\mathbf{M} \mathbf{TxtEx}^*$ -identifies \mathcal{L} . Then by the operator recursion theorem [Cas74], there exists a recursive 1-1 increasing p such that $W_{p(\cdot)}$ may be defined as follows.

Let φ -programs $p(0), p(1)$ enumerate $\langle 0, p(0) \rangle, \langle 1, p(1) \rangle$. Let σ_0 be such that $\text{content}(\sigma_0) = \{\langle 0, p(0) \rangle, \langle 1, p(1) \rangle\}$. Let $W_{p(\cdot)}^s$ denote $W_{p(\cdot)}$ enumerated before stage s . Go to stage 1. Note that there is no stage 0. We start at stage 1, just for the ease of writing the proof.

Stage s

1. Enumerate $W_{p(0)}^s \cup W_{p(1)}^s$ in $W_{p(0)}, W_{p(1)}, W_{p(2s)}$ and $W_{p(2s+1)}$. Enumerate $\langle 2, p(2s) \rangle$ in $W_{p(0)}, W_{p(2s)}$. Enumerate $\langle 2, p(2s+1) \rangle$ in $W_{p(1)}, W_{p(2s+1)}$. Let τ_0 be an extension of σ_s such that $\text{content}(\tau_0) = W_{p(0)}$ enumerated till now. Let τ_1 be an extension of σ_s such that $\text{content}(\tau_1) = W_{p(1)}$ enumerated till now.
2. Let $x = 0$. Dovetail (2a), (2b) until, if ever, step (2b) succeeds. If and when step (2b) succeeds, go to step (3).
 - 2a. Go to substage 0.

Substage s'

Enumerate $\langle 4, x \rangle$ in $W_{p(0)}, W_{p(2s)}$.

Enumerate $\langle 5, x \rangle$ in $W_{p(1)}, W_{p(2s+1)}$.

Let $x = x + 1$.

Go to substage $s' + 1$.

End substage s' .
 - 2b. Search for $i \in \{0, 1\}$, $n \in N$ such that $\mathbf{M}(\tau_i \diamond \langle 4+i, 0 \rangle \diamond \langle 4+i, 1 \rangle \diamond \dots \diamond \langle 4+i, n \rangle) \neq \mathbf{M}(\sigma_s)$.
3. If and when (2b) succeeds, let i and n be as found in (2b). Let

$S = (\{\langle 4+i, 0 \rangle, \langle 4+i, 1 \rangle, \dots, \langle 4+i, n \rangle\}) \cup$
 $(W_{p(1)} \text{ enumerated till now}) \cup$
 $(W_{p(0)} \text{ enumerated till now}).$
4. Let σ_{s+1} be an extension of $\tau_i \diamond \langle 4+i, 0 \rangle \diamond \langle 4+i, 1 \rangle \diamond \dots \diamond \langle 4+i, n \rangle$ such that $\text{content}(\sigma_{s+1}) = S$. Enumerate S in $W_{p(0)}$ and $W_{p(1)}$.
5. Go to stage $s + 1$.

End stage s .

Now we consider the following cases:

Case 1: All stages finish.

In this case let $L = W_{p(0)} = W_{p(1)} \in \mathcal{L}$. But \mathbf{M} on, $\bigcup_{s \in N} \sigma_s$, a text for L makes infinitely many mind changes (since only way infinitely many stages can exist is by the success of step (2b) infinitely often).

Case 2: Stage s , starts but does not finish.

In this case let $L_0 = W_{p(0)} = W_{p(2s)} \in \mathcal{L}$ and $L_1 = W_{p(1)} = W_{p(2s+1)} \in \mathcal{L}$. Note that L_0 and L_1 are infinitely different from each other. Let $T_0 = \tau_0 \diamond \langle 4, 0 \rangle \diamond \langle 4, 1 \rangle \diamond \dots$, and $T_1 = \tau_1 \diamond \langle 5, 0 \rangle \diamond \langle 5, 1 \rangle \diamond \dots$, where τ_0 and τ_1 are as defined in step (1), of stage s . Now T_0 is a text for L_0 and T_1 is a text for L_1 . However, \mathbf{M} converges to the same grammar on both T_0 and T_1 . Hence, \mathbf{M} does not **TextEx***-identify at least one of L_0 and L_1 .

The above cases imply that $\mathcal{L} \not\subseteq \mathbf{TxtEx}^*(\mathbf{M})$. ■

Corollary 41 *Suppose $i \in N$. For any L , define $L' = \{\langle j, x \rangle \mid j > 0 \wedge x \in L\} \cup \{\langle 0, x \rangle \mid x < 3i\}$. Let \mathcal{L} be as defined in the proof of Theorem 39. Let $\mathcal{L}' = \{L' \mid L \in \mathcal{L}\}$. Then $\mathcal{L}' \notin \mathbf{TxtEx}^*$.*

It is open at present if Theorem 39 can be extended to establish, $(\forall i \in N^+)[\mathbf{In}^{2i-1}\mathbf{TxtEx} - \mathbf{N}^i\mathbf{TxtEx}^* \neq \emptyset]$.

Theorem 42 $\mathbf{In}^*\mathbf{TxtEx} - \mathbf{N}^*\mathbf{TxtBc}^* \neq \emptyset$.

PROOF. For $D \in \mathbf{FIN}$, let $L_D = \{\langle 0, x \rangle \mid x > \max(D)\} \cup \{\langle j, x \rangle \mid j \geq 1 \wedge x \in D\}$. For $m \in N$, let $X_m = \{\langle 0, x \rangle \mid x > m\} \cup \{\langle j, x \rangle \mid j \geq 1 \wedge x \neq m\}$.

Let $\mathcal{L} = \{L_D \mid D \in \mathbf{FIN}\} \cup \{X_m \mid m \in N\}$.

It is easy to see that $\mathcal{L} \in \mathbf{In}^*\mathbf{TxtEx}$. Suppose by way of contradiction that $\mathbf{M} \mathbf{N}^*\mathbf{TxtBc}^*$ -identifies \mathcal{L} . Then we show how to obtain an $\mathbf{M}' \mathbf{TxtBc}^*$ -identifying $\mathcal{L}' = \{L \mid \text{card}(N - L) = 1\} \cup \mathbf{FIN}$. Let F be an effective mapping from \mathbf{SEQ} to \mathbf{SEQ} such that (a) $\sigma \subseteq \tau$ implies $F(\sigma) \subseteq F(\tau)$, and (b) $\text{content}(F(\sigma)) = \{\langle j, x \rangle \mid 1 \leq j \leq |\sigma| \wedge x \in \text{content}(\sigma)\} \cup \{\langle 0, x \rangle \mid x \leq |\sigma|\}$. Let f be a recursive function such that, for all i , $W_{f(i)} = \{x \mid \langle 1, x \rangle \in W_i\}$. Let $\mathbf{M}'(\sigma) = f(\mathbf{M}(F(\sigma)))$. It is easy to see that if $\mathbf{M} \mathbf{N}^*\mathbf{TxtBc}^*$ -identifies \mathcal{L} , then $\mathbf{M}' \mathbf{TxtBc}^*$ -identifies \mathcal{L}' . But this is not possible (Theorem 20). Thus $\mathcal{L} \notin \mathbf{N}^*\mathbf{TxtBc}^*$. ■

However,

Theorem 43 *For all $j \in N, a \in N \cup \{*\}$, $\mathbf{In}^*\mathbf{TxtEx}^a \subseteq \mathbf{Im}^j\mathbf{TxtEx}^{2a}$.*

PROOF. Suppose \mathbf{M} is given. We construct \mathbf{M}' such that $\mathbf{In}^*\mathbf{TxtEx}^a(\mathbf{M}) \subseteq \mathbf{Im}^j\mathbf{TxtEx}^{2a}(\mathbf{M}')$. Without loss of generality assume that \mathbf{M} is rearrangement independent. Let majority be a function such that, for all sequences g_1, g_2, \dots, g_l of grammars, $\text{majority}(g_1, g_2, \dots, g_l)$ is such that $W_{\text{majority}(g_1, g_2, \dots, g_l)} = \{x \mid \text{card}(\{i \mid 1 \leq i \leq l \wedge x \in W_{g_i}\}) > l/2\}$.

Let T be a j -imperfect information sequence for $L \in \mathbf{In}^*\mathbf{TxtEx}^a(\mathbf{M})$. If $a = *$, then let $k = 2j + 1$, else let k be such that $k \cdot a < (\frac{k}{2} - j)(2a + 1)$. (i.e., we choose k large enough to ensure the bound on the number of errors later). Let $n_0 < n_1 < n_3 < \dots < n_{2k-1}$ be such that the following is satisfied (where $S_i = \text{content}(T[n_i])$).

$$(\forall i < k)[\langle S_{2i+1} - S_{2i}, 2 \cdot n_{2i+1} \rangle \text{ is a stabilizing sequence for } \mathbf{M} \text{ on } \text{content}(T) - S_{2i}].$$

We claim that such n_0, \dots, n_{2k-1} exist. To see this let s be so large that $\text{content}(T) - \text{content}(T[s]) \subseteq L$. Now for any $n > s$, there exists an n' such that $\langle \text{content}(T[n']) - \text{content}(T[n]), 2n' \rangle$ is a stabilizing sequence for \mathbf{M} on $\text{content}(T) - \text{content}(T[n])$ (since \mathbf{M}

$\mathbf{In}^*\mathbf{TxtEx}^a$ -identifies L). Thus n_0, n_1, \dots can be easily picked to satisfy the requirements above.

Now we claim that $\text{majority}(\mathbf{M}(\langle S_1 - S_0, 2n_1 \rangle), \mathbf{M}(\langle S_3 - S_2, 2n_3 \rangle), \dots, \langle S_{2k-1} - S_{2k-2}, 2n_{2k-1} \rangle)$ is a grammar for a $2a$ -variant of L . To see this note that for each $i < k$, either $\mathbf{M}(\langle S_{2i+1} - S_{2i}, 2n_{2i+1} \rangle)$ is a grammar for an a -variant of L , or $S_{2i+1} - S_{2i}$ contains an element not in L (otherwise \mathbf{M} does not $\mathbf{In}^*\mathbf{TxtEx}^a$ -identify L). It follows that at least $k - j$ of $\mathbf{M}(\langle S_1 - S_0, 2n_1 \rangle), \mathbf{M}(\langle S_3 - S_2, 2n_3 \rangle), \dots, \mathbf{M}(\langle S_{2k-1} - S_{2k-2}, 2n_{2k-1} \rangle)$ are grammars for an a -variant of L . Thus, $\text{majority}(\mathbf{M}(\langle S_1 - S_0, 2n_1 \rangle), \mathbf{M}(\langle S_3 - S_2, 2n_3 \rangle), \dots, \mathbf{M}(\langle S_{2k-1} - S_{2k-2}, 2n_{2k-1} \rangle))$ is a grammar for a $2a$ -variant of L . (Note that k was chosen large enough to ensure the bound on the number of errors committed by majority) \blacksquare

Corollary 44 For all j, a , $\mathbf{In}^*\mathbf{TxtEx}^a \subseteq \mathbf{N}^j\mathbf{TxtEx}^{2a}$.

Having investigated noisy versus incomplete data, we now turn to the comparison of learning from incomplete versus imperfect information sequences. The reader should note that only approximate results are known; the exact relationship is open.

Theorem 45 $(\forall i \in \mathbf{N}^+)[\mathbf{In}^{3i-1}\mathbf{Ex} - \mathbf{Im}^{2i}\mathbf{Ex}^* \neq \emptyset]$.

PROOF. Consider the following collections of functions \mathcal{C}_i , for $i \in \mathbf{N}$:

$\mathcal{C}_i = \{f \in \mathcal{R} \mid \text{the following hold:}$

1. $\max(\{f(x) \mid x \in \mathbf{N}\}) < \infty$;
2. Let $S = \{x \mid \text{card}(\{y \mid f(y) \geq f(x)\}) \leq 3i\}$;

 - 2a. $\text{card}(S) = 3i$;
 - 2b. $(\forall x \in S)[\max(W_{f(x)}) < \infty \wedge \varphi_{\max(W_{f(x)})} = f]$.

(Note: S above represents the $3i$ points where f has the largest values.)

}

It is easy to see that $\mathcal{C}_i \in \mathbf{In}^{3i-1}\mathbf{Ex}$. We show that $\mathcal{C}_1 \notin \mathbf{Im}^2\mathbf{Ex}^*$. The proof can easily be generalized to establish $\mathcal{C}_i \notin \mathbf{Im}^{2i}\mathbf{Ex}^*$.

Suppose by way of contradiction, machine \mathbf{M} $\mathbf{Im}^2\mathbf{Ex}^*$ -identifies \mathcal{C}_1 . Then by the operator recursion theorem, there exists a recursive 1-1 increasing p , $p(0) > 1$, such that $W_{p(\cdot)}$ and $\varphi_{p(\cdot)}$ may be defined as follows.

Let $p(2) \in W_{p(0)}$. Let $\varphi_{p(2)}(0) = \varphi_{p(2)}(1) = \varphi_{p(2)}(2) = p(0)$. Let x_s denote the largest x such that $\varphi_{p(2)}(x)$ is defined before stage s . Let σ_0 be such that $\text{content}(\sigma_0) = \{(0, p(1)), (1, p(0)), (2, p(0))\}$. Go to stage 0.

Stage s

1. Enumerate $p(s+3)$ in $W_{p(1)}$.

For $1 \leq x \leq x_s$, let $\varphi_{p(s+3)}(x) = \varphi_{p(2)}(x)$.

Let $\varphi_{p(s+3)}(0) = \varphi_{p(s+3)}(x_s+1) = \varphi_{p(s+3)}(x_s+2) = p(1)$.

Let $x = x_s + 2$.

2. Dovetail steps (3), (4), and (5) until step (3) or step (4) succeeds. If step (3) succeeds (before step (4) does, if ever) then go to step (6); If step (4) succeeds (before step (3) does, if ever) then go to step (7).
3. Search for an $n > x_s + 3$ such that $\mathbf{M}(\sigma_s \diamond (x_s + 3, 0) \diamond (x_s + 4, 0) \diamond \cdots \diamond (n, 0)) \neq \mathbf{M}(\sigma_s)$.
4. Search for a $y \geq x_s + 3$ such that $\varphi_{\mathbf{M}(\sigma_s)}(y) \downarrow = 0$.
5. Go to substage 0.

Substage s'

Let $x = x + 1$, $\varphi_{p(s+3)}(x) = 0$.

Go to substage $s' + 1$.

End substage s' .

6. For n as found in step (3) let

$\varphi_{p(2)}(x) = 0$, $x_s < x \leq n$ and

$\sigma_{s+1} = \sigma_s \diamond (x_s + 3, 0) \diamond (x_s + 4, 0) \diamond \cdots \diamond (n, 0) \diamond (x_s + 1, 0) \diamond (x_s + 2, 0)$.

Go to stage $s + 1$.

7. For y as found in step (4) let

$\varphi_{p(2)}(x) = 1$, $x_s < x \leq y$.

$\sigma_{s+1} = \sigma_s \diamond (x_s + 1, 1) \diamond \cdots \diamond (y, 1)$.

Go to stage $s + 1$.

End Stage s

Now consider the following cases:

Case 1: All stages finish.

In this case, let $f = \varphi_{p(2)} \in \mathcal{C}_1$. Let $G = \bigcup_s \sigma_s$. Clearly, G is a 2-imperfect information sequence for f ($f(0)$ has been changed from $p(0)$ to $p(1)$ in the information sequence G).

Case 1a: \mathbf{M} on G changes its mind infinitely often.

In this case, \mathbf{M} does not $\mathbf{Im}^2\mathbf{Ex}^*$ -identify f .

Case 1b: \mathbf{M} on G converges.

In this case, the only way in which infinitely many stages can exist is by execution of step (7) infinitely often. But, then, $\varphi_{\mathbf{M}(G)}$ is infinitely different from f . Thus, \mathbf{M} does not $\mathbf{Im}^2\mathbf{Ex}^*$ -identify f .

Case 2: Stage s starts but does not finish.

In this case, let $f = \varphi_{p(s+3)} \in \mathcal{C}_1$. Now, on $G = \sigma_s \diamond (x_s + 3, 0) \diamond (x_s + 4, 0) \diamond \dots$, a 2-imperfect information sequence for f , \mathbf{M} converges to $\mathbf{M}(\sigma_s)$, and, for all but finitely many x , $\varphi_{\mathbf{M}(\sigma_s)}(x) \neq 0$. Thus, \mathbf{M} does not $\mathbf{Im}^2\mathbf{Ex}^*$ -identify f .

The above cases imply that \mathbf{M} does not $\mathbf{Im}^2\mathbf{Ex}^*$ -identify \mathcal{C}_1 . ■

Above proof can be easily modified to prove Theorem 46 below. We omit the details.

Theorem 46 $(\forall i, j \in N)[\mathbf{In}^{3i-1}\mathbf{Ex} - \mathbf{Im}^{2i}\mathbf{Bc}^j \neq \emptyset]$.

Theorem 47 below suggests that Theorem 45 is nearly ‘optimal.’ As already mentioned, the exact relationship between identification criteria for incomplete and imperfect information sequence is still open.

Theorem 47 $(\forall i, j \in N)[\mathbf{In}^{4i}\mathbf{Ex}^j \subseteq \mathbf{Im}^{2i}\mathbf{Ex}^{2j}]$.

PROOF. Suppose machine \mathbf{M} $\mathbf{In}^{4i}\mathbf{Ex}^j$ -identifies \mathcal{C} . Without loss of generality, let \mathbf{M} be rearrangement independent (thus we can assume that \mathbf{M} is given a finite set and length as its input). We construct \mathbf{M}' which $\mathbf{Im}^{2i}\mathbf{Ex}^{2j}$ -identifies \mathcal{C} . Let G be a $2i$ -imperfect information sequence for $f \in \mathcal{C}$. We assume that for no x, y and z such that $y \neq z$, both (x, y) and (x, z) appear in $\text{content}(G)$ (we can assume this since from a $2i$ -imperfect information sequence for any total function f , we can effectively construct a $2i$ -imperfect information sequence satisfying this property; F as described in Theorem 38 does this).

Let $n_1 < n_2 < \dots < n_{2i+2}$, $S, P, x_1, x_2, \dots, x_{j \cdot \text{card}(P)}, y_1, y_2, \dots, y_{j \cdot \text{card}(P)}$ (lexicographically least, if such exist) be such that $\text{card}(S) \leq i$ and conditions (A) to (D) below are satisfied.

(A) $(\forall k \mid 1 \leq k \leq i+1)(\forall S' \subseteq \text{content}(G[n_{2k-1}]) \mid \text{card}(S') \leq i) [\langle \text{content}(G[n_{2k}]) - S - S', 2 \cdot n_{2k} \rangle$ is a stabilizing sequence for \mathbf{M} on $\text{content}(G) - S - S'$].

(B) $P = \{\mathbf{M}(\langle \text{content}(G[n_{2k}]) - S - S', 2 \cdot n_{2k} \rangle) \mid (1 \leq k \leq i+1) \wedge (S' \subseteq \text{content}(G[n_{2k-1}])) \wedge (\text{card}(S') \leq i)\}$.

(C) $\{x \mid (\exists k, l \in P)[\varphi_k(x) \downarrow \neq \varphi_l(x) \downarrow]\} \subseteq \{x_1, x_2, \dots, x_{j \cdot \text{card}(P)}\}$.

(D) For all $k \in P$, $\text{card}(\{l \mid \varphi_k(x_l) \neq y_l, 1 \leq l \leq j \cdot \text{card}(P)\}) \leq j$.

Let Prog be a recursive function such that

$\varphi_{Prog(P, x_1, \dots, y_1, \dots)}(x)$

if $x = x_k$ for some k

then output y_k

else

search for $p \in P$ such that $\varphi_p(x) \downarrow$

output $\varphi_p(x)$ for first such p found

endif

end

Clearly, if there exist $n_1 < n_2 < \dots < n_{2i+2}$, $S, P, x_1, \dots, y_1, \dots$ such that (A) to (D) are satisfied then \mathbf{M}' can find the lexicographically least such values (in the limit) and thus output, on $G, Prog(P, x_1, \dots, y_1, \dots)$ in the limit.

We first argue that there exist such $n_1, n_2, \dots, S, \dots, P, x_1, \dots, y_1, \dots$. We then argue for such $n_1, \dots, S, P, x_1, \dots, y_1, \dots, \varphi_{Prog(P, x_1, \dots, y_1, \dots)} =^{2j} f$. This would prove the theorem.

We first show that such $S_1, \dots, n_1, \dots, P, x_1, \dots, y_1, \dots$ exist. Consider any $2i$ -imperfect information sequence G for f , such that for no x, y, z such that $y \neq z$ does (x, y) and (x, z) both are in $\text{content}(G)$. This implies that there are at most i elements (x, y) in the information sequence such that $f(x) \neq y$. Let $S = \{(x, y) \mid (x, y) \in \text{content}(G) \wedge f(x) \neq y\}$. Now since $\mathbf{M} \mathbf{In}^{4i} \mathbf{Ex}^j$ -identifies f , for all S' of cardinality at most i , \mathbf{M} on any information sequence for $\text{content}(G) - S - S'$ converges to a program which computes a j -variant of f . Thus there exist appropriate $n_1, \dots, P, x_1, \dots, y_1, \dots$ satisfying (A) to (D).

We now show that for such $S, n_1, \dots, P, x_1, \dots, y_1, \dots, \varphi_{Prog(P, x_1, \dots, y_1, \dots)} =^{2j} f$. Clearly, for any $p \in P$, $\text{card}(\{x \mid \varphi_{Prog(P, x_1, \dots, y_1, \dots)}(x) \neq \varphi_p(x) \downarrow\}) \leq j$. Thus our proof will be complete if we can prove that there is at least one program in P which computes a j -variant for f . Now for any k such that $1 \leq k \leq i + 1$, $S' \subseteq \text{content}(G[n_{2k-1}])$ such that $\text{card}(S') \leq i$, we have $\langle \text{content}(G[n_{2k}]) - S - S', 2 \cdot n_{2k} \rangle$ is a stabilizing sequence for \mathbf{M} on $\text{content}(G) - S - S'$. Thus, either $\{\mathbf{M}(\langle \text{content}(G[n_{2k}]) - S - S', 2 \cdot n_{2k} \rangle) \mid S' \subseteq \text{content}(G[n_{2k-1}]) \wedge \text{card}(S') \leq i\}$ contains a program which computes a j -variant of f or there exists a noisy element in $\text{content}(G[n_{2k}]) - \text{content}(G[n_{2k-1}])$ (otherwise, \mathbf{M} does not $\mathbf{In}^{4i} \mathbf{Ex}^j$ -identify f). Since there are at most i noisy elements in G , there must exist a k such that $\{\mathbf{M}(\langle \text{content}(G[n_{2k}]) - S - S', 2 \cdot n_{2k} \rangle) \mid S' \subseteq \text{content}(G[n_{2k-1}]) \wedge \text{card}(S') \leq i\}$ contains a program which computes a j -variant for f . ■

Slight modification of above proof can be used to show that

Theorem 48 $(\forall i, j \in N)[\mathbf{In}^{4i+1} \mathbf{Ex}^j \subseteq \mathbf{Im}^{2i+1} \mathbf{Ex}^{2j}]$.

Theorems 47 and 48 can actually be improved slightly to give

Theorem 49 $(\forall i, j \in N : i \geq 1)(\forall k \in \{0, 1\})[\mathbf{In}^{\max(\{3i, 4i-2\})+k} \mathbf{Ex}^j \subseteq \mathbf{In}^{2i+k} \mathbf{Ex}^{j+\min(\{j, 2i-1\})}]$.

We omit the proof of the above theorem. Note that the language learning analogue of the proof of Theorem 47 does not work since there is no corresponding procedure for *prog* and there is no limit effective way to check analogue of conditions (C) and (D).

Theorem 50 $\mathbf{In}^* \mathbf{Ex} - [\mathbf{In}^* \mathbf{Ex}^* \cup \bigcup_{j \in N} \mathbf{In}^* \mathbf{Bc}^j] \neq \emptyset$.

PROOF. For any $f \in \mathcal{R}$, we describe a function f' as follows:

Let $p_f = \text{MinProg}(f)$.

For $j < p_f$, let $\text{err}_j = \min(\{x \mid \varphi_j(x) \neq f(x)\})$.

For $x < \langle p_f, \langle \text{err}_0, \text{err}_1, \text{err}_2, \dots, \text{err}_{p_f-1} \rangle \rangle$, let $f'(\langle 0, x \rangle) = 1$.

For $x \geq \langle p_f, \langle \text{err}_0, \text{err}_1, \text{err}_2, \dots, \text{err}_{p_f-1} \rangle \rangle$, let $f'(\langle 0, x \rangle) = 0$.

For $k \geq 1, x \in N$, let $f'(\langle k, x \rangle) = f(x)$.

Let $\mathcal{C} = \{f' \mid f \in \mathcal{R}\}$. Now proceeding in a way similar to the proof of Theorem 34 it can be shown that $\mathcal{C} \in \mathbf{In}^* \mathbf{Ex} - [\mathbf{In}^* \mathbf{Ex}^* \cup \bigcup_{j \in N} \mathbf{In}^* \mathbf{Bc}^j]$. We leave the details to the reader. ■

4 Conclusions

In this paper we considered the effect of three types of inaccuracies in the input data. We showed several hierarchies and tradeoff results based on the three different kinds of inaccuracies. It is open to exactly characterize the relationship between the identification criteria based on imperfect and incomplete data (for both function and language identification). We showed partial, nearly optimal results, in this direction. It is also open whether $\mathbf{N}^* \mathbf{TxtEx} - \mathbf{In}^1 \mathbf{TxtBc}^*$ is empty or not.

5 Acknowledgements

We would like to thank Arun Sharma for helpful discussions and comments. Preliminary version of this paper appeared in [FJ89].

References

- [AS83] D. Angluin and C. Smith. A survey of inductive inference: Theory and methods. *Computing Surveys*, 15:237–289, 1983.
- [Bar74] J. M. Barzdin. Two theorems on the limiting synthesis of functions. *In Theory of Algorithms and Programs, Latvian State University, Riga*, 210:82–88, 1974. In Russian.
- [BB75] L. Blum and M. Blum. Toward a mathematical theory of inductive inference. *Information and Control*, 28:125–155, 1975.
- [Blu67] M. Blum. A machine-independent theory of the complexity of recursive functions. *Journal of the ACM*, 14:322–336, 1967.
- [Cas74] J. Case. Periodicity in generations of automata. *Mathematical Systems Theory*, 8:15–32, 1974.
- [Cas88] J. Case. The power of vacillation. In D. Haussler and L. Pitt, editors, *Proceedings of the Workshop on Computational Learning Theory*, pages 133–142. Morgan Kaufmann Publishers, Inc., 1988. Expanded in [Cas92].
- [Cas92] J. Case. The power of vacillation in language learning. Technical Report 93-08, University of Delaware, 1992. Expands on [Cas88]; journal article under review.
- [CL82] J. Case and C. Lynes. Machine inductive inference and language identification. In M. Nielsen and E. M. Schmidt, editors, *Proceedings of the 9th International Colloquium on Automata, Languages and Programming*, pages 107–115. Springer-Verlag, 1982. Lecture Notes in Computer Science 140.
- [CS83] J. Case and C. Smith. Comparison of identification criteria for machine inductive inference. *Theoretical Computer Science*, 25:193–220, 1983.
- [FJ89] M. A. Fulk and S. Jain. Learning in the presence of inaccurate information. In R. Rivest, D. Haussler, and M. K. Warmuth, editors, *Proceedings of the Second Annual Workshop on Computational Learning Theory, Santa Cruz, California*, pages 175–188. Morgan Kaufmann Publishers, Inc., August 1989.
- [FJO94] M. Fulk, S. Jain, and D. Osherson. Open problems in systems that learn. *Journal of Computer and System Sciences*, 49(3):589–604, December 1994.

- [Ful85] M. Fulk. *A Study of Inductive Inference Machines*. PhD thesis, SUNY at Buffalo, 1985.
- [Ful90] M. Fulk. Prudence and other conditions on formal language learning. *Information and Computation*, 85:1–11, 1990.
- [Gol67] E. M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.
- [KW80] R. Klette and R. Wiehagen. Research in the theory of inductive inference by GDR mathematicians – A survey. *Information Sciences*, 22:149–169, 1980.
- [MY78] M. Machtey and P. Young. *An Introduction to the General Theory of Algorithms*. North Holland, New York, 1978.
- [OSW82] D. Osherson, M. Stob, and S. Weinstein. Ideal learning machines. *Cognitive Science*, 6:277–290, 1982.
- [OSW84] D. Osherson, M. Stob, and S. Weinstein. Learning theory and natural language. *Cognition*, 17:1–28, 1984.
- [OSW86] D. Osherson, M. Stob, and S. Weinstein. *Systems that Learn, An Introduction to Learning Theory for Cognitive and Computer Scientists*. MIT Press, Cambridge, Mass., 1986.
- [OW82a] D. Osherson and S. Weinstein. Criteria of language learning. *Information and Control*, 52:123–138, 1982.
- [OW82b] D. Osherson and S. Weinstein. A note on formal learning theory. *Cognition*, 11:77–88, 1982.
- [Pin79] S. Pinker. Formal models of language learning. *Cognition*, 7:217–283, 1979.
- [Rog58] H. Rogers. Gödel numberings of partial recursive functions. *Journal of Symbolic Logic*, 23:331–341, 1958.
- [Rog67] H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, New York, 1967. Reprinted, MIT Press 1987.
- [SR86] G. Schäfer-Richter. Some results in the theory of effective program synthesis - learning by defective information. *Lecture Notes in Computer Science*, 215:219–225, 1986.

- [WC80] K. Wexler and P. Culicover. *Formal Principles of Language Acquisition*. MIT Press, Cambridge, Mass, 1980.
- [Wex82] K. Wexler. On extensional learnability. *Cognition*, 11:89–95, 1982.