Modelling Binary Classification with Computability Theory

Kerstin Karen Seidel

Universitätsdissertation
zur Erlangung des akademischen Grades

doctor rerum naturalium
(Dr. rer. nat.)

in der Wissenschaftsdisziplin
Theoretische Informatik

eingereicht an der
Digital-Engineering-Fakultät
der Universität Potsdam

Datum der Disputation: 11. November 2021

Betreuer

Prof. Dr. Tobias Friedrich
Hasso Plattner Institute, University of Potsdam

Gutachter

Prof. Dr. Vasco Brakka
Bundeswehr University Munich

Prof. Dr. Ekaterina Fokina
Technical University of Vienna

Published online on the
Publication Server of the University of Potsdam:
https://doi.org/10.25932/publishup-52998
https://nbn-resolving.org/urn:nbn:de:kobv:517-opus4-529988

Abstract

We investigate models for incremental binary classification, an example for supervised online learning. Our starting point is a model for human and machine learning suggested by E. M. Gold, [Gol67].

In the first part, we consider incremental learning algorithms that use all of the available binary labeled training data in order to compute the current hypothesis. For this model, we observe that the algorithm can be assumed to always terminate and that the distribution of the training data does not influence learnability. This is still true if we pose additional *delayable* requirements that remain valid despite a hypothesis output delayed in time [KP16]. Additionally, we consider the non-delayable requirement of *consistent* learning. Our corresponding results underpin the claim for delayability being a suitable structural property to describe and collectively investigate a major part of learning success criteria.

Our first theorem states the pairwise implications or incomparabilities between an established collection of delayable learning success criteria, the so-called complete map. Especially, the learning algorithm can be assumed to only change its last hypothesis in case it is inconsistent with the current training data. Such a learning behaviour is called *conservative* [Ang80].

By referring to learning functions, we obtain a *hierarchy* [Bār74], [CS83] of approximative learning success criteria. Hereby we allow an increasing finite number of *errors* of the hypothesized concept by the learning algorithm compared with the concept to be learned.

Moreover, we observe a *duality* depending on whether *vacillations* between infinitely many different correct hypotheses are still considered a successful learning behaviour. This contrasts the vacillatory hierarchy for learning from solely positive information [Cas99].

We also consider a hypothesis space located between the two most common hypothesis space types in the nearby relevant literature and provide the complete map.

In the second part, we model more efficient learning algorithms. These update

their hypothesis referring to the current datum and without direct regress to past training data. We focus on iterative (hypothesis based) [R W76] and **BMS** (state based) [Car+07] learning algorithms.

Iterative learning algorithms use the last hypothesis and the current datum in order to infer the new hypothesis. Past research analyzed, for example, the above mentioned pairwise relations between delayable learning success criteria when learning from purely positive training data, see [LZ91], [CM08a], [CK10], [Jai+16].

We compare delayable learning success criteria with respect to iterative learning algorithms, as well as learning from either exclusively positive or binary labeled data. The existence of concept classes that can be learned by an iterative learning algorithm but not in a conservative way had already been observed [JLZ07a], showing that conservativeness is restrictive. An additional requirement arising from cognitive science research is *U-shapedness* [SS82], stating that the learning algorithm does diverge from a correct hypothesis. We show that forbidding *U-shapes* also restricts iterative learners from binary labeled data.

In order to compute the next hypothesis, **BMS** learning algorithms refer to the currently observed datum and the actual state of the learning algorithm. For learning algorithms equipped with an infinite amount of states, we provide the *complete map*.

A learning success criterion is semantic if it still holds, when the learning algorithm outputs other parameters standing for the same classifier. Syntactic (non-semantic) learning success criteria, for example conservativeness and syntactic non-U-shapedness, restrict **BMS** learning algorithms. For proving the equivalence of the syntactic requirements, we refer to witness-based learning processes [KS16]. In these, every change of the hypothesis is justified by a later on correctly classified witness from the training data. Moreover, for every semantic delayable learning requirement, iterative and **BMS** learning algorithms are equivalent. In case the considered learning success criterion incorporates syntactic non-U-shapedness, **BMS** learning algorithms can learn more concept classes than iterative learning algorithms.

The proofs are combinatorial, inspired by investigating formal languages or employ results from computability theory, such as infinite recursion theorems (fixed point theorems) [Köt09].

Zusammenfassung

Wir untersuchen Modelle für inkrementelle binäre Klassifikation, ein Beispiel für überwachtes online Lernen. Den Ausgangspunkt bildet ein Modell für menschliches und maschinelles Lernen von E. M. Gold, [Gol67].

Im ersten Teil untersuchen wir inkrementelle Lernalgorithmen, welche zur Berechnung der Hypothesen jeweils die gesamten binär gelabelten Trainingsdaten heranziehen. Bezogen auf dieses Modell können wir annehmen, dass der Lernalgorithmus stets terminiert und die Verteilung der Trainingsdaten die grundsätzliche Lernbarkeit nicht beeinflusst. Dies bleibt bestehen, wenn wir zusätzliche Anforderungen an einen erfolgreichen Lernprozess stellen, die bei einer zeitlich verzögerten Ausgabe von Hypothesen weiterhin zutreffen, [KP16].

Weitherin untersuchen wir nicht verzögerbare konsistente Lernprozesse. Unsere Ergebnisse bekräftigen die Behauptung, dass Verzögerbarkeit eine geeignete strukturelle Eigenschaft ist, um einen Großteil der Lernerfolgskriterien zu beschreiben und gesammelt zu untersuchen.

Unser erstes Theorem klärt für dieses Modell die paarweisen Implikationen oder Unvergleichbarkeiten innerhalb einer etablierten Auswahl verzögerbarer Lernerfolgskriterien auf. Insbesondere können wir annehmen, dass der inkrementelle Lernalgorithmus seine Hypothese nur dann verändert, wenn die aktuellen Trainingsdaten der letzten Hypothese widersprechen. Ein solches Lernverhalten wird als *konservativ*, [Ang80], bezeichnet.

Ausgehend von Resultaten über Funktionenlernen erhalten wir eine strikte *Hierarchie* von approximativen Lernerfolgskriterien [BP73].

Weiterhin ergibt sich eine *Dualität* abhängig davon, ob das Oszillieren zwischen korrekten Hypothesen als erfolgreiches Lernen angesehen wird. Dies steht im Gegensatz zur oszillierenden Hierarchie, wenn der Lernalgorithmus von ausschließlich positiven Daten lernt, [Cas99].

Auch betrachten wir einen Hypothesenraum, der einen Kompromiss zwischen den beiden am häufigsten in der naheliegenden Literatur vertretenen Arten von Hypothesenräumen darstellt.

Im zweiten Teil modellieren wir effizientere Lernalgorithmen. Diese aktualisieren ihre Hypothese ausgehend vom aktuellen Datum, jedoch ohne Zugriff auf die zurückliegenden Trainingsdaten. Wir konzentrieren uns auf iterative (hypothesenbasierte) [R W76] and **BMS** (zustandsbasierte) [Car+07] Lernalgorithmen.

Iterative Lernalgorithmen nutzen ihre letzte Hypothese und das aktuelle Datum, um die neue Hypothese zu berechnen. Die bisherige Forschung klärt beispielsweise die oben erwähnten paarweisen Vergleiche zwischen den verzögerbaren Lernerfolgskriterien, wenn von ausschließlich positiven Trainingsdaten gelernt wird, siehe [LZ91], [CM08a], [CK10], [Jai+16].

Wir vergleichen verzögerbare Lernerfolgskriterien bezogen auf iterative Lernalgorithmen, sowie das Lernen von aussschließlich positiver oder binär gelabelten Daten. Bereits bekannt war die Existenz von Konzeptklassen, die von einem iterativen Lernalgorithmus gelernt werden können, jedoch nicht auf eine konservative Weise, [JLZ07a]. *U-shapedness* [SS82] ist ein in den Kognitionswissenschaften beobachtetes Phänomen, demzufolge der Lerner im Lernprozess von einer bereits korrekten Hypothese divergiert. Wir zeigen, dass iterative Lernalgorithmen auch durch das Verbieten von U-Shapes eingeschränkt werden.

Zur Berechnung der nächsten Hypothese nutzen **BMS**-Lernalgorithmen ergänzend zum aktuellen Datum den aktuellen Zustand des Lernalgorithmus. Für Lernalgorithmen, die über unendlich viele mögliche Zustände verfügen, leiten wir alle paarweisen Implikationen oder Unvergleichbarkeiten innerhalb der etablierten Auswahl verzögerbarer Lernerfolgskriterien her.

Ein Lernerfolgskriterium ist semantisch, wenn es weiterhin gilt, falls im Lernprozess andere Parameter ausgegeben werden, die jeweils für die gleichen Klassifikatoren stehen. Syntaktische (nicht-semantische) Lernerfolgskriterien, beispielsweise Konservativität und syntaktische Non-U-Shapedness, schränken **BMS**-Lernalgorithmen ein. Um die Äquivalenz der syntaktischen Lernerfolgskriterien zu zeigen, betrachten wir witness-based Lernprozesse, [KS16]. In diesen wird jeder Hypothesenwechsel durch einen später korrekt klassifizierten Zeugen in den Trainingsdaten gerechtfertig. Weiterhin sind iterative und **BMS**-Lernalgorithmen für die semantischen verzögerbaren Lernerfolgskriterien jeweils *äquivalent*. Ist syntaktische Non-U-Shapedness Teil des Lernerfolgskriteriums, sind **BMS**-Lernalgorithmen mächtiger als iterative Lernalgorithmen.

Die Beweise sind kombinatorisch, angelehnt an Untersuchungen zu formalen Sprachen oder nutzen Resultate aus dem Gebiet der Berechenbarkeitstheorie, beispielsweise unendliche Rekursionstheoreme (Fixpunktsätze) [Köt09].

Acknowledgments

First of all, I am grateful to Tobias Friedrich and Timo Koetzing for providing me with the necessary financial support to pursue these studies. I also owe them a debt of gratitude for always giving advice without enforcing a problem solution and doing their best to facilitate this kind of research. In this context I thank my mentor Felix Naumann as well for his encouragement.

Moreover, I am grateful to other researchers and PhD students I met on this journey which have been understanding and inspiring. In particular, André Nies for inviting me to his excellent research environment and pointing out the idea to study linear functions, as well as Noam Greenberg for his encouragement and the possibility to present parts of my work. Very importantly, I also thank Frank Stephan and Sanjay Jain for their constant support, availability and helpful feedback regardless geodesic distance. I also thank Ziyuan Gao and Rupert Hölzl for helping me around in Singapore and being such trustworthy collaborators. Moreover, I am grateful to Thomas Zeugmann and Sandra Zilles for pointers to prior research. I was also very pleased to meet Christine Gassner, Vasco Brattka, Arno Pauly and other researchers in Computability Theory.

Furthermore, I am also grateful to Eugen Hellmann and Peter Scholze for helpful feedback regarding isolated parts of the proof for the learnability of half-spaces. Also feedback of Simon Wietheger and initial discussions with Vanja Doskoč and Armin Wells have been helpful there. Moreover, I thank Ardalan Khazraei for interesting discussions and his essential contributions to Theorem 4.4 as well as the learnability of half-spaces.

Substantially, I also thank Martin Aschenbach who initiated parts of this study with his master thesis supervised by Timo Kötzing. I worked the initial results and proof ideas up by correcting proofs and embedding it into a broader research landscape with further results. Most of the ambitious results in this thesis would not exist without the infrequent but valuable discussions with Timo Kötzing and I am very much obliged for this.

I am as well grateful to the reviewers and committee members for the time they invest into the evaluation of my efforts.

Besides working on this thesis, I also had the opportunity to create a related MOOC on the OpenHPI platform. For this, I owe thanks to Tobias Friedrich for the idea and support, as well as Simon Wietheger for helping with the slides and in the forum. Moreover, I am grateful to Martin Taraz for implementing the machine learning algorithms and deep learning architectures I suggested.

I am also grateful for the opportunities to participate in teaching, which I enjoy a lot. Moreover, I am glad that my work and discussions with members of the research group encouraged quite some publications.

In addition, I am grateful to Katrin Heinrich, Timo Kötzing, Martin Krejca, Anna Melnichenko, Martin Schirneck, Maximillian Katzmann, Philipp Fischbeck, Pascal Lenzner, Stefan Neubert, Vanja Doskoč and others who helped me with administrative and other duties.

Moreover, I thank Aurelién Garivier, Claire Vernade, Melanie Schirmer, Bernhard Renard, Alice Wittig, Michael Meyer, Eva Müller-Hill, David Kollosche, Anna Melnichenko, Thomas Bläsius, Maximillian Katzmann, Philipp Fischbeck, Timo Kötzing, Gregor Lagodzinski, Louise Molitor, Andreas Goebel, Martin Krejca, Katrin Casel, Christopher Weyand, Francesco Quinzan, Davis Issac, Ágnes Cseh, Ziena Zeif, Aikaterini Niklanovits, Sarel Cohen, Omri Ben-Eliezer, Keyulu Xu, Jingling Li, Jure Leskovec, Otto Kißig, Martin Taraz, Ankhit Chauhan, Amy Siu and many others for mostly very pleasant scientific and non-scientific distractions from this studies. I hope to pursue some of the related topics and activities further.

A lot of people that I already mentioned helped me with their feedback and the possibility to present my work in Wellington, Potsdam, Hiddensee, Kyoto, Aachen, Singapore, Cambridge and Ghent. Also discussions with Steve Hanneke, Matthew de Brecht, Christoph Lippert, Matthias Kirchler and Bert Arnrich with his Research Group helped me to progress.

I am also grateful to the many other people at Hasso-Plattner-Institute and other Research Facilities, who supported me.

Last but not least, I am very grateful to my family and friends for their understanding, support and encouragement. In particular, I owe gratitude to my parents, grandparents, sister, parents-in-law, grandmother-in-law, siblings-in-law for bearing difficult phases and forgiving being so absent sometimes. The person I owe most to is Philipp Schlicht, who is not only supportive in any imaginable way but also a great dad for our son. Thanks, especially to both of you, for always reaching out!

Contents

1 Introduction

Humans, other life forms and in an rapidly growing amount also machines utilize prior knowledge, in order to generalize to and succeed in unseen situations. In the current era of machine learning, heuristics play a crucial role in order to design and improve algorithms that employ the growing amount of available data as well as impressive computing abilities. The engineering practices massively used in the technologies that are interweaved with society are far from understood. Research in *computational learning theory* discusses concrete initial mathematical models and algorithms for this complex subject, [Wig19].

We investigate models for *binary classification*, a special case of supervised machine learning, [GBC16]. As an illustrating example, let us assume we want to verify formally whether there is a learning algorithm for the collection of email spam filters. Here we are not concerned with the definition of spam as discussed elsewhere, for example [Cor08]. Many different machine learning algorithms have been applied to the challenging problem of designing spam filters, see [Dad+19] for a recent publication. The details of the algorithms used by email providers and companies focusing on spam filtering are not publicly available. We aim to explain our abstract terminology with this sample application. However, our models also suit this example well as they can take into account that in email spam filtering false positive and false negative predictions are not treated equally.

Let us for the start leave a lot of challenging details aside. A suitable online learning algorithm successively experiences more and more emails labeled to be spam or not spam by higher-order knowledge and the specific user. We call this sequence of emails the *training data*. Every time the learning algorithm observes a new labeled email, it outputs a classifier that also predicts whether future emails are spam or not. The learner is successful, if after some time the hypothesized classifier generalizes well with respect to the user's taste. Hence, we seek a learning algorithm that for every user succeeds on the following task: When comprising more and more data into the hypotheses, the sequence of suggested classifiers converges to an optimal one for this user. As we are not in

control of the user's taste, the evolution of higher-order knowledge or the order in which the emails arrive, the algorithm has to succeed for all possible email filters and for every distribution of the data.

We can think of an email as a sequence of characteristics, for example symbols, words or collections of the latter and other more sophisticated features. (Note that a reasonable feature extraction is at the core of machine learning.) Encoding transforms the target features into numeric values or vectors. Hence, for each email we obtain a sequence of the encoded features. Essentially, this *numerical feature array* or *tensor* is the input for the machine learning algorithm. This approach also covers the emerging field of Graph Neural Networks by additionally providing the corresponding graph structure, [Sca+08]. The edges represent interactions or similarities between the feature tensors, [Bro+17].

For our purposes, we encode the feature array and possibly more structural information associated to an email into *a single number*. This number represents the underlying email. The above encoding procedure is not specific to email classification but applies to several machine learning applications with their respective inputs, for example pattern recognition, [Bis06]. The actual encoding techniques in electronic devices differ.

In *inductive inference* the classifier is often given by an algorithm recursively enumerating all numbers representing spam emails, [Odi92]. With this mind-set, one can think of some email spam detector to be given by a set of rules such that every email derived by them is classified as spam.

On the other hand for real-world machine learning scenarios, one could define the hard classifier to be a computable function mapping numbers representing spam emails to one and all other emails to zero. A classifier of the second kind can easily be transformed into a classifier of the first kind. The second definition is more restrictive and hence, if not stated differently, we stick to *recursively enumerable sets*.

Following the terminology in [SB14], our concept class to be learned is the collection of all email spam detectors. Due to the connections of inductive inference to grammatical inference, we stay closer to the terminology in [Jai+99], where a concept class is referred to as a *language class*. With respect to our running example a concept or language corresponds to a spam filter for one specific user. Due to the ambiguity of some terminology when referring to applied machine learning, we stick to concepts in the introduction. However, all eligible classifiers output by the learning algorithm are required to be elements

of the so-called hypothesis space. The general hypothesis space of recursively enumerable sets is called *W-hypothesis space*, [RC94]. In our example each classifier is given by a computer program corresponding to an enumeration of all the spam emails. The W-hypothesis space comprises all other common hypothesis spaces.

As noted above, for a finite amount of data the learner produces a hypothesis that corresponds to a classifier. In the limit the hypothesized classifiers are supposed to *minimize the error* with respect to the concept to be learned. There are different formalizations of this requirement. Successful PAC-learning requires that for enough available data with high probability the error becomes small, [Val84]. The sample complexity associated to a given probability and error has been settled, [Han16].

We build on the model suggested by E. M. Gold, [Gol67], following which the learner is successful if it *eventually* settles on a *correct* classifier. For our running example a correct classifier is an email spam filter that perfectly reflects the users taste. In Section 3.3 we study versions of Gold's model, that consider approximations, [Cas99]. In addition also ideas from probability theory play an important role and have been incorporated into Gold's model by L. Pitt, [Pit84], and G. Barmpalias and F. Stephan, [BS17]. These are interesting directions to pursue further, especially in the context of learning algorithms not relying on all data available.

Fundamental machine learning algorithms for supervised binary classification like the perceptron [Ros58] and support vector machines [BGV92] use *linear classifiers* as hypothesis space. With a fixed computable kernel function even more learning tasks can be reduced to classifying with half-spaces. This is one of many examples for more uniform hypothesis spaces of classifiers, [LZZ08]. If we restrict ourselves to linear classifiers, we might ask whether the concept class of spam filters is learnable with the uniform hypothesis space of half-spaces.

The learnability of linear predictors has been investigated with respect to other learning models and respective research questions, see for example [SB14], [Sha15] and [Gao+17]. The concept class of linear classifiers forms a uniform hypothesis space, because a computer can enumerate the parameters corresponding to a linear inequality. In the spirit of *Occams Razor*, we obtain a successful learning algorithm when unbiasedly outputting the first enumerated half-space consistent with all of the training data.

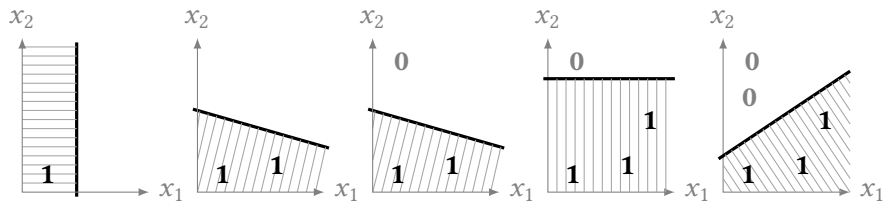

Figure 1.1: Example Learning Process when the hypotheses correspond to half-spaces.

This successful learning strategy is referred to as *learning by enumeration*, [Gol67]. If we pose a realizability assumption, [SB14], we obtain the learnability of the sub concept class of email spam filters. The learning by enumeration strategy works for every concept class assumed to be a subset of a uniform hypothesis space. Questions regarding efficiency are not considered here. We refer the interested reader to [Köt09].

A classical perceptron or support vector machine computes the current hypothesis by using all of the available data. We refer to learning algorithms with this property as *full-information* or *full-batch* learners and analyze them in Part I. Our results can also be found in [AKS18].

In Chapter 2 we follow [Gol67] and start with formalizing binary classification with full-batch learning algorithms by referring to Turing machines, one of the most fundamental mathematical models for computer algorithms, [Odi92]. As most computer programs rely on recursion, we do not require the learning algorithm to produce an output on every possible input. In the respective Section 2.2, the formalized notions include the data stream, learning algorithm, learning success criteria and a corresponding notation that allows to state how different models relate to each other, [Köt09]. We also give the definitions of an established collection of additional requirements that can be incorporated into the learning success criterion, [KP14], [KS16] and [Jai+16].

Thereafter, in Section 2.3, we observe that every learnable concept class can be learned by a total learner, namely a learning algorithm that terminates its computation on every input. Moreover, the underlying distribution of the data does not matter as we can provably assume that the data is presented in some canonical order. These observations hold for all learning success criteria invariant with respect to a time delayed output of the hypothesis. For example,

the learning algorithm will still converge to the optimal classifier if the hypothesis is output 2 time units later due to some technical issues. Such learning success criteria are called *delayable*, [KP16]. We prove that the above observations about the totality of the learning algorithm and a canonical presentation of the data hold for all delayable learning success criteria. In contrast, we show that these observations are no longer true when we require the output hypothesis of the learning algorithm to be consistent with the respective input. *Consistency* is not delayable, see for example the learning by enumeration strategy mentioned earlier.

In Section 2.4 we derive all 45 pairwise equivalences, proper implications or incomparabilities between the introduced delayable learning success criteria. By definition and previous results about one third of the 90 relevant implications have been known, see Section 2.2, [OSW86] and [LZK96]. In particular, we show that any learning algorithm can be assumed to only change its previously hypothesized classifier if the latter is inconsistent with the available training data. This so-called *conservative* learning behaviour, [Ang80], can be accomplished with our insights from Section 2.2 and a regularity property. The proof gives a deeper understanding of the W-hypothesis space and additionally covers other requirements. For example, conservativeness implies that the learning algorithm does not diverge from a correct hypothesis.

The following Chapter 3 is concerned with other models of successful learning in Inductive Inference.

In Section 3.2 we sharpen the comparison of learning from exclusively positive data with learning from labeled data in [LZ93] by posing the most restrictive additional requirements.

Thereafter, in Section 3.3, we consider approximations by allowing a finite number of *errors* of the concept hypothesized by the learning algorithm compared with the classifier to be learned. With appropriate representations as a total function or recursively enumerable set, we provide an equivalence between learning total classifiers from either exclusively positive or binary labeled data. This allows us to transfer the approximative hierarchy in [Bār74] and [CS83] to our setting of full-information learning from binary labeled data. Concretely, we show that increasing the number of allowed errors makes strictly more concept classes learnable. Furthermore, for a fixed error parameter, we provide a *duality* depending on whether vacillating between infinitely many different correct

hypotheses is still considered successful learning. This contrasts the hierarchies when learning from solely positive information, [Cas99].

In Section 3.4 we consider a hypothesis space between uniform hypothesis spaces of symmetric classifiers and the W-hypothesis space. Different variations of this approach are investigated for learning from solely positive data in [Ber+20a] and [Ber+20b]. For learning from binary labeled data, we immediately observe that the pairwise equivalences, proper implications and incomparabilities between the established collection of delayable learning success criteria are the same as for the W-hypothesis space considered in Section 2.4.

In contrast to classical machine learning algorithms, there is a growing interest in incremental implementations that do not access all training data to infer the new hypothesized classifier. For example, when training binary classifiers from standard libraries for one epoch, a parameter to be adjusted is the so-called batch size, specifying how many training examples are used for the computation of the next hypothesis. In Part II we consider models for these memory-efficient algorithms.

In Chapter 4 we focus on *iterative* algorithms, that compute the next hypothesis based on the current labeled datum and the last hypothesized classifier, [R W76]. With an easy locking sequence argument, [BB75], one can show that these iterative learners have strictly less learning power than the full-information variant. It has also been observed that the learning capability is not improved if the last $k > 1$ data are used in the computation of the next hypothesis from the previous one, [OSW86]. Our results can also be found in [KKS20]. This reference in addition includes a constructive iterative algorithm proving the learnability of the uniform concept class of half-spaces from labeled data. Assuming realizability, also the sub concept class of all spam filters is learnable by an iterative algorithm.

The essential terminology is recapitulated in Section 4.2. Thereafter, in Section 4.3, we provide a procedure to obtain concept classes learnable by a full-information algorithm from solely positive data but not by an iterative learning algorithm from positive and negative information. Hence, we observe that the aforementioned two settings are incomparable with respect to their learning capabilities.

In the next two Sections, we investigate the pairwise relations between the delayable learning success criteria for iterative learners. For learning from solely positive information these have been clarified in [LZ91], [CM08a], [CK10] and

[Jai+16]. For learning from binary labeled data, it was observed in [JLZ07a] that consistency and conservativeness are restrictive. A further additional requirement arising from cognitive science research is *U-shapedness* [SS82], stating that the learning algorithm does deviate from an optimal classifier. We already mentioned that conservativeness forbids U-shapes and hence naturally the question arises whether *non-U-shapedness* is also restrictive. We differentiate a semantic and a syntactic formalization of this phenomenon, where a learning success criterion is semantic, if it does still hold, when the learning algorithm outputs other parameters standing for the same classifier. On the one hand, in Section 4.4, we provide a lemma that might be helpful to settle the learning power of the semantic version. On the other hand, in Section 4.5, we show that forbidding non-semantic *U-shapes* also restricts iterative learning algorithms on binary labeled data.

In Chapter 5 we investigate the learning abilities of **BMS** learning algorithms that do refer to the currently observed datum and the actual state of the algorithm in order to compute the next hypothesis, [Car+07]. For successful learning the algorithm must stop using new states eventually. We provide the *complete map* of implications or incomparabilities between the established collection of delayable learning success criteria when learning from positive data.

In Section 5.2 we fix the notation that is also inspired by automata theory.

Building on this, in Section 5.3 we prove that **BMS** and iterative learning algorithms are equally powerful for all semantic delayable learning success criteria. This is also true for learning from binary labeled data.

In Section 5.4 we show the equivalence of syntactic (non-semantic) learning success criteria, for example conservativeness and syntactic non-U-shapedness. For this, we refer to witness-based learning processes, [KS16], in which every change of the hypothesis is justified by a later on correctly classified witness from the training data. Moreover, we observe that syntactic non-U-shapedness restricts **BMS** learning algorithms from positive data. Finally, we observe that with respect to learning success criteria incorporating syntactic non-U-shapedness, **BMS** learning algorithms can learn more concept classes than iterative learning algorithms.

Our insights have a strong mathematical flavor. We rely on results from Computability Theory, which round off the contributions towards a better understanding of Machine Learning by Linear Algebra, Calculus, Probability Theory,

Differential Geometry, Statistics and other areas of mathematics. Most notably, we employ infinite fixed-point-theorems, like a one-to-one version of Case's Operator Recursion Theorem, [Köt09].

Part I

Full-Batch Learning from Informant

2 Map of Delayable Learning Success Criteria

Learning from positive and negative information, so-called *informant*, being one of the models for human and machine learning introduced by E. M. Gold is investigated. Particularly, naturally arising questions about this learning setting, originating in results on learning from solely positive information, are answered.

By a carefully arranged argument learners can be assumed to only change their hypothesis in case it is inconsistent with the data (such a learning behavior is called *conservative*). The deduced main theorem states the relations between the most important delayable learning success criteria, being the ones not ruined by a delayed in time hypothesis output.

Additionally, our investigations concerning the non-delayable requirement of consistent learning underpin the claim for *delayability* being the right structural property to gain a deeper understanding concerning the nature of learning success criteria.

2.1 Introduction

Research in the area of *inductive inference* aims at investigating the learning of formal languages and has connections to computability theory, complexity theory, cognitive science, machine learning, and more generally artificial intelligence. Setting up a classification program for deciding whether a given word belongs to a certain language can be seen as a problem in supervised machine learning, where the machine experiences labeled data about the target language. The label is 1 if the datum is contained in the language and 0 otherwise. The machine's task is to infer some rule in order to generate words in the language of interest and thereby generalize from the training samples. This so-called *learning from informant* was introduced in [Gol67] and further investigated in several publications, including [BB75], [OSW86] and [LZK96].

According to [Gol67] the learner is modelled by a computable function, successively receiving sequences incorporating more and more data. The source of labeled data is called an *informant*, which is supposed to be *complete in the limit*,

i.e., every word in the language must occur at least once. Thereby, the learner possibly updates the current description of the target language (its hypothesis). Learning is considered successful, if after some finite time the learner settles on exactly one correct hypothesis, which precisely captures the words in the language to be learned. As a single language can easily be learned, the interesting question is whether there is a learner successful on all languages in a fixed collection of languages.

Example. Consider $\mathcal{L} = \{ \mathbb{N} \setminus X \mid X \subseteq \mathbb{N} \text{ finite} \}$, the collection of all co-finite sets of natural numbers. Clearly, there is a computable function p mapping finite subsets $X \subseteq \mathbb{N}$ to $p(X)$, such that $p(X)$ encodes a program which stops if and only if the input is not in X. We call $p(X)$ an *index* for $\mathbb{N} \setminus X$. The learner is successful if for every finite $X \subseteq \mathbb{N}$ it infers $p(X)$ from a possibly very large but finite number of samples labeled according to $\mathbb{N} \setminus X$.

Regarding this example, let us assume the first two samples are $(60, 1)$ and $(2, 0)$. The first datum still leaves all options with $60 \notin X$. As the second datum tells us that $2 \in X$, we may make the learner guess $p(\{2\})$ until possibly more negative data is available. Thus, the collection of all co-finite sets of natural numbers is **Ex**-learnable from informant, simply by making the learner guess the complement of all negative information obtained so far. Since after finitely many steps all elements of the finite complement of the target language have been observed, the learner will be correct from that point onward.

It is well-known that this collection of languages cannot be learned from purely positive information. Intuitively, at any time the learner cannot distinguish the whole set of natural numbers from all other co-finite sets which contain all natural numbers presented to the learner until this point.

Learning from solely positive information, so-called *text*, has been studied extensively, including many learning success criteria and other variations. Some results are summed up in [Jai+99] and [Cas16]. We address the naturally arising question what difference it makes to learn from positive and negative information.

For learning from text there are entire maps displaying the pairwise relations of different well-known learning success criteria, see [KP14], [KS16] and [Jai+16]. We give an equally informative map for **Ex**-learning from informant.

The most important requirements on the learning process when learning from informant are *conservativeness* (**Conv**), where only inconsistent hypotheses

are allowed to be changed; *strong decisiveness* (**SDec**), forbidding to ever return semantically to a withdrawn hypothesis; *strong monotonicity* (**SMon**), requiring that in every step the hypothesis incorporates the former one; *monotonicity* (**Mon**), fulfilled if in every step the set of correctly inferred words incorporates the formerly correctly guessed; *cautiousness* (**Caut**), for which never a strict subset of earlier conjectures is guessed. In [LZK96] it was observed that requiring monotonicity is restrictive and that under the assumption of strong monotonicity even fewer collections of languages can be learned from informant. We complete the picture by answering the following questions regarding **Ex**-learning from informant positively:

1. Is every learnable collection of languages also learnable in a conservative and strongly decisive way?

2. Are monotonic and cautious learning incomparable?

The above mentioned observations in [LZK96] follow from positively answering the second question.

A diagram incorporating the resulting map is depicted in Figure 2.1. The complete map can be found in Figure 2.2.

Answering the first question builds on providing the two *normal forms* of (1) requiring learning success only on the information presented in the canonical order and (2) assuming the learner to be defined on all input sequences. Further, a regularity property borrowed from text learning plays a crucial role in the proof.

Requiring all of the learners guesses to be *consistent* with the positive and the negative information being presented to it so far makes learning harder. Next to this we also observe that the above normal forms cannot be assumed when the learner is required to act consistently. On the one hand, it is easier to find a learner for a collection of languages that consistently learns each of them only from the canonical presentation than finding one consistently learning them from arbitrary informant. On the other hand finding a total learner consistently **Ex**-learning a collection of languages is harder than finding a partial one.

We further transfer the concept of a learning success criterion to be invariant under time-delayed outputs of the hypotheses, introduced for learning from text in [KP16] and generalized in [KSS17], to the setting of learning from informant. Consistency is not *delayable* since a hypothesis which is consistent now might

be inconsistent later due to new data. As this is the only requirement not being delayable, the results mentioned in the last paragraph justify the conjecture of delayability being the right property to proof more results that at once apply to all learning success criteria but consistency.

While in [LZ94] variously restricted learning of collections of recursive languages with a uniform decision procedure are considered, the above mentioned results also apply to arbitrary collections of recursively enumerable sets. Further, our results are as strong as possible, meaning that negative results are stated for indexable families, if possible, and positive results for all collections of languages.

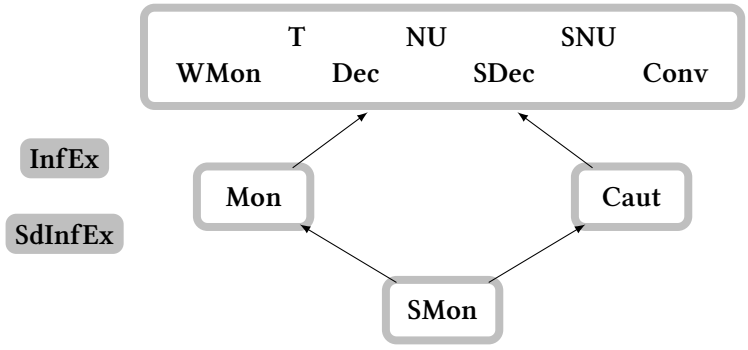

Figure 2.1: Relations between delayable learning restrictions in **Ex**-learning from informants. Implications are represented as black lines from bottom to top. Two learning settings are equivalent if and only if they lie in the same grey outlined zone.

In contrast to our observations, it has been shown in [Ang80] that requiring a conservative learning process is a restriction when learning from text. Further, this is equivalent to cautious learning as shown in [KP16]. That monotonic learning is restrictive and incomparable to both of them in the text learning setting follows from [LZK96], [KS95], [JS98] and [KP16]. Further, when learning from text, strong monotonicity is again the most restrictive assumption by [LZK96]. Strong decisiveness is restrictive, see [Bal+08], and further is restricted by cautiousness/conservativeness on the one hand and monotonicity on the other hand by [KP16]. In the latter visualizations and a detailed discussion are provided.

When the learner does not have access to the order of presentation but knows the number of samples, the map remains the same as observed in [KS16].

In case the learner makes its decisions only based on the set of presented samples and ignores any information about the way it is presented, it is called *set-driven* (**Sd**). For such set-driven learners, when learning from text, conservative, strongly decisive and cautious learning are no longer restrictive and the situation with monotonic and strong monotonic learning remains unchanged by [KS95] and [KP16].

We observe that for delayable informant learning all three kinds of learners yield the same map. Thus, our results imply that negative information compensates for the lack of information set-driven learners have to deal with.

[Gol67] was already interested in the above mentioned normal forms and proved that they can be assumed without loss of generality in the basic setting of pure **Ex**-learning, whereas our results apply to all delayable learning success criteria.

The name "delayability" refers to tricks in order to delay mind changes of the learner which were used to obtain polynomial computation times for the learners hypothesis updates as discussed in [Pit89] and [CK09]. Moreover, it should not be confused with the notion of δ-delay, [AZ08], which allows satisfaction of the considered learning restriction δ steps later than in the un-δ-delayed version.

In [OSW86] several restrictions for learning from informant are analyzed and mentioned that cautious learning is a restriction to learning power; we extend this statement with our Proposition 2.22 in which we give one half of the answer to the second question above by providing a family of languages not cautiously but monotonically **Ex**-learnable from informant.

Furthermore, [OSW86] consider a version of *conservativeness* where mind changes are only allowed if there is *positive* data contradicting the current hypothesis, which they claim to restrict learning power. In this thesis, we stick to the more common definition in [BB75] and [Bār77], according to which mind changes are allowed also when there is negative data contradicting the current hypothesis.

In Section 2.2 the setting of learning from informant is formally introduced by transferring fundamental definitions and —as far as possible— observations from the setting of learning from text. In Section 2.3 in order to derive the

entire map of pairwise relations between delayable **Ex**-learning success criteria, normal forms and a regularity property for such learning from informant are provided. Further, consistent learning is being investigated. In Section 2.4 we answer the questions above and present all pairwise relations of learning criteria in Theorem 2.24.

All sections build on Section 2.2. Additionally, Section 2.4 builds on Section 2.3.

2.2 Informant Learning

We formally introduce the notion of an informant and transfer concepts and fundamental results from the setting of learning from text to learning from informant. This includes the learner itself, convergence criteria, locking sequences, learning restrictions and success criteria as well as a compact notation for comparing different learning settings. In the last subsection delayability as the central property of learning restrictions and learning success criteria is formally introduced.

As far as possible, notation and terminology on the learning theoretic side follow [Jai+99], whereas on the computability theoretic side we refer to [Odi99].

We let $\mathbb{N}$ denote the *natural numbers* including 0 and write ∞ for an *infinite cardinality*. Moreover, for a function f we write $\mathrm{dom}(f)$ for its *domain* and $\mathrm{ran}(f)$ for its *range*. If we deal with (a subset of) a cartesian product, we are going to refer to the *projection functions* to the first or second coordinate by pr_1 and pr_2, respectively. For sets X, Y and $a \in \mathbb{N}$ we write $X =^a Y$, if X *equals* Y *with* a *anomalies*, i.e., $|(X \setminus Y) \cup (Y \setminus X)| \leq a$, where $|.|$ denotes the *cardinality function*. In this spirit we write $X =^* Y$, if there exists some $a \in \mathbb{N}$ such that $X =^a Y$. Further, $X^{<\omega}$ denotes the *finite sequences* over X and X^ω stands for the *countably infinite sequences* over X. Additionally, $X^{\leq\omega} := X^{<\omega} \cup X^\omega$ denotes the set of all *countably finite or infinite sequences* over X. For every $f \in X^{\leq\omega}$ and $t \in \mathbb{N}$, we let $f[t] := \{(s, f(s)) \mid s < t\}$ denote the *restriction of* f *to* t. For sequences $\sigma, \tau \in X^{<\omega}$ their concatenation is denoted by $\sigma^\frown\tau$ and we write $\sigma \preceq \tau$, if σ is an initial segment of τ, i.e., there is some $t \in \mathbb{N}$ such that $\sigma = \tau[t]$. Finally, we write $\mathrm{last}(\sigma)$ for the last element of σ , $\sigma(|\sigma| - 1)$, and σ^- for the initial segment of σ without $\mathrm{last}(\sigma)$, i.e. $\sigma[|\sigma| - 1]$. Clearly, $\sigma = \sigma^-{}^\frown\mathrm{last}(\sigma)$. In our setting, we typically have $X = \mathbb{N} \times \{0, 1\}$. Without demanding computability, we denote

by $\mathfrak{P}$ and $\mathfrak{R}$ the set of all partial functions $f : \mathrm{dom}(f) \subseteq \mathbb{N}\times\{0,1\}^{<\omega} \to \mathbb{N}$ and total functions $f : \mathbb{N}\times\{0,1\}^{<\omega} \to \mathbb{N}$, respectively.

Let $L \subseteq \mathbb{N}$. If L is recursively enumerable, we call L a *language*. In case its characteristic function is computable, we say it is a *recursive language*. Moreover, we call $\mathcal{L} \subseteq \mathrm{Pow}(\mathbb{N})$ a *collection of (recursive) languages*, if every $L \in \mathcal{L}$ is a (recursive) language. In case there exists an enumeration $\{L_\xi \mid \xi \in \Xi\}$ of $\mathcal{L}$, where $\Xi \subseteq \mathbb{N}$ is recursive and a computable function f with $\mathrm{ran}(f) \subseteq \{0,1\}$ such that $x \in L_\xi \Leftrightarrow f(x,\xi) = 1$ for all $\xi \in \Xi$ and $x \in \mathbb{N}$, we say $\mathcal{L}$ is an *indexable family of recursive languages*. By definition indexable families are collections of recursive languages with a uniform decision procedure.

Further, we fix a programming system φ as introduced in [RC94]. Briefly, in the φ-system, for a natural number p, we denote by φ_p the partial computable function with program code p. We call p an *index* for $W_p := \mathrm{dom}(\varphi_p)$. For a finite set $X \subseteq \mathbb{N}$ we denote by $\mathrm{ind}(X)$ a canonical index for X. In reference to a Blum complexity measure, for all $p, t \in \mathbb{N}$, we denote by $W_p^t \subseteq W_p$ the recursive set of all natural numbers less or equal to t, on which the machine executing p halts in at most t steps. Moreover, by s-m-n we refer to a well-known recursion theoretic observation, which gives finite and infinite recursion theorems, like Case's Operator Recursion Theorem **ORT**, [Cas74]. Inuitively, it states that for every recursive operator there is a computable function that is a fixed point of the action of the operator on the φ-system. Formally, a 1-1 version of this result reads as follows.

1-1 Operator Recursion Theorem ([Köt09]). Let $\Theta : \mathcal{P} \to \mathcal{P}$ be a computable operator, namely a function mapping partial computable functions to partial computable functions. Then there is a 1-1 computable function $h \in \mathcal{P}$ such that $\forall n, x\big(\varphi_{h(n)}(x) = \Theta(h)(n,x)\big)$.

For our purposes the operator Θ will always be implicit. The first application of ORT is in Proposition 2.18 and it occurs in many different variants in other proofs. For further intuitions see for example [Cas94].

Finally, we let $H = \{\, p \in \mathbb{N} \mid \varphi_p(p)\downarrow \,\}$ denote the halting problem.

2.2.1 Informant and Learners

Intuitively, for any natural number x an *informant for a language* L answers the question whether $x \in L$ in finite time. More precisely, for every natural

number x the informant I has either $(x,1)$ or $(x,0)$ in its range, where the first is interpreted as $x \in L$ and the second as $x \notin L$, respectively.

Definition 2.1. *(i) Let $f \in (\mathbb{N}\times\{0,1\})^{\leq\omega}$. We denote by*

$$\operatorname{pos}(f) := \{y \in \mathbb{N} \mid \exists x \in \mathbb{N}\colon \operatorname{pr}_1(f(x)) = y \wedge \operatorname{pr}_2(f(x)) = 1\},$$
$$\operatorname{neg}(f) := \{y \in \mathbb{N} \mid \exists x \in \mathbb{N}\colon \operatorname{pr}_1(f(x)) = y \wedge \operatorname{pr}_2(f(x)) = 0\}$$

the sets of all natural numbers, about which f gives some positive or negative information, respectively.

(ii) Let L be a language. We call every function $I : \mathbb{N} \to \mathbb{N}\times\{0,1\}$ such that $\operatorname{pos}(I) \cup \operatorname{neg}(I) = \mathbb{N}$ and $\operatorname{pos}(I) \cap \operatorname{neg}(I) = \varnothing$ an informant. *Further, we denote by* **Inf** *the set of all informant and the* set of all informant for the language L *is defined as*

$$\mathbf{Inf}(L) := \{I \in \mathbf{Inf} \mid \operatorname{pos}(I) = L\}.$$

(iii) Let I be an informant. If for every time $t \in \mathbb{N}$ the informant I reveals information about t itself, for short $\operatorname{pr}_1(I(t)) = t$, we call I a canonical informant.

It is immediate, that $\operatorname{neg}(I) = \mathbb{N}\setminus L$ for every $I \in \mathbf{Inf}(L)$. In [Gol67] a canonical informant is referred to as *methodical informant.*

We employ Turings model for human computers which is the foundation of all modern computers to model the processes in human and machine learning.

Definition 2.2. *A* learner *is a (partial) computable function*

$$M : \operatorname{dom}(M) \subseteq (\mathbb{N}\times\{0,1\})^{<\omega} \to \mathbb{N}.$$

The set of all partial computable functions $M : \operatorname{dom}(M) \subseteq (\mathbb{N}\times\{0,1\})^{<\omega} \to \mathbb{N}$ and total computable functions $M : (\mathbb{N}\times\{0,1\})^{<\omega} \to \mathbb{N}$ are denoted by $\mathcal{P}$ and $\mathcal{R}$, respectively.

2.2.2 Convergence Criteria and Locking Sequences

Convergence criteria tell us what quality of the approximation and syntactic accuracy of the learners' eventual hypotheses are necessary to call learning

successful. Further, we prove that learning success implies the existence of sequences on which the learner is locked in a way corresponding to the convergence criterion. We will use locking sequences to show that a collection of languages cannot be learned in a certain way.

Definition 2.3. *Let M be a learner and $\mathcal{L}$ a collection of languages. Further, let $a \in \mathbb{N} \cup \{*\}$ and $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$.*

(i) Let $L \in \mathcal{L}$ be a language and $I \in \mathbf{Inf}(L)$ an informant for L presented to M.

a) We call $h = (h_t)_{t \in \mathbb{N}} \in \mathbb{N}^\omega$, where $h_t := M(I[t])$ for all $t \in \mathbb{N}$, the learning sequence of M on I.

b) M learns L from I with a anomalies and vacillation number b in the limit*, for short M $\mathbf{Ex}^a_b$-learns L from I or $\mathbf{Ex}^a_b(M, I)$, if there is a time $t_0 \in \mathbb{N}$ such that $|\{ h_t \mid t \geq t_0 \}| \leq b$ and for all $t \geq t_0$ we have $W_{h_t} =^a L$.*

(ii) M learns $\mathcal{L}$ with a anomalies and vacillation number b in the limit*, for short M $\mathbf{Ex}^a_b$-learns $\mathcal{L}$, if $\mathbf{Ex}^a_b(M, I)$ for every $L \in \mathcal{L}$ and every $I \in \mathbf{Inf}(L)$.*

The intuition behind *(i)(b)* is that, sensing I, M eventually only vacillates between at most b-many hypotheses, where the case $b = *$ stands for eventually finitely many different hypotheses. In accordance with the literature, we omit the superscript 0 and the subscript 1.

Ex-learning, also known as *explanatory learning*, is the most common definition for successful learning and corresponds to the notion of identifiability in the limit by [Gol67], where the learner eventually decides on one correct hypothesis. On the other end of the hierarchy of convergence criteria is *behaviorally correct learning*, for short **Bc**- or $\mathbf{Ex}_\infty$-learning, which only requires the learner to be eventually correct, but allows infinitely many syntactically different hypotheses in the limit. Behaviorally correct learning was introduced in [OW82]. The general definition of $\mathbf{Ex}^a_b$-learning for $a \in \mathbb{N} \cup \{*\}$ and $b \in \mathbb{N}_{>0} \cup \{*\}$ was first mentioned in [Cas99].

In our setting, we also allow $b = \infty$ and subsume all $\mathbf{Ex}^a_b$ under the notion of a *convergence criterion*, since they determine in which semi-topological sense the learning sequence needs to have L as its limit, in order to succeed in learning L.

In the following we transfer an often employed observation in [BB75] to the setting of learning from informant and generalize it to all convergence criteria

introduced in Definition 2.3. For this we first recall the notion of consistency of a sequence with a set according to [BB75] and [Bār77].

Definition 2.4. *Let $f \in (\mathbb{N}\times\{0,1\})^{\leq\omega}$ and $A \subseteq \mathbb{N}$. We define*

$$\mathbf{Cons}(f, A) \quad :\Leftrightarrow \quad \mathrm{pos}(f) \subseteq A \ \wedge \ \mathrm{neg}(f) \subseteq \mathbb{N} \setminus A$$

and say f is consistent with *A.*

Definition 2.5. *Let M be a learner, L a language and $a \in \mathbb{N} \cup \{*\}$ as well as $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$. We call $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ a $\mathbf{Ex}^a_b$*-locking sequence for *M on L, if $\mathbf{Cons}(\sigma, L)$ and*

$$\exists D \subseteq \mathbb{N} \, (\, |D| \leq b \ \wedge \ \forall \tau \in (\mathbb{N}\times\{0,1\})^{<\omega} \\ \big(\mathbf{Cons}(\tau, L) \Rightarrow \big(M(\sigma^\frown\tau)\downarrow \wedge W_{M(\sigma^\frown\tau)} =^a L \wedge M(\sigma^\frown\tau) \in D \big) \big) \,)$$

Further, a locking sequence for *M on L is a* **Ex**-*locking sequence for M on L.*

Intuitively, the learner M is locked by the sequence σ onto the language L in the sense that no presentation consistent with L can circumvent M guessing admissible approximations to L and additionally all guesses based on an extension of σ are captured by a finite set of size at most b.

Note that the definition implies $M(\sigma)\downarrow$, $W_{M(\sigma)} =^a L$ and $M(\sigma) \in D$.

Lemma 2.6. *Let M be a learner, $a \in \mathbb{N} \cup \{*\}$, $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$ and L a language $\mathbf{Ex}^a_b$-identified by M. Then there is a $\mathbf{Ex}^a_b$-locking sequence for M on L.*

Proof. This is a contradictory argument. Without loss of generality M is defined on $\varnothing$. Assume towards a contradiction for every σ with $\mathbf{Cons}(\sigma, L)$, $M(\sigma)\downarrow$ and $W_{M(\sigma)} =^a L$ and for every finite $D \subseteq \mathbb{N}$ with at most b elements there exists a sequence $\tau^D_\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ with

$$\mathbf{Cons}(\tau^D_\sigma, L) \wedge \Big(M(\sigma^\frown\tau^D_\sigma)\uparrow \vee \neg W_{M(\sigma^\frown\tau^D_\sigma)} =^a L \vee M(\sigma^\frown\tau^D_\sigma) \notin D \Big).$$

Let I_L denote the canonical informant for L. We obtain an informant for L on which M does not $\mathbf{Ex}^a_b$-converge by letting

$$I := \bigcup_{n\in\mathbb{N}} \sigma_n, \text{ with}$$

$$\sigma_0 := I_L[1],$$
$$\sigma_{n+1} := \sigma_n \frown \tau_{\sigma_n}^{D_n} \frown I_L(n+1)$$

for all $n \in \mathbb{N}$, where in $D_n := \{ M(\sigma_i^-) \mid \max\{0, n-b+1\} \leq i \leq n \}$ we collect M's at most b-many last relevant hypotheses. Since I is an informant for L by having interlaced the canonical informant for L, the learner M $\mathbf{Ex}_b^a$-converges on I. Therefore, let n_0 be such that for all t with $\sigma_{n_0}^- \trianglelefteq I[t]$ we have $h_t{\downarrow}$ and $W_{h_t} =^a L$. Then certainly $\{ M(\sigma_i^-) \mid n_0 \leq i \leq n_0 + b \}$ has cardinality $b+1$, a contradiction. □

Obviously, an appropriate version also holds when learning from text is considered.

2.2.3 Learning Success Criteria

We list the most common requirements that combined with a convergence criterion define when a learning process is considered successful.

The choice of additional requirements in the following definition is justified by prior investigations of the corresponding criteria, when learning from text, see [KP16], [KS16] and [Jai+16].

Definition 2.7. *Let M be a learner, $I \in \mathbf{Inf}$ an informant and $h = (h_t)_{t\in\mathbb{N}} \in \mathbb{N}^\omega$ the learning sequence of M on I. We write*

(i) $\mathbf{Cons}(M, I)$ *([Ang80]), if M is* consistent *on I, i.e., for all t*

$$\mathbf{Cons}(I[t], W_{h_t}).$$

(ii) $\mathbf{Conv}(M, I)$ *([Ang80]), if M is* conservative *on I, i.e., for all s, t with $s \leq t$*

$$\mathbf{Cons}(I[t], W_{h_s}) \implies h_s = h_t.$$

(iii) $\mathbf{Dec}(M, I)$ *([OSW82]), if M is* decisive *on I, i.e., for all r, s, t with $r \leq s \leq t$*

$$W_{h_r} = W_{h_t} \implies W_{h_r} = W_{h_s}.$$

(iv) $\mathbf{Caut}(M, I)$ *([OSW86]), if M is* cautious *on I, i.e., for all s, t with $s \leq t$*

$$\neg W_{h_t} \subsetneq W_{h_s}.$$

(v) **WMon**(M, I) *([Jan91],[Wie91]), if* M *is* weakly monotonic on I*, i.e., for all* s, t *with* $s \leq t$

$$\mathbf{Cons}(I[t], W_{h_s}) \implies W_{h_s} \subseteq W_{h_t}.$$

(vi) **Mon**(M, I) *([Jan91],[Wie91]), if* M *is* monotonic on I*, i.e., for all* s, t *with* $s \leq t$

$$W_{h_s} \cap \mathrm{pos}(I) \subseteq W_{h_t} \cap \mathrm{pos}(I).$$

(vii) **SMon**(M, I) *([Jan91],[Wie91]), if* M *is* strongly monotonic on I*, i.e., for all* s, t *with* $s \leq t$

$$W_{h_s} \subseteq W_{h_t}.$$

(viii) **NU**(M, I) *([Bal+08]), if* M *is* non-U-shaped on I*, i.e., for all* r, s, t *with* $r \leq s \leq t$

$$W_{h_r} = W_{h_t} = \mathrm{pos}(I) \implies W_{h_r} = W_{h_s}.$$

(ix) **SNU**(M, I) *([CM11]), if* M *is* strongly non-U-shaped on I*, i.e., for all* r, s, t *with* $r \leq s \leq t$

$$W_{h_r} = W_{h_t} = \mathrm{pos}(I) \implies h_r = h_s.$$

(x) **SDec**(M, I) *([KP16]), if* M *is* strongly decisive on I*, i.e., for all* r, s, t *with* $r \leq s \leq t$

$$W_{h_r} = W_{h_t} \implies h_r = h_s.$$

The following lemma states the implications between almost all of the above defined learning restrictions, which form the foundation of our research. Figure 2.2 includes the resulting backbone, which is slightly different from the one for learning from text, since **WMon** does not necessarily imply **NU** in the context of learning from informant.

Lemma 2.8. *Let* M *be a learner and* $I \in$ **Inf** *an informant. Then*

(i) **Conv**(M, I) *implies* **SNU**(M, I) *and* **WMon**(M, I).

(ii) **SDec**(M, I) *implies* **Dec**(M, I) *and* **SNU**(M, I).

(iii) **SMon**(M, I) *implies* **Caut**(M, I), **Dec**(M, I), **Mon**(M, I) *and* **WMon**(M, I).

(iv) **Dec**(M, I) *and* **SNU**(M, I) *each imply* **NU**(M, I).

(v) $\mathbf{WMon}(M,I)$ *does* not *imply* $\mathbf{NU}(M,I)$ *in general.*

Proof. Verifying the claimed implications is straightforward. In order to verify *(v)*, consider $L = 2\mathbb{N}$. Fix $p, q \in \mathbb{N}$ such that $W_p = 2\mathbb{N} \cup \{1\}$ and $W_q = 2\mathbb{N}$ and define the learner M for all $\sigma \in \mathbb{N} \times \{0,1\}^{<\omega}$ by

$$M(\sigma) = \begin{cases} p, & \text{if } 1 \in \mathrm{neg}(\sigma) \wedge 2 \notin \mathrm{pos}(\sigma); \\ q, & \text{otherwise.} \end{cases}$$

In order to prove $\mathbf{WMon}(M,I)$ for every $I \in \mathbf{Inf}(L)$, let I be an informant for L and $\mathfrak{s}_I(x) := \min\{t \in \mathbb{N} \mid \mathrm{pr}_1(I(t)) = x\}$, i.e., $\mathfrak{s}_I(1)$ and $\mathfrak{s}_I(2)$ denote the first occurance of $(1,0)$ and $(2,1)$ in $\mathrm{ran}(I)$, respectively. Then we have for all $t \in \mathbb{N}$

$$W_{h_t} = \begin{cases} 2\mathbb{N} \cup \{1\}, & \text{if } \mathfrak{s}_I(1) < t \leq \mathfrak{s}_I(2); \\ 2\mathbb{N}, & \text{otherwise.} \end{cases}$$

We have $W_{h_s} = W_{M(I[s])} = 2\mathbb{N} \cup \{1\}$ as well as $1 \in \mathrm{neg}(I[t])$ for all $s, t \in \mathbb{N}$ with $\mathfrak{s}_I(1) < s \leq \mathfrak{s}_I(2)$ and $t > \mathfrak{s}_I(2)$. Therefore, $\neg\mathbf{Cons}(I[t], W_{h_s})$ because of $\mathrm{neg}(I[t]) \not\subseteq \mathbb{N} \setminus W_{h_s}$. We obtain $\mathbf{WMon}(M,I)$ since whenever $s \leq t$ in $\mathbb{N}$ are such that $\mathbf{Cons}(I[t], W_{h_s})$, we know that $W_{h_s} = 2\mathbb{N} \cup \{1\}$ can only hold if likewise $\mathfrak{s}_I(1) < t \leq \mathfrak{s}_I(2)$ and hence $W_{h_t} = 2\mathbb{N} \cup \{1\}$, which yields $W_{h_s} \subseteq W_{h_t}$. Furthermore, if $W_{h_s} = 2\mathbb{N}$ all options for W_{h_t} satisfy $W_{h_s} \subseteq W_{h_t}$. Otherwise, in case M observes the canonical informant I for L, we have $W_{h_0} = W_{h_1} = 2\mathbb{N}$, $W_{h_2} = 2\mathbb{N} \cup \{1\}$ and $W_{h_t} = 2\mathbb{N}$ for all $t > 2$, which shows $\neg\mathbf{NU}(M,I)$. □

By the next definition, in order to characterize what successful learning means, we choose a convergence criterion from Definition 2.3 and may pose additional learning restrictions from Definition 2.7.

Definition 2.9. *Let* $\mathbf{T} := \mathfrak{P} \times \mathbf{Inf}$ *denote the whole set of pairs of possible learners and informant. We denote by*

$$\begin{aligned} \Delta := \{\, & \mathbf{Caut}, \mathbf{Cons}, \mathbf{Conv}, \mathbf{Dec}, \mathbf{SDec}, \\ & \mathbf{WMon}, \mathbf{Mon}, \mathbf{SMon}, \mathbf{NU}, \mathbf{SNU}, \mathbf{T} \,\} \end{aligned}$$

the set of admissible learning restrictions *and by*

$$\Gamma := \{\, \mathbf{Ex}^a_b \mid a \in \mathbb{N} \cup \{*\} \ \wedge\ b \in \mathbb{N}_{>0} \cup \{*, \infty\} \,\}$$

the set of convergence criteria. *Further, if*

$$\beta \in \{ \bigcap_{i=0}^{n} \delta_i \cap \gamma \mid n \in \mathbb{N}, \forall i \leq n\, (\delta_i \in \Delta) \text{ and } \gamma \in \Gamma \} \subseteq \mathfrak{P} \times \mathbf{Inf},$$

we say that β *is a* learning success criterion.

Note that every convergence criterion is indeed a learning success criterion by letting $n = 0$ and $\delta_0 = \mathbf{T}$, where the latter stands for no restriction. In the literature convergence criteria are also called identificaton criteria and then denoted by I or ID.

We refer to all $\delta \in \{\mathbf{Caut}, \mathbf{Cons}, \mathbf{Dec}, \mathbf{Mon}, \mathbf{SMon}, \mathbf{WMon}, \mathbf{NU}, \mathbf{T}\}$ also as *semantic* learning restrictions, as they allow for proper semantic convergence.

2.2.4 Comparing the Learning Power of Learning Settings

In order to state observations about how two ways of defining learning success relate to each other, the learning power of the different settings is encapsulated in notions $[\alpha\mathbf{Inf}\beta]$ defined as follows.

Definition 2.10. *Let* $\alpha \subseteq \mathcal{P}$ *be a property of partial computable functions from the set* $(\mathbb{N}\times\{0,1\})^{<\omega}$ *to* $\mathbb{N}$ *and* β *a learning success criterion. We denote by* $[\alpha\mathbf{Inf}\beta]$ *the set of all collections of languages that are* β*-learnable from informant by a learner* M *with the property* α*.*

In case the learner only needs to succeed on canonical informant, we denote the corresponding set of collections of languages by $[\alpha\mathbf{Inf}_{\text{can}}\beta]$*.*

In the learning success criterion at position β, the learning restrictions to meet are denoted in alphabetic order, followed by a convergence criterion.

At position α, we restrict the set of admissible learners by requiring for example totality. The properties stated at position α are *independent of learning success.*

For example, a collection of languages $\mathcal{L}$ lies in $[\mathcal{R}\mathbf{Inf}_{\text{can}}\mathbf{ConvSDecEx}]$ if and only if there is a total learner M conservatively, strongly decisively **Ex**-learning every $L \in \mathcal{L}$ from canonical informant. The latter means that for every canonical informant I for some $L \in \mathcal{L}$ we have $\mathbf{Conv}(M, I)$, $\mathbf{SDec}(M, I)$ and $\mathbf{Ex}(M, I)$.

Note that it is also conventional to require M's hypothesis sequence to fulfill certain learning restrictions, not asking for the success of the learning process.

For instance, we are going to show that there is a collection of languages $\mathcal{L}$ such that:

- there is a learner which behaves consistently on all $L \in \mathcal{L}$ and **Ex**-learns all of them, for short $\mathcal{L} \in [\mathbf{InfConsEx}]$.
- there is no learner which **Ex**-learns every $L \in \mathcal{L}$ and behaves consistently on all languages, for short $\mathcal{L} \notin [\mathbf{ConsInfEx}]$.

The existence of $\mathcal{L}$ is implicit when writing $[\mathbf{ConsInfEx}] \subsetneq [\mathbf{InfConsEx}]$.

This notation makes it also possible to distinguish the mode of information presentation. If the learner observes the language as solely positive information, we write $[\alpha\mathbf{Txt}\beta]$ for the collections of languages β-learnable by a learner with property α from text. Of course for α and β the original definitions for the setting of learning from text have to be used. All formal definitions for learning from text can be found in [KP14].

2.3 Delayability versus Consistency

In order to facilitate smooth proofs later on, we discuss normal forms for learning from informant. First, we consider the notion of set-drivenness. In Lemma 2.14 we show for delayable learning success criteria, that every collection of languages that is learnable from canonical informant is also learnable by a set-driven learner from arbitrary informant. By Proposition 2.15 this does not hold for consistent **Ex**-learning. This also implies that consistency is a restriction when learning from informant. Moreover, in Lemma 2.17 we observe that only considering total learners does not alter the learnability of a collection of languages in case of a delayable learning success criterion. This does not hold for consistent **Ex**-learning by Proposition 2.18.

2.3.1 Delayability

We now introduce a property of learning restrictions and learning success criteria, which allows general observations, not bound to the setting of **Ex**-learning, since it applies to all of the learning restrictions introduced in Definition 2.7 except consistency.

Definition 2.11. *Denote the set of all unbounded and non-decreasing functions by* $\mathfrak{S}$*, i.e.,* $\mathfrak{S} := \{ \mathfrak{s} : \mathbb{N} \to \mathbb{N} \mid \forall x \in \mathbb{N}\, \exists t \in \mathbb{N} \colon \mathfrak{s}(t) \geq x \text{ and } \forall t \in \mathbb{N} \colon \mathfrak{s}(t+1) \geq \mathfrak{s}(t) \}$*. Then every* $\mathfrak{s} \in \mathfrak{S}$ *is a so called* admissible simulating function.

A predicate $\beta \subseteq \mathfrak{P} \times \mathcal{I}$ *is* delayable*, if for all* $\mathfrak{s} \in \mathfrak{S}$*, all* $I, I' \in \mathcal{I}$ *and all partial functions* $M, M' \in \mathfrak{P}$ *holds: Whenever we have* $\mathrm{pos}(I'[t]) \supseteq \mathrm{pos}(I[\mathfrak{s}(t)])$*,* $\mathrm{neg}(I'[t]) \supseteq \mathrm{neg}(I[\mathfrak{s}(t)])$ *and* $M'(I'[t]) = M(I[\mathfrak{s}(t)])$ *for all* $t \in \mathbb{N}$*, from* $\beta(M, I)$ *we can conclude* $\beta(M', I')$*.*

The unboundedness of the simulating function guarantees $\mathrm{pos}(I) = \mathrm{pos}(I')$ and $\mathrm{neg}(I) = \mathrm{neg}(I')$.

In order to give an intuition for delayability, think of β as a learning restriction or learning success criterion and imagine M to be a learner. Then β is delayable if and only if it carries over from M together with an informant I to all learners M' and informant I' representing a delayed version of M on I. More concretely, as long as the learner M' conjectures $h_{\mathfrak{s}(t)} = M(I[\mathfrak{s}(t)])$ at time t and has, in form of $I'[t]$, at least as much data available as was used by M for this hypothesis, M' with I' is considered a delayed version of M with I.

The next result guarantees that arguing with the just defined properties covers all of the considered learning restrictions but consistency.

Lemma 2.12. *(i) Let* $\delta \in \Delta$ *be a learning restriction. Then* δ *is delayable if and only if* $\delta \neq$ **Cons**.

(ii) Every convergence criterion $\gamma \in \Gamma$ *is delayable.*

(iii) The intersection of finitely many delayable predicates on $\mathfrak{P} \times \mathcal{I}$ *is again delayable. Especially, every learning success criterion* $\beta = \bigcap_{i=0}^{n} \delta_i \cap \gamma$ *with* $\delta_i \in \Delta \setminus \{\mathbf{Cons}\}$ *for all* $i \leq n$ *and* $\gamma \in \Gamma$*,* β *is delayable.*

Proof. We approach (i) by showing, that **Cons** is not delayable. To do so, consider $\mathfrak{s} \in \mathfrak{S}$ with $\mathfrak{s}(t) := \lfloor \frac{t}{2} \rfloor$, $I, I' \in \mathcal{I}$ defined by $I(x) := (\lfloor \frac{x}{2} \rfloor, \mathbb{1}_{2\mathbb{N}}(\lfloor \frac{x}{2} \rfloor))$ and $I'(x) := (x, \mathbb{1}_{2\mathbb{N}}(x))$, where $\mathbb{1}_{2\mathbb{N}}$ stands for the characteristic function of all even natural numbers. By s-m-n there are learners M and M' such that for all $\sigma \in (\mathbb{N} \times \{0,1\})^{<\omega}$

$$W_{M(\sigma)} = \{x \in \mathbb{N} \mid (x \text{ even } \wedge\, x \leq \lfloor \frac{|\sigma|}{2} \rfloor) \vee (x \text{ odd } \wedge\, x > \lfloor \frac{|\sigma|}{2} \rfloor)\}$$

$$W_{M'(\sigma)} = \{x \in \mathbb{N} \mid (x \text{ even } \wedge x \leq \lfloor \frac{|\sigma|}{4} \rfloor) \vee (x \text{ odd } \wedge x > \lfloor \frac{|\sigma|}{4} \rfloor)\}.$$

Further, $\mathbf{Cons}(M, I)$ is easily verified since for all $t \in \mathbb{N}$

$$\text{pos}(I[t]) = \{x \in \mathbb{N} \mid x \text{ even } \wedge x \leq \lfloor \frac{t-1}{2} \rfloor\} \subseteq W_{M(I[t])}$$
$$\text{neg}(I[t]) = \{x \in \mathbb{N} \mid x \text{ odd } \wedge x \leq \lfloor \frac{t-1}{2} \rfloor)\} \subseteq \mathbb{N} \setminus W_{M(I[t])}$$

but on the other hand $\neg\mathbf{Cons}(M', I')$ since for all $t > 2$

$$\text{pos}(I'[t]) = \{x \in \mathbb{N} \mid x \text{ even } \wedge x < t\}$$
$$\not\subseteq \{x \in \mathbb{N} \mid (x \text{ even } \wedge x \leq \lfloor \frac{t}{4} \rfloor) \vee (x \text{ odd } \wedge x > \lfloor \frac{t}{4} \rfloor)\} = W_{M'(I'[t])}.$$

The remaining proofs for (i) and (ii) are straightforward. Basically, for **Dec**, **SDec**, **SMon** and **Caut**, the simulating function $\mathfrak{s}$ being non-decreasing and $M'(I'[t]) = M(I[\mathfrak{s}(t)])$ for all $t \in \mathbb{N}$ would suffice, while for **NU**, **SNU** and **Mon** one further needs that the informant I and I' satisfy $\text{pos}(I) = \text{pos}(I')$. The proof for **WMon** and **Conv** to be delayable, requires all assumptions, but $\mathfrak{s}$'s unboundedness. Last but not least, in order to prove that every convergence criterion $\gamma = \mathbf{Ex}^a_b$, for some $a \in \mathbb{N} \cup \{*\}$ and $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$, carries over to delayed variants, one essentially needs both characterizing properties of $\mathfrak{s}$ and of course $M'(I'[t]) = M(I[\mathfrak{s}(t)])$. Finally, (iii) is obvious. □

2.3.2 Set-driven Learners and Canonical Informant

We start by formally capturing the intuition for a learner being set-driven, given in the introduction.

Definition 2.13 ([WC80]). *A learner M is* set-driven*, for short* $\mathbf{Sd}(M)$*, if for all* $\sigma, \tau \in \mathbb{N} \times \{0,1\}^{<\omega}$

$$(\,\text{pos}(\sigma) = \text{pos}(\tau) \wedge \text{neg}(\sigma) = \text{neg}(\tau)\,) \;\Rightarrow\; M(\sigma) = M(\tau).$$

[Sch84] and [Ful85] showed that set-drivenness is a restriction when learning only from positive information and also the relation between the learning restrictions differ as observed in [KP16].

In the next Lemma we observe that, by contrast, set-drivenness is not a restriction in the setting of learning from informant. Concurrently, we generalize [Gol67]'s observation, stating that considering solely canonical informant to determine learning success does not give more learning power, to arbitrary delayable learning success criteria.

Lemma 2.14. *Let β be a delayable learning success criterion. Then*

$$[\mathbf{Inf}_{\mathrm{can}}\beta] = [\mathbf{SdInf}\beta].$$

Proof. Clearly, we have $[\mathbf{Inf}_{\mathrm{can}}\beta] \supseteq [\mathbf{SdInf}\beta]$. For the other inclusion, let $\mathcal{L}$ be β-learnable by a learner M from canonical informant. We proceed by formally showing that rearranging the input on the initial segment of $\mathbb{N}$, we already have complete information about at that time, is an admissible simulation in the sense of Definition 2.11. Let $L \in \mathcal{L}$ and $I' \in \mathcal{I}(L)$. For every $f \in (\mathbb{N}\times\{0,1\})^{\leq\omega}$, thus especially for I' and all its initial segments, we define $\mathfrak{s}_f \in \mathfrak{S}$ for all t for which $f[t]$ is defined, by

$$\mathfrak{s}_f(t) = \sup\{x \in \mathbb{N} \mid \forall w < x \colon w \in \mathrm{pos}(f[t]) \cup \mathrm{neg}(f[t])\},$$

i.e., the largest natural number x such that for all $w < x$ we know, whether $w \in \mathrm{pos}(f)$. In the following f will either be I' or one of its initial segments, which in any case ensures $\mathrm{pos}(f[t]) \subseteq L$ for all appropriate t. By construction, $\mathfrak{s}_f$ is non-decreasing and if we consider an informant I, since $\mathrm{pos}(I)\cup\mathrm{neg}(I) = \mathbb{N}$, $\mathfrak{s}_I$ is also unbounded. In order to employ the delayability of β, we define an operator $\mathbf{Meth}\colon (\mathbb{N}\times\{0,1\})^{\leq\omega} \to (\mathbb{N}\times\{0,1\})^{\leq\omega}$ such that for every $f \in (\mathbb{N}\times\{0,1\})^{\leq\omega}$ in form of $\mathbf{Meth}(f)$ we obtain a canonically sound version of f. $\mathbf{Meth}(f)$ is defined on all $t < \mathfrak{s}_f(|f|)$ in case f is finite and on every $t \in \mathbb{N}$ otherwise by

$$\mathbf{Meth}(f)(t) := \begin{cases} (t,0), & \text{if } (t,0) \in \mathrm{ran}(f); \\ (t,1), & \text{otherwise.} \end{cases}$$

Intuitively, in $\mathbf{Meth}(f)$ we sortedly and without repetitions sum up all information contained in f up to the largest initial segment of $\mathbb{N}$, f without interruption informs us about. For a finite sequence σ the canonical version $\Sigma(\sigma)$ has length $\mathfrak{s}_\sigma(|\sigma|)$. Now consider the learner M' defined by

$$M'(\sigma) = M(\mathbf{Meth}(\sigma)).$$

Since $I := \mathbf{Meth}(I')$ is a canonical informant for L, we have $\beta(M, I)$. Moreover, for all $t \in \mathbb{N}$ holds $\mathrm{pos}(I[\mathfrak{s}_{I'}(t)]) \subseteq \mathrm{pos}(I'[t])$ and $\mathrm{neg}(I[\mathfrak{s}_{I'}(t)]) \subseteq \mathrm{neg}(I'[t])$ by the definitions of $\mathfrak{s}_{I'}$ and of I using **Meth**. Finally,

$$M'(I'[t]) = M(\mathbf{Meth}(I'[t])) = M(\mathbf{Meth}(I')[\mathfrak{s}_{I'}(t)]) = M(I[\mathfrak{s}_{I'}(t)])$$

and the delayability of β yields $\beta(M', I')$. □

Therefore, while considering delayable learning from informant, looking only at canonical informant already yields the full picture also for set-driven learners. Clearly, the picture is also the same for so-called *partially set-driven learners* that base their hypotheses only on the set and the number of samples.

The next proposition answers the arising question, whether Lemma 2.14 also holds, when requiring the non-delayable learning restriction of consistency, negatively. H denotes the halting problem.

Proposition 2.15. *For* $\mathcal{L} := \{2H \cup 2(H \cup \{x\}) + 1 \mid x \in \mathbb{N}\}$ *holds*

$$\mathcal{L} \in [\mathcal{R}\mathbf{Inf}_{\mathrm{can}}\mathbf{ConsConvSDecSMonEx}] \setminus [\mathbf{InfConsEx}].$$

Particularly, $[\mathbf{InfConsEx}] \subsetneq [\mathbf{Inf}_{\mathrm{can}}\mathbf{ConsEx}]$.

Proof. Let $p : \mathbb{N} \to \mathbb{N}$ be computable such that $W_{p(x)} = 2H \cup 2(H \cup \{x\}) + 1$ for every $x \in \mathbb{N}$ and let h be an index for $2H \cup 2H + 1$. Consider the total learner M defined by

$$M(\sigma) = \begin{cases} p(x), & \text{if } x \text{ with } 2x \in \mathrm{neg}(\sigma) \text{ and } 2x+1 \in \mathrm{pos}(\sigma) \text{ exists;} \\ h, & \text{otherwise} \end{cases}$$

for every $\sigma \in (\mathbb{N} \times \{0,1\})^{<\omega}$. Clearly, M conservatively, strongly decisively and strongly monotonically **Ex**-learns $\mathcal{L}$ from informant and on canonical informant for languages in $\mathcal{L}$ it is consistent.

Now, assume there is a learner M such that $\mathcal{L} \in \mathbf{InfConsEx}(M)$. By Lemma 2.6 there is a locking sequence σ for $2H \cup 2H + 1$. By s-m-n there is a computable function

$$\chi(x) = \begin{cases} 1, & \text{if } M(\sigma) = M(\sigma\hat{}(2x+1, 1)); \\ 0, & \text{otherwise.} \end{cases}$$

By the consistency of M on $\mathcal{L}$, we immediately obtain that χ is the characteristic function for H, a contradiction. □

Note, that there must not be an indexable family witnessing the difference stated in the previous proposition, since every indexable family is consistently and conservatively **Ex**-learnable by enumeration.

Further, *request informant for M and L* are introduced in [Gol67]. As the name already suggests, there is an interaction between the learner and the informant in the sense that the learner decides, about which natural number the informant should inform it next. His observation $[\mathbf{InfEx}] = [\mathbf{Inf}_{\text{can}}\mathbf{Ex}] = [\mathbf{Inf}_{\text{req}}\mathbf{Ex}]$ seems to hold true when facing arbitrary delayable learning success criteria, but fails in the context of the non-delayable learning restriction of consistency.

Since $\mathcal{L}$ in Proposition 2.15 lies in $[\mathbf{Inf}_{\text{can}}\mathbf{Ex}]$, which by Lemma 2.14 equals $[\mathbf{InfEx}]$, we gain that for learning from informant consistent **Ex**-learning is weaker than **Ex**-learning, i.e., $[\mathbf{InfConsEx}] \subsetneq [\mathbf{InfEx}]$.

We now show that, as observed for learning from text in [Jai+99], a consistent behavior regardless learning success cannot be assumed in general, when learning from informant.

Proposition 2.16. *For $\mathcal{L} := \{\mathbb{N}, H\}$ holds*

$$\mathcal{L} \in [\mathcal{R}\mathbf{InfConsConvSDecEx}] \setminus [\mathbf{ConsInfEx}].$$

In particular, $[\mathbf{ConsInfEx}] \subsetneq [\mathbf{InfConsEx}]$.

Proof. Fix an index h for H and an index p for $\mathbb{N}$. The total learner M with

$$M(\sigma) = \begin{cases} p, & \text{if } \operatorname{neg}(\sigma) = \varnothing; \\ h, & \text{otherwise} \end{cases}$$

for every $\sigma \in (\mathbb{N} \times \{0,1\})^{<\omega}$ clearly consistently, conservatively and strongly decisively **Ex**-learns $\mathcal{L}$.

Aiming at the claimed proper inclusion, assume there is a consistent learner M for $\mathcal{L}$ from informant. Since M learns H, by Lemma 2.6, we gain a locking

sequence $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ for M on H, which means $\mathbf{Cons}(\sigma, H)$, $W_{M(\sigma)} = H$ and for all $\tau \in (\mathbb{N}\times\{0,1\})^{<\omega}$ with $\mathbf{Cons}(\tau, H)$ holds $M(\sigma^\frown\tau)\downarrow = M(\sigma)$. By letting

$$\chi(x) := \begin{cases} 1, & \text{if } M(\sigma^\frown(x,1)) = M(\sigma); \\ 0, & \text{otherwise} \end{cases}$$

for all $x \in \mathbb{N}$, we can decide H by the global consistency of M, a contradiction. □

2.3.3 Total Learners

Similar to full-information learning from text we show that for delayable learning restrictions totality is not a restrictive assumption. Basically, the total learner simulates the original learner on the longest initial segment of the input, on which the convergence of the original learner is already visible.

Lemma 2.17. *Let β be a delayable learning success criterion. Then*

$$[\mathbf{Inf}\beta] = [\mathcal{R}\mathbf{Inf}\beta].$$

Proof. Let $\mathcal{L} \in [\mathbf{Inf}\beta]$ and M be a learner witnessing this. Without loss of generality we may assume that $\varnothing \in \mathrm{dom}(M)$. We define the total learner M' by letting $\mathfrak{s}_M : (\mathbb{N}\times\{0,1\})^{<\omega} \to \mathbb{N}$,

$$\sigma \mapsto \sup\{s \in \mathbb{N} \mid s \leq |\sigma| \text{ and } M \text{ halts on } \sigma[s] \text{ after at most } |\sigma| \text{ steps}\}$$

and

$$M'(\sigma) := M(\sigma[\mathfrak{s}_M(\sigma)]).$$

The convention $\sup(\varnothing) = 0$ yields that $\mathfrak{s}_M$ is total and it is computable, since for M only the first $|\sigma|$-many steps have to be evaluated on σ's finitely many initial segments. One could also employ a Blum complexity measure here. Hence, M' is a total computable function.

In order to observe that M' $\mathbf{Inf}\beta$-learns $\mathcal{L}$, let $L \in \mathcal{L}$ and I be an informant for L. By letting $\mathfrak{s}(t) := \mathfrak{s}_M(I[t])$, we clearly obtain an unbounded non-decreasing function, hence $\mathfrak{s} \in \mathfrak{S}$. Moreover, for all $t \in \mathbb{N}$ from $\mathfrak{s}(t) \leq t$ immediately follows

$$\begin{aligned} &\mathrm{pos}(I[\mathfrak{s}(t)]) \subseteq \mathrm{pos}(I[t]),\ \mathrm{neg}(I[\mathfrak{s}(t)]) \subseteq \mathrm{neg}(I[t]) \text{ as well as} \\ &\quad M'(I[t]) = M(I[\mathfrak{s}_M(I[t])]) = M(I[\mathfrak{s}(t)]). \end{aligned}$$

By the delayability of β and with $I' = I$, we finally obtain $\beta(M', I)$. □

By the next proposition also for learning from informant requiring the learner to be total is a restrictive assumption for the non-delayable learning restriction of consistency. For learning from text this was observed in [WZ95] and generalized to δ-delayed consistent learning from text in [AZ08].

Proposition 2.18. *There is a* collection of decidable languages *witnessing*

$$[\mathcal{R}\mathbf{InfConsEx}] \subsetneq [\mathbf{InfConsEx}].$$

Proof. Let o be an index for $\emptyset$ and define for all $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ the learner M by

$$M(\sigma) := \begin{cases} o, & \text{if } \operatorname{pos}(\sigma) = \emptyset; \\ \varphi_{\max(\operatorname{pos}(\sigma))}(\langle\sigma\rangle), & \text{otherwise.} \end{cases}$$

We argue that $\mathcal{L} := \{ L \subseteq \mathbb{N} \mid L \text{ is decidable and } L \in \mathbf{InfConsEx}(M) \}$ is not consistently learnable by a total learner from informant. Assume towards a contradiction M' is such a learner. For a sequence σ of natural numbers we denote by $\overline{\sigma}$ the corresponding canonical finite informant sequence, ending with the highest value σ takes. Further, for a natural number x we denote by $\operatorname{seq}(x)$ the unique element of $(\mathbb{N}\times\{0,1\})^{<\omega}$ with $\langle \operatorname{seq}(x)\rangle = x$. Then by 1-1 ORT there are $e, z \in \mathbb{N}$ and functions $a, b : \mathbb{N}^{<\omega} \to \mathbb{N}$, such that

$$\forall \sigma, \tau \in \mathbb{N}^{<\omega} \; (\sigma \lhd \tau \Rightarrow \max\{a(\sigma), b(\sigma)\} < \min\{a(\tau), b(\tau)\}), \tag{2.1}$$

with the property that for all $\sigma \in \mathbb{N}^{<\omega}$ and all $i \in \mathbb{N}$

$$\begin{aligned} \sigma_0 &= \emptyset; \\ \sigma_{i+1} &= \sigma_i \, ^\frown \begin{cases} a(\sigma_i), & \text{if } M'(\overline{\sigma_i \, ^\frown a(\sigma_i)}) \neq M'(\overline{\sigma_i}); \\ b(\sigma_i), & \text{otherwise;} \end{cases} \\ W_e &= \bigcup_{i\in\mathbb{N}} \operatorname{pos}(\overline{\sigma_i}); \\ \varphi_z(y) &= \begin{cases} 1, & \text{if } y \in \operatorname{pos}(\overline{\sigma_y}); \\ 0, & \text{otherwise;} \end{cases} \end{aligned} \tag{2.2}$$

$$\varphi_{a(\sigma)}(x) = \begin{cases} e, & \text{if } M'(\overline{\sigma \frown a(\sigma)}) \neq M'(\overline{\sigma}) \text{ and} \\ & \forall y \in \mathrm{pos}(\mathrm{seq}(x))\ \varphi_z(y) = 1 \wedge \\ & \forall y \in \mathrm{neg}(\mathrm{seq}(x))\ \varphi_z(y) = 0; \\ \mathrm{ind}(\mathrm{pos}(\mathrm{seq}(x))), & \text{otherwise;} \end{cases}$$

$$\varphi_{b(\sigma)}(x) = \begin{cases} e, & \text{if } \forall y \in \mathrm{pos}(\mathrm{seq}(x))\ \varphi_z(y) = 1 \wedge \\ & \forall y \in \mathrm{neg}(\mathrm{seq}(x))\ \varphi_z(y) = 0; \\ \mathrm{ind}(\mathrm{pos}(\mathrm{seq}(x))), & \text{otherwise;} \end{cases}$$

The operator Θ as stated in 1-1 **ORT** on page 17 is implicit in the equalities. Further, h is also implicitly given by $h(0) = e$, $h(1) = z$ and a, b defined on all remaining even and odd numbers, respectively. To be formally correct, the functions a and b rely on a computable encoding function with computable inverse mapping sequences $\sigma \in \mathbb{N}^{<\omega}$ to natural numbers and vice versa.

Let us now observe why the existence of such e, z, a, b is contradictory. Note that φ_z witnesses W_e's decidability by (2.1) and with this whether $\varphi_{a(\sigma)}$ and $\varphi_{b(\sigma)}$ output e or stick to p depends on $\mathbf{Cons}(\mathrm{seq}(x), W_e)$. Clearly, we have $W_e \in \mathcal{L}$ and thus M' also **InfConsEx**-learns W_e. By the **Ex**-convergence there are $e', j \in \mathbb{N}$, where j is minimal, such that $W_{e'} = W_e$ and for all $i \geq j$ we have $M'(\overline{\sigma_i}) = e'$ and hence $M'(\overline{\sigma_i \frown a(\sigma_i)}) = M'(\overline{\sigma_i})$ by (2.2).

We now argue that $L := \mathrm{pos}(\overline{\sigma_j}) \cup \{a(\sigma_j)\} \in \mathcal{L}$. Let I be an informant for L and $t \in \mathbb{N}$. By (4.1) we observe that M is consistent on I as

$$M(I[t]) = \varphi_{\max(\mathrm{pos}(I[t]))}(\langle I[t] \rangle) = \begin{cases} e, & \text{if } \mathbf{Cons}(I[t], W_e); \\ \mathrm{ind}(\mathrm{pos}(I[t])), & \text{otherwise.} \end{cases}$$

Further, by the choice of j as well as (2.1) and (2.2) we have

$$a(\sigma_j) \notin W_e = W_{e'}, \tag{2.3}$$

and with this $W_{M(I[t])} = L$, if $\mathrm{pos}(I[t]) = L$.

On the other hand M' does not consistently learn L as by the choice of j we obtain $M'(\overline{\sigma_j \frown a(\sigma_j)}) = M'(\overline{\sigma_j}) = e'$ and $\neg\mathbf{Cons}(\overline{\sigma_j \frown a(\sigma_j)}, W_{e'})$ by (2.3), a contradiction. □

2.4 Relations between Delayable Learning Success Criteria

In order to reveal the relations between the delayable learning restrictions in **Ex**-learning from informant, we provide a regularity property of learners, called *syntactic decisiveness*, for **Ex**-learning in Lemma 2.20.

Most importantly, in Proposition 2.21 we acquire that conservativeness and strongly decisiveness do not restrict informant learning. After this, Propositions 2.23 and 2.22 provide that cautious and monotonic learning are incomparable, implying that both these learning settings are strictly stronger than strongly monotonic learning and strictly weaker than unrestricted learning. The overall picture is summarized in Figure 2.2 and stated in Theorem 2.24.

2.4.1 Syntactically Decisive Learning

A further beneficial property, requiring a learner never to *syntactically* return to an abandoned hypothesis, is supplied.

Definition 2.19 ([KP16]). *Let M be a learner, L a language and I an informant for L. We write*

$\mathbf{SynDec}(M, I)$, *if M is* syntactically decisive *on I, i.e.,*

$$\forall r, s, t : (r \leq s \leq t \wedge h_r = h_t) \Rightarrow h_r = h_s.$$

The following easy observation shows that this variant of decisiveness can always be assumed in the setting of **Ex**-learning from informant. This is employed in the proof of our essential Proposition 2.21, showing that conservativeness and strong decisiveness do not restrict **Ex**-learning from informant.

Lemma 2.20. *We have* $[\mathbf{InfEx}] = [\mathbf{SynDecInfEx}]$.

Proof. Since obviously $[\mathbf{SynDecInfEx}] \subseteq [\mathbf{InfEx}]$, it suffices to show that every **InfEx**-learnable collection of languages is also **SynDecEx**-learnable from informant. For, let $\mathcal{L} \in [\mathbf{InfEx}]$ and M witnessing this. In the definition of the learner M', we make use of a one-one computable padding function $\mathrm{pad} : \mathbb{N}\times\mathbb{N} \to \mathbb{N}$ such that $W_p = \mathrm{dom}(\varphi_p) = \mathrm{dom}(\varphi_{\mathrm{pad}(p,x)}) = W_{\mathrm{pad}(p,x)}$ for all $p, x \in \mathbb{N}$. Now,

consider M' defined by

$$M'(\sigma) := \begin{cases} \mathrm{pad}(M(\sigma), |\sigma|), & \text{if } M(\sigma^-) \neq M(\sigma); \\ M'(\sigma), & \text{otherwise.} \end{cases}$$

M' behaves almost like M with the crucial difference, that whenever M performs a mind change, M' semantically guesses the same language as M did, but syntactically its hypothesis is different from all former ones. The padding function's defining property and the assumption that M **InfEx**-learns $\mathcal{L}$ immediately yield the **SynDecInfEx**-learnability of $\mathcal{L}$ by M'. □

Note that **SDec** implies **SynDec**, which is again a delayable learning restriction. Thus by Lemma 2.14, in the proof of Lemma 2.20 we could have also restricted our attention to canonical informant. It is further easy to see that Lemma 2.20 also holds for all other convergence criteria introduced and the simulation does not destroy any of the learning restrictions introduced in Definition 2.7.

2.4.2 Conservative and Strongly Decisive Learning

The following proof for **ConvSDecEx**-learning being equivalent to **Ex**-learning from informant builds on the normal forms of canonical presentations and totality provided in Section 2.3 as well as the regularity property introduced in the last subsection.

Proposition 2.21. *We have* [**InfEx**] = [**InfConvSDecEx**].

Proof. Obviously [**InfEx**] ⊇ [**InfConvSDecEx**] and by the Lemmas 2.14, 2.17 and 2.20 it suffices to show [$\mathcal{R}$**SynDecInfEx**] ⊆ [**Inf**$_{\mathrm{can}}$**ConvSDecEx**].

In the following for every set X and $t \in \mathbb{N}$, let $X[t]$ denote the canonical informant sequence of the first t elements of $\mathbb{N}$.

Now, let $\mathcal{L} \in$ [$\mathcal{R}$**SynDecInfEx**] and M a learner witnessing this. In particular, M is total and on informant for languages in $\mathcal{L}$ we have that M never returns to a withdrawn hypothesis. We want to define a learner M' which mimics the behavior of M, but modified such that, if σ is a locking sequence, then the hypothesis of M' codes the same language as the guess of M. However, if σ is not a locking sequence, then the language guessed by M' should not include data that M changes its mind on in the future. Thus, carefully in form of a recursively defined $\subseteq$-increasing sequence $(A^t_\sigma)_{t\in\mathbb{N}}$ in the guess of M' we only

include the elements of the hypothesis of M that do not cause a mind change of M when looking more and more computation steps ahead. The following formal definitions make sure, this can be done in a computable way.

For every $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$, $t \in \mathbb{N}$ with $t \geq |\sigma|$ and $D \subseteq W^t_{M(\sigma)}$, we let

$$\mathbb{r}^t_\sigma(D) = \min\left\{ |\sigma| \leq r \leq t \;\middle|\; D \subseteq W^r_{M(\sigma)} \right\}.$$

Moreover, we define[1]

$$\mathcal{X}^t_\sigma(D) = \{\, X \subseteq W^t_{M(\sigma)} \mid \max(X) < \inf(W^t_{M(\sigma)} \setminus X), D \subsetneq X \text{ and } M(\sigma) = M(\, W^t_{M(\sigma)}[\, \mathbb{r}^t_\sigma(X)+1 \,]\,) \,\}.$$

In the following we abbreviate $X \subseteq W^t_{M(\sigma)}$ and $\max(X) < \inf(W^t_{M(\sigma)} \setminus X)$ by $X \trianglelefteq W^t_{M(\sigma)}$ and say that X is an initial subset of $W^t_{M(\sigma)}$.

Aiming at providing suitable hypotheses $p(\sigma)$ for the conservative strongly decisive learner M', given σ, we carefully enumerate more and more elements included in $W_{M(\sigma)}$. We are going to start with the positive information provided by σ. Having obtained A^t_σ with $\mathcal{X}^t_\sigma(A^t_\sigma)$ we have a set at hand that contains all initial subsets X of $W^t_{M(\sigma)}$ strictly incorporating A^t_σ, for which M does not differentiate between σ and the appropriate initial segment $W^t_{M(\sigma)}[\, \mathbb{r}^t_\sigma(X)+1 \,]$ of the canonical informant of M's guess on σ. Thus $\mathcal{X}^t_\sigma(A^t_\sigma)$ contains our candidate sets for extending A^t_σ. The length $\mathbb{r}^t_\sigma(X)+1$ of the initial segment is minimal such that X is a subset of $W^{\mathbb{r}^t_\sigma(X)}_{M(\sigma)}$ and at least $|\sigma|$ to assure **Ex**-convergence of the new learner.

For an arbitrary $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ this reads as follows

$$A^0_\sigma = \mathrm{pos}(\sigma);$$

$$\forall t \in \mathbb{N} : A^{t+1}_\sigma = \begin{cases} W^t_{M(\sigma)}, & \text{if } \mathrm{neg}(\sigma) \cap A^t_\sigma \neq \emptyset; \\ \max_\subseteq \mathcal{X}^t_\sigma(A^t_\sigma), & \text{else if } \mathcal{X}^t_\sigma(A^t_\sigma) \neq \emptyset; \\ A^t_\sigma, & \text{otherwise.} \end{cases}$$

Furthermore, using s-m-n, we define $p : (\mathbb{N}\times\{0,1\})^{<\omega} \to \mathbb{N}$ as a one-one

1 We suppose $\inf(\emptyset) = \infty$ for convenience.

function, such that for all $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$

$$W_{p(\sigma)} = \bigcup_{t\in\mathbb{N}} A^t_\sigma. \tag{2.4}$$

In the following, for all $\tau \in (\mathbb{N}\times\{0,1\})^{<\omega}$ we denote by τ' the largest initial segment of τ for which $M'(\tau') = M'(\tau)$, i.e., the last time M' performed a mind change. Finally, we define our new learner M' by

$$M'(\sigma) = \begin{cases} p(\sigma), & \text{if } |\sigma| = 0; \\ p(\sigma), & \text{else if } M((\sigma^-)') \neq M(\sigma) \wedge \neg\mathbf{Cons}(\sigma, A^{|\sigma|}_{(\sigma^-)'}); \\ M'(\sigma^-), & \text{otherwise.} \end{cases}$$

That is, M' follows the mind changes of M once a suitably inconsistent hypothesis has been seen. All hypotheses of M are poisoned in a way to ensure that we can decide inconsistency.

Let us first observe that M' **Ex**-learns every $L \in \mathbf{InfEx}(M)$ from informant. For, let t_0 be minimal such that, for all $t \geq t_0$, $M(L[t]) = M(L[t_0])$. Thus, $e := M(L[t_0])$ is a correct hypothesis for L.

If M' does not make a mind change in or after t_0, then M' converged already before that mind change of M. Thus, let $s_0 < t_0$ be minimal such that for all $t \geq s_0$, $e' := M'(L[s_0]) = M'(L[t])$. As p is one-one and M learns syntactically decisive, we have $M(L[s_0]) \neq M(L[t])$ for all $t \geq t_0$. From $(L[t-1])' = L[s_0]$ and the definition of M' we get $\mathbf{Cons}(L[t], A^t_{L[s_0]})$ for all $t \geq t_0$. Thus, $W_{e'} = L$, because the final hypothesis $W_{e'}$ of M' contains all elements of L and no other by Equation (2.4).

In case M' makes a mind change in or after t_0, let $t_1 \geq t_0$ be the time of that mind change. As M does not perform mind changes after t_0, the learner M' cannot make further mind changes and therefore converges to $e' := p(L[t_1])$. By construction we have $A^t_{L[t_1]} \subseteq W_e = L$ for all $t \in \mathbb{N}$ and with it $W_{e'} \subseteq L$ by Equation 2.4. Towards a contradiction, suppose $W_{e'} \subsetneq L$ and let $x \in L \setminus W_{e'}$ be minimal. By letting s_0 such that $\mathrm{pos}(L[x]) \subseteq A^{s_0}_{L[t_1]}$ and $x \in W^{s_0}_e$, every initial subset of $W^{s_0}_e$ extending $A^{s_0}_{L[t_1]}$ would necessarily contain x. Therefore we have $A^s_{L[t_1]} = A^{s_0}_{L[t_1]}$ and $\mathcal{X}^s_{L[t_1]}(A^{s_0}_{L[t_1]}) = \emptyset$ for all $s \geq s_0$. We obtain the **Ex**-convergence of M' by constructing $s_2 \geq s_0$ with $\mathcal{X}^{s_2}_{L[t_1]}(A^{s_0}_{L[t_1]}) \neq \emptyset$. For this, let $y := \max(A^{s_0}_{L[t_1]} \cup \{x\})$ which implies $A^{s_0}_{L[t_1]} \subsetneq \mathrm{pos}(L[y+1])$. Moreover,

let $s_1 \geq t_1$ be large enough such that $L[y+1] = W_e^{s_1}[y+1]$. Thus, by letting $r := \mathbb{r}^{s_1}_{L[t_1]}(\mathrm{pos}(L[y+1])) + 1$ we gain $r = \mathbb{r}^{s}_{L[t_1]}(\mathrm{pos}(L[y+1])) + 1$ for all $s \geq s_1$, where the latter denotes the time window considered in the third requirement for $\mathrm{pos}(L[y+1]) \in \mathcal{X}^{s}_{L[t_1]}(A^{s_0}_{L[t_1]})$. Furthermore, let $s_2 \geq s_1$ with $L[r] = W_e^{s_2}[r]$. By the definition of r we have $r > t_1 \geq t_0$ and gain

$$\mathrm{pos}(L[y+1]) \trianglelefteq W_e^{s_2}, \quad A^{s_2}_{L[t_1]} = A^{s_0}_{L[t_1]} \subsetneq \mathrm{pos}(L[y+1]) \quad \text{and}$$
$$M(L[t_1]) = e = M(L[r]) = M(\, W_e^{s_2}[r]\,),$$

for short $\mathrm{pos}(L[y+1]) \in \mathcal{X}^{s_2}_{L[t_1]}(A^{s_0}_{L[t_1]})$, implying $\mathcal{X}^{s_2}_{L[t_1]}(A^{s_0}_{L[t_1]}) \neq \emptyset$.

Now we come to prove that M' is conservative on every $L \in \mathbf{InfEx}(M)$. For, let t be such that $M'(L[t]) \neq M'(L[t+1])$. Let $e' := M'(L[t])$ and let $t' \leq t$ be minimal such that $M'(L[t']) = e'$. From the mind change of M' we get $\neg\mathbf{Cons}(L[t+1], A^{t+1}_{L[t']})$. In case it holds $\mathrm{neg}(L[t+1]) \cap A^{t+1}_{L[t']} \neq \emptyset$, since $A^{t+1}_{L[t']} \subseteq W_{e'}$, we would immediately observe $\neg\mathbf{Cons}(L[t+1], W_{e'})$. Therefore, we may assume $\mathrm{pos}(L[t+1]) \setminus A^{t+1}_{L[t']} \neq \emptyset$. Suppose, by way of contradiction, $W_{e'}$ is consistent with $L[t+1]$, i.e., $\mathrm{pos}(L[t+1]) \subseteq W_{e'}$ and $\mathrm{neg}(L[t+1]) \cap W_{e'} = \emptyset$. Then we have $\mathrm{neg}(L[t+1]) \cap A^{s}_{L[t']} = \emptyset$ for all $s \in \mathbb{N}$. Since $\mathrm{pos}(L[t+1]) \subseteq W_{e'}$, there is t_0 minimal such that

$$L[t+1] = A^{t_0+1}_{L[t']}[t+1]. \tag{2.5}$$

We have $\mathrm{neg}(L[t']) \cap A^{t_0}_{L[t']} = \emptyset$ as otherwise $\neg\mathbf{Cons}(L[t+1], W_{e'})$. Because t_0 was minimal, we have $A^{t_0}_{L[t']} \subsetneq A^{t_0+1}_{L[t']}$ and with this $A^{t_0+1}_{L[t']} \in \mathcal{X}^{t_0}_{L[t']}(A^{t_0}_{L[t']})$ by the definition of $A^{t_0+1}_{L[t']}$. In particular, this tells us

$$A^{t_0+1}_{L[t']} \trianglelefteq W^{t_0}_{M(L[t'])} \quad \text{and} \tag{2.6}$$
$$M(L[t']) = M(\, W^{t_0}_{M(L[t'])}[\, \mathbb{r}^{t_0}_{L[t']}(A^{t_0+1}_{L[t']}) + 1\,]\,). \tag{2.7}$$

and therefore with

$$L[t'] \trianglelefteq L[t+1] \overset{(2.5)}{\trianglelefteq} A^{t_0+1}_{L[t']} \overset{(2.6)}{\trianglelefteq} W^{t_0}_{M(L[t'])}[\, \mathbb{r}^{t_0}_{L[t']}(A^{t_0+1}_{L[t']}) + 1\,]$$

by Equation (2.7) and M's syntactic decisivenes we get $M(L[t']) = M(L[t+1])$. Therefore, M' did not make a mind change in $t+1$, a contradiction. □

2.4.3 Completing the Picture of Delayable Learning

The next two propositions show that monotonic and cautious **Ex**-learning are incomparable on the level of indexable families. With Proposition 2.21 this yields all relations between delayable **Ex**-learning success criteria as stated in Theorem 2.24.

We extend the observation of [OSW86] for cautious learning to restrict learning power with the following result. The positive part has already been discussed in the example in the introduction.

Proposition 2.22. *For the* indexable family $\mathcal{L} := \{\mathbb{N} \setminus X \mid X \subseteq \mathbb{N} \text{ finite}\}$ *holds*

$$\mathcal{L} \in [\mathbf{InfMonEx}] \setminus [\mathbf{InfCautBc}].$$

Particularly, $[\mathbf{InfCautEx}] \subsetneq [\mathbf{InfEx}]$.

Proof. In order to approach $\mathcal{L} \notin [\mathbf{InfCautBc}]$, let M be a **InfBc**-learner for $\mathcal{L}$ and I_0 the canonical informant for $\mathbb{N}$. Moreover, let t_0 be such that $W_{M(I_0[t_0])} = \mathbb{N}$. Let I_1 be the canonical informant for $L_1 := \mathbb{N} \setminus \{t_0\}$. Since M learns L_1, there is $t_1 > t_0$ such that $W_{M(I_1[t_1])} = L_1$. We have $I_1[t_0] = I_0[t_0]$ and hence M is not cautiously learning L_1 from I_1.

We now show the **MonEx**-learnability. By s-m-n there is a computable function $p : \mathbb{N} \to \mathbb{N}$ such that for all finite sets X holds $W_{p(\langle X \rangle)} = \mathbb{N} \setminus X$, where $\langle X \rangle$ denotes a canonical code for X as already employed in the proof of Proposition 2.23. We define the learner M by letting for all $\sigma \in \mathbb{N} \times \{0,1\}^{<\omega}$

$$M(\sigma) = p(\langle \mathrm{neg}(\sigma) \rangle).$$

The corresponding intuition is that M includes every natural number in its guess, not explicitly excluded by σ. Clearly, M learns $\mathcal{L}$ and behaves monotonically on $\mathcal{L}$, since for every $X \subseteq \mathbb{N}$ finite, every informant I for $\mathbb{N} \setminus X$ and every $t \in \mathbb{N}$, we have $W_{M(I[t])} \supseteq \mathbb{N} \setminus X$ and therefore $W_{M(I[t])} \cap \mathbb{N} \setminus X = \mathbb{N} \setminus X$. □

This reproves $[\mathbf{InfSMonEx}] \subsetneq [\mathbf{InfMonEx}]$ observed in [LZK96] also on the level of indexable families.

In the next proposition the learner can even be assumed cautious on languages it does not identify. Thus, according to Definition 2.10 we write this success independent property of the learner on the left side of the mode of presentation.

Proposition 2.23. *For the* indexable family

$$\mathcal{L} := \{2X \cup (2(\mathbb{N} \setminus X) + 1) \mid X \subseteq \mathbb{N} \textit{ finite or } X = \mathbb{N}\}$$

holds $\mathcal{L} \in [\mathbf{CautInfEx}] \setminus [\mathbf{InfMonBc}]$.
Particularly, $[\mathbf{InfMonEx}] \subsetneq [\mathbf{InfEx}]$.

Proof. We first show $\mathcal{L} \notin [\mathbf{InfMonBc}]$. Let M be a **InfBc**-learner for $\mathcal{L}$. Further, let I_0 be the canonical informant for $L_0 := 2\mathbb{N} \in \mathcal{L}$. Then there exists t_0 such that $W_{M(I_0[2t_0])} = 2\mathbb{N}$. Moreover, consider the canonical informant I_1 for

$$L_1 := 2\{0, \ldots, t_0\} \cup (2(\mathbb{N} \setminus \{0, \ldots, t_0\}) + 1) \ \in \mathcal{L}$$

and let $t_1 > t_0$ such that $W_{M(I_1[2t_1])} = L_1$. Similarly, we let I_2 be the canonical informant for

$$L_2 := 2\{0, \ldots, t_0, t_1 + 1\} \cup (2(\mathbb{N} \setminus \{0, \ldots, t_0, t_1 + 1\}) + 1) \ \in \mathcal{L}$$

and choose $t_2 > t_1$ with $W_{M(I_2[2t_2])} = L_2$. Since $2(t_1 + 1) \in (L_0 \cap L_2) \setminus L_1$ and by construction $I_2[2t_0] = I_0[2t_0]$ as well as $I_2[2t_1] = I_1[2t_1]$, we obtain

$$2(t_1 + 1) \in W_{M(I_2[2t_0])} \cap L_2 \quad \text{and} \quad 2(t_1 + 1) \notin W_{M(I_2[2t_1])} \cap L_2$$

and therefore M does not learn L_2 monotonically from I_2.

Let us now adress $\mathcal{L} \in [\mathbf{CautInfEx}]$. Fix $p \in \mathbb{N}$ such that $W_p = 2\mathbb{N}$. Further, by s-m-n there is a computable function $q : \mathbb{N} \to \mathbb{N}$ with $W_{q(\langle X \rangle)} = X \cup (2\mathbb{N} \setminus X) + 1$, where $\langle X \rangle$ stands for a canonical code of the finite set X. We define the learner M for all $\sigma \in \mathbb{N} \times \{0, 1\}^{<\omega}$ by

$$M(\sigma) = \begin{cases} p, & \text{if } \operatorname{pos}(\sigma) \subseteq 2\mathbb{N}; \\ q(\langle \operatorname{pos}(\sigma) \cap 2\mathbb{N} \rangle), & \text{otherwise.} \end{cases}$$

Intuitively, M guesses $2\mathbb{N}$ as long as no odd number is known to be in the language L to be learned. If for sure $L \neq 2\mathbb{N}$, then M assumes that all even numbers known to be in L so far are the only even numbers therein.

It is easy to verify that M is computable and by construction it learns $\mathcal{L}$. For establishing the cautiousness, let L be any language, I an informant for L and $s \leq t$. Furthermore, assume $W_{M(I[s])} \neq W_{M(I[t])}$. In case $\operatorname{pos}(I[s]) \not\subseteq 2\mathbb{N}$, we

have $x \in (\mathrm{pos}(I[t]) \cap 2\mathbb{N})$ with $x \notin (\mathrm{pos}(I[s]) \cap 2\mathbb{N})$ and therefore as desired $W_{M(I[t])} \setminus W_{M(I[s])} \neq \emptyset$. Then $\mathrm{pos}(I[s]) \subseteq 2\mathbb{N}$ implies $W_{M(I[s])} = 2\mathbb{N}$ and thus again $W_{M(I[t])} \setminus W_{M(I[s])} \neq \emptyset$. □

We sum up the preceding results in the next theorem and also represent them in Figure 2.2.

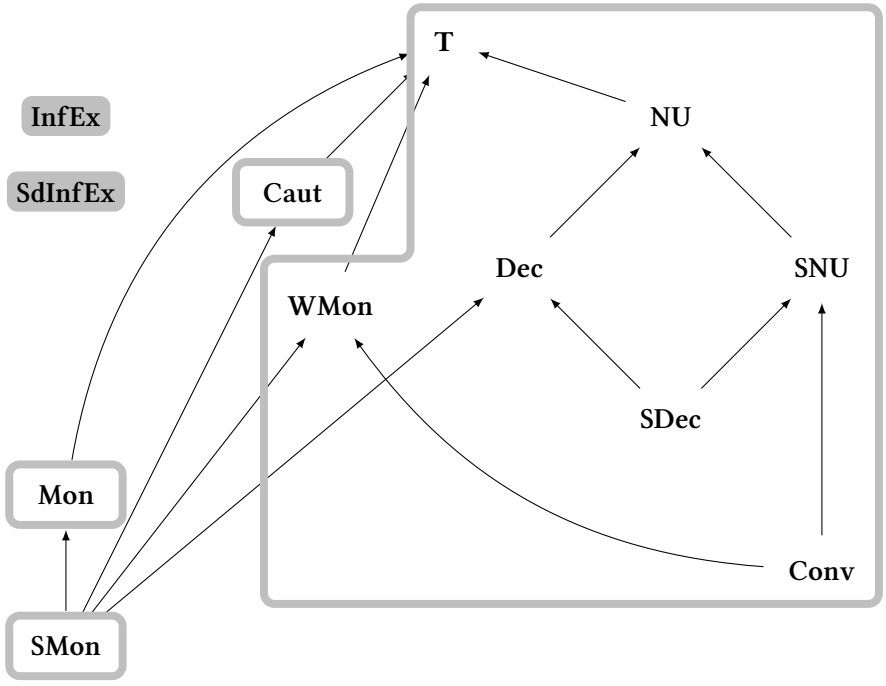

Figure 2.2: Relations between delayable learning restrictions in full-information (explanatory) **Ex**-learning of languages from informant. The implications according to Lemma 2.8 are represented as arrows from bottom to top. Two learning settings are equivalent if and only if they lie in the same grey outlined zone as stated in Theorem 2.24.

Theorem 2.24. *We have*

(i) $\forall \delta \in \{\mathbf{Conv}, \mathbf{Dec}, \mathbf{SDec}, \mathbf{WMon}, \mathbf{NU}, \mathbf{SNU}\}$:

$$[\mathbf{Inf}\delta\mathbf{Ex}] = [\mathbf{InfEx}].$$

(ii) [**InfMonEx**] $\perp$ [**InfCautEx**].

Proof. The first part is an immediate consequence of Proposition 2.21 and so is the second part of the Propositions 2.22 and 2.23. □

2.5 Further Research

According to [OSW86] requiring the learner to base its hypothesis only on the previous one and the current datum, makes **Ex**-learning harder. While the relations between the delayable learning restrictions for these so called *iterative learners* in the presentation mode of solely positive information has been investigated in [Jai+16], so far this has not been done when learning from informant. For indexable families, this was already of interest to [ST92], [LG03] and [JLZ07b]. Moreover, **Conv** restricts iterative learning from informant, [JLZ07b], and in Chapter 4 we show that also **SNU** does. Memory-restricted learning, as investigated in Part II, is of special interest as it models the behavior of neural networks and other machine learning paradigms.

Further improvements to the model would be more problem specific hypothesis spaces, a probabilistic presentation of the data and other convergence criteria. We address some of these issues in Chapter 3.

3 Approximations, Vacillations and another Hypothesis Space

We continue our investigations of learning from informant, a model for human and machine learning introduced by E. M. Gold. We answer naturally arising questions originating in results on learning functions and learning formal languages from solely positive information.

More concretely, we show that a highly restricted form of learning formal languages from informant does not imply the learnability from solely positive information.

We also obtain an anomalous *hierarchy* when allowing for an increasing finite number of *anomalies* of the hypothesized language by the learner compared with the language to be learned.

In contrast to the vacillatory hierarchy for learning from solely positive information, we observe a *duality* depending on whether infinitely many *vacillations* between different (almost) correct hypotheses are still considered a successful learning behavior.

Finally, we suggest a hypothesis space more suitable for symmetric classification tasks and observe the relations between the corresponding delayable learning success criteria.

3.1 Introduction

We are doing research in the area of *inductive inference* and investigate the learnability of formal languages. This branch of algorithmic learning theory has connections to computability theory, complexity theory, cognitive science, machine learning, and more generally artificial intelligence. The task is to generalize from labeled training samples by providing a classifier for deciding whether a given word belongs to a certain concept. *Learning from informant* was introduced in [Gol67] and further investigated in several publications, including [BB75], [Bār77], [Höl+17] and [Gao+19].

Following [Gol67] the learner is modelled by a computable function. It successively receives sequences incorporating more and more data and outputs a

hypothesis every time. This source of labeled data is called an *informant*, which is supposed to be *complete in the limit*. Learning is considered successful, if after some finite time the learners' hypotheses yield good enough approximations to the target language. The original and most common learning success criterion is called **Ex**-*learning* and additionally requires that the learner eventually settles on exactly one correct hypothesis, which precisely captures the words in the language to be learned. We also consider approximations. As a single language can be learned by a constant learner, we wonder whether there is a learner successful on all languages in a fixed concept class.

For example, the concept class $\mathcal{L} = \{ \mathbb{N} \setminus X \mid X \subseteq \mathbb{N} \text{ finite} \}$ is learnable from informant but not from purely positive information.

Learning from exclusively positive information, so-called *text*, plays a prominent role in Inductive Inference and a lot of references are given in Section 2.1.

We add to a careful investigation on how informant and text learning relate to each other in [LZ93]. We show that even for the most restrictive delayable learning success criterion when **Ex**-learning from informant there is a collection of recursive languages learnable in this setting that is not learnable from text.

Regarding approximations, admitting for finitely many anomalies a, i.e. elements of the symmetric difference of the hypothesized and the target concept, yields the anomalous hierarchy for learning from text in [CL82]. We provide an equivalence between learning collections of functions from enumerations of their graphs to learning the languages encoding their graphs from informant. This allows us to transfer the anomalous hierarchy when learning functions in [Bār74] and [CS83] to the setting of learning from informant and therefore obtain a hierarchy

$$[\mathbf{InfEx}] \subsetneq \ldots \subsetneq [\mathbf{InfEx}^a] \subsetneq [\mathbf{InfEx}^{a+1}] \subsetneq \ldots.$$

[Cas99] observed the vacillatory hierarchy for learning from text. Thereby in the limit a vacillation between b many (almost) correct descriptions is allowed, where $b \in \mathbb{N}_{>0} \cup \{\infty\}$. In contrast we observe a duality by showing that, when learning from informant, requiring the learner to eventually output exactly one correct enumeration procedure is as powerful as allowing any finite number of correct descriptions in the limit. Furthermore, $b = \infty$, known as behaviorally correct (**Bc**) learning, gives us strictly more learning power. In particular, we

obtain for all $b \in \mathbb{N}_{>0}$

$$[\mathbf{InfEx}] = \ldots = [\mathbf{InfEx}_b] = [\mathbf{InfEx}_{b+1}] = \ldots \subsetneq [\mathbf{InfBc}].$$

We also compare learning settings in which both a and b do not take their standard values 0 and 1, respectively.

While most research in inductive inference regarding learning formal languages focuses on the W-hypothesis space we argue that the hypothesis space of total computable functions might be more suitable for symmetric machine learning tasks. We derive the complete map with respect to this setting. It equals the one observed for the W-hypothesis space derived in Section 2.4.

In Section 3.2 we generalize the above mentioned result in [Gol67], namely **Ex**-learning from text to be harder than **Ex**-learning from informant by further restricting learning from informant. In Section 3.3 we provide the aforementioned anomalous hierarchy and vacillatory duality. Section 3.4 contains the relations between the delayable learning success criteria for the more symmetric hypothesis space of recursive languages.

We kept every section as self-contained as possible. Unavoidably, all sections build on Section 2.2 in Chapter 2.

3.2 Outperforming Learning from Text

Already in [Gol67] it was observed that $[\mathbf{TxtEx}] \subsetneq [\mathbf{InfEx}]$. Later on in [LZ93] the interdependencies when considering the different monotonicity learning restrictions were investigated. For instance, they showed that there exists an indexable family $\mathcal{L} \in [\mathbf{InfMonEx}] \setminus [\mathbf{TxtEx}]$ and in contrast that for indexable families **InfSMonEx**-learnability implies **TxtEx**-learnability. We show that this inclusion fails on the level of families of recursive languages even with all learning restrictions at hand.

Proposition 3.1. *For the class of recursive languages*

$$\mathcal{L} := \{2(L \cup \{x\}) \cup 2L + 1 \mid L \text{ is recursive} \wedge W_{\min(L)} = L \wedge x \geq \min(L)\}$$

holds $\mathcal{L} \in [\mathbf{InfConvSDecSMonEx}] \setminus [\mathbf{TxtEx}]$.

Proof. Let p_m denote an index for $2W_m \cup 2W_m + 1$ and $p_{m,x}$ an index for $2(W_m \cup \{x\}) \cup 2W_m + 1$. The learner M will look for the minimum of the presented set

and moreover try to detect the exception x, in case it exists. Thus, it checks for all m such that $2m \in \mathrm{pos}(\sigma)$ or $2m+1 \in \mathrm{pos}(\sigma)$ whether for all $k < m$ holds $2k \in \mathrm{neg}(\sigma)$ or $2k+1 \in \mathrm{neg}(\sigma)$. In case m has this property relative to σ, we write $\min_L(m,\sigma)$ as m might be the minimum of the language presented. Further, M tries to find x such that $2x \in \mathrm{pos}(\sigma)$ and $2x+1 \in \mathrm{neg}(\sigma)$ and we abbreviate by $\mathrm{exc}_L(x,\sigma)$ that x is such an exception. Consider the learner M for all $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ defined by

$$M(\sigma) = \begin{cases} \mathrm{ind}(\varnothing), & \text{if there is no } m \text{ with } \min_L(m,\sigma); \\ p_m, & \text{if } \min_L(m,\sigma) \text{ and there is no } x \text{ with } \mathrm{exc}_L(x,\sigma); \\ p_{m,x}, & \text{if } \min_L(m,\sigma) \text{ and } x \text{ is minimal with } \mathrm{exc}_L(x,\sigma). \end{cases}$$

Clearly, M conservatively, strongly decisively and strongly monotonically **Ex**-learns $\mathcal{L}$.

To observe $\mathcal{L} \notin [\mathbf{TxtEx}]$, assume there exists M such that $\mathcal{L} \in \mathbf{TxtEx}(M)$. By ORT there exists $e \in \mathbb{N}$ such that for all $i \in \mathbb{N}$

$$A_\sigma(i) = \{\, k \in \mathbb{N} \mid M(\sigma) \neq M(\sigma^\frown(2e+4i)^k) \,\};$$
$$B_\sigma(i) = \{\, k \in \mathbb{N} \mid M(\sigma) \neq M(\sigma^\frown(2e+4i+2)^k) \,\};$$
$$\sigma_0 = (2e, 2e+1);$$
$$\sigma_{i+1} = \begin{cases} \sigma_i, & \text{if } A_{\sigma_i}(i) = B_{\sigma_i}(i) = \varnothing \\ & \text{or } i > 0 \wedge \sigma_{i-1} = \sigma_i; \\ \sigma_i^\frown(2e+4i)^{\inf(A_{\sigma_i}(i))^\frown}(2e+4i+1), & \text{if } A_{\sigma_i}(i) \neq \varnothing \\ & \wedge \inf(A_{\sigma_i}(i)) \leq \\ & \qquad \inf(B_{\sigma_i}(i))); \\ \sigma_i^\frown(2e+4i+2)^{\inf(B_{\sigma_i}(i))^\frown}(2e+4i+3), & \text{if } B_{\sigma_i}(i) \neq \varnothing \\ & \wedge \inf(B_{\sigma_i}(i)) < \\ & \qquad \inf(A_{\sigma_i}(i)); \end{cases}$$
$$W_e = \bigcup_{i\in\mathbb{N}} \{n \mid 2n+1 \in \mathrm{ran}(\sigma_i)\}.$$

Intuitively, the program on input n successively computes σ_i until it finds the minimal $x \geq 2n+1$ in its range; it halts if and only if x is found and $x = 2n+1$.

W_e is recursive, because we can decide it along the construction of the σ_i.

Thus, $2W_e \cup 2W_e + 1 \in \mathcal{L}$. If for some index i holds $\sigma_{i+1} = \sigma_i$, then M fails to learn $2(W_e \cup \{e + 2i\}) \cup 2W_e + 1$ or $2(W_e \cup \{e + 2i + 1\}) \cup 2W_e + 1$. On the other hand, if there is no such i, by letting $T := \bigcup_{i \in \mathbb{N}} \sigma_i$ we obtain a text for $2W_e \cup 2W_e + 1$, on which M performs infinitely many mindchanges. □

3.3 Anomalous Hierarchy and Vacillatory Duality

We compare the convergence criteria $\mathbf{Ex}^a_b$ from Definition 2.3 for different parameters $a \in \mathbb{N} \cup \{*\}$ and $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$. The duality depending on whether $b = \infty$ for fixed a follow from the Propositions 3.4 and 3.5.

3.3.1 Anomalous Hierarchy

Beneficial for analyzing the anomalous hierarchy for informant learning are results from function learning. When learning collections of recursive functions, a text for the graph of the respective function f is presented to the learner and it wants to infer a program code p such that φ_p is a good enough approximation to f. More formally, $f =^a \varphi_p$ if and only if $|\{ x \mid f(x) \neq \varphi_p(x) \}| \leq a$. We denote the associated learning criteria in the form $[\mathbf{FnEx}^a_b]$.

By the next lemma, collections of functions separating two convergence criteria in the associated setting yield a separating collection for the respective convergence criteria, when learning languages from informant.

In the following we make use of a computable bijection $\langle ., . \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ with its computable inverses $\pi_1, \pi_2 : \mathbb{N} \to \mathbb{N}$ such that $x = \langle \pi_1(x), \pi_2(x) \rangle$ for all $x \in \mathbb{N}$.

Lemma 3.2. *For $f \in \mathcal{R}$ let $L_f := \{ \langle x, f(x) \rangle \mid x \in \mathbb{N} \}$ denote the language encoding its graph. Let $a \in \mathbb{N} \cup \{*\}$ and $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$. Then for every $\mathcal{F} \subseteq \mathcal{R}$ we define $\mathcal{L}_\mathcal{F} = \{ L_f \mid f \in \mathcal{F} \}$ and obtain*

$$\mathcal{F} \in [\mathbf{FnEx}^a_b] \quad \Leftrightarrow \quad \mathcal{L}_\mathcal{F} \in [\mathbf{InfEx}^a_b].$$

Proof. Let a, b and $\mathcal{F}$ be as stated. First, assume there is a learner M on function sequences such that $\mathcal{F} \in \mathbf{FnEx}^a_b(M)$. In order to define the learner M' acting on informant sequences and returning W-indices, we employ the following procedure for obtaining a W-code $G(p)$ for L_{φ_p}, when given a φ-code p:

> Given input n, interpreted as $\langle x, y\rangle$, let the program encoded by p run on $x = \pi_1(n)$. If it halts and returns $y = \pi_2(n)$, then halt; otherwise loop.

The learner M' acts on $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ by

$$M'(\sigma) := G(M(\text{decode}(\text{pos}(\sigma)))),$$

where $\text{decode}(\text{pos}(\sigma))$ denotes the from σ uniformly computable sequence τ with $\tau(i) = (\pi_1(n_i), \pi_2(n_i))$ for all $i < |\text{pos}(\sigma)| = |\tau|$, where $(n_i)_{i<|\text{pos}(\sigma)|}$ denotes the enumeration of $\text{pos}(\sigma)$ according to σ. By construction, $\mathcal{L}_{\mathcal{F}} \in \mathbf{InfEx}^a_b(M')$ as G preserves the number of anomalies.

For the other claimed direction let M be a learner on informant sequences with $\mathcal{L}_{\mathcal{F}} \in \mathbf{InfEx}^a_b(M)$. As above we employ a computable function that for every $f \in \mathcal{R}$ transforms a W-index p for L_f into a φ-index $H(p)$ such that $\varphi_{H(p)} = f$. Thereby, we interpret each natural number i as $\langle\langle u, v\rangle, t\rangle$ and check whether φ_p halts on $\langle u, v\rangle$ in at most t steps of computation. If so, we check whether u is the argument x we want to compute $f(x)$ for and in case the answer is yes, we return v.

> Given input x, for $i = 0$ till ∞ do the following: If $\Phi_p(\pi_1(i)) \leq \pi_2(i)$ and $\pi_1(\pi_1(i)) = x$, then return $\pi_2(\pi_1(i))$; otherwise increment i.

Before defining M', we argue that H preserves the number of anomalies. Let $x, p \in \mathbb{N}$ be such that $f(x) \neq \varphi_{H(p)}(x)$. Then $\langle x, \varphi_{H(p)}(x)\rangle \notin L_f$. On the other hand, by the definition of H we have $\Phi_p(\langle x, \varphi_{H(p)}(x)\rangle)\downarrow$ and therefore $\langle x, \varphi_{H(p)}(x)\rangle \in W_p \setminus L_f$.

We define the learner M' on $\sigma \in (\mathbb{N}\times\mathbb{N})^{<\omega}$ by

$$M'(\sigma) := H(M(\hat{\sigma})),$$

where we transform $\sigma = ((x_0, f(x_0)), \ldots, (x_{|\sigma|-1}, f(x_{|\sigma|-1})))$ into an informant sequence $\hat{\sigma}$ of length $\widehat{|\sigma|} := \max\{j \mid \forall i < j\ \pi_1(i) < |\sigma|\}$ by letting

$$\hat{\sigma}(i) := \begin{cases} (\langle x_{\pi_1(i)}, \pi_2(i)\rangle, 1) & \text{if } \sigma(\pi_1(i)) = (x_{\pi_1(i)}, \pi_2(i)) \\ (\langle x_{\pi_1(i)}, \pi_2(i)\rangle, 0) & \text{otherwise} \end{cases}$$

for all $i < \widehat{|\sigma|}$. Note that for every $f \in \mathcal{R}$ and every $T \in \mathbf{Txt}(f)$ by letting $I_T := \bigcup_{j\in\mathbb{N}} \widehat{T[j]}$, we obtain an informant for L_f. We show $(\langle x, f(x)\rangle, 1) \in I_T$ for

every $x \in \mathbb{N}$ and leave the other details to the reader. Let $x \in \mathbb{N}$ and i minimal, such that $(x, f(x)) \in \operatorname{ran}(T[i])$, i.e., $x_{i-1} = x$. Further, let j be such that $i \leq \hat{j}$. Then clearly

$$I_T(\langle i-1, f(x)\rangle) = \widehat{T[j]}(\langle i-1, f(x)\rangle) = (\langle x, f(x)\rangle, 1).$$

In a nutshell, $\mathcal{F} \in \mathbf{FnEx}^a_b(M')$ as H preserves the number of anomalies. □

With this we obtain a hierarchy of learning restrictions.

Proposition 3.3. *Let $b \in \{1, \infty\}$. Then*

(i) for all $a \in \mathbb{N}$ holds $[\mathbf{InfEx}^a_b] \subsetneq [\mathbf{InfEx}^{a+1}_b]$,

(ii) $\bigcup_{a\in\mathbb{N}} [\mathbf{InfEx}^a_b] \subsetneq [\mathbf{InfEx}^*_b]$,

(iii) $[\mathbf{InfEx}^*] \subsetneq [\mathbf{InfBc}]$.

Proof. By Lemma 3.2 this results transfer from the corresponding observations for function learning in [Bār74] and [CS83]. □

In particular, we have

$$\begin{aligned}[\mathbf{InfEx}] \subsetneq \ldots \subsetneq [\mathbf{InfEx}^a] \subsetneq [\mathbf{InfEx}^{a+1}] \subsetneq \ldots \\ \subsetneq \bigcup_{a\in\mathbb{N}} [\mathbf{InfEx}^a] \subsetneq [\mathbf{InfEx}^*] \\ \subsetneq [\mathbf{InfBc}] \subsetneq \ldots \subsetneq [\mathbf{InfBc}^a] \subsetneq [\mathbf{InfBc}^{a+1}] \subsetneq \ldots \\ \subsetneq \bigcup_{a\in\mathbb{N}} [\mathbf{InfBc}^a] \subsetneq [\mathbf{InfEx}^*_\infty].\end{aligned}$$

Lemma 3.2 obviously also holds when considering $\mathbf{TxtEx}^a_b$-learning languages, where the construction of the text sequence from the informant sequence is folklore. This reproves the results in [CL82].

3.3.2 Duality of the Vacillatory Hierarchy

In Proposition 3.3 we already observed a hierarchy, when varying the number of anomalies and will now show that allowing the learner to vacillate between finitely many correct hypothesis in the limit does not give more learning power. On the contrary, only requiring semantic convergence, i.e., allowing infinitely

many correct hypotheses in the limit, does allow to learn more collections of languages even with an arbitrary semantic learning restriction at hand. This contrasts the results in language learning from text in [Cas99], observing for every $a \in \mathbb{N} \cup \{*\}$ a hierarchy

$$[\mathbf{TxtEx}^a] \subsetneq \ldots \subsetneq [\mathbf{TxtEx}^a_b] \subsetneq [\mathbf{TxtEx}^a_{b+1}] \subsetneq \ldots \subsetneq \bigcup_{b \in \mathbb{N}_{>0}} [\mathbf{TxtEx}^a_b] \subsetneq [\mathbf{TxtEx}^a_*] \subseteq [\mathbf{TxtBc}^a].$$

We separate **InfEx**- and **InfBc**-learning at the level of families of recursive languages. As every indexable family of recursive languages is **Ex**-learnable from informant by enumeration, the result is optimal.

Proposition 3.4. *For the collection of recursive languages*

$$\mathcal{L} = \{L \cup \{x\} \mid L \subseteq \mathbb{N} \text{ is recursive} \wedge W_{\min(L)} = L \wedge x \geq \min(L)\}$$

holds $\mathcal{L} \in [\mathbf{InfSMonBc}] \setminus [\mathbf{InfEx}]$.

Proof. By Lemma 2.14 it suffices to show

$$\mathcal{L} \in [\mathbf{Inf}_{\text{can}}\mathbf{SMonBc}] \setminus [\mathbf{Inf}_{\text{can}}\mathbf{Ex}].$$

By s-m-n there are $p : \mathbb{N} \times \{0,1\}^{<\omega} \times \mathbb{N} \to \mathbb{N}$ and a learner M such that for all $\sigma \in \mathbb{N} \times \{0,1\}^{<\omega}$ and $x \in \mathbb{N}$

$$W_{p(\sigma,x)} = W_{\min(\text{pos}(\sigma))} \cup \{x\} \text{ and}$$

$$M(\sigma) = \begin{cases} o, & \text{if } \text{pos}(\sigma) = \emptyset; \\ \min(\text{pos}(\sigma)), & \text{else if } \text{pos}(\sigma) \setminus W^{|\sigma|}_{\min(\text{pos}(\sigma))} = \emptyset; \\ p(\sigma, x), & \text{else if } x = \min(\text{pos}(\sigma) \setminus W^{|\sigma|}_{\min(\text{pos}(\sigma))}); \end{cases}$$

where o refers to the canonical index for the empty set. Let $L \cup \{x\} \in \mathcal{L}$ with $L \subseteq \mathbb{N}$ recursive, $W_{\min(L)} = L$ and $x \geq \min(L)$ and let I be the canonical informant for $L \cup \{x\}$. Then for all $t > \min(L)$ we have $W_{\min(\mathrm{pos}(I[t]))} = W_{\min(L)} = L$. Further, let m be minimal such that $\{y \in L \mid y < x\} \subseteq W^m_{\min(L)}$. Since $x \geq \min(L)$ the construction yields for all $t \in \mathbb{N}$

$$W_{h_t} = \begin{cases} \varnothing, & \text{if } t \leq \min(L); \\ L, & \text{else if } \min(L) \leq t < \max\{x+1, m\}; \\ L \cup \{x\}, & \text{otherwise.} \end{cases}$$

This can be easily verified, since in case $y \in L$ we have $L = L \cup \{y\}$ and establishes the $\mathbf{Inf}_{\mathrm{can}}\mathbf{SMonBc}$-learnability of $\mathcal{L}$ by M.

In order to approach $\mathcal{L} \notin [\mathbf{Inf}_{\mathrm{can}}\mathbf{Ex}]$, assume to the contrary that there is a learner M that $\mathbf{Inf}_{\mathrm{can}}\mathbf{Ex}$-learns $\mathcal{L}$. By Lemma 2.17 M can be assumed total. We first define a recursive language L with $W_{\min(L)} = L$ helpful for showing that not all of $\mathcal{L}$ is $\mathbf{Inf}_{\mathrm{can}}\mathbf{Ex}$-learned by M. In order to do so, for every canonical $\sigma \in \mathbb{N}\times\{0,1\}^{<\omega}$ we define sets $A^0_\sigma, A^1_\sigma \subseteq \mathbb{N}$. For this let I^0_σ stand for the canonical informant of $\mathrm{pos}(\sigma)$, whereas I^1_σ denotes the canonical informant of $\mathrm{pos}(\sigma) \cup \{|\sigma|\}$. In A^0_σ we collect all $t > |\sigma|$ for which M's hypothesis on $I^0_\sigma[t]$ is different from $M(\sigma)$. Similarly, in A^1_σ we capture all $t > |\sigma|$ such that M on $I^1_\sigma[t]$ makes a guess different from $M(\sigma)$. Formally, this reads as follows

$$A^0_\sigma := \{\, t \in \mathbb{N} \mid t > |\sigma| \wedge M(I^0_\sigma[t]) \neq M(\sigma) \,\},$$
$$A^1_\sigma := \{\, t \in \mathbb{N} \mid t > |\sigma| \wedge M(I^1_\sigma[t]) \neq M(\sigma) \,\}.$$

As σ is canonical, for every $t > |\sigma|$

$$I^0_\sigma[t] = \sigma^\frown(\,(|\sigma|,0),(|\sigma|{+}1,0),\ldots,(t-1,0)\,),$$
$$I^1_\sigma[t] = \sigma^\frown(\,(|\sigma|,1),(|\sigma|{+}1,0),\ldots,(t-1,0)\,).$$

By ORT there exists $p \in \mathbb{N}$ such that[2]

$$\sigma_0 = (\,(0,0),\ldots,(p-1,0),(p,1)\,),$$
$$\forall i \in \mathbb{N}\colon \sigma_{i+1} = \begin{cases} \sigma_i, & \text{if } A^0_{\sigma_i} = A^1_{\sigma_i} = \emptyset; \\ I^0_{\sigma_i}[\min(A^0_{\sigma_i})], & \text{if } \inf(A^0_{\sigma_i}) \leq \inf(A^1_{\sigma_i}); \\ I^1_{\sigma_i}[\min(A^1_{\sigma_i})], & \text{otherwise;} \end{cases}$$

$$W_p = \bigcup_{i \in \mathbb{N}} \mathrm{pos}(\sigma_i).$$

Intuitively, the program p on input x halts if $x = p$ or in the successive construction of the sequence $(\sigma_i)_{i \in \mathbb{N}}$ there is j with $|\sigma_j| > x$ and $\sigma_j(x) = (x, 1)$. Hence, $p = \min(W_p)$ and W_p is recursive, which immediately yields $L := W_p \in \mathcal{L}$. Further, for every $i \in \mathbb{N}$ from $\sigma_i \neq \sigma_{i+1}$ follows $M(\sigma_i) \neq M(\sigma_{i+1})$. Aiming at a contradiction, let I be the canonical informant for L, which implies $\bigcup_{i \in \mathbb{N}} \sigma_i \trianglelefteq I$. Since M **Ex**-learns L and thus does not make infinitely many mind changes on I, there exists $i_0 \in \mathbb{N}$ such that for all $i \geq i_0$ we have $\sigma_i = \sigma_{i_0}$. But then for all $t > |\sigma_{i_0}|$ holds

$$M(I^0_{\sigma_{i_0}}[t]) = M(\sigma_{i_0}) = M(I^1_{\sigma_{i_0}}[t]),$$

thus M does not learn at least one of $L = \mathrm{pos}(\sigma_{i_0})$ and $L \cup \{|\sigma_{i_0}|\}$ from their canonical informant. On the other hand both of them lie in $\mathcal{L}$ and therefore, M had not existed in the beginning. □

Since allowing infinitely many different correct hypotheses in the limit gives more learning power, the question arises, whether finitely many hypotheses already allow to learn more collections of languages. The following proposition shows that, as observed in [BP73] and [CS83] for function learning, the hierarchy of vacillatory learning collapses when learning languages from informant.

Proposition 3.5. *Let $a \in \mathbb{N} \cup \{*\}$. Then $[\mathbf{InfEx}^a] = [\mathbf{InfEx}^a_*]$.*

Proof. Clearly, $[\mathbf{InfEx}^a] \subseteq [\mathbf{InfEx}^a_*]$. For the other inclusion let $\mathcal{L}$ be in $[\mathbf{InfEx}^a_*]$ and M a learner witnessing this. By Lemma 2.17 we assume that M is total. In the construction of the $\mathbf{Ex}^a$-learner M', we employ the recursive function $\Xi : (\mathbb{N} \times \{0,1\})^{<\omega} \times \mathbb{N} \to \mathbb{N}$, which given $\sigma \in (\mathbb{N} \times \{0,1\})^{<\omega}$ and $p \in \mathbb{N}$ alters p such that $W^{|\sigma|}_{\Xi(\sigma,p)} \cap \mathrm{neg}(\sigma) = \emptyset$ and moreover, if $\sigma \trianglelefteq \tau$ are such that $W^{|\sigma|}_p \cap \mathrm{neg}(\sigma) = W^{|\tau|}_p \cap \mathrm{neg}(\tau)$, then $\Xi(\sigma, p) = \Xi(\tau, p)$. One way to do this is by letting $\Xi(\sigma, p)$ denote the unique program, which given x successively checks, whether $x = y_i$, where $(y_i)_{i < |\mathrm{neg}(\sigma)|}$ is the increasing enumeration of $\mathrm{neg}(\sigma)$. As soon as the answer is positive, the program goes into a loop. Otherwise it

2 Again we use the convention $\inf(\emptyset) = \infty$.

executes the program encoded in p on x, which yields

$$\varphi_{\Xi(\sigma,p)}(x) = \begin{cases} \uparrow, & \text{if } x \in \mathrm{neg}(\sigma); \\ \varphi_p(x), & \text{otherwise.} \end{cases}$$

Now, M' works as follows:

I. Compute $p_i := M(\sigma[i])$ for all $i \leq |\sigma|$.

II. Withdraw all p_i with the property $|\mathrm{neg}(\sigma) \cap W_{p_i}^{|\sigma|}| > a$.

III. Define $M'(\sigma)$ to be a code for the program coresponding to the union vote of all $\Xi(\sigma, p_i)$, for which p_i was not withdrawn in the previous step:

> Given input x, for n from 0 till ∞ do the following: If $i := \pi_1(n) \leq |\sigma|$, $|\mathrm{neg}(\sigma) \cap W_{p_i}^{|\sigma|}| \leq a$ and $\Phi_{\Xi(\sigma,p_i)}(x) \leq \pi_2(n)$, then return 0; otherwise increment n.

This guarantees

$$\varphi_{M'(\sigma)}(x) = \begin{cases} 0, & \text{if } \exists\, i \leq |\sigma| \,(\, |\mathrm{neg}(\sigma) \cap W_{p_i}^{|\sigma|}| \leq a \;\wedge\; \varphi_{\Xi(\sigma,p_i)}(x)\!\downarrow\,); \\ \uparrow, & \text{otherwise.} \end{cases}$$

Intuitively, $M'(\sigma)$ eliminates all membership errors in guesses of M on initial segments of σ, not immediately violating the allowed number of anomalies, and then asks whether one of them converges on the input, which implies

$$W_{M'(\sigma)} = \bigcup_{i \leq |\sigma|, |\mathrm{neg}(\sigma) \cap W_{p_i}^{|\sigma|}| \leq a} W_{\Xi(\sigma, M(\sigma[i]))}.$$

In order to show $\mathcal{L} \in \mathbf{InfEx}^a(M')$, let $L \in \mathcal{L}$ and $I \in \mathbf{Inf}(L)$. As $\mathcal{L} \in \mathbf{Ex}_*^a(M)$, there is t_0 such that all of M's hypotheses on I are in $\{h_s \mid s \leq t_0\}$ and additionally $|\, W_{h_s}^{t_0} \cap \mathbb{N} \setminus L\,| > a$ for all $s \leq t_0$ with $|\, W_{h_s} \cap \mathbb{N} \setminus L| > a$. Moreover, we can assume that for all $s \leq t_0$ with $|\, W_{h_s} \cap \mathbb{N} \setminus L\,| \leq a$ we have observed all commission errors in at most t_0 steps, which formally reads as $W_{h_s} \cap \mathbb{N} \setminus L = W_{h_s}^{t_0} \cap \mathbb{N} \setminus L$.

Then for all $t \geq t_0$ we obtain the same set of indices

$$A := \{\, \Xi(I[t], p_i) \mid i \leq t \wedge |\mathrm{neg}(I[t]) \cap W_{p_i}^t| \leq a \,\}$$

and therefore M' will return syntactically the same hypothesis, namely, h'_{t_0}.

It remains to argue for $W_{h'_{t_0}} =^a L$. By construction and the choice of t_0 there are no commission errors, i.e., $W_{h'_{t_0}} \cap \mathbb{N} \setminus L = \emptyset$. Further, since $\varphi_{h'_{t_0}}(x)$ exists in case there is at least one $p \in A$ such that $\varphi_p(x)$ exists, there are at most a arguments, on which $\varphi_{h'_{t_0}}$ is undefined. □

For learning from informant we gain for every $a \in \mathbb{N} \cup \{*\}$ a duality

$$\begin{aligned}[\mathbf{InfEx}^a] = \ldots = [\mathbf{InfEx}^a_b] = [\mathbf{InfEx}^a_{b+1}] = \ldots \\ = \bigcup_{b \in \mathbb{N}_{>0}} [\mathbf{InfEx}^a_b] = [\mathbf{InfEx}^a_*] \subsetneq [\mathbf{InfBc}^a].\end{aligned}$$

3.4 Learning Characteristic Functions of Collections of Recursive Languages

We now turn to the setting in which we want to learn a set of Boolean classifiers. In Machine Learning the input is usually considered a labeled element of $\mathbb{R}^d$. It is reasonable to consider only the countably many d-tuples x of computable reals $\mathbb{R}^d_{\text{comp}}$. By fixing a (non-computable) enumeration $\mathbb{R}^d_{\text{comp}} = \langle x_i \mid i < \mathbb{N} \rangle$, we might as a first attempt identify i with x_i. Then our hypothesis space is the set of all Boolean functions. We will later restrict ourselves to total computable Boolean functions.

Definitions 2.1 for informant and 2.2 for the learner are independent of the interpretation of the hypothesis. The Definition 2.3 of convergence criteria has to be slightly modified as follows.

Definition 3.6. *Let M be a learner and $\mathcal{L}$ a collection of recursive languages. Further, let $a \in \mathbb{N} \cup \{*\}$ and $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$.*

(i) Let $L \in \mathcal{L}$ be a language and $I \in \mathbf{Inf}(L)$ an informant for L presented to M.

a) We call $h = (h_t)_{t \in \mathbb{N}} \in (\mathbb{N} \cup \{?\})^\omega$, where $h_t := M(I[t])$ for all $t \in \mathbb{N}$, the learning sequence of M on I.

b) M learns *L from I with a* anomalies *and* vacillation number *b in the limit, for short M $\mathbf{Ex}^a_{C^b}$-learns L from I or $\mathbf{Ex}^a_{C^b}(M, I)$, if there is a time $t_0 \in \mathbb{N}$ such that $|\{ h_t \mid t \geq t_0 \}| \leq b$ and for all $t \geq t_0$ we have $\mathrm{Diff}_L(h_t) = \{x \in \mathbb{N} \mid \varphi_{h_t}(x) \neq \chi_L(x)\}$ has at most size a.*

(ii) M learns $\mathcal{L}$ with a anomalies and vacillation number b in the limit, *for short* M $\mathbf{Ex}_{C_b^a}$*-learns* $\mathcal{L}$, *if* $\mathbf{Ex}_{C_b^a}(M, I)$ *for every* $L \in \mathcal{L}$ *and every* $I \in \mathbf{Inf}(L)$.

This is also equivalent to learning the characteristic function of L from text. We also have to adjust the Definition 2.5 of locking sequences.

Definition 3.7. *Let* M *be a learner,* L *a language and* $a \in \mathbb{N} \cup \{*\}$ *as well as* $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$. *We call* $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ *an* $\mathbf{Ex}_{C_b^a}$-locking sequence for M on L, *if* $\mathbf{Cons}(\sigma, L)$ *and*

$$\exists D \subseteq \mathbb{N} \Big(|D| \leq b \ \wedge \ \forall \tau \in (\mathbb{N}\times\{0,1\})^{<\omega} \Big(\mathbf{Cons}(\tau, L) \Rightarrow \\ (M(\sigma\hat{}\tau)\downarrow \wedge |\mathrm{Diff}_L(M(\sigma\hat{}\tau))| \leq a \wedge M(\sigma\hat{}\tau) \in D)) \Big)$$

Then the proof of Lemma 2.6 immediately transfers and we obtain the following lemma.

Lemma 3.8. *Let* M *be a learner,* $a \in \mathbb{N} \cup \{*\}$, $b \in \mathbb{N}_{>0} \cup \{*, \infty\}$ *and* L *a language* $\mathbf{Ex}_{C_b^a}$*-identified by* M. *Then there is a* $\mathbf{Ex}_{C_b^a}$*-locking sequence for* M *on* L.

We also have to adjust the Definition 2.4 of consistency in the following way.

Definition 3.9. *Let* φ *be a Boolean computable function. We define*

$$\mathrm{pos}(\varphi) = \{x \in \mathbb{N} \mid \varphi(x)\downarrow = 1\};$$
$$\mathrm{neg}(\varphi) = \{x \in \mathbb{N} \mid \varphi(x)\downarrow = 0\}.$$

Let $f \in (\mathbb{N}\times\{0,1\})^{\leq\omega}$. *We say* f is consistent with φ, *for short* $\mathbf{Cons}(f, \varphi)$, *if*

$$\mathrm{pos}(f) \subseteq \mathrm{pos}(\varphi) \ \wedge \ \mathrm{neg}(f) \subseteq \mathrm{neg}(\varphi).$$

Let C_{h_t} denote $\mathrm{pos}(\varphi_{h_t})$. By replacing W_{h_t} by C_{h_t}, the definitions of the learning restrictions in Definition 2.7, learning success criteria in Definition 2.9 and learning criteria in Definition 2.10 remain the same. The implications (independent of the learning success criterion at hand) between the delayable learning restrictions as stated in Lemma 2.8 hold accordingly.

Moreover, the Definition 2.11 and basic Lemma 2.12 concerning delayability remain unchanged. Also Lemma 2.14 about considering canonical informant being sufficient and Lemma 2.17 about totality being no restriction for delayable

learning success criteria still hold as the proofs only refer to the concept of delayability.

To our knowledge Machine Learning algorithms only hypothesize total classifiers. Denote the set of encoded programs for total Boolean functions on $\mathbb{N}$ by $C\mathbf{Ind}$. From now on we will allow the learner M to hypothesize elements of $C\mathbf{Ind}$ on data consistent with some classifier to be learned only. We denote by $[\mathbf{Inf}C\mathbf{IndEx}_C]$ the collection of all recursive languages $\mathbf{Ex}_C$-learnable by such a learner M from informant. In Definition 2.9 in the learning success criterion at position β, we write $C\mathbf{Ind}$ between the learning restrictions to be met and the convergence criterion.

With $[C\mathbf{IndInfEx}_C]$ we refer to the collection of all recursive languages $\mathbf{Ex}_C$-learnable by a learner with range contained in $C\mathbf{Ind}$. These learners only output hypotheses for total computable Boolean functions and in Definition 2.9 we write $C\mathbf{Ind}$ as part of α.

Later we might consider appropriately chosen subsets of $C\mathbf{Ind}$ as hypothesis space.

In this setting we can assume the learner to output only hypotheses consistent with the input on relevant data. This is done by patching the hypothesis according to the finitely many training data points the learner has received so far.

Proposition 3.10. *We have*

$$[\mathbf{Inf}_{\text{can}}C\mathbf{IndEx}_C] = [\mathbf{SdInfCons}C\mathbf{IndEx}_C].$$

Proof. We use the idea from Lemma 2.14. Thus, the new learner outputs M's hypothesis h on the largest complete canonical informant with information only from the current input σ. As h is an index for a total function, we can, in a uniformly computable way, obtain a hypothesis h_σ from h such that

(i) φ_{h_σ} is consistent with all data in σ and

(ii) $h_\sigma = h$ if σ is consistent with φ_h.

More precisely, the computable operator maps an index h of a computable function $\varphi_h : \mathbb{N} \to \{0, 1\}$ and a finite informant sequence σ to an index h_σ of a

computable function φ_{h_σ} with

$$\varphi_{h_\sigma}(x) = \begin{cases} 1, & \text{if } x \in \mathrm{pos}(\sigma); \\ 0, & \text{else if } x \in \mathrm{neg}(\sigma); \\ \varphi_h(x), & \text{otherwise.} \end{cases}$$

The simulation only requires information about $\mathrm{pos}(\sigma) \cup \mathrm{neg}(\sigma)$ and thus the learner is set-driven. Further, $h_\sigma = h$ whenever φ_h is consistent with σ. As M converges on the canonical informant and we only alter h in case at least one datum in σ is inconsistent with φ_h, we obtain the convergence of the new learner. Clearly, it is consistent by construction. □

Summing up, as consistency of the input data with a hypothesized total computable Boolean functions is decidable, C**Ind**-learners can be assumed consistent while learning. By the same argument $\tau(C\mathbf{Ind})$-learners can be assumed $\tau(\mathbf{Cons})$.

It is easy to see that **Ex** can be replaced by every convergence criterion (and also **Mon**, **NU**, **SNU**).

On the other hand, it is easy to adapt the proof of Proposition 2.18 as follows.

Proposition 3.11. *There is a* collection of decidable languages *witnessing*

$$[\mathcal{R}\mathbf{InfConsC}\mathbf{IndEx}_C] \subsetneq [\mathbf{InfConsC}\mathbf{IndEx}_C].$$

Proof. Let o be an index for the everywhere 0-function. Further, define for all $\sigma \in (\mathbb{N}\times\{0,1\})^{<\omega}$ the learner M by

$$M(\sigma) := \begin{cases} o, & \text{if } \mathrm{pos}(\sigma) = \varnothing; \\ \varphi_{\max(\mathrm{pos}(\sigma))}(\langle\sigma\rangle), & \text{otherwise.} \end{cases}$$

We argue that $\mathcal{L} := \{ L \subseteq \mathbb{N} \mid L \text{ is decidable and } L \in \mathbf{InfConsEx}_C(M) \}$ is not consistently learnable by a total learner from informant. Assume towards a contradiction M' is such a learner. For a sequence σ of natural numbers we denote by $\overline{\sigma}$ the corresponding canonical finite informant sequence, ending with the highest value σ takes. Further, for a natural number x we denote by $\tau(x)$ the unique element of $\mathbb{N}^{<\omega}$ with $\langle\tau(x)\rangle = x$. Then by padded ORT there are

$e, z \in \mathbb{N}$ and functions $a, b : \mathbb{N}^{<\omega} \to \mathbb{N}$, such that

$$\forall \sigma, \tau \in \mathbb{N}^{<\omega} \, (\, \sigma \lhd \tau \Rightarrow \max\{a(\sigma), b(\sigma)\} < \min\{a(\tau), b(\tau)\} \,), \tag{3.1}$$

with the property that for all $\sigma \in \mathbb{N}^{<\omega}$ and all $i \in \mathbb{N}$

$$\begin{aligned}
\sigma_0 &= \varnothing; \\
\sigma_{i+1} &= \sigma_i \, ^\frown \begin{cases} a(\sigma_i), & \text{if } M'(\overline{\sigma_i \, ^\frown a(\sigma_i)}) \neq M'(\overline{\sigma_i}); \\ b(\sigma_i), & \text{otherwise;} \end{cases} \\
\varphi_e(y) &= \begin{cases} 1, & \text{if } y \in \operatorname{pos}(\overline{\sigma_y}); \\ 0, & \text{otherwise;} \end{cases} \\
\varphi_{a(\sigma)}(x) &= \begin{cases} e, & \text{if } \mathbf{Cons}(\tau(x), \varphi_e) \text{ and } M'(\overline{\sigma \, ^\frown a(\sigma)}) \neq M'(\overline{\sigma}); \\ \operatorname{ind}(\operatorname{pos}(\tau(x))), & \text{otherwise;} \end{cases} \\
\varphi_{b(\sigma)}(x) &= \begin{cases} e, & \text{if } \mathbf{Cons}(\tau(x), \varphi_e); \\ \operatorname{ind}(\operatorname{pos}(\tau(x))), & \text{otherwise;} \end{cases}
\end{aligned} \tag{3.2}$$

Consider the decidable language $L_e = \operatorname{pos}(\varphi_e)$. Clearly, we have $L_e \in \mathcal{L}$ and thus M' also $\mathbf{InfConsEx}_C$-learns L_e. By the $\mathbf{Ex}_C$-convergence there are $e', j \in \mathbb{N}$, where j is minimal, such that $\varphi_{e'} = \varphi_e$ and for all $i \geq j$ we have $M'(\overline{\sigma_i}) = e'$ and hence $M'(\overline{\sigma_i \, ^\frown a(\sigma_i)}) = M'(\overline{\sigma_i})$ by (3.2).

We now argue that $L := \operatorname{pos}(\overline{\sigma_j}) \cup \{a(\sigma_j)\} \in \mathcal{L}$. Let I be an informant for L and $t \in \mathbb{N}$. By (3.2) we observe that M is consistent on I as

$$M(I[t]) = \varphi_{\max(\operatorname{pos}(I[t]))}(\langle I[t] \rangle) = \begin{cases} e, & \text{if } \mathbf{Cons}(I[t], \varphi_e); \\ \operatorname{ind}(\operatorname{pos}(I[t])), & \text{otherwise.} \end{cases}$$

Further, by the choice of j we have $\neg\mathbf{Cons}(\, (a(\sigma_j), 1), \varphi_e \,)$. If $\operatorname{pos}(I[t]) = L$, we obtain $\varphi_{M(I[t])} = \operatorname{ind}_L$. On the other hand M' does not consistently learn L as by the choice of j we obtain $M'(\overline{\sigma_j \, ^\frown a(\sigma_j)}) = M'(\overline{\sigma_j}) = e'$ and $\neg\mathbf{Cons}(\overline{\sigma_j \, ^\frown a(\sigma_j)}, L_e)$, a contradiction. □

Thus, learning algorithms not defined on all inputs have strictly more learning power.

As we clearly can do a padding-trick for C-indices, similar to Lemma 2.20,

we might assume the learner to be syntactically decisive. Furthermore, the separations of **Caut**, **Mon** and **SMon** are still valid as they are witnessed by indexable families. Thus, the interesting question is whether **Conv** and **SDec** are also not restrictive for binary classifiers. We now observe that this still holds true but the proof is much simpler than for W-indices, because the consistency of data with hypotheses is decidable.

Theorem 3.12. *For $\delta \in \{\mathbf{T}, \mathbf{Mon}\}$ holds*

$$[\mathbf{Inf}\delta C\mathbf{IndBc}_C] = [\mathbf{InfConvSDec}\delta C\mathbf{IndEx}_C].$$

Proof. By the comment after Proposition 3.10 we assume $\delta \subseteq \mathbf{Cons}$. Let $\mathcal{L} \in [\mathbf{Inf}\delta C\mathbf{IndBc}_C]$ and the learner M witnessing this. It is an easy exercise to check that the following learner acts as required, where σ is a finite informant sequence and $\xi \in \mathbb{N} \times \{0,1\}$:

$$M'(\varnothing) = M(\varnothing);$$
$$M'(\sigma\hat{}\xi) = \begin{cases} M(\sigma\hat{}\xi), & \text{if } \neg\mathbf{Cons}(\sigma\hat{}\xi, M'(\sigma)); \\ M'(\sigma), & \text{otherwise.} \end{cases}$$

Note that the consistency of M on $\mathcal{L}$ is only employed to obtain **SDec**. □

Corollary 3.13. $[\mathbf{Inf}_{\mathrm{can}}C\mathbf{IndBc}_C] = [\mathbf{InfConsConv}C\mathbf{IndEx}_C]$.

For $\tau(C\mathbf{Ind})$-learners the simulation in Theorem 3.12 preserves totality.

In a nutshell for learners only outputting C-indices, we obtain the same map as for W-indices. In contrast, **Cons** is not a restriction anymore.

Moreover, $\mathbf{Bc}_C$-learning is not weaker than explanatory learning and thus the vacillatory hierarchy collapses.

3.5 Further Research

From Section 3.2 arises the question how memory-restricted variants of learning from informant relate to learning from text. We give some initial results in Section 4.3.

Moreover, future investigations could address the relationships between the different delayable learning restrictions for some of the investigated convergence criteria, where the general results in Section 2.3 may be helpful. Some relevant references, especially for behaviourally correct learning, are [Cas16], [KSS17], [DK20], [DK21a] and [DK21b].

Another open question regards the relation between learning recursive functions from text for their graphs and learning languages from either informant or text. It seems like delayability plays a crucial role in order to obtain normal forms and investigate how learning restrictions relate in each setting. It is yet not clear, whether delayability is the right assumption to generalize Lemma 3.2. The survey [ZZ08] and the standard textbook [Jai+99] contain more results in the setting of function learning which may transfer to learning collections of languages from informant with such a generalization.

Along this line would also be reproving that consistent learning from canonically ordered data is easier than consistent learning from every possible order of presentation, with the observation $[\mathbf{ConsFnEx}] \subsetneq [\mathbf{ConsFn}_{\text{can}}\mathbf{Ex}]$ by [JB80] and a generalization of Lemma 3.2 to consistent learning.

Part II

Memory-Efficient Learning

4 Iterative Learning from Informant

In order to model an efficient learning paradigm, iterative learning algorithms access data one by one, updating the current hypothesis without regress to past data. Prior research investigating the impact of additional requirements on iterative learners left many questions open, especially in learning from informant, where the input is binary labeled.

We first compare learning from positive information (text) with learning from informant. We provide different concept classes learnable from text but not by an iterative learner from informant. Further, we show that totality restricts iterative learning from informant.

Towards a map of iterative learning from informant, we prove that strongly non-U-shaped learning is restrictive and that iterative learners from informant can be assumed canny for a wide range of learning criteria. Finally, we compare two syntactic learning requirements.

4.1 Introduction

We are interested in the problem of algorithmically learning a description for a formal language (a computably enumerable subset of the set of natural numbers) when presented successively all information about that language; this is sometimes called *inductive inference*, a branch of (algorithmic) learning theory.

Many criteria for deciding whether a learner M is *successful* on a language L have been proposed in the literature. Gold, in his seminal paper [Gol67], gave a first, simple learning criterion, **Ex**-*learning*[3], where a learner is *successful* iff, on every complete information about L it eventually stops changing its conjectures, and its final conjecture is a correct description for the input sequence. Trivially, each single, describable language L has a suitable constant function as a **Ex**-learner (this learner constantly outputs a description for L). As we want algorithms for more than a single learning task, we are interested in analyzing for which *classes of languages* $\mathcal{L}$ there is a *single learner* M learning *each* member

3 **Ex** stands for *explanatory*.

of $\mathcal{L}$. This framework is also sometimes known as *language learning in the limit* and has been studied using a wide range of learning criteria in the flavor of **Ex**-learning (see, for example, the textbook [Jai+99]).

One major criticism of the model suggested by Gold, see for example [CM08b], is its excessive use of memory: for each new hypothesis the entire history of past data is available. Iterative learning [R W76], is the most common variant of learning in the limit which addresses memory constraints: the memory of the learner on past data is just its current hypothesis. Due to the padding lemma, this memory is still not void, but finitely many data can be memorized in the hypothesis.

Prior work on iterative learning [CK10; CM08b; Jai+16; Jai+99; JMZ13] focused on learning from text, that is, from positive data only. Hence, in **TxtEx**-learning the complete information is a listing of all and only the elements of L. In this chapter we are mainly interested in the paradigm of learning from both positive and negative information. For example, when learning half-spaces, one could see data declaring that $\langle 1, 1\rangle$ is in the target half-space, further is $\langle 3, 2\rangle$, but $\langle 1, 7\rangle$ is not, and so on. This setting is called *learning from informant* (in contrast to learning from *text*).

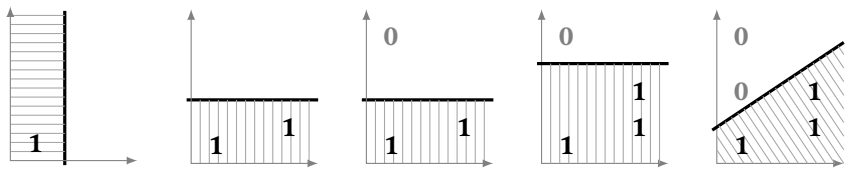

Figure 4.1: Example Learning Process with binary labeled data and half-spaces as hypotheses.

Iterative learning from informant was analyzed by [JLZ07b], where various natural restrictions have been considered and the authors focused on the case of learning indexable families (classes of languages which are uniformly decidable). In this chapter we are looking at other established restrictions and also consider learning of arbitrary classes of computably enumerable languages.

In Section 4.3 we consider the two aforementioned restrictions on learning from informant: learning from text and learning iteratively. Both restrictions

render fewer classes of languages learnable; in fact, the two restrictions yield two incomparable sets of language classes being learnable, which also shows that learning iteratively from text is weaker than supposing just one of the two restrictions.

Towards a better understanding of iterative learners we analyze which normal forms can be assumed in Section 4.4. First we show that, analogously to the case of learning from text (as analyzed in [CM09]), we cannot assume learners to be total (i.e. always giving an output).

However, from [CM08b] we know that we can assume iterative text learners to be *canny* (also defined in Section 4.4); we adapt this normal form for the case of iterative learning from informant and show that it can be assumed to hold for iterative learners generally.

Many works in inductive inference, see for example [Jai+16], [KP16], [KS16], [KSS17], focus on relating different additional learning requirements for a fixed learning model. In particular, [Jai+16] mapped out all pairwise relations for an established choice of learning restrictions for iterative learning from text. The complete map of all pairwise relations between for full-information learners from informant can be found in [AKS18] and Chapter 2. A similar map for the case of iterative learning from informant is not known. Canniness is central in investigating the learning power of iterative learning from text. Hence, it is an important stepping stone to understand iterative learners better and determine such pairwise relations. We argue in Lemma 4.18 that the normal form of canniness still can be assumed in case we pose additional semantic learning requirements.

In Section 4.5 we collect all previously known results for such a map, see [ST92], [JLZ07b]. We observe that it decreases learning power to require the learner to never change its hypothesis, once it is correct. The proof for separating this notion, called strong non-U-shapedness, relies on the ORT recursion theorem [Cas74]. We close this section by comparing two syntactic learning requirements for iterative learners from informant that proved important to derive the equivalence of all syntactic requirements for iterative learners from text.

We continue this chapter with some mathematical preliminaries in Section 4.2 before discussing our results in more detail.

4.2 Iterative Learning from Informant

Notation and terminology on the learning theoretic side follow [OSW86], [Jai+99] and [LZZ08], whereas on the computability theoretic side we refer to [Odi99] and [Rog67]. For both we also recommend [Köt09].

A *language* L is a recursively enumerable subset of $\mathbb{N}$. We denote the characteristic function for $L \subseteq \mathbb{N}$ by $f_L : \mathbb{N} \to \{0, 1\}$.

Gold in his seminal paper [Gol67], distinguished two major different kinds of information presentation. A function $I : \mathbb{N} \to \mathbb{N} \times \{0, 1\}$ is an *informant for language* L, if there is a surjection $n : \mathbb{N} \to \mathbb{N}$ such that $I(t) = (n(t), f_L(n(t)))$ holds for every $t \in \mathbb{N}$. Moreover, for an informant I let

$$\begin{aligned} \operatorname{pos}(I) &:= \{y \in \mathbb{N} \mid \exists x \in \mathbb{N} \colon \operatorname{pr}_1(I(x)) = y \wedge \operatorname{pr}_2(I(x)) = 1\} \text{ and} \\ \operatorname{neg}(I) &:= \{y \in \mathbb{N} \mid \exists x \in \mathbb{N} \colon \operatorname{pr}_1(I(x)) = y \wedge \operatorname{pr}_2(I(x)) = 0\} \end{aligned}$$

denote the sets of all natural numbers, about which I gives some positive or negative information, respectively. A *text for language* L is a function $T : \mathbb{N} \to \mathbb{N} \cup \{\#\}$ with range L after removing #. The symbol # is interpreted as pause symbol.

Therefore, when learning from informant, the *set of admissible inputs to the learning algorithm* $\mathbb{S}$ is the set of all finite sequences

$$\sigma = ((n_0, y_0), \ldots, (n_{|\sigma|-1}, y_{|\sigma|-1}))$$

of *consistently* binary labeled natural numbers. When learning from text (positive data only), we encounter inputs to the learning algorithm from the set $\mathbb{T}$ of finite sequences $\tau = (n_0, \ldots, n_{|\tau|-1})$ of natural numbers and the pause symbol #. The initial subsequence relation is denoted by $\trianglelefteq$.

A set $\mathcal{L} = \{L_i \mid i \in \mathbb{N}\}$ of languages is called *indexable family* if there is a computer program that on input $(i, n) \in \mathbb{N}^2$ returns 1 if $n \in L_i$ and 0 otherwise. Examples are **Fin** and **CoFin**, the set of all finite subsets of $\mathbb{N}$ and the set of all complements of finite subsets of $\mathbb{N}$, respectively.

Let $\mathcal{L}$ be a collection of languages we seek a provably correct learning algorithm for. We will refer to $\mathcal{L}$ as the *concept class* which will often be an indexable family. Further, let $\mathcal{H} = \{L_i \mid i \in \mathbb{N}\}$ with $\mathcal{L} \subseteq \mathcal{H}$ be a collection of languages called the *hypothesis space*. In general we do *not* assume that for every $L \in \mathcal{L}$

there is a unique index $i \in \mathbb{N}$ with $L_i = L$. Indeed, ambiguity in the hypothesis space helps memory-resticted learners to remember data.

A *learner M from informant (text)* is a computable function

$$M : \mathbb{S} \to \mathbb{N} \cup \{?\} \qquad (M : \mathbb{T} \to \mathbb{N} \cup \{?\})$$

with the output i interpreted with respect to $\mathcal{H} = \{L_i \mid i \in \mathbb{N}\}$, a prefixed hypothesis space. The output ? often serves as initial hypothesis or is interpreted as no new hypothesis. Often $\mathcal{H}$ is an indexable class or the established W-hypothesis space defined in Subsection 4.4.

Let I be an informant (T be a text) for L and $\mathcal{H} = \{L_i \mid i \in \mathbb{N}\}$ a hypothesis space. A learner $M : \mathbb{S} \to \mathbb{N} \cup \{?\}$ ($M : \mathbb{T} \to \mathbb{N} \cup \{?\}$) is *successful on I (on T)* if it eventually settles on $i \in \mathbb{N}$ with $L_i = L$. This means that when receiving increasingly long finite initial segments of I (of T) as inputs, it will from some time on be correct and not change the output on longer initial segments of I (of T). *M learns L wrt $\mathcal{H}$* if it is successful on every informant I (on every text T) for L. *M learns $\mathcal{L}$* if there is a hypothesis space $\mathcal{H}$ such that M learns every $L \in \mathcal{L}$ wrt $\mathcal{H}$. We denote the collection of all $\mathcal{L}$ learnable from informant (text) by [**InfEx**] ([**TxtEx**]). If we fix the hypothesis space, we denote this by a subscript for **Ex**.

According to [R W76], [LZ96], [Cas+99] a learner M is *iterative* if its output on $\sigma \in \mathbb{S}$ ($\tau \in \mathbb{T}$) only depends on the last input last(σ) and the hypothesis $M(\sigma^-)$ after observing σ without its last element last(σ). The collection of all concept classes $\mathcal{L}$ learnable by an iterative learner from informant (text) is denoted by [**ItInfEx**] ([**ItTxtEx**]).

The s-m-n theorem gives finite and infinite recursion theorems, see [Cas94], [Odi92]. We will refer to Case's Operator Recursion Theorem ORT in its 1-1-form, see [Cas74], [Jai+99], [Köt09].

4.3 Comparison with Learning from Text

By ignoring negative information every informant incorporates a text for the language presented and we gain [**ItTxtEx**] $\subseteq$ [**ItInfEx**].

It has been observed in [OSW86] that the superfinite language class **Fin** $\cup \{\mathbb{N}\}$ is in [**InfEx**] $\setminus$ [**ItInfEx**]. With $L_k = 2\mathbb{N} \cup \{2k+1\}$ and $L'_k = L_k \setminus \{2k\}$ the indexable family $\mathcal{L} = \{2\mathbb{N}\} \cup \{L_k, L'_k \mid k \in \mathbb{N}\}$ lies in [**TxtEx**] $\cap$ [**ItInfEx**] but

not in [**ItTxtEx**]. In [Jai+99] the separations are witnessed by the indexable family $\{\mathbb{N} \setminus \{0\}\} \cup \{D \cup \{0\} : D \in \mathbf{Fin}\}$.

It can easily be verified that $\mathbf{CoFin} \in [\mathbf{ItInfEx}] \setminus [\mathbf{TxtEx}]$ and with the next result [**ItInfEx**] and [**TxtEx**] are incomparable by inclusion.

Lemma 4.1. *There is an indexable family in* [**TxtEx**] \ [**ItInfEx**].

Proof. As there is a computable bijection between $\mathbb{N}$ and $\mathbb{N} \times \mathbb{N}$, we can also consider subsets of $\mathbb{N} \times \mathbb{N}$ as languages. Denote by $L_{S,D} = S \times (D \cup \{0\}) \cup (\mathbb{N} \setminus S) \times (\mathbb{N} \setminus \{0\}) \subseteq \mathbb{N} \times \mathbb{N}$ the language with $D \cup \{0\}$ in all rows numbered by an $s \in S$ and $\mathbb{N} \setminus \{0\}$ in all other rows. Consider the indexable family

$$\mathcal{L} = \{L_{S,D} \mid S, D \in \mathbf{Fin}\}.$$

$\mathcal{L}$ is clearly an indexable family, as there is a computable enumaration of all pairs (S, D) where S is a finite subset of $\mathbb{N}$ and D is a finite subset of $\mathbb{N} \setminus \{0\}$. Moreover, there is a uniform procedure to check whether (n_1, n_2) is in $L_{S,D}$.

$\mathcal{L} \in$ [TxtEx]: Maintain full information at step n of the entire sequence $T[n]$ read from text. Conjecture $S' := \{x | (x, 0) \in T[n]\}$ and $D' := \{y | \exists x \in S' : (x, y) \in T[n]\}$. S' will eventually converge to S as all $(x, 0)$ will be received by the learner at some point for all $x \in S$. After $S' = S$, we can say that D' will also converge to D (if it has not already) because at some point all (x, y) will have been received for all $x \in S$.

$\mathcal{L} \notin$ [ItInfEx]: Suppose an iterative learner M learns $\mathcal{F}$ from informant. Let σ be a locking sequence of M for $\mathbb{N} \times (\mathbb{N} \setminus \{0\})$. Let x_0 be such that $(x_0, 0)$ is not labeled in σ. Such an x_0 must exist because there are infinitely many $(x, 0)$ but σ is a finite sequence. Define $D := \{y | (x_0, y) \in \mathrm{pos}(\sigma)\}$. $L := \{x_0\} \times (D \cup \{0\}) \cup (\mathbb{N} \setminus \{x_0\}) \times (\mathbb{N} \setminus \{0\})$ is then consistent with σ, so let $\sigma' \sqsupseteq \sigma$ be a locking sequence for L. Define y_0 such that $y_0 > \max(\{0\} \cup \{y | \exists x : (x, y) \in \mathrm{pos}(\sigma') \cup \mathrm{neg}(\sigma')\})$. The element (x_0, y_0) is consistent with $\mathbb{N} \times (\mathbb{N} \setminus \{0\})$ if and only if it is labeled positively and with L if and only if it is labeled negatively. Because σ is a locking sequence for $\mathbb{N} \times (\mathbb{N} \setminus \{0\})$ and $((x_0, y_0), 1))$ is consistent with it, $M(\sigma((x_0, y_0), 1)) = M(\sigma) = e_1$ such that $W_{e_1} = \mathbb{N} \times (\mathbb{N} \setminus \{0\})$ so by iterativeness of M we have that if $\tau := \sigma((x_0, y_0), 1)(\sigma' - \sigma)$ where $\sigma' - \sigma$ is the subsequence of σ' starting after σ ends, then $M(\tau) = M(\sigma')$ meaning τ is also a locking sequence for L. This is a contradiction because if I is an informant for L, then $J := I \setminus \{((x_0, y_0), 0)\}$ is also consistent with L so for all $\ell \geq 0$ we have $M(\tau J[\ell]) = M(\sigma') = e_2$ such that

$W_{e_2} = L$ but τJ is an informant for $L' := \{x_0\} \times (D \cup \{(x_0, y_0)\} \cup \{0\}) \cup (\mathbb{N} \setminus \{x_0\}) \times (\mathbb{N} \setminus \{0\}) \in \mathcal{F}$ and $L' \neq L$, a contradiction. □

Summing up, we know $[\mathbf{ItTxtEx}] \subsetneq [\mathbf{TxtEx}] \perp [\mathbf{ItInfEx}] \subsetneq [\mathbf{InfEx}]$, where $\perp$ stands for incomparability with respect to set inclusion, meaning (1) there is a concept class learnable from text but not by an iterative learner from informant and (2) there is a concept class learnable by an iterative learner from informant but not from text.

In the following we give a procedure to generate more separating classes in $[\mathbf{TxtEx}] \setminus [\mathbf{ItInfEx}]$. With the help of the Boolean function $\mathbf{f}$ being defined in Definition 4.2 we obtain from an indexable family $\mathcal{L} \in [\mathbf{InfEx}] \setminus [\mathbf{ItInfEx}]$ an indexable family $\mathbf{f}(\mathcal{L}) \in [\mathbf{TxtEx}] \setminus [\mathbf{ItInfEx}]$.

The idea is to apply the Boolean function $\mathbf{f}$, defined in the following, to an indexable family, a set of informant and to a hypothesis space being a candidate to witness the learnability. With this notation we can draw conclusions from the learnability in the setting before applying $\mathbf{f}$ to the setting after applying $\mathbf{f}$ and vice versa.

Definition 4.2. *We refer to the function* $\mathbf{f} : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$ *defined by*

$$(2n \in \mathbf{f}(L) \Leftrightarrow n \in L) \wedge (2n+1 \in \mathbf{f}(L) \Leftrightarrow n \notin L)$$

as the Boolean mapping. For a set of languages $\mathcal{L}$ *we define* $\mathbf{f}(\mathcal{L}) = \{\mathbf{f}(L) | L \in \mathcal{L}\}$.

Note that for an indexable class $\mathcal{L}$ the image $f(\mathcal{L})$ is again an indexable class.

To obtain a result also applicable in other context, we generalize the notation. Let $\mathcal{I}$ be a set of informant (text), for example the ones containing each information only once or infinitely often. *M learns L from* $\mathcal{I}$ if it is successful on every $I \in \mathcal{I}$ for L. *M learns* $\mathcal{L}$ *from* $\mathcal{I}$ if it learns every $L \in \mathcal{L}$ from $\mathcal{I}$. We denote the collection of all $\mathcal{L}$ learnable from $\mathcal{I}$ by $[\mathcal{I}\mathbf{Ex}]$.

The idea is to apply the Boolean function $\mathbf{f}$ to an indexable family, a set of informant and a hypothesis space possibly witnessing the learnability. With this notation we can draw conclusions from the learnability in the setting before applying $\mathbf{f}$ to the setting after applying $\mathbf{f}$ and vice versa.

Definition 4.3. *We refer to the function* $\mathbf{f} : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$ *defined by*

$$(2n \in \mathbf{f}(L) \Leftrightarrow n \in L) \wedge (2n+1 \in \mathbf{f}(L) \Leftrightarrow n \notin L)$$

as the Boolean mapping. For a set of languages $\mathcal{L}$ *we define* $\mathbf{f}(\mathcal{L}) = \{\mathbf{f}(L) | L \in \mathcal{L}\}$. *For an informant* I *for* L *we obtain an informant* $\mathbf{f}(I)$ *for* $\mathbf{f}(L)$ *by interweaving* I_+ *and* I_- *where*

$$I_+(t) = \begin{cases} (2n_t, 1), & \text{if } I(t) = (n_t, 1); \\ (2n_t + 1, 1), & \text{if } I(t) = (n_t, 0). \end{cases} \quad \text{and}$$

$$I_-(t) = \begin{cases} (2n_t + 1, 0), & \text{if } I(t) = (n_t, 1); \\ (2n_t, 0), & \text{if } I(t) = (n_t, 0). \end{cases}$$

Moreover, the projection of I_+ *to the first coordinate yields a text for* $\mathbf{f}(L)$. *For a set of informant* $\mathcal{I}$ *we define the corresponding sets of informant* $\mathbf{f}(\mathcal{I})$ *and text* $T_{\mathbf{f}}(\mathcal{I})$ *by*

$$\mathbf{f}(\mathcal{I}) = \{\mathbf{f}(I) \mid I \in \mathcal{I}\} \quad \text{and} \quad T_{\mathbf{f}}(\mathcal{I}) = \{\mathrm{pr}_1 \circ I_+ \mid I \in \mathcal{I}\}.$$

Note that for an indexable class $\mathcal{L}$ the image $f(\mathcal{L})$ is again an indexable class.

We will apply the following result to the full set of informant but state it more generally for arbitrary sets of informant $\mathcal{I}$.

Theorem 4.4. *Let* $\mathcal{I}$ *be a set of informant,* $\mathcal{L} \subseteq \{pos(I) | I \in \mathcal{I}\}$ *a concept class and* $\mathcal{H}$ *an indexable family as suitable fixed hypothesis space. Consider the Boolean mapping* $\mathbf{f}$ *from Definition 4.2.*

If $\mathcal{L} \in [\mathcal{I}\mathbf{Ex}_{\mathcal{H}}]$, *then* $\mathbf{f}(\mathcal{L}) \in [T_{\mathbf{f}}(\mathcal{I})\mathbf{Ex}_{\mathbf{f}(\mathcal{H})}]$.

Moreover, if $\mathcal{I}$ *is upwards closed with respect to the subsequence relation, then* $\mathcal{L} \in [(\mathbf{It})\mathcal{I}\mathbf{Ex}_{\mathcal{H}}]$ *is equivalent to* $\mathbf{f}(\mathcal{L}) \in [(\mathbf{It})\mathbf{f}(\mathcal{I})\mathbf{Ex}_{\mathbf{f}(\mathcal{H})}]$.

Proof. Let $\mathbf{f}$, $\mathcal{I}$, $\mathcal{L}$ and $\mathcal{H}$ be as stated above.

$\mathcal{L} \in [\mathcal{I}\mathbf{Ex}_{\mathcal{H}}] \Rightarrow \mathbf{f}(\mathcal{L}) \in [T_{\mathbf{f}}(\mathcal{I})\mathbf{Ex}_{\mathbf{f}(\mathcal{H})}]$: Let M be a learner for $\mathcal{L}$ from $\mathcal{I}$. Let $\mathbf{f}(L) \in \mathbf{f}(\mathcal{L})$ and $T \in T_{\mathbf{t}}(\mathcal{I})$ a text for $\mathbf{f}(L)$. Then there is an informant $I \in \mathcal{I}$ for L such that $T = \mathrm{pr}_1 \circ I_+$. If for every $t \in \mathbb{N}$ we denote the first and second coordinate of $I(t)$ by n_t and λ_t, respectively, we obtain $T = (2n_t + 1 - \lambda_t)_{t \in \mathbb{N}}$. Therefore, we can in a computable way reconstruct $I[t]$ from $T[t]$. We define a learner M' which simulates M by $M'(T[t]) = M(I[t])$. It is easy to see that M' learns $\mathbf{f}(\mathcal{L})$ from $T_{\mathbf{f}}(\mathcal{I})$.

If $\mathcal{I}$ is upwards closed with respect to the subsequence relation, $\mathcal{L} \in [\mathbf{It}\mathcal{I}\mathbf{Ex}_{\mathcal{H}}]$ implies $f(\mathcal{L}) \in [\mathbf{It}\mathbf{f}(\mathcal{I})\mathbf{Ex}_{\mathbf{f}(\mathcal{H})}]$: The proof is very similar to the last paragraph. Let M be a learner for $\mathcal{L}$ from $\mathcal{I}$. Let $\mathbf{f}(L) \in \mathbf{f}(\mathcal{L})$ and $I' \in \mathbf{f}(\mathcal{I})$ an informant

for $\mathbf{f}(L)$. Then there is an informant $I \in \mathcal{I}$ for L such that I' results from interweaving I_+ and I_-. We compute $\tilde{I}(t) = (\lfloor \frac{x_t}{2} \rfloor, (x_t - w_t) \mod 2)$ from $I'(t) = (x_t, w_t)$ and define M' by $M'(I'[t]) = M(\tilde{I}[t])$. Because $\tilde{I}$ contains I as a subsequence, we obtain $\tilde{I} \in \mathcal{I}$. Again, it is easily verified that M' learns $\mathbf{f}(\mathcal{L})$ from $\mathbf{f}(\mathcal{I})$. Moreover, it easy to see that M' is iterative, in case M is.

$f(\mathcal{L}) \in [\mathbf{Itf}(\mathcal{I})\mathbf{Ex}_{\mathbf{f}(\mathcal{H})}] \Rightarrow \mathcal{L} \in [\mathbf{It}\mathcal{I}\mathbf{Ex}_{\mathcal{H}}]$: We proceed in a similar fashion. Let M' be a learner for $\mathbf{f}(\mathcal{L})$ from $\mathbf{f}(\mathcal{I})$. Let $L \in \mathcal{L}$ and I an informant for L. We recursively construct initial segments σ_t with $|\sigma_t| = 2t$ for the informant $\mathbf{f}(I)$ for $\mathbf{f}(L)$ from I as follows: $\sigma_0 = \emptyset$; if σ_t is defined and $I(t) = (n_t, \lambda_t)$ then let $\sigma_{t+1} = \sigma_t(2n_t + 1 - \lambda_t, 1)(2n_t + \lambda_t, 0)$. Clearly, $\mathbf{f}(I) = \bigcup_{t \in \mathbb{N}} \sigma_t$. The learner $M(I[t]) = M'(\sigma_t)$ learns $\mathcal{L}$ from $\mathcal{I}$. Finally, if M' is iterative, so is M. □

If $\mathcal{I}$ is the set of all informant for $\mathcal{L}$, then $T_{\mathbf{f}}(\mathcal{I})$ is the set of all text for $\mathbf{f}(\mathcal{L})$. $\mathbf{f}(\mathcal{I})$ is the set of all informant for $\mathbf{f}(\mathcal{L})$ that have the positive and negative information in the order given by interweaving.

Corollary 4.5. *Consider the Boolean mapping* $\mathbf{f}$ *from Definition 4.2. Let* $\mathcal{L}$ *be an indexable concept class and require that learnability is witnessed by indexable hypothesis spaces. Then* $\mathcal{L} \in [\mathbf{InfEx}]$ *implies* $\mathbf{f}(\mathcal{L}) \in [\mathbf{TxtEx}]$. *Moreover, from* $\mathbf{f}(\mathcal{L}) \in [\mathbf{ItInfEx}]$ *we can conclude* $\mathcal{L} \in [\mathbf{ItInfEx}]$.

Proof. For the second implication note that $\mathbf{f}(\mathcal{L}) \in [\mathbf{ItInfEx}] \Rightarrow \mathbf{f}(\mathcal{L}) \in [\mathbf{It}\,\mathbf{f}(\mathbf{Inf})\mathbf{Ex}] \Rightarrow \mathcal{L} \in [\mathbf{ItInfEx}]$. □

Therefore, every set of languages separating [**ItInfEx**] and [**InfEx**] yields a separating class for [**ItInfEx**] and [**TxtEx**].

4.4 Total and Canny Learners

For the rest of this chapter, without further notation, all results are understood with respect to the W-hypothesis space defined in the following. We fix a programming system φ as introduced in [RC94]. Briefly, in the φ-system, for a natural number p, we denote by φ_p the partial computable function with program code p. We also call p an *index* for W_p defined as $\mathrm{dom}(\varphi_p)$. In reference to a Blum complexity measure, for all $p, t \in \mathbb{N}$, we denote by $W_p^t \subseteq W_p$ the recursive set of all natural numbers less or equal to t, on which the machine executing p halts in at most t steps.

The question whether excluding partial functions as learners, denoted by $\mathcal{R}$, makes some sets of languages unlearnable has been investigated. Allowing only total learners does not restrict full-information learning from informant and text, i.e. $[\mathcal{R}\mathbf{InfEx}] = [\mathbf{InfEx}]$ and $[\mathcal{R}\mathbf{TxtEx}] = [\mathbf{TxtEx}]$. On the other hand [CM09] showed $[\mathcal{R}\mathbf{ItTxtEx}] \subsetneq [\mathbf{ItTxtEx}]$.

We show that totality, denoted by $\mathcal{R}$, restricts iterative learning from informant. The proof uses an easy ORT argument.

Theorem 4.6. $[\mathbf{ItInfEx}] \setminus [\mathcal{R}\mathbf{ItInfEx}] \neq \emptyset$.

Proof. Let o be an index for $\emptyset$ and define the iterative learner M for all $\xi \in \mathbb{N}\times\{0,1\}$ by

$$M(\emptyset) = o;$$
$$h_M(h,\xi) = \begin{cases} \varphi_{\mathrm{pr}_1(\xi)}(0), & \text{if } \mathrm{pr}_2(\xi) = 1 \text{ and } h \notin \mathrm{ran}(\mathrm{ind}); \\ h, & \text{otherwise.} \end{cases}$$

We argue that $\mathcal{L} := \{ L \subseteq \mathbb{N} \mid L \in \mathbf{ItInfEx}(M) \}$ is not learnable by a total learner from informant. Assume towards a contradiction M' is such a learner.

For a finite informant sequence σ we denote by $\overline{\sigma}$ the corresponding canonical finite informant sequence, ending with σ's datum with highest first coordinate. Then by 1-1 ORT there are $e \in \mathbb{N}$ and a strictly increasing computable function $a : \mathbb{N}^{<\omega} \to \mathbb{N}$, such that for all $\sigma \in \mathbb{N}^{<\omega}$ and all $i \in \mathbb{N}$

$$\begin{aligned} \sigma_0 &= \emptyset; \\ \sigma_{i+1} &= \sigma_i \frown \begin{cases} (a(\sigma_i),1), & \text{if } M'(\overline{\sigma_i \frown (a(\sigma_i),1)}) \neq M'(\overline{\sigma_i}); \\ \emptyset, & \text{otherwise;} \end{cases} \\ W_e &= \bigcup_{i\in\mathbb{N}} \mathrm{pos}(\overline{\sigma_i}); \\ \varphi_{a(\sigma)}(x) &= \begin{cases} e, & \text{if } M'(\overline{\sigma \frown (a(\sigma),1)}) \neq M'(\overline{\sigma}); \\ \mathrm{ind}_{\mathrm{pos}(\sigma)\cup\{a(\sigma)\}}, & \text{otherwise;} \end{cases} \end{aligned} \tag{4.1}$$

Clearly, we have $W_e \in \mathcal{L}$ and thus M' also $\mathbf{InfEx}$-learns W_e. By the $\mathbf{Ex}$-convergence there are $e', t_0 \in \mathbb{N}$, where t_0 is minimal, such that $W_{e'} = W_e$ and for all $t \geq t_0$

we have $M'(\bigcup_{i\in\mathbb{N}} \overline{\sigma_i}[t]) = e'$ and hence by (4.1) for all i with $|\overline{\sigma_i}| \geq t_0$

$$M'(\overline{\sigma_i{}^\frown(a(\sigma_i),1)}) = M'(\overline{\sigma_i}) = M'(\overline{\sigma_i{}^\frown(a(\sigma_i),0)}).$$

It is easy to see, that $W_e = \mathrm{pos}(\sigma_i)$ and $W_e \cup \{a(\sigma_i)\} \in \mathcal{L}$. On the other hand M' is iterative and hence does not learn W_e and $W_e \cup \{a(\sigma_i)\}$, a contradiction. □

The following definition is central in investigating the learning power of iterative learning from text, see [CM07] and [Jai+16]. We transfer it to learning from informant.

Definition 4.7. *A learner M from informant is called* canny *in case for every finite informant sequence σ holds*

(i) if $M(\sigma)$ is defined then $M(\sigma) \in \mathbb{N}$;

(ii) for every $x \in \mathbb{N} \setminus (\mathrm{pos}(\sigma) \cup \mathrm{neg}(\sigma))$ and $i \in \{0,1\}$ a mind change $M(\sigma^\frown(x,i)) \neq M(\sigma)$ implies for all finite informant sequences τ with $\sigma^\frown(x,i) \trianglelefteq \tau$ that $M(\tau^\frown(x,i)) = M(\tau)$.

Hence, the learner is canny in case it always outputs a hypotheses and no datum twice causes a mind change of the learner. Also for learning from informant the learner can be assumed canny by a simulation argument.

Lemma 4.8. *For every iterative learner M, there exists a canny iterative learner M' such that*

$$\mathbf{InfEx}(M) \subseteq \mathbf{InfEx}(M').$$

Proof. Let f be a computable 1-1 function mapping every finite informant sequence σ to a natural number encoding a program with $W_{f(\sigma)} = W_{M(\sigma)}$ if $M(\sigma) \in \mathbb{N}$ and $W_{f(\sigma)} = \emptyset$ otherwise. Clearly, σ can be reconstructed from $f(\sigma)$. We define the canny learner M' by letting

$$M'(\emptyset) = f(\emptyset)$$

$$h_{M'}(f(\sigma),(x,i)) = \begin{cases} f(\sigma^\frown(x,i)), & \text{if } x \notin \mathrm{pos}(\sigma) \cup \mathrm{neg}(\sigma) \wedge \\ & M(\sigma^\frown(x,i))\downarrow \neq M(\sigma)\downarrow; \\ f(\sigma), & \text{if } M(\sigma^\frown(x,i))\downarrow = M(\sigma)\downarrow \vee \\ & x \in (\mathrm{pos}(\sigma) \cup \mathrm{neg}(\sigma)); \\ \uparrow, & \text{otherwise.} \end{cases}$$

M' mimics M via f on a possibly finite informant subsequence of the originally presented informant with ignoring data not causing mind changes of M or that has already caused a mind change.

Let $L \in \mathbf{InfEx}(M)$ and $I' \in \mathbf{Inf}(L)$. As M has to learn L from every informant for it, M' will always be defined. Further, let $\sigma_0 = \varnothing$ and

$$\sigma_{t+1} = \begin{cases} \sigma_t{}^\frown I'(t), & \text{if } I'(t) \notin \operatorname{ran}(\sigma_t) \wedge M(\sigma_t{}^\frown I'(t))\!\downarrow \neq M(\sigma_t)\!\downarrow; \\ \sigma_t, & \text{otherwise.} \end{cases}$$

Then by induction for all $t \in \mathbb{N}$ holds $M'(I'[t]) = f(\sigma_t)$.

The following function translates between the two settings

$$\begin{aligned} \mathfrak{r}(0) &= 0; \\ \mathfrak{r}(t+1) &= \min\{r > \mathfrak{r}(t) \mid I'(r-1) \notin \operatorname{ran}(\sigma_{\mathfrak{r}(t)})\}. \end{aligned}$$

Intuitively, the infinite range of $\mathfrak{r}$ captures all points in time r at which a datum that has not caused a mind change so far, is seen and a mind-change of M' is possible. Thus, the mind change condition is of interest in order to decide whether $\sigma_{\mathfrak{r}(t+1)} \neq \sigma_{\mathfrak{r}(t)}$. Note that $\sigma_r = \sigma_{\mathfrak{r}(t)}$ for all r with $\mathfrak{r}(t) \leq r < \mathfrak{r}(t+1)$.

Let $I(t) = I'(\mathfrak{r}(t+1) - 1)$ for all $t \in \mathbb{N}$. Since only already observed data is ommited, I is an informant for L.

We next argue that $M(I[t]) = M(\sigma_{\mathfrak{r}(t)})$ for all $t \in \mathbb{N}$. As $I[0] = \varnothing = \sigma_0$, the claim holds for $t = 0$. Now we assume $M(I[t]) = M(\sigma_{\mathfrak{r}(t)})$ and obtain $M(I[t+1]) = M(I[t]{}^\frown I(t)) = M(\sigma_{\mathfrak{r}(t)}{}^\frown I(t))$. The equality $M(\sigma_{\mathfrak{r}(t)}{}^\frown I(t)) = M(\sigma_{\mathfrak{r}(t+1)})$ is true with the following arguments. By the definitions of I and $\mathfrak{r}$ we have $I(t) = I'(\mathfrak{r}(t+1) - 1) \notin \operatorname{ran}(\sigma_{\mathfrak{r}(t)})$. Hence, there are two cases:

(i) If $M(\sigma_{\mathfrak{r}(t)}{}^\frown I(t)) = M(\sigma_{\mathfrak{r}(t)})$, then from $\sigma_{\mathfrak{r}(t+1)-1} = \sigma_{\mathfrak{r}(t)}$ and the definition of M' we obtain $\sigma_{\mathfrak{r}(t+1)} = \sigma_{\mathfrak{r}(t)}$. Putting both together, the claimed equality $M(\sigma_{\mathfrak{r}(t)}{}^\frown I(t)) = M(\sigma_{\mathfrak{r}(t+1)})$ follows.

(ii) If $M(\sigma_{\mathfrak{r}(t)}{}^\frown I(t)) \neq M(\sigma_{\mathfrak{r}(t)})$, the definition of M' yields $\sigma_{\mathfrak{r}(t+1)} = \sigma_{\mathfrak{r}(t)}{}^\frown I(t)$. Hence the claimed equality also holds in this case.

We now argue that M' explanatory learns L from I'. In order to see this, first observe $\sigma_{\mathfrak{r}(t+1)} = \sigma_{\mathfrak{r}(t)}$ if and only if $M(I[t+1]) = M(I[t])$ for every $t \in \mathbb{N}$. This is because

$$\sigma_{\mathfrak{r}(t+1)} = \sigma_{\mathfrak{r}(t)} \Leftrightarrow M(\,\sigma_{\mathfrak{r}(t)}{}^\frown I(t)\,) = M(\sigma_{\mathfrak{r}(t)})$$

$$\Leftrightarrow M(I[t]^\frown I(t)) = M(I[t])$$
$$\Leftrightarrow M(I[t+1]) = M(I[t]).$$

As I is an informant for L, the learner M explanatory learns L from I. Hence there exists some t_0 such that $W_{M(I[t_0])} = L$ and for all $t \geq t_0$ holds $M(I[t]) = M(I[t_0])$. With this follows $\sigma_{\mathfrak{r}(t)} = \sigma_{\mathfrak{r}(t_0)}$ for all $t \geq t_0$. As for every r there exists some t with $\mathfrak{r}(t) \leq r$ and $\sigma_r = \sigma_{\mathfrak{r}(t)}$, we obtain $\sigma_r = \sigma_{\mathfrak{r}(t_0)}$ for all $r \geq \mathfrak{r}(t_0)$. We conclude $M'(I'[t]) = f(\sigma_t) = f(\sigma_{\mathfrak{r}(t_0)})$ for all $t \geq \mathfrak{r}(t_0)$ and by the definition of f finally $W_{f(\sigma_{\mathfrak{r}(t_0)})} = W_{M(\sigma_{\mathfrak{r}(t_0)})} = W_{M(I[t_0])} = L$. □

4.5 Additional Requirements

In the following we review additional properties one might require the learning process to have in order to consider it successful. For this, we employ the following notion of consistency when learning from informant.

As in [LZZ08] according to [BB75] and [Bār77] for $A \subseteq \mathbb{N}$ we define

$$\mathbf{Cons}(f, A) \quad :\Leftrightarrow \quad \mathrm{pos}(f) \subseteq A \ \wedge \ \mathrm{neg}(f) \subseteq \mathbb{N} \setminus A$$

and say *f is consistent with A* or *f is compatible with A*.

Learning restrictions incorporate certain desired properties of the learners' behavior relative to the information being presented. We state the definitions for learning from informant here.

Definition 4.9. *Let M be a learner and I an informant. We denote by $h_t = M(I[t])$ the hypothesis of M after observing $I[t]$ and write*

(i) $\mathbf{Conv}(M, I)$ *([Ang80]), if M is* conservative on I*, i.e., for all s, t with $s \leq t$ the consistency* $\mathbf{Cons}(I[t], W_{h_s})$ *implies $h_s = h_t$.*

(ii) $\mathbf{Dec}(M, I)$ *([OSW82]), if M is* decisive on I*, i.e., for all r, s, t with $r \leq s \leq t$ the semantic equivalence $W_{h_r} = W_{h_t}$ implies the semantic equivalence $W_{h_r} = W_{h_s}$.*

(iii) $\mathbf{Caut}(M, I)$ *([OSW86]), if M is* cautious on I*, i.e., for all s, t with $s \leq t$ holds $\neg W_{h_t} \subsetneq W_{h_s}$.*

(iv) $\mathbf{WMon}(M, I)$ *([Jan91],[Wie91]), if M is* weakly monotonic on I*, i.e., for all s, t with $s \leq t$ holds* $\mathbf{Cons}(I[t], W_{h_s}) \ \Rightarrow \ W_{h_s} \subseteq W_{h_t}$.

(v) **Mon**(M, I) *([Jan91],[Wie91]), if* M *is* monotonic on I*, i.e., for all* s, t *with* $s \leq t$ *holds* $W_{h_s} \cap \mathrm{pos}(I) \subseteq W_{h_t} \cap \mathrm{pos}(I)$.

(vi) **SMon**(M, I) *([Jan91],[Wie91]), if* M *is* strongly monotonic on I*, i.e., for all* s, t *with* $s \leq t$ *holds* $W_{h_s} \subseteq W_{h_t}$.

(vii) **NU**(M, I) *([Bal+08]), if* M *is* non-U-shaped on I*, i.e., for all* r, s, t *with* $r \leq s \leq t$ *the semantic success* $W_{h_r} = W_{h_t} = \mathrm{pos}(I)$ *implies the semantic equivalence* $W_{h_r} = W_{h_s}$.

(viii) **SNU**(M, I) *([CM11]), if* M *is* strongly non-U-shaped on I*, i.e., for all* r, s, t *with* $r \leq s \leq t$ *the semantic success* $W_{h_r} = W_{h_t} = \mathrm{pos}(I)$ *implies the syntactic equality* $h_r = h_s$.

(ix) **SDec**(M, I) *([KP14]), if* M *is* strongly decisive on I*, i.e., for all* r, s, t *with* $r \leq s \leq t$ *the semantic equivalence* $W_{h_r} = W_{h_t}$ *implies the syntactic equality* $h_r = h_s$.

It is easy to observe that **Conv**(M, I) implies **SNU**(M, I) and **WMon**(M, I); **SDec**(M, I) implies **Dec**(M, I) and **SNU**(M, I); **SMon**(M, I) implies **Caut**(M, I), **Dec**(M, I), **Mon**(M, I), **WMon**(M, I) and finally **Dec**(M, I) and **SNU**(M, I) imply **NU**(M, I).

When additional requirements apply to the definition of learning success, we write them between **Inf** and **Ex**. For example, Theorem 4.6 proves

$$[\mathbf{ItInfConvSDecSMonEx}] \setminus [\mathcal{R}\mathbf{ItInfEx}] \neq \emptyset$$

because the non-total learner acts conservatively, strongly decisively and strongly monotonically when learning $\mathcal{L}$.

The text variants can be found in [Jai+16] where all pairwise relations $=$, $\subsetneq$ or $\perp$ between the sets [**ItTxt**δ**Ex**] (iterative learners from text) for $\delta \in \Delta$, where $\Delta = \{\mathbf{Conv}, \mathbf{Dec}, \mathbf{Caut}, \mathbf{WMon}, \mathbf{Mon}, \mathbf{SMon}, \mathbf{NU}, \mathbf{SNU}, \mathbf{SDec}\}$, are depicted. Moreover, they can be found in Chapter 5. The complete map of all pairwise relations between the sets [**Inf**δ**Ex**] (full-information learners from informant) for $\delta \in \Delta$ can be found in Section 2.4. We sum up the current status regarding the map for iterative learning from informant in the following.

Recall the indexable family $\mathcal{L} = \{2\mathbb{N}\} \cup \{L_k, L'_k \mid k \in \mathbb{N}\}$ with $L_k = 2\mathbb{N} \cup \{2k+1\}$ and $L'_k = L_k \setminus \{2k\}$, separating [**ItTxtEx**] from [**TxtEx**]. Clearly, $\mathcal{L} \in [\mathcal{R}\mathbf{ItInfConvSDecMonEx}]$.

With a locking sequence argument we observe for all $\delta \in \Delta \setminus \{\mathbf{SMon}\}$ that $[\mathbf{ItInfSMonEx}] \setminus [\mathbf{ItInf}\delta\mathbf{Ex}] \neq \emptyset$.

Let $\mathbf{Inf}_{\text{can}}$ denote the set of all informant labelling the natural numbers according to their canonical order. Then $\mathbf{Fin} \cup \{\mathbb{N}\} \in [\mathcal{R}\mathbf{ItInf}_{\text{can}}\mathbf{ConsConvSDecMonEx}]$ and thus in contrast to full-information learning from informant $[\mathbf{ItInf}_{\text{can}}\mathbf{Ex}] \neq [\mathbf{ItInfEx}]$, see [AKS18] or Section 2.3.

Theorem 4.6 can be restated as.

Theorem 4.10. $[\mathbf{ItInfConvSDecSMonEx}] \setminus [\mathcal{R}\mathbf{ItInfEx}] \neq \emptyset$.

It has been observed that requiring a monotonic behavior of the learner is restrictive.

Theorem 4.11. *[ST92] There exists an indexable family in* $[\mathbf{ItInfMonEx}] \subsetneq [\mathbf{ItInfEx}]$.

It is easy to see that requiring a cautious behavior of the learner is also restrictive.

Theorem 4.12. *There exists an indexable family in* $[\mathbf{ItInfCautEx}] \subsetneq [\mathbf{ItInfEx}]$.

Proof. The indexable family $\{\mathbb{N}\} \cup \{\mathbb{N} \setminus \{x\} \mid x \in \mathbb{N}\}$ is clearly not cautiously learnable but conservatively, strongly decisively and monotonically learnable by a total iterative learner from informant. □

Corollary 4.13. $[\mathbf{ItInfCautEx}] \perp [\mathbf{ItInfMonEx}]$

Moreover, requiring a conservative learning behavior is also restrictive.

Theorem 4.14. *[JLZ07b] There exists an indexable family in* $[\mathbf{ItInfConvEx}] \subsetneq [\mathbf{ItInfEx}]$.

Indeed, they provide an indexable family in $[\mathbf{ItInfCautWMonNUDecEx}] \setminus [\mathbf{ItInfConvEx}]$ and an indexable family that lies in $[\mathcal{R}\mathbf{ItTxtCautConvSDecEx}] \setminus [\mathbf{ItInfMonEx}]$.

Hence the map differs from the map on iterative learning from text in [Jai+16] as **Caut** is restrictive and also from the map of full-information learning in [AKS18] from informant as **Conv** is restrictive too. It has been open how **WMon**, **Dec**, **NU**, **SDec** and **SNU** relate to each other and the other requirements. We show that also **SNU** restricts **ItInfEx** with an intricate **ORT**-argument.

Theorem 4.15. [**ItInfSNUEx**] $\subsetneq$ [**ItInfEx**]

Proof. Let M be a learner as follows, where the initial hypothesis is o, an index for $\emptyset$. We consider input data x with given label $\ell \in \{0,1\}$.

$$\forall e, x, \ell : h_M(e,(x,\ell)) = \begin{cases} e, & \text{if } e = o \wedge \ell = 0; \\ \mathrm{pad}(\varphi_x(0), x), & \text{else if } e = o \wedge \ell = 1; \\ \mathrm{pad}(\varphi_y(\langle e', x, \ell\rangle), y), & \text{else, with } e = \mathrm{pad}(e', y). \end{cases}$$

Let $\mathcal{L}$ be what M learns and suppose M' learns $\mathcal{L}$ also SNU.

We define strictly increasing computable functions $a, b, e_1, e_2 : \mathbb{N} \to \mathbb{N}$ and $e_0 \in \mathbb{N}$ by ORT. Thereby, we interpret a and b as data streams and for all k, t the numbers e_0, $e_1(\langle k,t\rangle)$ and $e_2(\langle k,t\rangle)$ as hypotheses. We start with defining a and b by letting for all $i, k \in \mathbb{N}$

$$\varphi_{a(i)}(z) = \begin{cases} e_1(\langle k,k\rangle), & \text{if } z = \langle e_0, b(k), 1\rangle; \\ e_0, & \text{else if } z = 0 \vee z = \langle e_0, x, \ell\rangle; \\ e_1(\langle k,t\rangle), & \text{else if } z = \langle e_1(\langle k,s\rangle), a(t), 1\rangle \wedge t \geq s \wedge \\ & W^t_{e_0}[k] \neq W^s_{e_0}[k]; \\ e_2(\langle k,k\rangle), & \text{else if } z = \langle e_1(\langle k,s\rangle), a(t), 0\rangle \wedge t \geq k; \\ e_2(\langle k,t\rangle), & \text{else if } z = \langle e_2(\langle k,s\rangle), a(t), \ell\rangle \wedge t \geq s \wedge \\ & W^t_{e_0}[k] \neq W^s_{e_0}[k]; \\ e, & \text{else if } z = \langle e, x, \ell\rangle; \end{cases}$$

$$\varphi_{b(k)}(z) = \begin{cases} e_1(\langle k,k\rangle), & \text{if } z = 0; \\ e_1(\langle k,t\rangle), & \text{else if } z = \langle e_1(\langle k,s\rangle), a(t), 1\rangle \wedge t \geq s \wedge \\ & W^t_{e_0}[k] \neq W^s_{e_0}[k]; \\ e_2(\langle k,k\rangle), & \text{else if } z = \langle e_1(\langle k,s\rangle), a(t), 0\rangle \wedge t \geq k; \\ e_2(\langle k,t\rangle), & \text{else if } z = \langle e_2(\langle k,s\rangle), a(t), \ell\rangle \wedge t \geq s \wedge \\ & W^t_{e_0}[k] \neq W^s_{e_0}[k]; \\ e, & \text{else if } z = \langle e, x, \ell\rangle; \end{cases}$$

Before we define W_{e_0}, $W_{e_1(\langle k,t\rangle)}$ and $W_{e_2(\langle k,t\rangle)}$, note that, while M sees only negatively labeled data, it sticks to o as hypothesis. Once a positive a-datum is seen, it sticks to e_0 as hypothesis. The first positive $b(k)$-datum makes it change its mind to $e_1(\langle k,k\rangle)$. Any *negative* a-datum after the positive $b(k)$-datum leads to $e_2(\langle k,k\rangle)$. As the second coordinate in $\langle k,t\rangle$ will tell us which canonical informant sequence $W^t_{e_0}[k]$ we consider, we enlarge it whenever neccessary in order to guarantee $W^t_{e_0}[k] = W_{e_0}[k]$ in the limit.

We give the definitions of what to list into W_{e_0}, $W_{e_1(\langle k,t\rangle)}$ and $W_{e_2(\langle k,t\rangle)}$ as algorithms.

In W_{e_0} we enumerate all $a(i)$ on which M' changes its mind when labeled positively while M' observes the canonical informant for W_{e_0}. For convenience, in the definition of W_{e_0} we let $a(-1) = -1$ and denote by $[u,w]$ the set of all integers v with $u \leq v \leq w$.

Algorithm 1: The definition of e_0 in the ORT-argument.

```
e ← initial hypothesis of M';
for i = 0 to ∞ do
    if h*_{M'}(e, [a(i−1)+1, a(i)−1] × {0}^(a(i), 1)) ↓≠ e then
        e ← h_{M'}(e, [a(i−1)+1, a(i)−1] × {0}^(a(i), 1));
        list a(i) into W_{e_0};
    else if h*_{M'}(e, [a(i−1)+1, a(i)−1] × {0}^(a(i), 0)) ↓≠ e then
        e ← h_{M'}(e, [a(i−1)+1, a(i)−1] × {0}^(a(i), 0));
```

As M learns W_{e_0}, also M' has to learn it. Let I be the canonical informant for W_{e_0} and k be such that $M'(I[i]) = M'(I[k])$ for all $i \geq k$ and $W_{M'(I[k])} = W_{e_0}$.

Algorithm 2: The definition of $e_1(\langle k, t\rangle)$ and $e_2(\langle k, t\rangle)$ in the ORT-argument.

Input: $\langle k, t\rangle$;
$e \leftarrow M'(W^t_{e_0}[k](b(k), 1))$;
$i \leftarrow k$;
list $b(k)$ and the positive information in $W^t_{e_0}[k]$ into W_{e_1} and W_{e_2};
for $s = 0$ ***to*** ∞ **do**
    while $h_{M'}(e, (a(i), 1)) = e$ ***and*** $h_{M'}(e, (a(i), 0)) = e$ **do**
        list $a(i)$ into W_{e_1};
        $i \leftarrow i + 1$;
    list all of what is already listed in W_{e_1} into W_{e_2};
    if $h_{M'}(e, (a(i), 1)) \neq e$ **then**
        list $a(i)$ into W_{e_1} and W_{e_2};
        $e \leftarrow h_{M'}(e, (a(i), 1))$;
    else
        $j \leftarrow i$;
        $i \leftarrow i + 1$;
        while $h_{M'}(e, (a(i), 1)) = e$ **do**
            list $a(i)$ into W_{e_1} and W_{e_2};
            $i \leftarrow i + 1$;
        list $a(i)$ into W_{e_1} and W_{e_2};
        list $a(j)$ into W_{e_1} and W_{e_2};
        $e \leftarrow h^*_{M'}(e, (a(i), 1)(a(j), 1))$;
    $i \leftarrow i + 1$;

For all k, t, t' with $W^t_{e_0}[k] = W^{t'}_{e_0}[k]$ holds $W_{e_1(\langle k,t\rangle)} = W_{e_1(\langle k,t'\rangle)}$ and $W_{e_2(\langle k,t\rangle)} = W_{e_2(\langle k,t'\rangle)}$.

We will now argue that for t minimal with $W^t_{e_0}[k] = I[k]$ every possible outcome of Algorithm 2 is contradictory.

(i) If all stages s are visited, then $W_{e_1(\langle k,t\rangle)} = W_{e_2(\langle k,t\rangle)}$ contains essentially all $a(i)$ with $i \geq k$. Hence M will eventually output the correct hypothesis $e_1(\langle k, t\rangle)$ while M' makes infinitely many mind changes on a suitable informant I'. More precisely, the informant I' starts with $I[k](b(k), 1)$ and afterwards enumerates all $a(i)$ with $i \geq k$ in the order they were listed into $W_{e_1(\langle k,t\rangle)}$.

(ii) If the first while loop does not terminate for some stage s, then $W_{e_1(\langle k,t\rangle)}$ and $W_{e_2(\langle k,t\rangle)}$ are different. As $W_{e_2(\langle k,t\rangle)}$ is finite, M learns it by changing its mind on some negative a-datum. On the other hand $W_{e_1(\langle k,t\rangle)}$ contains all $a(i)$ with $i \geq k$ and M learns it by not changing its mind. Let e_{s-1} denote the current value of variable e when entering the stage s. By the case assumption, M' does not perform a mind-change on any further positive or negative a-datum. Therefore, we must have $W_{e_1(\langle k,t\rangle)} = W_{e_{s-1}} = W_{e_2(\langle k,t\rangle)}$, a contradiction.

(iii) If the second while loop does not terminate for some stage s, then we have $W_{e_1(\langle k,t\rangle)} = W_{e_2(\langle k,t\rangle)}$ and it contains all $a(i)$ with $i \geq k$ but $a(j_s)$. This is learned by M from any informant (though with different final hypotheses, depending on the informant). Again, we let e_{s-1} denote the current value of e when entering stage s. By the choice of k for all $j \geq k$ holds $M'(I[k]^\frown(a(j), 1)) = M'(I[k])$ and $M'(I[k]^\frown(a(j), 0)) = M'(I[k])$. Hence M' on the informant

$$I'' = I[k](a(j_s), 0)(b(k), 1)((a(i), 1))_{i\geq k, i\neq j_s}$$

for $W_{e_1(\langle k,t\rangle)}$ outputs e_{s-1} and therefore e_{s-1} must be correct. On the other hand e_{s-1} cannot be correct, since M' is SNU and changing its mind on the negative information $(a(j_s), 0)$ in the informant

$$I''' = I[k](b(k), 1)((a(i), 1))_{i<j_s}(a(j_s), 0)((a(i), 1))_{i>j_s}$$

for $W_{e_1(\langle k,t\rangle)}$. □

We are now attempting to clarify in which sense precisely **Conv** is a restriction and more specifically, where exactly and how often there are separations in the implication chains **Conv** $\Rightarrow$ **WMon** $\Rightarrow$ **T**, **Conv** $\Rightarrow$ **SNU** $\Rightarrow$ **NU** $\Rightarrow$ **T** and **SDec** $\Rightarrow$ **Dec** $\Rightarrow$ **NU** $\Rightarrow$ **T**. In the following we provide a lemma that might help to investigate **WMon**, **Dec** and **NU**.

Definition 4.16. *Denote the set of all unbounded and non-decreasing functions by* $\mathfrak{S}$*, i.e.,*

$$\mathfrak{S} := \{\, \mathfrak{s} : \mathbb{N} \to \mathbb{N} \mid \forall x \in \mathbb{N}\, \exists t \in \mathbb{N} \colon \mathfrak{s}(t) \geq x \text{ and } \forall t \in \mathbb{N} \colon \mathfrak{s}(t+1) \geq \mathfrak{s}(t) \,\}.$$

Then every $\mathfrak{s} \in \mathfrak{S}$ *is a so called* admissible simulating function.

A predicate $\beta \subseteq \mathfrak{P} \times \mathcal{I}$ *is* semantically delayable*, if for all* $\mathfrak{s} \in \mathfrak{S}$*, all* $I, I' \in \mathcal{I}$ *and all learners* $M, M' \in \mathfrak{P}$ *holds: Whenever we have* $\mathrm{pos}(I'[t]) \supseteq \mathrm{pos}(I[\mathfrak{s}(t)])$, $\mathrm{neg}(I'[t]) \supseteq \mathrm{neg}(I[\mathfrak{s}(t)])$ *and* $W_{M'(I'[t])} = W_{M(I[\mathfrak{s}(t)])}$ *for all* $t \in \mathbb{N}$*, from* $\beta(M, I)$ *we can conclude* $\beta(M', I')$*.*

Lemma 4.17. *Let* δ *be a semantic learning restriction, i.e.* $\delta \in \{$**Caut**, **Dec**, **WMon**, **Mon**, **SMon**, **NU**$\}$*. Then* δ *is semantically delayable.*

Lemma 4.8 can be generalized as follows.

Lemma 4.18. *For every iterative learner* M *and every semantically delayable learning restriction* δ*, there exists a canny iterative learner* M' *such that* $\mathbf{Inf}\delta\mathbf{Ex}(M) \subseteq \mathbf{Inf}\delta\mathbf{Ex}(M')$*.*

Proof. We add δ in front of **Ex** in the proof of Lemma 4.8. Further, we define a simulating function (Definition 4.16) by

$$\mathfrak{s}(t) = \max\{s \in \mathbb{N} \mid \mathfrak{r}(s) \leq t\}.$$

It is easy to check that $\mathfrak{s}$ is unbounded and clearly it is non-decreasing. Then by the definitions of I and $\mathfrak{s}$ we have $\mathrm{pos}(I[\mathfrak{s}(t)]) \subseteq \mathrm{pos}(I'[\mathfrak{r}(\mathfrak{s}(t))]) \subseteq \mathrm{pos}(I'[t])$ and similarly $\mathrm{neg}(I[\mathfrak{s}(t)]) \subseteq \mathrm{neg}(I'[t])$ for all $t \in \mathbb{N}$. As $M'(I'[t]) = f(\sigma_t)$ and $M(\sigma_{\mathfrak{r}(\mathfrak{s}(t))}) = M(I[\mathfrak{s}(t)])$ for all $t \in \mathbb{N}$, in order to obtain $W_{M'(I'[t])} = W_{M(I[\mathfrak{s}(t)])}$ it suffices to show $W_{f(\sigma_t)} = W_{M(\sigma_{\mathfrak{r}(\mathfrak{s}(t))})}$. Since $W_{f(\sigma_t)} = W_{M(\sigma_t)}$ for all $t \in \mathbb{N}$, this can be concluded from $\sigma_t = \sigma_{\mathfrak{r}(\mathfrak{s}(t))}$. But this obviously holds because $\mathfrak{r}(\mathfrak{s}(t)) \leq t < \mathfrak{r}(\mathfrak{s}(t)+1)$ follows from the definition of $\mathfrak{s}$.

Finally, from $\delta(M, I)$ we conclude $\delta(M', I')$. □

Two other learning restrictions that might be helpful to understand the syntactic learning criteria **SNU**, **SDec** and **Conv** better are the following.

Definition 4.19. *Let M be a learner and I an informant. We denote by $h_t = M(I[t])$ the hypothesis of M after observing $I[t]$ and write*

(i) $\mathbf{LocConv}(M, I)$ *([JLZ07b]), if M is* locally conservative *on I, i.e., for all t the mind-change $h_t \neq h_{t+1}$ implies* $\mathbf{Cons}(I(t), W_{h_t})$.

(ii) $\mathbf{Wb}(M, I)$ *([KS16]), if M is* witness-based *on I, i.e., for all r, s, t with $r < s \leq t$ the mind-change $h_r \neq h_s$ implies* $\operatorname{pos}(I[s]) \cap W_{h_t} \setminus W_{h_r} \neq \emptyset \vee \operatorname{neg}(I[s]) \cap W_{h_r} \setminus W_{h_t} \neq \emptyset$.

Hence, in a locally conservative learning process every mind-change is justified by the datum just seen. Moreover, a in witness-based learning process each mind-change is witnessed by some false negative or false positive datum. Obviously, **LocConv** $\Rightarrow$ **Conv** and **Wb** $\Rightarrow$ **Conv**.

As for learning from text, see [Jai+16], we gain that every concept class locally conservatively learnable by an iterative learner from informant is also learnable in a witness-based fashion by an iterative learner.

Theorem 4.20. $[\mathbf{ItInfLocConvEx}] \subseteq [\mathbf{ItInfWbEx}]$

Proof. Let $\mathcal{L}$ be a concept class learned by the iterative learner M in a locally conservative manner. As we are interested in a witness-based learner N, we always enlarge the guess of M by all data witnessing a mind-change in the past. As we want N to be iterative, this is done via padding the set of witnesses to the hypothesis and a total computable function g adding this information to the hypothesis of M as follows:

$$W_{g(\operatorname{pad}(h,\langle MC\rangle))} = (W_h \cup \operatorname{pos}[MC]) \setminus \operatorname{neg}[MC];$$
$$N(\emptyset) = g(\operatorname{pad}(M(\emptyset), \langle\emptyset\rangle));$$
$$h_N(g(\operatorname{pad}(h, \langle MC\rangle)), \xi) = \begin{cases} g(\operatorname{pad}(h, \langle MC\rangle)), & \text{if } h_M(h, \xi) = h \vee \\ & \quad \xi \in MC; \\ g(\operatorname{pad}(h_M(h, \xi), & \\ \quad \langle MC \cup \{\xi\}\rangle)), & \text{otherwise.} \end{cases}$$

Clearly, N is iterative. Further, whenever M is locked on h and $W_h = L$, since MC is consistent with L, we also have $W_{g(\operatorname{pad}(f(h),\langle MC\rangle))} = L$. As N simulates M on

an informant omitting all data that already caused a mind-change beforehand, N does explanatory learn $\mathcal{L}$. As M learns locally conservatively and by employing g, the learner N acts witness-based. □

4.6 Suggestions for Future Research

Future work should address the complete map for iterative learners from informant. It remains open, whether the syntactic learning criteria **SNU**, **SDec** and **Conv** have the same learning power. Theorem 4.20 might be helpful regarding the latter. Further, it seems like settling **NU**, **Dec** and **WMon** requires completely new techniques. We hope that Lemma 4.18 is a helping hand in this endeavour.

Maps for other models of memory-limited learning, such as **BMS**, see [Car+07], or **Bem**, see [FJO94], [LZ96] and [Cas+99], would help to rate models. We address the map for **BMS** algorithms when learning from text in the next chapter.

5 Map for BMS-Learning from Text

We investigate learning collections of languages from text by an inductive inference machine with access to the current datum and a bounded memory in form of states. Such a bounded memory states (**BMS**) learner is considered successful in case it eventually settles on a correct hypothesis while exploiting only finitely many different states.

We give the complete map of all pairwise relations for an established collection of criteria of successfull learning. Most prominently, we show that non-U-shapedness is not restrictive, while conservativeness and (strong) monotonicity are. Some results carry over from iterative learning by a general lemma showing that, for a wealth of restrictions (the *semantic* restrictions), iterative and bounded memory states learning are equivalent. We also give an example of a non-semantic restriction (strongly non-U-shapedness) where the two settings differ.

5.1 Introduction

We are interested in the problem of algorithmically learning a description for a formal language (a computably enumerable subset of the set of natural numbers) when presented successively all and only the elements of that language; this is sometimes called *inductive inference*, a branch of (algorithmic) learning theory. For example, a learner M might be presented more and more even numbers. After each new number, M outputs a description for a language as its conjecture. The learner M might decide to output a program for the set of all multiples of 4, as long as all numbers presented are divisible by 4. Later, when M sees an even number not divisible by 4, it might change this guess to a program for the set of all multiples of 2.

Many criteria for deciding whether a learner M is *successful* on a language L have been proposed in the literature. Gold, in his seminal paper [Gol67], gave a first, simple learning criterion, **TxtEx**-*learning*[4], where a learner is *successful*

4 **Txt** stands for learning from a *text* of positive examples; **Ex** stands for *explanatory*.

iff, on every *text* for L (listing of all and only the elements of L) it eventually stops changing its conjectures, and its final conjecture is a correct description for the input sequence. Trivially, each single, describable language L has a suitable constant function as an **TxtEx**-learner (this learner constantly outputs a description for L). Thus, we are interested in analyzing for which *classes of languages* $\mathcal{L}$ there is a *single learner* M learning *each* member of $\mathcal{L}$. Sometimes, this framework is called *language learning in the limit* and has been studied extensively, using a wide range of learning criteria similar to **TxtEx**-learning (see, for example, the textbook [Jai+99]).

One major criticism of the model suggested by Gold is its excessive use of memory, see for example [CM08a]: for each new hypothesis the entire history of past data is available. Iterative learning is the most common variant of learning in the limit which addresses memory constraints: the memory of the learner on past data is just its current hypothesis. Due to the padding lemma [Jai+99], this memory is not necessarily void, but only finitely many data can be memorized in the hypothesis. There is a comprehensive body of work on iterative learning, see, e.g., [CK10; CM08a; Jai+16; Jai+99; JMZ13].

Another way of modelling restricted memory learning is to grant the learner access to not their current hypothesis, but a *state* which can be used in the computation of the next hypothesis (and next state). This was introduced in [Car+07] and called *bounded memory states (BMS)* learning. It is a reasonable assumption to have a countable reservoir of states. Assuming a computable enumeration of these states, we use natural numbers to refer to them. Note that allowing arbitrary use of all natural numbers as states would effectively allow a learner to store all seen data in the state, thus giving the same mode as Gold's original setting. Probably the minimal way to restrict the use of states is to demand for successful learning that a learner must stop using new states eventually (but may still traverse among the finitely many states produced so far, and may use infinitely many states on data for a non-target language). It was claimed that this setting is equivalent to iterative learning [Car+07, Remark 38] (this restriction is called *ClassBMS* there, we refer to it by $\mathbf{TxtBMS}_*\mathbf{Ex}$). However, this was only remarked for the plain setting of explanatory learning; for further restrictions, the setting is completely unknown, only for explicit constant state bounds a few scattered results are known, see [Car+07; CK13].

We consider a wealth of restrictions, described in detail in Section 5.2 (after an introduction to the general notation). Following the approach of giving *maps*

of pairwise relations suggested in [KS16], we give a complete map in Figure 5.1. We note that this map is the same as the map for iterative learning given in [Jai+16], but partially for different reasons.

In Lemma 5.10 we show that, for many restrictions (the so-called *semantic* restrictions, where only the semantics of hypotheses are restricted) the learning setting with bounded memory states is equivalent to learning iteratively. This proves and generalizes the aforementioned remark in [Car+07] to a wide class of restrictions. The iterative learner uses the hypotheses of the $\mathbf{BMS}_*$-learner on an equivalent text and additionally pads a subgraph of the translation diagram to it. It keeps track of all states visited so far together with the datum which caused the first transfer to the respective state. This way we can reconstruct the last first-time-visited state while observing the equivalent text sequence. Moreover, the equivalent text prevents the iterative learner from returning to a previously visited state but the last one and hence enables the required convergence.

However, if restrictions are not semantic, then iterative and bounded memory states learning can differ. We show this concretely for the case of so-called *strongly non-U-shaped* learning in Theorem 5.16. Inspired by cognitive science research [SS82], [Mar+92] a semantic version of this requirement was defined in [Bal+08] and later the syntactic variant was introduced in [CM11]. Both requirements have been extensively studied, see [CC13] for a survey and moreover [CK13], [CK16], [KSS17]. The proof of Theorem 5.16 uses an intricate ORT-argument, which might suggest that the two settings, while different, are very similar nonetheless. It is based on the proof that strong non-U-shapedness restricts $\mathbf{BMS}_*\mathbf{Ex}$-learning. The proof of the latter result combines the techniques for showing that strong non-U-shapedness restricts iterative learning, as proved in [CK13, Theorem 5.7], and that not every class strongly monotonically learnable by an iterative learner is strongly non-U-shapedly learnable by an iterative learner, see [Jai+16, Theorem 5]. Moreover, it relies on showing that state decisiveness can be assumed in Lemma 5.12.

The remainder of Section 5.4 completes the map given in Figure 5.1 for the case of syntactic restrictions (since these do not carry over from the setting of iterative learning). All syntactic learning requirements are closely related to strongly locking learners. The fundamental concept of a locking sequence was introduced by [BB75]. For a similar purpose than ours [Jai+16] introduced strongly locking learners. We generalize their construction for certain syntactically restricted

iterative learners from a strongly locking iterative learner. Finally, we obtain that all non-semantic learning restrictions also coincide for $\mathbf{BMS}_*$-learning.

5.2 Learners, Success Criteria and other Terminology

As far as possible, we follow [Jai+99] on the learning theoretic side and [Odi99] for computability theory. We recall the most essential notation and definitions.

We let $\mathbb{N}$ denote the *natural numbers* including 0. For a function f we write $\mathrm{dom}(f)$ for its *domain* and $\mathrm{ran}(f)$ for its *range*. If we deal with (a subset of) a cartesian product, we are going to refer to the *projection functions* to the first or second coordinate by pr_1 and pr_2, respectively.

Further, $X^{<\omega}$ denotes the *finite sequences* over the set X and X^{ω} stands for the *countably infinite sequences* over X. For every $\sigma \in X^{<\omega}$ and $t \leq |\sigma|$, $t \in \mathbb{N}$, we let $\sigma[t] := \{(s, \sigma(s)) \mid s < t\}$ denote the *restriction of* σ *to* t. Moreover, for sequences $\sigma, \tau \in X^{<\omega}$ their concatenation is denoted by $\sigma^\frown\tau$. Finally, we write $\mathrm{last}(\sigma)$ for the last element of σ , $\sigma(|\sigma| - 1)$, and σ^- for the initial segment of σ without $\mathrm{last}(\sigma)$, i.e. $\sigma[|\sigma| - 1]$. Clearly, $\sigma = \sigma^-{}^\frown\mathrm{last}(\sigma)$.

For a finite set $D \subseteq \mathbb{N}$ and a finite sequence $\sigma \in X^{<\omega}$, we denote by $\langle D \rangle$ and $\langle \sigma \rangle$ a canonical index for D or σ, respectively. Further, we fix a Goedel pairing function $\langle ., . \rangle$ with two arguments.

Let $L \subseteq \mathbb{N}$. We interpret every $n \in \mathbb{N}$ as a code for a word. If L is recursively enumerable, we call L a *language*.

We fix a programming system φ as introduced in [RC94]. Briefly, in the φ-system, for a natural number p, we denote by φ_p the partial computable function with program code p. We call p an *index* for W_p defined as $\mathrm{dom}(\varphi_p)$.

In reference to a Blum complexity measure Φ_p, for all $p, t \in \mathbb{N}$, we denote by $W_p^t \subseteq W_p$ the recursive set of all natural numbers less or equal to t, on which the machine executing p halts in at most t steps, i.e.

$$W_p^t = \{x \mid x \leq t \ \wedge \ \Phi_p(x) \leq t\}.$$

Moreover, the well-known s-m-n theorem gives finite and infinite recursion theorems, see [Cas74], [Cas94], [Odi92]. We will refer to Case's Operator Recursion Theorem ORT in its 1-1-form, see for example [Köt09] and Section 2.2.

We let $\Sigma = \mathbb{N} \cup \{\#\}$ be the input alphabet with $n \in \mathbb{N}$ interpreted as code for a word in the language and # interpreted as pause symbol, i.e. no new information. Further, let $\Omega = \mathbb{N} \cup \{?\}$ be the output alphabet with $p \in \mathbb{N}$ interpreted as φ-index and ? as no hypothesis or repetition of the last hypothesis, if existent. A function with range Ω is called a hypothesis generating function.

A *learner* is always a (partial) computable function

$$M : \mathrm{dom}(M) \subseteq \Sigma^{<\omega} \to \Omega.$$

The set of all total computable functions $M : \Sigma^{<\omega} \to \Omega$ is denoted by $\mathcal{R}$.

Let $f \in \Sigma^{<\omega} \cup \Sigma^{\omega}$, then the *content of* f, defined as $\mathrm{content}(f) := \mathrm{ran}(f) \setminus \{\#\}$, is the set of all natural numbers, about which f gives some positive information. The *set of all text for the language* L is defined as

$$\mathbf{Txt}(L) := \{T \in \Sigma^{\omega} \mid \mathrm{content}(T) = L\}.$$

Definition 5.1. *Let M be a learner. M is an* iterative learner *or* **It**-learner*, for short $M \in$ **It**, if there is a computable (partial) hypothesis generating function $h_M : \Omega \times \Sigma \to \Omega$ such that $M = h_M^{\ddagger}$ where $h_M^{\ddagger}$ is defined on finite sequences by*

$$\begin{aligned} h_M^{\ddagger}(\epsilon) &= ?; \\ h_M^{\ddagger}(\sigma^\frown x) &= h_M(h_M^{\ddagger}(\sigma), x). \end{aligned}$$

Definition 5.2. *Let M be a learner. M is a* bounded memory states learner *or* **BMS**-learner*, for short $M \in$ **BMS**, if there are a computable (partial) hypothesis generating function $h_M : \mathbb{N} \times \Sigma \to \Omega$ and a computable (partial) state transition function $s_M : \mathbb{N} \times \Sigma \to \mathbb{N}$ such that $\mathrm{dom}(h_M) = \mathrm{dom}(s_M)$ and $M = h_M^*$ where h_M^* and s_M^* are defined on finite sequences by*

$$\begin{aligned} s_M^*(\epsilon) &= 0; \\ h_M^*(\sigma^\frown x) &= h_M(s_M^*(\sigma), x); \\ s_M^*(\sigma^\frown x) &= s_M(s_M^*(\sigma), x). \end{aligned}$$

Note that every iterative learner gives a **BMS**-learner by identifying the hypothesis space Ω with the set of states via a computable bijection between $\mathbb{N}$ and Ω. The resulting **BMS**-learner will succeed on the same languages the iterative learner does learn. Further, as the set of visited states contains exactly all

hypotheses the learner puts out, this **BMS**-learner only uses finitely many states on all text for languages it explanatory learns. In [Car+07, Rem. 38] it is claimed that $\mathbf{BMS}_*$-learners and iterative learners are equally powerful on text. This also follows from our more general Lemma 5.10. The above intuition is formalized in the corresponding proof.

Definition 5.2 may be stated more generally for arbitrary finite or infinite sets of states Q, instead of $\mathbb{N}$. Moreover, s^*_M and h^*_M can easily be generalized to functions taking also a starting state s as input by

$$\begin{aligned} s^*_M(s, \epsilon) &= s; \\ h^*_M(s, \sigma^\frown x) &= h_M(s^*_M(s, \sigma), x); \\ s^*_M(s, \sigma^\frown x) &= s_M(s^*_M(s, \sigma), x). \end{aligned}$$

We now clarify what we mean by successful learning.

Definition 5.3. *Let M be a learner and $\mathcal{L}$ a collection of languages.*

(i) Let $L \in \mathcal{L}$ be a language and $T \in \mathbf{Txt}(L)$ a text for L presented to M.

a) We call $h = (h_t)_{t\in\mathbb{N}} \in \Omega^\omega$, where $h_t := M(T[t])$ for all $t \in \mathbb{N}$, the learning sequence of M on T.

b) M learns L from T in the limit, *for short M $\mathbf{Ex}$-learns L from T or $\mathbf{Ex}(M, T)$, if there exists $t_0 \in \mathbb{N}$ such that $W_{h_{t_0}} = \mathrm{content}(T)$ and $\forall t \geq t_0 \left(h_t \neq ? \implies h_t = h_{t_0} \right)$.*

(ii) M learns $\mathcal{L}$ in the limit, *for short M $\mathbf{Ex}$-learns $\mathcal{L}$, if $\mathbf{Ex}(M, T)$ for every $L \in \mathcal{L}$ and every $T \in \mathbf{Txt}(L)$.*

Definition 5.4. *Let $\mathcal{L}$ be a collection of languages. $\mathcal{L}$ is* learnable in the limit *or* $\mathbf{Ex}$-learnable, *if there exists a learner M that $\mathbf{Ex}$-learns $\mathcal{L}$.*

Ex-learning is the most common definition for successful learning in inductive inference and corresponds to the notion of identifiability in the limit by [Gol67], where the learner eventually decides on one correct hypotheses.

In our investigations, the most important additional requirement on a successful learning process for a **BMS**-learner is to use finitely many states only, as stated in the following definition.

Definition 5.5. *Let M be a* **BMS***-learner and $T \in \mathbf{Txt}$. We say that M* uses finitely many memory states on T*, for short* $\mathbf{BMS}_*(M,T)$*, if* $\{ s^*_M(T[t]) \mid t \in \mathbb{N} \}$ *is finite.*

We list the most common additional requirements regarding the learning sequence, which may tag a learning process. For this we first recall the notion of consistency of a sequence with a set.

Definition 5.6. *Let $f \in \Sigma^{<\omega} \cup \Sigma^{\omega}$ and $A \subseteq \Sigma$. We define*

$$\mathbf{Cons}(f,A) \quad :\Leftrightarrow \quad \mathrm{content}(f) \subseteq A$$

and say f is consistent with *A.*

The listed properties of the learning sequence have been at the center of different investigations. Studying how they relate to one another did begin in [KP16], [KS16], [Jai+16] and [AKS18].

Definition 5.7. *Let M be a learner, $T \in \mathbf{Txt}$ and $h = (h_t)_{t\in\mathbb{N}} \in \Omega^{\omega}$ the learning sequence of M on T, i.e. $h_t = M(T[t])$ for all $t \in \mathbb{N}$.. We write*

(i) $\mathbf{Conv}(M,T)$ *([Ang80]), if M is* conservative on *T, i.e., for all s,t with $s \leq t$ holds* $\mathbf{Cons}(T[t], W_{h_s}) \implies h_s = h_t$.

(ii) $\mathbf{Dec}(M,T)$ *([OSW82]), if M is* decisive on *T, i.e., for all r,s,t with $r \leq s \leq t$ holds* $W_{h_r} = W_{h_t} \implies W_{h_r} = W_{h_s}$.

(iii) $\mathbf{Caut}(M,T)$ *([OSW86]), if M is* cautious on *T, i.e., for all s,t with $s \leq t$ holds* $\neg W_{h_t} \subsetneq W_{h_s}$.

(iv) $\mathbf{WMon}(M,T)$ *([Jan91],[Wie91]), if M is* weakly monotonic on *T, i.e., for all s,t with $s \leq t$ holds* $\mathbf{Cons}(T[t], W_{h_s}) \implies W_{h_s} \subseteq W_{h_t}$.

(v) $\mathbf{Mon}(M,T)$ *([Jan91],[Wie91]), if M is* monotonic on *T, i.e., for all s,t with $s \leq t$ holds* $W_{h_s} \cap \mathrm{content}(T) \subseteq W_{h_t} \cap \mathrm{pos}(T)$.

(vi) $\mathbf{SMon}(M,T)$ *([Jan91],[Wie91]), if M is* strongly monotonic on *T, i.e., for all s,t with $s \leq t$ holds* $W_{h_s} \subseteq W_{h_t}$.

(vii) $\mathbf{NU}(M,T)$ *([Bal+08]), if M is* non-U-shaped on *T, i.e., for all r,s,t with $r \leq s \leq t$ holds* $W_{h_r} = W_{h_t} = \mathrm{content}(T) \implies W_{h_r} = W_{h_s}$.

(viii) $\mathbf{SNU}(M,T)$ *([CM11]), if M is* strongly non-U-shaped on *T, i.e., for all r,s,t with $r \leq s \leq t$ holds* $W_{h_r} = W_{h_t} = \mathrm{content}(T) \implies h_r = h_s$.

(ix) **SDec**(M, T) *([KP16]), if* M *is* strongly decisive on T*, i.e., for all* r, s, t *with* $r \leq s \leq t$ *holds* $W_{h_r} = W_{h_t} \implies h_r = h_s$.

(x) **Wb**(M, T) *([KS16]), if* M *is* witness-based on T*, i.e., for all* r, t *such that for some* s *with* $r < s \leq t$ *holds* $h_r \neq h_s$ *we have* $\mathrm{content}(T[s]) \cap (W_{h_t} \setminus W_{h_r}) \neq \emptyset$.

As for learning from informant in Section 2.2, **Conv**(M, T) implies **SNU**(M, T), **WMon**(M, T); **SDec**(M, T) implies **Dec**(M, T), **SNU**(M, T); **SMon**(M, T) implies **Caut**(M, T), **Dec**(M, T), **Mon**(M, T), **WMon**(M, T), **Dec**(M, T); **SNU**(M, T) implies **NU**(M, T). Moreover, **WMon**(M, T) also implies **NU**(M, T). Figure 5.1 includes the resulting backbone with arrows indicating the aforementioned implications. Further, **Wb**(M, T) implies **Conv**(M, T), **SDec**(M, T) and **Caut**(M, T).

In order to characterize what successful learning means, these predicates may be combined with the explanatory convergence criterion. For this, we let $\Delta := \{$ **Caut**, **Conv**, **Dec**, **SDec**, **WMon**, **Mon**, **SMon**, **NU**, **SNU**, **T** $\}$ denote the set of *admissible learning restrictions*, with **T** standing for no restriction. Further, a *learning success criterion* is a predicate being the intersection of the convergence criterion **Ex** with arbitrarily many admissible learning restrictions. This means that the sequence of hypotheses has to converge and in addition has the desired properties. Therefore, the collection of all learning success criteria is

$$\{ \bigcap_{i=0}^{n} \delta_i \cap \mathbf{Ex} \mid n \in \mathbb{N}, \forall i \leq n (\delta_i \in \Delta) \}.$$

Note that plain explanatory convergence is a learning success criterion by letting $n = 0$ and $\delta_0 = \mathbf{T}$.

We refer to all $\delta \in \{$**Caut**, **Cons**, **Dec**, **Mon**, **SMon**, **WMon**, **NU**, **T**$\}$ also as *semantic* learning restrictions, as they do not require the learner to settle on exactly one hypothesis.

In order to state observations about how two ways of defining learning success relate to each other, the learning power of the different settings is encapsulated in notions $[\alpha\mathbf{Txt}\beta]$ defined as follows.

Definition 5.8. *Let* α *be a property of partial computable functions from the set* $\Sigma^{<\omega}$ *to* $\mathbb{N}$ *and* β *a learning success criterion. We denote by* $[\alpha\mathbf{Txt}\beta]$ *the set of all collections of languages that are* β*-learnable from text by a learner* M *with the property* α.

At position α, we restrict the set of admissible learners for example by requiring them to be iterative or finite bounded memory states learners. The properties stated at position α are *independent of learning success.* In contrast, at position β, the required learning behavior and convergence criterion are specified. We do not use separators in the notation to stay consistent with established notation in the field that was inspired by [Jai+99].

For example, a collection of languages $\mathcal{L}$ lies in [**BMSTxtBMS$_*$ConvEx**] if and only if there is a bounded memory states learner M conservatively explanatory learning every $L \in \mathcal{L}$ from text while using only finite memory. More concretely, for all $L \in \mathcal{L}$ and for every text $T \in \mathbf{Txt}(L)$ we have $\mathbf{Conv}(M, T)$, $\mathbf{BMS}_*(M, T)$ and $\mathbf{Ex}(M, T)$.

The proofs of Lemmata 5.10 and 5.12 employ the following property of learning requirements and learning success criteria, that applies to all such considered in this chapter.

Definition 5.9. *Denote the set of all unbounded and non-decreasing functions by* $\mathfrak{S}$*, i.e.,*

$$\mathfrak{S} := \{\, \mathfrak{s} : \mathbb{N} \to \mathbb{N} \mid \forall x \in \mathbb{N}\, \exists t \in \mathbb{N} \colon \mathfrak{s}(t) \geq x \text{ and } \forall t \in \mathbb{N} \colon \mathfrak{s}(t+1) \geq \mathfrak{s}(t) \,\}.$$

Then every $\mathfrak{s} \in \mathfrak{S}$ *is a so called* admissible simulating function.

A predicate β *on pairs of learners and text* allows for simulation on equivalent text*, if for all simulating functions* $\mathfrak{s} \in \mathfrak{S}$*, all text* $T, T' \in \mathbf{Txt}$ *and all learners* M, M' *holds: Whenever we have* $\mathrm{content}(T'[t]) = \mathrm{content}(T[\mathfrak{s}(t)])$ *and* $M'(T'[t]) = M(T[\mathfrak{s}(t)])$ *for all* $t \in \mathbb{N}$*, from* $\beta(M, T)$ *we can conclude* $\beta(M', T')$*.*

Intuitively, as long as the learner M' conjectures $h'_t = h_{\mathfrak{s}(t)} = M(T[\mathfrak{s}(t)])$ at time t and has, in form of $T'[t]$, the same data available as was used by M for this hypothesis, M' on T' is considered to be a simulation of M on T.

It is easy to see that all learning success criteria considered in this chapter allow for simulation on equivalent text.

5.3 Relations between Semantic Learning Requirements

We show that bounded memory states learners and iterative learners have equal learning power, when a semantic learning requirement is added to the standard convergence criterion. With this the results from iterative learning are transferred to this setting.

The following lemma formally establishes the equal learning power of iterative and $\mathbf{BMS}_*$-learning for all learning success criteria but **Conv**, **SDec** and **SNU**. We are going to prove in Section 5.4 that even for the three aforementioned non-semantic additional requirements we obtain the same behavior.

Lemma 5.10. *Let δ allow for simulation on equivalent text.*

(i) We have $[\mathbf{TxtBMS}_*\delta\mathbf{Ex}] \supseteq [\mathbf{ItTxt}\delta\mathbf{Ex}]$.

(ii) If δ is semantic then $[\mathbf{TxtBMS}_*\delta\mathbf{Ex}] = [\mathbf{ItTxt}\delta\mathbf{Ex}]$.

Proof. While (i) and "$\supseteq$" in (ii) are easy to verify by using the hypotheses as states, the other inclusion in (ii) is more challenging. The iterative learner constructed from the **BMS**-learner M uses the hypotheses of M on an equivalent text and additionally pads a subgraph of the translation diagram of M to it.

(1) and "$\supseteq$" of (2). Let M be an iterative learner, i.e. there is a computable function $h_M : \Omega \times \Sigma \to \Omega$ with $M = h_M^{\ddagger}$ where $h_M^{\ddagger}(\epsilon) = ?$ and $h_M^{\ddagger}(\sigma\hat{}x) = h_M(h_M^{\ddagger}(\sigma), x)$ for all $\sigma \in \Sigma^{<\omega}$ and $x \in \Sigma$. We show that M can be obtained as a state driven learner by using the hypotheses also as states. For this, we fix the computable bijection $\pi : Q \to \Omega$ with computable inverse, defined by $\pi(0) = ?$ and $\pi(i) = i - 1$ for all $i > 0$. Then the learner $N = h_N^*$ with $\langle s_N, h_N\rangle(q, x) = (\pi^{-1}(h_M(\pi(q), x)), h_M(\pi(q), x))$ is as wished because the state corresponds via π directly to the last hypothesis of M and so the learners M and N act identically.

Formally, this follows by an induction showing for every $\tau \in \Sigma^{<\omega}$ that $s_N^*(\tau) = \pi^{-1}(M(\tau))$ and moreover if $|\tau| > 0$ we have $N(\tau) = M(\tau)$. The claim holds for $\tau = \epsilon$, because of $s_N^*(\epsilon) = 0 = \pi^{-1}(M(\epsilon))$. In case there are $\sigma \in \Sigma^{<\omega}$ and $x \in \Sigma$ such that $\tau = \sigma\hat{}x$, we may assume $s_N^*(\sigma) = \pi^{-1}(M(\sigma))$ and obtain

$$s_N^*(\tau) \overset{\text{Def. } s_N^*}{=} s_N(s_N^*(\sigma), x) \overset{s_N^*(\sigma)=\pi^{-1}(M(\sigma))}{=} s_N(\pi^{-1}(M(\sigma)), x)$$

$$\begin{aligned} & \overset{\text{Def. } s_N}{=} \pi^{-1}(h_M(M(\sigma), x)) \overset{M=h_M^{\ddagger}}{=} \pi^{-1}(M(\tau)), \\ N(\tau) \overset{N=h_N^*}{=} h_N(s_N^*(\sigma), x) & \overset{s_N^*(\sigma)=\pi^{-1}(M(\sigma))}{=} h_N(\pi^{-1}(M(\sigma)), x) \\ & \overset{\text{Def. } h_N}{=} h_M(M(\sigma), x)) \overset{M=h_M^{\ddagger}}{=} M(\tau). \end{aligned}$$

That M in case of learning success uses only finitely many states follows immediately from the **Ex**-convergence, implying to output only finitely many pairwise distinct hypotheses.

"$\subseteq$" of (2). Let $\mathcal{L} \in [\mathbf{TxtBMS}_*\delta\mathbf{Ex}]$ be witnessed by the learner M, i.e., there is $\langle s_M, h_M \rangle : Q \times \Sigma \to Q \times \Omega$ such that $M = h_M^*$. Further, we may assume that for all $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$ the set of visited states $s_M^*[\{T[t] \mid t \in \mathbb{N}\}]$ is finite and M $\delta\mathbf{Ex}$-learns L from T.

Intuitively, the iterative learner $M_{\mathbf{It}}$ uses the hypotheses of M on an equivalent text $\hat{T}$ and additionally pads a subgraph visit(σ) of the translation diagram of the **BMS**-learner M to it. In visit(σ), which is being build after having observed σ, we keep track of all states visited so far together with the datum which caused the first transfer to the respective state. In order to assure **Ex**-convergence, we do not change the subgraph in case the new state had already been visited after some proper initial segment of σ was observed. From visit(σ) we can reconstruct the last first-time-visited state $s_{M_{\mathbf{It}}}^*(\sigma)$ of M while observing the equivalent sequence corresponding to σ. Moreover, we build the equivalent text $\hat{T}$ by inserting a path of already observed data leading to state $s_{M_{\mathbf{It}}}^*(\sigma)$, in case this is necessary to prevent the learner $M_{\mathbf{It}}$ from returning to a previously visited state but the last one. With this strategy we make sure that the last state is the one we are currently in, as keeping track of the current state while observing the original text may destroy the **Ex**-convergence.

Formally, we define functions pump : $\Sigma^{<\omega} \setminus \{\epsilon\} \times \mathbb{N} \to \Sigma^{<\omega}$ and visit : $\Sigma^{<\omega} \to \Sigma^{<\omega}$ by

$$\text{pump}(\text{visit}(\sigma), x) = \begin{cases} x, & \text{if } s_M(s_{M_{\mathbf{It}}}^*(\sigma), x) \notin \\ & \text{pr}_1[\text{visit}(\sigma)]; \\ x \hat{} \text{path}(s_M(s_{M_{\mathbf{It}}}^*(\sigma), x), s_{M_{\mathbf{It}}}^*(\sigma)), & \text{otherwise;} \end{cases}$$

$$\text{visit}(\epsilon) = \epsilon;$$

$$\mathrm{visit}(\sigma\hat{}x) = \begin{cases} \mathrm{visit}(\sigma)\hat{}\langle s_M(s^*_{M_{\mathbf{It}}}(\sigma), x), x\rangle, & \text{if } s_M(s^*_{M_{\mathbf{It}}}(\sigma), x) \notin \\ & \mathrm{pr}_1[\mathrm{visit}(\sigma)]; \\ \mathrm{visit}(\sigma), & \text{otherwise;} \end{cases}$$

with the application of the projection to the first coordinate extracting the set of visited states. Moreover, for states $s_0, s_1 \in S$ with $\mathrm{path}(s_0, s_1)$ we refer to the unique sequence $(\sigma(i), \sigma(i+1), \ldots, \sigma(j))$ of second coordinates in $\mathrm{visit}(\sigma)$ such that $(s_0, \sigma(i))\hat{} \ldots \hat{}(s_1, \sigma(j))$ is an intermediate sequence in $\mathrm{visit}(\sigma)$. The learner $M_{\mathbf{It}}$ is now defined by

$$M_{\mathbf{It}}(\sigma\hat{}x) = \mathrm{pad}(h^*_M(s^*_{M_{\mathbf{It}}}(\sigma), \mathrm{pump}(\mathrm{visit}(\sigma), x)), \mathrm{visit}(\sigma\hat{}x)).$$

By construction $s^*_{M_{\mathbf{It}}}(\sigma) = \mathrm{last}(\mathrm{pr}_1(\mathrm{visit}(\sigma)))$ and therefore the hypothesis of $M_{\mathbf{It}}$ on some sequence $\sigma\hat{}x$ is always only based on $\mathrm{visit}(\sigma)$ and x, which makes $M_{\mathbf{It}}$ iterative.

The text $\hat{T} = \bigcup_{t\in\mathbb{N}} \tau_t$ with $\tau_0 = \epsilon$ and $\tau_{t+1} = \tau_t\hat{}\mathrm{pump}(\mathrm{visit}(T[t]), T(t))$ is a text for L. Let $\mathfrak{s} : \mathbb{N} \to \mathbb{N}, t \mapsto |\tau_t|$ be the corresponding simulating function. As for all $t \in \mathbb{N}$ holds $\mathrm{content}(T[t]) = \mathrm{content}(\hat{T}[\mathfrak{s}(t)])$ and $M_{\mathbf{It}}(T[t]) = \mathrm{pad}(M(\hat{T}[\mathfrak{s}(t)]), \mathrm{visit}(T[t]))$, we obtain $W_{M_{\mathbf{It}}(T[t])} = W_{M(\hat{T}[\mathfrak{s}(t)])}$ and because δ is semantic and afsoet, we conclude the semantic δ-convergence of $M_{\mathbf{It}}$ on T. Having in mind that M uses only finitely many pairwise distinct states $\mathrm{visit}(T[t])$ stabilizes. Paired with the **Ex**-convergence of M on $\hat{T}$ we conclude the **Ex**-convergence of $M_{\mathbf{It}}$ on T. □

Note that obviously the proof is identical for learning from positive and negative information, introduced by [Gol67]. In this learning model the information the learner receives is labeled, like in binary classification, and has to be complete in the limit. See [AKS18] for a formal definition, a summary of results on this model and the complete map.

With Lemma 5.10 the following results transfer from learning with iterative learners and it remains to investigate the relations to and between the non-semantic requirements **Conv**, **SDec** and **SNU**.

Theorem 5.11. *(i)* $[\mathbf{TxtBMS}_*\mathbf{NUEx}] = [\mathbf{TxtBMS}_*\mathbf{Ex}]$

(ii) $\forall \delta \in \{\mathbf{Dec}, \mathbf{WMon}, \mathbf{Caut}\}\ [\mathbf{TxtBMS}_*\delta\mathbf{Ex}] = [\mathbf{TxtBMS}_*\mathbf{Ex}]$

(iii) $[\mathbf{TxtBMS}_*\mathbf{MonEx}] \subsetneq [\mathbf{TxtBMS}_*\mathbf{Ex}]$

(iv) $[\mathbf{TxtBMS}_*\mathbf{SMonEx}] \subsetneq [\mathbf{TxtBMS}_*\mathbf{MonEx}]$

Proof. The respective results for iterative learners are [CM08a, Theorem 2], [Jai+16, Theorem 10], [Jai+16, Theorem 3] and [Jai+16, Theorem 2]. □

5.4 Relations to and between Syntactic Learning Requirements

The following lemma establishes that we may assume $\mathbf{BMS}_*$-learners to never go back to withdrawn states. This is essential in almost all of the following proofs. It can also be used to simplify the proof of Lemma 5.10.

Lemma 5.12. *Let β be a learning success criterion allowing for simulation on equivalent text and $\mathcal{L} \in [\mathbf{TxtBMS}_*\beta]$. Then there is a* **BMS***-learner N such that N never returns to a withdrawn state and $\mathbf{BMS}_*\beta$-learns $\mathcal{L}$ from text.*

Proof. Let M be a **BMS**-learner with $\mathcal{L} \in \mathbf{TxtBMS}_*\beta(M)$. We employ a construction similar to the one in the proof of Theorem 5.10. Again for visit $\in Q \times \Sigma^{<\omega}$ with pairwise distinct first coordinates and $s' \in \mathrm{pr}_1[\mathrm{visit}]$ by $\mathrm{path}(\mathrm{visit}, s')$ we denote the unique sequence of second coordinates $x_0{}^\frown \ldots {}^\frown x_\xi$ of visit such that $(s', x_0)^\frown \ldots {}^\frown(\mathrm{last}(\mathrm{pr}_1[\mathrm{visit}]), x_\xi)$ is a final segment of visit. The **BMS** learner N is initialized with state $\mathrm{pad}(0, (0, \#))$ and for every $s \in Q$, visit $\in Q \times \Sigma^{<\omega}$ and $x \in \Sigma$ defined by

$$s_N(\langle s, \mathrm{visit}\rangle, x) = \begin{cases} \langle s, \mathrm{visit}\rangle, & \text{if } s_M(s,x) \in \mathrm{pr}_1[\mathrm{visit}]; \\ \langle s_M(s,x), \mathrm{visit}^\frown(s_M(s,x), x)\rangle, & \text{otherwise;} \end{cases}$$

$$h_N(\langle s, \mathrm{visit}\rangle, x) = \begin{cases} h_M^*(s, x^\frown \mathrm{path}(\mathrm{visit}, s_M(s,x))), & \text{if } s_M(s,x) \in \mathrm{pr}_1[\mathrm{visit}]; \\ h_M(s,x), & \text{otherwise.} \end{cases}$$

By construction N is a $\mathbf{BMS}_*$-learner, as it only uses states $\langle s, \mathrm{visit}\rangle$ where $s = \mathrm{pr}_1(\mathrm{last}(\mathrm{visit}))$ is a state used by M and for every $s \in Q$, visited by M, there is exactly one sequence visit $\in Q \times \Sigma^{<\omega}$ such that $\langle s, \mathrm{visit}\rangle$ is used by N. The learner N simulates M on an equivalent text as in the proof of Theorem 5.10. □

We show that strongly monotonically $\mathbf{BMS}_*$-learnability does not imply strongly non-U-shapedly $\mathbf{BMS}_*$-learnability.

Theorem 5.13. $[\mathbf{TxtBMS}_*\mathbf{SMonEx}] \not\subseteq [\mathbf{TxtBMS}_*\mathbf{SNUEx}]$

Proof. We define a self-learning **BMS**-learner M and with a tailored ORT-argument there can not be a **BMS**-learner strongly non-U-shapedly learning all languages that M learns strongly monotonically.

Consider the **BMS**-learner M initialized with state $\langle ?, \langle\varnothing\rangle\rangle$ and h_M and s_M for every $e \in \Omega$, $D \subseteq \mathbb{N}$ finite and $x \in \Sigma$ defined by:

$$s_M(\langle e, \langle D\rangle\rangle, x) = \begin{cases} \langle e, \langle D\rangle\rangle, & \text{if } x \in D \cup \{\#\} \ \vee \ \varphi_x(e) = e; \\ \langle \varphi_x(e), \langle D \cup \{x\}\rangle\rangle, & \text{else if } \varphi_x(e) \neq e; \\ \uparrow, & \text{otherwise.} \end{cases}$$

$$h_M(\langle e, \langle D\rangle\rangle, x) = \begin{cases} e, & \text{if } x \in D \cup \{\#\} \ \vee \ \varphi_x(e) = e; \\ \varphi_x(e), & \text{else if } \varphi_x(e) \neq e; \\ \uparrow, & \text{otherwise.} \end{cases}$$

Thus, M is self-learning by interpreting the datum x as a program and the conjectures are generated by applying this program to the last hypothesis. (We identify φ_x with the function obtained by using a bijection from $\mathbb{N}$ to Ω.) Further, in form of the states, the last hypothesis as well as exactly the data that already lead to a mind change of M is stored.

Let $\mathcal{L} = \mathbf{TxtBMS}_*\mathbf{SMonEx}(M)$.

Assume there is a $\mathbf{BMS}_*$-learner N with hypothesis generating function h_N and state transition function s_N, such that $\mathcal{L} \subseteq \mathbf{TxtBMS}_*\mathbf{SNUEx}(N)$. By Lemma 5.12 we assume that N does not return to withdrawn states.

We are going to obtain a language $L \in \mathcal{L}$ not strongly non-U-shapedly learned by N by applying 1-1 ORT and thereby refering to the c.e. predicates MC and NoMC defined for fixed $a, b \in \mathcal{R}$, all $k \in \mathbb{N}$ and $\sigma \in \Sigma^{<\omega}$ with the help of the formulas $\psi_k(\ell)$, expressing that the $\mathbf{BMS}_*$-learner N does not perform a mind- or state-change on the text $a[k]\hat{}\,b(k)\hat{}\,\#^\infty$ after having observed $a[k]\hat{}\,b(k)\hat{}\,\#^\ell$. The predicates state that N does converge and (not) make a mind-change when observing σ after having observed $a[k]\hat{}\,a(k)\hat{}\,\#^{\ell_k}$, with ℓ_k being the least ℓ with $\psi_k(\ell)$.

$$\begin{aligned} \psi_k(\ell) \Leftrightarrow\ & N(a[k]\hat{}\,b(k)\hat{}\,\#^\ell) = N(a[k]\hat{}\,b(k)\hat{}\,\#^{\ell+1}) \ \wedge \\ & s_N^*(a[k]\hat{}\,b(k)\hat{}\,\#^\ell) = s_N^*(a[k]\hat{}\,b(k)\hat{}\,\#^{\ell+1}); \\ \mathrm{NoMC}(k, \sigma) \Leftrightarrow\ & \exists \ell_k \in \mathbb{N}\ (\ \psi_k(\ell_k) \ \wedge \ \forall \ell < \ell_k\ \neg\psi_k(\ell) \ \wedge \end{aligned}$$

$$\begin{aligned} & N(a[k]\hat{}b(k)\hat{}\#^{\ell_k}\hat{}\sigma)\downarrow = N(a[k]\hat{}b(k)\hat{}\#^{\ell_k})\,); \\ \mathrm{MC}(k,\sigma) \Leftrightarrow \exists \ell_k \in \mathbb{N}\,(\,& \psi_k(\ell_k) \;\wedge\; \forall \ell < \ell_k\, \neg\psi_k(\ell) \wedge \\ & N(a[k]\hat{}b(k)\hat{}\#^{\ell_k}\hat{}\sigma)\downarrow \neq N(a[k]\hat{}b(k)\hat{}\#^{\ell_k})\,). \end{aligned}$$

Now, let p be an index for the program which on inputs $k \in \mathbb{N}$ and $\sigma \in \Sigma^{<\omega}$ searches for ℓ_k. In case ℓ_k exists, the program encoded in p runs N on $a[k]\hat{}b(k)\hat{}\#^{\ell_k}\hat{}\sigma$. Hence, $\Phi_p(k,\sigma)$ stands for the number of computation steps the program just described needs on input k,σ. By the definition of p we have $\Phi_p(k,\sigma)\uparrow$ if and only if $\ell_k\uparrow$ or $N(a[k]\hat{}b(k)\hat{}\#^{\ell_k}\hat{}\sigma)\uparrow$.

We abbreviate with $({}^{<\omega}a,i) = {}^{<\omega}{}_{\leq i}(\mathrm{ran}(a[i]) \cup \{\#\})$ the set of all finite sequences over $\mathrm{ran}(a[i]) \cup \{\#\}$ with length at most i. Moreover, we employ a well-order $<_a$ on $({}^{<\omega}\mathrm{ran}(a))$ by letting $\rho <_a \sigma$ if and only if for the unique i_ρ such that $\rho \in ({}^{<\omega}a, i_\rho+1) \setminus ({}^{<\omega}a, i_\rho)$ holds $\sigma \notin ({}^{<\omega}a, i_\rho+1)$ or else $\sigma \notin ({}^{<\omega}a, i_\rho)$ and at the same time $\langle\rho\rangle < \langle\sigma\rangle$.

For constructing L we will also make use of the c.e. sets

$$\begin{aligned} E_k = \{\, a(i) \mid \forall \sigma \in ({}^{<\omega}a, i)\ & \mathrm{NoMC}(k,\sigma) \vee \\ & (\,\exists \sigma \forall \rho <_a \sigma\, \mathrm{NoMC}(k,\rho) \;\wedge\; \Phi_p(k,\sigma) > i\,)\,\}. \end{aligned}$$

It is easy to see that E_k is finite and equals $\{\, a(i) \mid i < \max(\{i_{\sigma_0}\} \cup \{\Phi_p(k,\sigma) \mid \sigma \leq_a \sigma_0\})\,\}$ if and only if for $\sigma_0 \in ({}^{<\omega}\mathrm{ran}(a))$ holds $\mathrm{MC}(k,\sigma_0)$ and $\mathrm{NoMC}(k,\sigma)$ for all $\sigma <_a \sigma_0$. Otherwise $E_k = \mathrm{ran}(a)$.

By 1-1 ORT there are $a, b, e_1, e_2 \in \mathcal{R}$ with pairwise disjoint ranges and $e_0 \in \mathbb{N}$, such that

$$\varphi_{a(i)}(e) = \begin{cases} e_0 & \text{if } e \in \{?, e_0\}; \\ e_2(k) & \text{else if } e = e_1(k) \text{ for some } k \leq i; \\ e, & \text{otherwise;} \end{cases}$$

$$\varphi_{b(k)}(e) = \begin{cases} e_1(k) & \text{if } e \in \{?, e_0\}; \\ e, & \text{otherwise;} \end{cases}$$

$$W_{e_0} = \begin{cases} \mathrm{ran}(a[t_0]) & \text{if } t_0 \text{ is minimal with } \forall t \geq t_0 \\ & (\,N(a[t]) = N(a[t_0]) \wedge s^*_N(a[t]) = s^*_N(a[t_0])\,); \\ \mathrm{ran}(a), & \text{no such } t_0 \text{ exists.;} \end{cases}$$

$$W_{e_1(k)} = \mathrm{content}(a[k]) \cup \{b(k)\} \cup \begin{cases} E_k & \text{if } \exists \sigma_0 \, (\, \mathrm{MC}(k, \sigma_0) \wedge \\ & \forall \sigma <_a \sigma_0 \, \mathrm{NoMC}(k, \sigma) \,); \\ \emptyset, & \text{otherwise;} \end{cases}$$

$$W_{e_2(k)} = \mathrm{content}(a[k]) \cup \{b(k)\} \cup E_k.$$

As $W_{e_0} \in \mathcal{L}$ by construction, N has to learn it and hence t_0 exists.

We first observe that there exists σ_0 such that $\mathrm{MC}(t_0, \sigma_0)$ and $\mathrm{NoMC}(t_0, \sigma)$ for all $\sigma <_a \sigma_0$. Assume otherwise, then one of the following is true: $\ell_{t_0}{\uparrow}$ or for all $\sigma \in ({}^{<\omega}\mathrm{ran}(a))$ holds $\mathrm{NoMC}(t_0, \sigma)$ or for σ_0 minimal with $\neg\mathrm{NoMC}(t_0, \sigma_0)$ we have $N(a[t_0]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}{}^\frown \sigma_0){\uparrow}$. Anyhow, this would mean $E_{t_0} = \mathrm{ran}(a)$. By the definition of e_1, e_2 and our converse assumption we obtain $W_{e_1(t_0)} = \mathrm{content}(a[t_0]) \cup \{b(t_0)\}$ and $W_{e_2(t_0)} = \mathrm{ran}(a) \cup \{b(t_0)\}$. It can be easily checked that $W_{e_1(t_0)}$ and $W_{e_2(t_0)}$ are strongly monotonically learned by M and hence lie in $\mathcal{L}$. As N has to learn $W_{e_1(t_0)}$ from the text $a[t_0]{}^\frown b(t_0){}^\frown \#^\infty$, we know $\ell_{t_0}{\downarrow}$ and moreover $W_{N(a[t_0]{}^\frown b(t_0){}^\frown \#^\ell)} = W_{e_1(t_0)}$ holds for all $\ell \geq \ell_{t_0}$. Moreover, N has to learn $W_{e_2(t_0)}$ from all the text $a[t_0]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}{}^\frown \sigma{}^\frown a$ with $\sigma \in ({}^{<\omega}\mathrm{ran}(a))$. Thus, $N(a[t_0]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}{}^\frown \sigma){\downarrow}$ for all $\sigma \in ({}^{<\omega}\mathrm{ran}(a))$. Because of our converse assumption, the only option left is $\mathrm{NoMC}(t_0, \sigma)$ for all $\sigma \in ({}^{<\omega}\mathrm{ran}(a))$. Since this is equivalent to $N(a[t_0]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}{}^\frown \sigma) = N(a[t_0]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}})$ for all $\sigma \in ({}^{<\omega}\mathrm{ran}(a))$, N cannot learn both $W_{e_1(t_0)}$ and $W_{e_2(t_0)}$. Hence σ_0 exists.

By the choice of t_0 and σ_0 we obtain $E_{t_0} = \mathrm{content}(a[t_1])$ for $t_1 = \max(\{i_{\sigma_0}\} \cup \{\Phi_p(k, \sigma) \mid \sigma \leq_a \sigma_0\}) \in \mathbb{N}$. Let $\hat{t} = \max\{t_0, t_1\}$ and $L = \mathrm{content}(a[\hat{t}]) \cup \{b(t_0)\}$. Then $W_{e_1(t_0)} = W_{e_2(t_0)} = L \in \mathcal{L}$ and by construction of E_{t_0} we have $\mathbf{Cons}(\sigma_0, L)$. Because of $\hat{t} \geq t_0$, we obtain $s^*_N(a[\hat{t}]) = s^*_N(a[t_0])$. With this and the choice of t_0 we conclude $N(a[\hat{t}]{}^\frown b(t_0){}^\frown \#^\ell) = N(a[t_0]{}^\frown b(t_0){}^\frown \#^\ell)$ for all $\ell \in \mathbb{N}$. Further, as N learns L from the text $a[\hat{t}]{}^\frown b(t_0){}^\frown \#^\infty$ we have $W_{N(a[\hat{t}]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}})} = L$. On the other hand by $\mathrm{MC}(t_0, \sigma_0)$ we obtain $N(a[\hat{t}]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}) \neq N(a[\hat{t}]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}{}^\frown \sigma_0)$, which forces N to perform a syntactic U-shape on the text $a[\hat{t}]{}^\frown b(t_0){}^\frown \#^{\ell_{t_0}}{}^\frown \sigma_0{}^\frown \#^\infty$ for L. □

For inferring the relations between the syntactic learning requirements **SNU**, **SDec** and **Conv**, we refer to **Wb**. All these criteria are closely related to strongly locking learners, which we define in the following.

It was observed by [BB75] that the learnability of every language L by a learner M is witnessed by a sequence σ, consistent with L, such that $M(\sigma)$ is an index for L and no extension of σ consistent with L will lead to a mind-change of M. Such

a sequence σ is called *(sink-)locking sequence for M on L*. For a similar purpose as ours [Jai+16] introduced strongly locking learners. A learner M acts strongly locking on a language L, if for every text T for L there is an initial segment σ of T that is a locking sequence for M on L.

The proof of the following theorem generalizes the construction of a conservative and strongly decisive iterative learner from a strongly locking iterative learner in [Jai+16, Theorem 8]. With it we obtain in the Corollary thereafter, that all non-semantic learning restrictions coincide.

Theorem 5.14. *Let $\mathcal{L}$ be a set of languages* $\mathbf{BMS}_*\mathbf{Ex}$*-learned by a strongly locking* $\mathbf{BMS}$*-learner. Then*

$$\mathcal{L} \in [\mathbf{TxtBMS}_*\mathbf{WbEx}].$$

Proof. Let $\mathcal{L} \in [\mathbf{TxtBMS}_*\mathbf{Ex}]$ be learned by the strongly locking learner M. By Lemma 5.12 we may assume that M does not return to withdrawn states.

We proceed in two steps. First we construct a learner M' conservatively $\mathbf{BMS}_*\mathbf{Ex}$-learning at least $\mathcal{L}$ in a strong sense, i.e.,

$$\forall \sigma \in \Sigma^{<\omega}\, \forall x \in \Sigma\, (\, M'(\sigma\hat{}x) \neq M'(\sigma) \implies x \notin W_{M'(\sigma)}\,). \tag{5.1}$$

That we require the last datum to violate consistency with the former hypothesis fits the setting of $\mathbf{BMS}$-learners and is also called locally conservative by [JLZ06]. Second, with such a learner at hand, we are going to construct a learner N which $\mathbf{BMS}_*\mathbf{Ex}$-learns $\mathcal{L}$ in a witness-based fashion. We will do this by keeping track of all data having caused a mind-change so far. More concretely, we alter the text by excluding mind-change data causing another mind-change and make sure that the witness for the mind-change is contained in all future hypotheses.

For defining the strongly conservative learner M', we employ a one-one function $f : \mathbb{N} \times Q \to \Omega$ satisfying

$$W_{f(e,s)} = \bigcup_{t \in \mathbb{N}} \begin{cases} W_e^t, & \text{if } \forall x \in W_e^t (\, h_M(s,x) = e \ \wedge\ s_M(s,x) = s\,); \\ \varnothing, & \text{otherwise} \end{cases}$$

for every hypothesis $e \in \mathbb{N} \subseteq \Omega$ and state $s \in Q$. The existence of f is granted by the smn theorem. Thus, f takes into account only the initial part of W_e not necessary to possibly justify a mind-change or state-change later on. Now define

for all $\sigma \in \Sigma^{<\omega}$

$$M'(\sigma) = f(M(\sigma), s^*_M(\sigma)).$$

As M never returns to withdrawn states and behaves strongly locking while $\mathbf{BMS}_*\mathbf{Ex}$-learning $\mathcal{L}$, M' also $\mathbf{Ex}$-learns $\mathcal{L}$. For $\sigma \neq \epsilon$ the values of $M(\sigma)$ and $s^*_M(\sigma)$ only depend on $s^*_M(\sigma^-)$ and $\mathrm{last}(\sigma)$ and hence M' is a $\mathbf{BMS}_*$-learner with $s_{M'} = s_M$. Moreover, by construction it is conservative in the strong sense defined in (5.1).

We now define the witness-based learner N. In addition to thinning out the hypotheses of M', as we did with the hypotheses of M when constructing M' from M, we patch all data causing mind-changes to it. This data is stored in the states used by N. Further, we only alter our old hypothesis in case we can guarantee the existence of a witness justifying the possible mind-change. To do this in a computable way, we need to store also the last hypothesis of M' in the states of N.

For every datum $x \in \Sigma$, data-sequence $\sigma \in \Sigma^{<\omega}$, hypothesis $e \in \mathbb{N} \subseteq \Omega$ and every finite sequence MC of natural numbers, interpreted as pairs of hypotheses and data, we define a state transition function s_N, auxiliary hypothesis generating function M, recursive function $g : \mathbb{N}^2 \to \Omega$ and the learner N by

$$h(\langle s, \langle \mathrm{MC} \rangle\rangle, x) = \begin{cases} h_{M'}(s, \#), & \text{if } x \in \mathrm{pr}_2[\mathrm{MC}]; \\ h_{M'}(s, x), & \text{otherwise;} \end{cases}$$

$$s_N(\langle s, \langle \mathrm{MC} \rangle\rangle, x) = \begin{cases} \langle s_{M'}(s, \#), \langle \mathrm{MC} \rangle\rangle, & \text{if } x \in \mathrm{pr}_2[\mathrm{MC}] \wedge h_{M'}(s, \#) = \mathrm{pr}_1(\mathrm{last}(\mathrm{MC})); \\ \langle s_{M'}(s, \#), \langle \mathrm{MC}^\frown \langle h_{M'}(s, \#), \# \rangle\rangle\rangle, & \text{if } x \in \mathrm{pr}_2[\mathrm{MC}] \wedge h_{M'}(s, \#) \neq \mathrm{pr}_1(\mathrm{last}(\mathrm{MC})); \\ \langle s_{M'}(s, x), \langle \mathrm{MC} \rangle\rangle, & \text{else if } h_{M'}(s, x) = \mathrm{pr}_1(\mathrm{last}(\mathrm{MC})); \\ \langle s_{M'}(s, x), \langle \mathrm{MC}^\frown \langle h_{M'}(s, x), x \rangle\rangle\rangle, & \text{otherwise;} \end{cases}$$

$$W_{g(e, \langle s, \langle \mathrm{MC} \rangle\rangle)} = \mathrm{pr}_2[\mathrm{MC}] \cup W_e;$$

$$N(\sigma\hat{}x) = \begin{cases} ?, & \text{if } h^*(\sigma\hat{}x) = ?; \\ g(h^*(\sigma\hat{}x), s^*_N(\sigma\hat{}x)), & \text{else if } h^*(\sigma\hat{}x) \neq \\ & \mathrm{pr}_1(\mathrm{last}(\mathrm{decode}(\mathrm{pr}_2(s^*_N(\sigma))))); \\ N(\sigma), & \text{otherwise.} \end{cases}$$

Thus with the help of g the data stored in the second coordinates of MC is patched to the language encoded in e. Further, N only makes a mind-change if h^* does, as $h^*(\sigma) = \mathrm{pr}_1(\mathrm{last}(\mathrm{decode}(\mathrm{pr}_2(s^*_N(\sigma))))))$. The learner h^* behaves like M' on the text, in which every datum repeatedly causing a mind-change is replaced by the pause symbol.

Let $L \in \mathcal{L}$ and $T \in \mathbf{Txt}(L)$. It is easy to see that for the text T' recursively defined by

$$T'(t) = \begin{cases} \#, & \text{if } \exists s < t \ (\, T(s) = T(t) \ \wedge \ M'(T'[s]\hat{}T(s)) \neq M'(T'[s]) \,); \\ T(t), & \text{otherwise,} \end{cases}$$

holds $h^*(T[t]) = M'(T'[t])$ for all $t \in \mathbb{N}$. This follows with a simultaneous induction also showing $\mathrm{pr}_1(s^*_N(T[t])) = s^*_{M'}(T'[t])$. Hence h^* on T behaves like M' on $T' \in \mathbf{Txt}(L)$.

Because M' **Ex**-converges on T', it makes only finitely many mind-changes and uses only finitely many states, which implies that N also only uses finitely many states. Let $e = M'(T'[t_0])$ be the final correct hypothesis of M' on T' with $t_0 \in \mathbb{N}$ chosen appropriately. Because M' never returns to withdrawn states, the states of N also stabilize. Moreover, $N(T[t_0])$ has to be correct since $\mathrm{pr}_2[\mathrm{MC}] \subseteq W_e$.

As already mentioned, N learns every $L \in \mathcal{L}$ witness-based because M' is strongly conservative. Every time N performs a mind-change on T, so does M' on T'. Therefore, there is a responsible datum x which was not in the former hypothesis of M' and also has not occured so far, as no datum in T' causes more than one mind-change. This datum x will be contained in all languages hypothesized by N in the future. □

With the latter theorem it is straightforward to observe that in the $\mathbf{BMS}_*\mathbf{Ex}$-setting conservative, strongly decisive and strongly non-U-shaped **Ex**-learning are equivalent.

Corollary 5.15. $\forall \gamma, \delta \in \{\mathbf{Conv}, \mathbf{SDec}, \mathbf{SNU}\}$ $[\mathbf{TxtBMS}_*\gamma\mathbf{Ex}] = [\mathbf{TxtBMS}_*\delta\mathbf{Ex}]$.

Proof. On the one hand a conservative or strongly decisive learning behavior is also a strongly non-U-shaped learning behavior. On the other hand, a learner behaving strongly non-U-shaped proceeds strongly locking and, by Theorem 5.14, from a strongly locking learner we may construct a learner with at least equal learning power, acting witness-based and hence also conservatively and strongly decisively. □

By [Jai+16, Theorem 2] and Lemma 5.10 ((i)) we obtain

$$[\mathbf{TxtBMS_*ConvEx}] \not\subseteq [\mathbf{TxtBMS_*SMonEx}].$$

From this we conclude with Theorem 5.13 and Corollary 5.15 the following incomparability

$$[\mathbf{TxtBMS_*ConvEx}] \perp [\mathbf{TxtBMS_*SMonEx}].$$

Similarly, with [Jai+16, Theorem 3] and again Lemma 5.10 ((i)) we obtain $[\mathbf{TxtBMS_*ConvEx}] \not\subseteq [\mathbf{TxtBMS_*MonEx}]$. Moreover, Theorem 5.13 implies $[\mathbf{TxtBMS_*MonEx}] \not\subseteq [\mathbf{TxtBMS_*SNUEx}]$ and with Corollary 5.15 follows

$$[\mathbf{TxtBMS_*ConvEx}] \perp [\mathbf{TxtBMS_*MonEx}].$$

Because Theorem 5.13 also reproves $[\mathbf{TxtBMS_*SNUEx}] \subsetneq [\mathbf{TxtBMS_*Ex}]$, first observed in [CK13, Th. 3.10], we completed the map for $\mathbf{BMS_*Ex}$-learning from text. An overview is depicted in Figure 5.1.

As this map equals the one for **It**-learning, naturally the question arises, whether a result similar to Lemma 5.10 can be observed for the syntactic learning criteria. In the following we show that this is not the case.

Theorem 5.16. $[\mathbf{ItTxtSNUEx}] \subsetneq [\mathbf{TxtBMS_*SNUEx}]$

Proof. By Lemma 5.10 we have $[\mathbf{ItTxtSNUEx}] \subseteq [\mathbf{TxtBMS_*SNUEx}]$.

We consider the **BMS**-learner M initialized with state $\langle\langle ?, 0\rangle, \langle\varnothing\rangle\rangle$ and h_M and

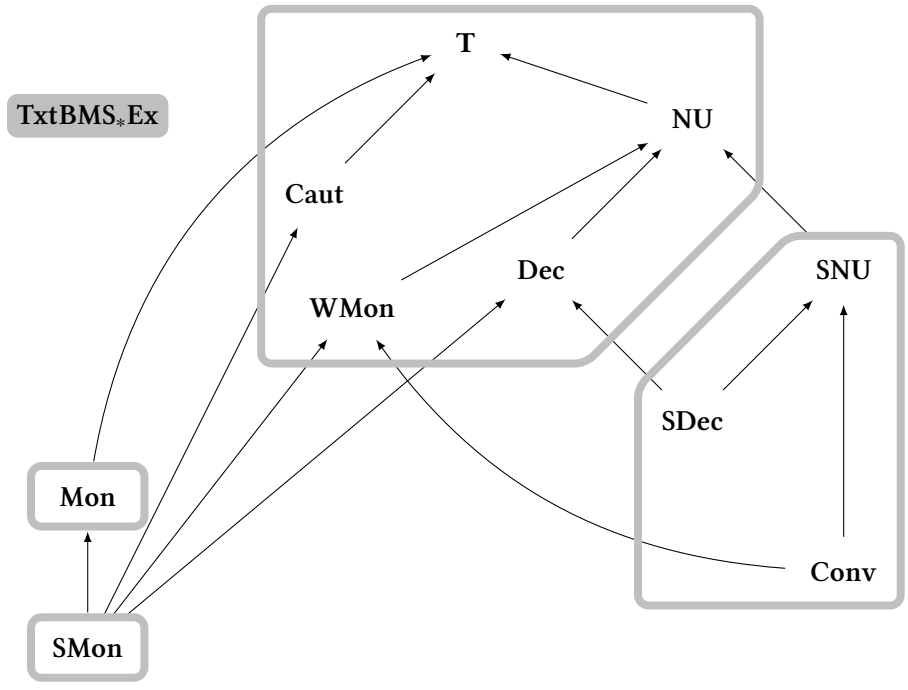

Figure 5.1: Relations between delayable learning restrictions in explanatory finitely bounded memory states learning of languages from informant. The arrows represent implications independent of the model. The outlined areas stand for equivalence classes with respect to learning power, when the underlying model is **TxtBMS$_*$Ex**.

s_M for every $\langle e, \xi\rangle \in \Omega$, $D \subseteq \mathbb{N}$ finite and $x \in \Sigma$ defined by:

$$s_M(\langle\langle e, \xi\rangle, \langle D\rangle\rangle, x) = \begin{cases} \langle\langle e, \xi\rangle, \langle D\rangle\rangle, & \text{if } x \in D \cup \{\#\} \vee \\ & \quad \mathrm{pr}_1(\,\varphi_x(\langle e, \xi\rangle)\downarrow\,) = e; \\ \langle\varphi_x(\langle e, \xi\rangle), \langle D \cup \{x\}\rangle\rangle, & \text{else if } \mathrm{pr}_1(\,\varphi_x(\langle e, \xi\rangle)\downarrow\,) \neq e; \\ \uparrow, & \text{otherwise.} \end{cases}$$

$$h_M(\langle\langle e,\xi\rangle,\langle D\rangle\rangle,x)=\begin{cases} e, & \text{if } x\in D\cup\{\#\}\ \vee \\ & \quad \mathrm{pr}_1(\,\varphi_x(\langle e,\xi\rangle)\!\downarrow\,)=e; \\ \mathrm{pr}_1(\,\varphi_x(\langle e,\xi\rangle)\,), & \text{else if } \mathrm{pr}_1(\,\varphi_x(\langle e,\xi\rangle)\!\downarrow\,)\neq e; \\ \uparrow, & \text{otherwise.}\end{cases}$$

Additionally to the last hypothesis as well as exactly the data that already lead to a mind-change of M, some parameter ξ is stored, indicating whether a further mind-change may cause a syntactic U-shape.

Let $\mathcal{L}=\mathbf{TxtBMS}_*\mathbf{SNUEx}(M)$. We will show that there is no iterative learner **ItTxtSNUEx**-learning $\mathcal{L}$. Assume N is an iterative learner with hypothesis generating function h_N and $\mathcal{L}\subseteq\mathbf{ItTxtEx}(N)$.

We obtain $L\in\mathcal{L}\setminus\mathbf{ItTxtSNUEx}(N)$ by applying 1-1 ORT [Cas74] referring to the Σ_1-predicates MC and NoMC, expressing that N does (not) perform a mind-change on a text built from parameters $a,b\in\mathcal{R}$. More specifically, the predicates state that N does converge and (not) make a mind-change when observing $\sigma\in\Sigma^{<\omega}$ after having observed $a[i]\hat{}\,b(i)\hat{}\,\#^{\ell_i}$, with $i\in\mathbb{N}$.

$$\begin{aligned}\psi_i(\ell)&\Leftrightarrow N(a[i]\hat{}\,b(i)\hat{}\,\#^{\ell})=N(a[i]\hat{}\,b(i)\hat{}\,\#^{\ell+1});\\ \mathrm{NoMC}(i,\sigma)&\Leftrightarrow\exists\ell_i\in\mathbb{N}\,(\,\psi_i(\ell_i)\ \wedge\ \forall\ell<\ell_i\,\neg\psi_i(\ell)\,\wedge\\ &\qquad\qquad N(a[i]\hat{}\,b(i)\hat{}\,\#^{\ell_i}\hat{}\,\sigma)\!\downarrow=N(a[i]\hat{}\,b(i)\hat{}\,\#^{\ell_i})\,);\\ \mathrm{MC}(i,\sigma)&\Leftrightarrow\exists\ell_i\in\mathbb{N}\,(\,\psi_i(\ell_i)\ \wedge\ \forall\ell<\ell_i\,\neg\psi_i(\ell)\,\wedge\\ &\qquad\qquad N(a[i]\hat{}\,b(i)\hat{}\,\#^{\ell_i}\hat{}\,\sigma)\!\downarrow\neq N(a[i]\hat{}\,b(i)\hat{}\,\#^{\ell_i})\,).\end{aligned}$$

By 1-1 ORT, applied to the recursive operator implicit in the following case distinction, there are recursive total functions a,b,e_1,e_2 with pairwise disjoint ranges and $e_0\in\mathbb{N}$, such that for all $i,\xi\in\mathbb{N}$, $e\in\Omega$

$$\varphi_{a(i)}(\langle e,\xi\rangle)=\begin{cases}\langle e_0,\xi\rangle, & \text{if } e\in\{?,e_0\};\\ \langle e_1(k),1\rangle, & \text{else if } \xi=0, i \text{ even and } \exists k\le i\,(\,e=e_1(k)\,);\\ \langle e_1(k),2\rangle, & \text{else if } \xi=0, i \text{ odd and } \exists k\le i\,(\,e=e_1(k)\,);\\ \langle e_2(k),0\rangle, & \text{else if } \xi=1, i \text{ odd and } \exists k\le i\,(\,e=e_1(k)\,);\\ \langle e_2(k),0\rangle, & \text{else if } \xi=2, i \text{ even and } \exists k\le i\,(\,e=e_1(k)\,);\\ \langle e,\xi\rangle, & \text{otherwise;}\end{cases}$$

$$\varphi_{b(i)}(\langle e, \xi\rangle) = \begin{cases} \langle e_1(i), \xi\rangle, & \text{if } e \in \{?, e_0\}; \\ \langle e, \xi\rangle, & \text{otherwise;} \end{cases}$$

$$W_{e_0} = \begin{cases} \operatorname{ran}(a[t_0]), & \text{if } t_0 \text{ is minimal with } \forall t \geq t_0\, N(a[t]) = N(a[t_0]); \\ \operatorname{ran}(a), & \text{no such } t_0 \text{ exists;} \end{cases}$$

$$W_{e_1(i)} = \operatorname{ran}(a[i]) \cup \{b(i)\} \cup \begin{cases} \{a(j)\} & \text{for first } j \geq i \text{ found} \\ & \text{with } \mathrm{MC}(i, a(j)); \\ \varnothing, & \text{no such } j \text{ exists;} \end{cases}$$

$$W_{e_2(i)} = \operatorname{ran}(a) \cup \{b(i)\}.$$

As the learner constantly puts out e_0 on every text for W_{e_0}, we have $W_{e_0} \in \mathcal{L}$. Thus, also N learns the finite language W_{e_0} and t_0 exists. Note that by the iterativeness of N we obtain $N(a[t_0]) = N(a[t_0]\frown a(i))$ for all $i \geq t_0$ and with this $N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}) = N(a[t_0]\frown a(i)\frown b(t_0)\frown \#^{\ell_{t_0}})$ for all $i \geq t_0$.

$W_{e_1(t_0)}$ and $W_{e_2(t_0)}$ also lie in $\mathcal{L}$. To see that M explanatory learns both of them, note that, after having observed $b(t_0)$, M only changes its mind from $e_1(t_0)$ to $e_2(t_0)$ after having seen $a(i)$ and $a(j)$ with $i, j \geq t_0$ and $i \in 2\mathbb{N}$ as well as $j \in 2\mathbb{N}+1$. This clearly happens for every text for the infinite language $W_{e_2(t_0)}$. As $|W_{e_1(t_0)} \setminus (\operatorname{content}(a[t_0]) \cup \{b(t_0)\})| \leq 1$, this mind change never occurs for any text for $W_{e_1(t_0)}$.

The syntactic non-U-shapedness of M's learning processes can be easily seen as for all $k, l \in \mathbb{N}$ the languages W_{e_0}, $W_{e_1(k)}$ and $W_{e_2(l)}$ are pairwise distinct, the learner never returns to an abandoned hypothesis and M only leaves hypothesis $\langle e_1(k), 0\rangle$ for $\langle e_1(k), \xi\rangle$, $\xi \neq 0$, if $W_{e_1(k)}$ is not correct.

Next, we show the existence of $j \geq t_0$ with $\mathrm{MC}(t_0, a(j))$. Assume towards a contradiction that j does not exist. Then $W_{e_1(t_0)} = \operatorname{content}(a[t_0]) \cup \{b(t_0)\}$. As M learns this language from the text $a[t_0]\frown b(t_0)\frown \#^\infty$, so does N. The convergence of N implies the existence of ℓ_{t_0}. Thus, for every $j \in \mathbb{N}$ we either have $N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a(j)) = N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}})$ or the computation of $N(a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a(j))$ does not terminate. Because N is iterative and learns $W_{e_2(t_0)}$, it may not be undefined and therefore always the latter is the case. But then N will not learn $W_{e_1(t_0)}$ and $W_{e_2(t_0)}$ as they are different but N does not make a mind-change on the text $a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}\frown a$ after having observed the initial segment $a[t_0]\frown b(t_0)\frown \#^{\ell_{t_0}}$, due to its iterativeness. Hence, j exists and $W_{e_1(t_0)} = \operatorname{ran}(a[t_0]) \cup \{b(t_0), a(j)\}$.

Finally, by the choice of j, the learner N does perform a syntactic U-shape on the text $a[t_0]^\frown a(j)^\frown b(t_0)^\frown \#^{\ell_{t_0}} {}^\frown a(j)^\frown \#^\infty$ for $W_{e_1(t_0)}$. More precisely, t_0 and ℓ_{t_0} were chosen such that $N(a[t_0]^\frown a(j)^\frown b(t_0)^\frown \#^{\ell_{t_0}})$ has to be correct and the characterizing property of j assures

$$N(a[t_0]^\frown a(j)^\frown b(t_0)^\frown \#^{\ell_{t_0}}) \neq N(a[t_0]^\frown a(j)^\frown b(t_0)^\frown \#^{\ell_{t_0}} {}^\frown a(j)).$$

Thus, no iterative learner can explanatory syntactically non-U-shapedly learn the language $\mathcal{L}$. □

By Corollary 5.15 we also obtain [**ItTxtSDecEx**] $\subsetneq$ [**TxtBMS$_*$SDecEx**] and [**ItTxtConvEx**] $\subsetneq$ [**TxtBMS$_*$ConvEx**].

5.5 Related Open Problems

We have given a complete map for learning with bounded memory states, where, on the way to success, the learner must use only finitely many states. Future work can address the complete maps for learning with an a priori bounded number of memory states, which needs very different combinatorial arguments. Results in this regard can be found in [Car+07] and [CK13]. We expect to see trade-offs, for example allowing for more states may make it possible to add various learning restrictions (just as non-deterministic finite automata can be made deterministic at the cost of an exponential state explosion).

Also memory-restricted learning from positive and negative data (so-called informant) has only partially been investigated for iterative learners and to our knowledge not at all for other models of memory-restricted learning. Very interesting also in regard of 1-1 hypothesis spaces that prevent coding tricks is the **Bem**-hierarchy, see [FJO94], [LZ96] and [Cas+99].

6 Conclusions & Outlook

In this thesis, we investigated different models for incremental binary classification. All of them are based on a formal model for incremental binary classification by E. M. Gold. First, we analyzed full-information learning algorithms from informant and showed that they can be assumed total for all delayable learning success criteria and that we can assume the information is presented in a canonical order. Moreover, both of these observations fail for the non-delayable requirement of consistency. We also derived all pairwise relations between established delayable learning success criteria, where conservativeness, cautiousness, monotonicity and strong monotonicity are representatives of the equivalence classes. With this we also showed that syntactic and semantic non-U-shapedness are not restrictive with respect to this model.

In addition, we observed that learning from text is strictly weaker, even with all delayable additional restrictions required from the informant learner in order to be considered successful. We also showed that when allowing for finitely many errors, we obtain a strict hierarchy depending on the permitted number of such. This does not hold if instead we vary the number of correct hypothesis the learner is allowed to vacillate between. Furthermore, we proved a duality where explanatory and behaviourally correct learning represent the equivalence classes. The complete map with respect to the hypothesis space of recursive binary classifiers instead of the W-hypothesis space closed the first part.

We then investigated iterative learning algorithms from informant and showed that their learning capability is incomparable with that of full-information learning algorithms from text. Afterwards, we showed that we can no longer assume totality but without loss of generality the iterative learning algorithms can be assumed canny. Finally, we separated non-U-shapedness.

In the final chapter, we considered bounded memory states (**BMS**) learning algorithms and derived the complete map when learning from text. In particular we showed that for all semantic learning success criteria iterative learning algorithms and **BMS** learning algorithms are equally strong. On the other hand

we observed that this is not the case for non-U-shapedness. Still as for iterative algorithms learning from text, the syntactic criteria are equivalent and restrictive.

We add some more suggestions for future research on top of what we mentioned at the end of the respective chapters.

For automatic structures as alternative approach to model a learner, there have been investigations on how different types of text effect the **Ex**-learnability, see [JLS10] and [Höl+17]. The latter started investigating how learning from canonical informant and learning from text relate to one another in the automatic setting. A lot of questions answered for the Turing machine model in this thesis are open for learning with automatic learners.

The learnability of computably presentable structures from informant has been initiated in [FKS19] and pursued further in [Bel+20] and [BFS20]. It might be interesting to look at the results again from the perspective of memory-efficiency or additional requirements on the learning process.

Regarding *iterative learning algorithms* the incomparability of **Caut** and **Mon**, as well as the separation of **Conv** are still valid for C-indices. C-Indices have been investigated for learning from positive information in [Ber+20a] and [Ber+20b], where also memory-efficiency is addressed. Still the investigations leave a lot of questions open, especially for learning from informant or with **BMS** algorithms.

Last but not least we encourage to investigate the learnability of indexable classes motivated by real-world machine learning and cognitive science research. For example, uniform families of formal languages serve as a illustrating example [JLZ07b], [LZZ08] and we discussed the learnability of half-spaces. This might also involve using techniques from other areas of mathematics and conducting more sophisticated experiments.

Bibliography

[AKS18] M. Aschenbach, T. Kötzing, and K. Seidel. **Learning from Informant: Relations between Learning Success Criteria**. *arXiv preprint arXiv: 1801.10502* (2018) (see pages 4, 65, 77, 78, 91, 96).

[Ang80] D. Angluin. **Inductive inference of formal languages from positive data**. *Information and control* 45:2 (1980), 117–135 (see pages i, iii, 5, 14, 21, 75, 91).

[AZ08] Y. Akama and T. Zeugmann. **Consistent and coherent learning with δ-delay**. *Information and Computation* 206:11 (2008), 1362–1374 (see pages 15, 32).

[Bal+08] G. Baliga, J. Case, W. Merkle, F. Stephan, and R. Wiehagen. **When Unlearning Helps**. *Information and Computation* 206 (2008), 694–709 (see pages 14, 22, 76, 87, 91).

[Bār74] J. Bārzdiņš. **Two Theorems on the Limiting Synthesis of Functions**. *In Theory of Algorithms and Programs, Latvian State University, Riga* 210 (1974), 82–88 (see pages i, 5, 44, 49).

[Bār77] J. Bārzdiņš. **Inductive Inference of Automata, Functions and Programs**. In: *Amer. Math. Soc. Transl.* 1977, 107–122 (see pages 15, 20, 43, 75).

[BB75] L. Blum and M. Blum. **Toward a Mathematical Theory of Inductive Inference**. *Information and Control* 28 (1975), 125–155 (see pages 6, 11, 15, 19, 20, 43, 75, 87, 100).

[Bel+20] D. Belanger, Z. Gao, S. Jain, W. Li, and F. Stephan. **Learnability and positive equivalence relations**. *arXiv preprint arXiv:2012.01466* (2020) (see page 110).

[Ber+20a] J. Berger, M. Böther, V. Doskoč, J. G. Harder, N. Klodt, T. Kötzing, W. Lötzsch, J. Peters, L. Schiller, L. Seifert, et al. **Learning Languages with Decidable Hypotheses**. *arXiv preprint arXiv:2011.09866* (2020) (see pages 6, 110).

[Ber+20b] J. Berger, M. Böther, V. Doskoč, J. G. Harder, N. Klodt, T. Kötzing, W. Lötzsch, J. Peters, L. Schiller, L. Seifert, et al. **Maps for Learning Indexable Classes**. *arXiv preprint arXiv:2010.09460* (2020) (see pages 6, 110).

[BFS20] N. Bazhenov, E. Fokina, and L. San Mauro. **Learning families of algebraic structures from informant**. *Information and Computation* 275 (2020), 104590 (see page 110).

[BGV92] B. E. Boser, I. M. Guyon, and V. N. Vapnik. **A training algorithm for optimal margin classifiers**. In: *Proceedings of the fifth annual workshop on Computational learning theory*. 1992, 144–152 (see page 3).

[Bis06] C. M. Bishop. **Pattern recognition and machine learning**. springer, 2006 (see page 2).

[BP73] J. Bārzdiņš and K. Podnieks. **The Theory of Inductive Inference**. In: *Mathematical Foundations of Computer Science*. 1973 (see pages iii, 52).

[Bro+17] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. **Geometric deep learning: going beyond euclidean data**. *IEEE Signal Processing Magazine* 34:4 (2017), 18–42 (see page 2).

[BS17] G. Barmpalias and F. Stephan. **Algorithmic learning of probability distributions from random data in the limit**. *arXiv preprint arXiv: 1710.11303* (2017) (see page 3).

[Car+07] L. Carlucci, J Case, S. Jain, and F. Stephan. **Results on memory-limited U-shaped learning**. *Information and Computation* 205 (2007), 1551–1573 (see pages ii, iv, 7, 84, 86, 87, 90, 108).

[Cas+99] J. Case, S. Jain, S. Lange, and T. Zeugmann. **Incremental Concept Learning For Bounded Data Mining**. *Information and Computation* 152 (1999), 74–110 (see pages 67, 84, 108).

[Cas16] J. Case. "Gold-Style Learning Theory." In: *Topics in Grammatical Inference*. 2016, 1–23 (see pages 12, 60).

[Cas74] J. Case. **Periodicity in generations of automata**. *Mathematical Systems Theory* 8:1 (1974), 15–32 (see pages 17, 65, 67, 88, 106).

[Cas94] J. Case. **Infinitary Self-Reference in Learning Theory**. *Journal of Experimental and Theoretical Artificial Intelligence* 6 (1994), 3–16 (see pages 17, 67, 88).

[Cas99] J. Case. **The power of vacillation in language learning**. *SIAM Journal on Computing* 28:6 (1999), 1941–1969 (see pages i, iii, 3, 6, 19, 44, 50).

[CC13] L. Carlucci and J. Case. **On the Necessity of U-Shaped Learning**. *Topics in Cognitive Science* 5 (2013). Invited for Special Issue on Formal Learning Theory; see `dx.doi.org/10.1111/tops.12002` for html form, 56–88 (see page 87).

[CK09] J. Case and T. Kötzing. **Difficulties in Forcing Fairness of Polynomial Time Inductive Inference**. In: *Proc. of Algorithmic Learning Theory*. 2009, 263–277 (see page 15).

[CK10] J. Case and T. Kötzing. **Strongly Non-U-Shaped Learning Results by General Techniques**. In: *COLT 2010*. Ed. by Adam Tauman Kalai and Mehryar Mohri. 2010, 181–193 (see pages ii, iv, 6, 64, 86).

[CK13] J. Case and T. Kötzing. **Memory-limited non-U-shaped learning with solved open problems**. *Theoretical Computer Science* 473 (2013), 100–123 (see pages 86, 87, 104, 108).

[CK16] J. Case and T. Kötzing. **Strongly non-U-shaped language learning results by general techniques**. *Information and Computation* 251 (2016), 1–15 (see page 87).

[CL82] J. Case and C. Lynes. **Machine Inductive Inference and Language Identification**. In: *Proc. of ICALP (International Colloquium on Automata, Languages and Programming)*. 1982, 107–115 (see pages 44, 49).

[CM07] J. Case and S. Moelius. **U-Shaped, Iterative, and Iterative-with-Counter Learning**. In: *Proceedings of the 20th Annual Conference on Learning Theory (COLT'07)*. Ed. by N. Bshouty and C. Gentile. Vol. 4539. Lecture Notes in Artificial Intelligence. 2007, 172–186 (see page 73).

[CM08a] J. Case and S. Moelius. **U-shaped, iterative, and iterative-with-counter learning**. *Machine Learning* 72 (2008), 63–88 (see pages ii, iv, 6, 86, 97).

[CM08b] J. Case and S. E. Moelius. **U-shaped, iterative, and iterative-with-counter learning**. *Machine Learning* 72 (2008), 63–88 (see pages 64, 65).

[CM09] J. Case and S. Moelius. **Parallelism increases iterative learning power**. *Theoretical Computer Science* 410:19 (2009), 1863–1875 (see pages 65, 72).

[CM11] J. Case and S. Moelius. **Optimal language learning from positive data**. *Information and Computation* 209 (2011), 1293–1311 (see pages 22, 76, 87, 91).

[Cor08] G. V. Cormack. **Email spam filtering: A systematic review**. Now Publishers Inc, 2008 (see page 1).

[CS83] J. Case and C. Smith. **Comparison of identification criteria for machine inductive inference**. *Theoretical Computer Science* 25:2 (1983), 193–220 (see pages i, 5, 44, 49, 52).

[Dad+19] E. G. Dada, J. S. Bassi, H. Chiroma, A. O. Adetunmbi, O. E. Ajibuwa, et al. **Machine learning for email spam filtering: review, approaches and open research problems**. *Heliyon* 5:6 (2019), e01802 (see page 1).

[DK20] V. Doskoč and T. Kötzing. **Cautious Limit Learning**. In: *Algorithmic Learning Theory (ALT)*. 2020, 251–276 (see page 60).

[DK21a] V. Doskoč and T. Kötzing. **Mapping Monotonic Restrictions in Inductive Inference**. In: *Computability in Europe (CiE)*. 2021 (see page 60).

[DK21b] V. Doskoč and T. Kötzing. **Normal Forms for Semantically Witness-Based Learners in Inductive Inference**. In: *Computability in Europe (CiE)*. 2021 (see page 60).

[FJO94] M. Fulk, S. Jain, and D. Osherson. **Open Problems in Systems That Learn**. *Journal of Computer and System Sciences* 49:3 (Dec. 1994), 589–604 (see pages 84, 108).

[FKS19] E. Fokina, T. Kötzing, and L. San Mauro. **Limit learning equivalence structures**. In: *Algorithmic Learning Theory*. PMLR. 2019, 383–403 (see page 110).

[Ful85] M. Fulk. **A Study of Inductive Inference Machines**. PhD thesis. SUNY at Buffalo, 1985 (see page 27).

[Gao+17] Z. Gao, C. Ries, H. U. Simon, and S. Zilles. **Preference-based teaching**. *The Journal of Machine Learning Research* 18:1 (2017), 1012–1043 (see page 3).

[Gao+19] Z. Gao, S. Jain, B. Khoussainov, W. Li, A. Melnikov, K. Seidel, and F. Stephan. **Random Subgroups of Rationals**. *arXiv preprint arXiv: 1901.04743* (2019) (see page 43).

[GBC16] I. Goodfellow, Y. Bengio, and A. Courville. **Deep Learning**. http://www.deeplearningbook.org. MIT Press, 2016 (see page 1).

[Gol67] E. Gold. **Language Identification in the Limit**. *Information and Control* 10 (1967), 447–474 (see pages i, iii, 3, 4, 11, 15, 18, 19, 28, 30, 43, 45, 63, 66, 85, 90, 96).

[Han16] S. Hanneke. **The optimal sample complexity of PAC learning**. *The Journal of Machine Learning Research* 17:1 (2016), 1319–1333 (see page 3).

[Höl+17] R. Hölzl, S. Jain, P. Schlicht, K. Seidel, and F. Stephan. **Automatic Learning from Repetitive Texts**. In: *Proc. of Algorithmic Learning Theory*. 2017, 129–150 (see pages 43, 110).

[Jai+16] S. Jain, T. Kötzing, J. Ma, and F. Stephan. **On the role of update constraints and text-types in iterative learning**. *Information and Computation* 247 (2016), 152–168 (see pages ii, iv, 4, 7, 12, 21, 42, 64, 65, 73, 76, 78, 83, 86, 87, 91, 97, 101, 104).

[Jai+99] S. Jain, D. Osherson, J. Royer, and A. Sharma. **Systems that Learn: An Introduction to Learning Theory**. Second. MIT Press, Cambridge, Massachusetts, 1999 (see pages 2, 12, 16, 30, 60, 64, 66–68, 86, 88, 93).

[Jan91] K. P. Jantke. **Monotonic and Nonmonotonic Inductive Inference of Functions and Patterns**. In: *Nonmonotonic and Inductive Logic, 1st International Workshop, Proc.* 1991, 161–177 (see pages 22, 75, 76, 91).

[JB80] K. Jantke and H. Beick. **Combining postulates of naturalness in inductive inference**. Humboldt-Universität zu Berlin. Sektion Mathematik, 1980 (see page 60).

[JLS10] S. Jain, Q. Luo, and F. Stephan. **Learnability of Automatic Classes.** In: *LATA*. 2010, 321–332 (see page 110).

[JLZ06] S. Jain, S. Lange, and S. Zilles. **Towards a Better Understanding of Incremental Learning**. In: *ALT*. Vol. 4264. Lecture Notes in Computer Science. 2006, 169–183 (see page 101).

[JLZ07a] S. Jain, S. Lange, and S. Zilles. **Some natural conditions on incremental learning**. *Information and Computation* 205:11 (2007), 1671–1684 (see pages ii, iv, 7).

[JLZ07b] S. Jain, S. Lange, and S. Zilles. **Some natural conditions on incremental learning**. *Information and Computation* 205 (2007), 1671–1684 (see pages 42, 64, 65, 77, 83, 110).

[JMZ13] S. Jain, S. Moelius, and S. Zilles. **Learning without coding**. *Theoretical Computer Science* 473 (2013), 124–148 (see pages 64, 86).

[JS98] S. Jain and A. Sharma. **Generalization and Specialization Strategies for Learning r.e. Languages**. *Annals of Mathematics and Artificial Intelligence* 23:1-2 (1998), 1–26 (see page 14).

[KKS20] A. Khazraei, T. Kötzing, and K. Seidel. **Learning Half-Spaces and other Concept Classes in the Limit with Iterative Learners**. *arXiv preprint arXiv:2010.03227* (2020) (see page 6).

[Köt09] T. Kötzing. **Abstraction and Complexity in Computational Learning in the Limit**. PhD thesis. University of Delaware, 2009 (see pages ii, iv, 4, 8, 17, 66, 67, 88).

[KP14] T. Kötzing and R. Palenta. **A map of update constraints in inductive inference**. In: *Algorithmic Learning Theory*. 2014, 40–54 (see pages 4, 12, 25, 76).

[KP16] T. Kötzing and R. Palenta. **A map of update constraints in inductive inference**. *Theoretical Computer Science* 650 (2016), 4–24 (see pages i, iii, 5, 13–15, 21, 22, 27, 34, 65, 91, 92).

[KS16] T. Kötzing and M. Schirneck. **Towards an Atlas of Computational Learning Theory**. In: *33rd Symposium on Theoretical Aspects of Computer Science*. 2016 (see pages ii, iv, 4, 7, 12, 15, 21, 65, 83, 87, 91, 92).

[KS95] E. Kinber and F. Stephan. **Language learning from texts: mindchanges, limited memory, and monotonicity**. *Information and Computation* 123:2 (1995), 224–241 (see pages 14, 15).

[KSS17] T. Kötzing, M. Schirneck, and K. Seidel. **Normal Forms in Semantic Language Identification**. In: *Proc. of Algorithmic Learning Theory*. PMLR, 2017, 493–516 (see pages 13, 60, 65, 87).

[LG03] S. Lange and G. Grieser. **Variants of iterative learning**. *Theoretical computer science* 292:2 (2003), 359–376 (see page 42).

[LZ91] S. Lange and T. Zeugmann. **Monotonic versus non-monotonic language learning**. In: *International Workshop on Nonmonotonic and Inductive Logic*. 1991, 254–269 (see pages ii, iv, 6).

[LZ93] S. Lange and T. Zeugmann. **Monotonic versus Non-monotonic Language Learning**. In: *Proc. of Nonmonotonic and Inductive Logic*. 1993, 254–269 (see pages 5, 44, 45).

[LZ94] S. Lange and T. Zeugmann. **Characterization of language learning from informant under various monotonicity constraints**. *Journal of Experimental & Theoretical Artificial Intelligence* 6:1 (1994), 73–94 (see page 14).

[LZ96] S. Lange and T. Zeugmann. **Incremental Learning from Positive Data**. *Journal of Computer and System Sciences* 53 (1996), 88–103 (see pages 67, 84, 108).

[LZK96] S. Lange, T. Zeugmann, and S. Kapur. **Monotonic and dual monotonic language learning**. *Theoretical Computer Science* 155:2 (1996), 365–410 (see pages 5, 11, 13, 14, 39).

[LZZ08] S. Lange, T. Zeugmann, and S. Zilles. **Learning indexed families of recursive languages from positive data: A survey**. *Theoretical Computer Science* 397:1 (2008), 194–232 (see pages 3, 66, 75, 110).

[Mar+92] G. Marcus, S. Pinker, M. Ullman, M. Hollander, T.J. Rosen, and F. Xu. **Overregularization in Language Acquisition**. Monographs of the Society for Research in Child Development, vol. 57, no. 4. Includes commentary by H. Clahsen. University of Chicago Press, 1992 (see page 87).

[Odi92] P. Odifreddi. **Classical recursion theory: The theory of functions and sets of natural numbers**. Elsevier, 1992 (see pages 2, 4, 67, 88).

[Odi99] P. Odifreddi. **Classical Recursion Theory**. Vol. II. Elsivier, Amsterdam, 1999 (see pages 16, 66, 88).

[OSW82] D. Osherson, M. Stob, and S. Weinstein. **Learning Strategies**. *Information and Control* 53 (1982), 32–51 (see pages 21, 75, 91).

[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 (see pages 5, 6, 11, 15, 21, 39, 42, 66, 67, 75, 91).

[OW82] D. Osherson and S. Weinstein. **Criteria of Language Learning**. *Information and Control* 52 (1982), 123–138 (see page 19).

[Pit84] L. Pitt. **A Characterization of Probabilistic Inference**. PhD thesis. Yale University, 1984 (see page 3).

[Pit89] L. Pitt. **Inductive Inference, DFAs, and Computational Complexity**. In: *Proc. of AII (Analogical and Inductive Inference)*. 1989, 18–44 (see page 15).

[R W76] R. Wiehagen. **Limes-Erkennung rekursiver Funktionen durch spezielle Strategien**. *J. Inf. Process. Cybern.* 12 (1-2) (1976), 93–99 (see pages ii, iv, 6, 64, 67).

[RC94] J. Royer and J. Case. **Subrecursive Programming Systems: Complexity and Succinctness**. Research monograph in *Progress in Theoretical Computer Science*. Birkhäuser Boston, 1994 (see pages 3, 17, 71, 88).

[Rog67] H. Rogers. **Theory of Recursive Functions and Effective Computability**. Reprinted, MIT Press, 1987. McGraw Hill, New York, 1967 (see page 66).

[Ros58] F. Rosenblatt. **The perceptron: a probabilistic model for information storage and organization in the brain.** *Psychological review* 65:6 (1958), 386 (see page 3).

[SB14] S. Shalev-Shwartz and S. Ben-David. **Understanding machine learning: From theory to algorithms**. Cambridge university press, 2014 (see pages 2–4).

[Sca+08] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. **The graph neural network model**. *IEEE transactions on neural networks* 20:1 (2008), 61–80 (see page 2).

[Sch84] G. Schäfer-Richter. **Über Eingabeabhängigkeit und Komplexität von Inferenzstrategien**. PhD thesis. RWTH Aachen, 1984 (see page 27).

[Sha15] O. Shamir. **The sample complexity of learning linear predictors with the squared loss**. *The Journal of Machine Learning Research* 16:1 (2015), 3475–3486 (see page 3).

[SS82] S. Strauss and R. Stavy, eds. **U-Shaped Behavioral Growth**. Developmental Psychology Series. Academic Press, NY, 1982 (see pages ii, iv, 7, 87).

[ST92] S. Lange and T. Zeugmann. **Types of Monotonic Language Learning and Their Characterization**. In: *Proc. 5th Annual ACM Workshop on Comput. Learning Theory*. New York, NY, 1992, 377–390 (see pages 42, 65, 77).

[Val84] L. Valiant. **A theory of the Learnable**. *Communications of the ACM* 27 (1984), 1134–1142 (see page 3).

[WC80] K. Wexler and P. Culicover. **Formal Principles of Language Acquisition**. MIT Press, Cambridge, Massachusetts, 1980 (see page 27).

[Wie91] R. Wiehagen. **A Thesis in Inductive Inference**. In: *Nonmonotonic and Inductive Logic, 1st International Workshop, Proc.* 1991, 184–207 (see pages 22, 75, 76, 91).

[Wig19] A. Wigderson. **Mathematics and Computation: A Theory Revolutionizing Technology and Science**. Princeton University Press, 2019 (see page 1).

[WZ95] R. Wiehagen and T. Zeugmann. "Learning and consistency." In: *Algorithmic Learning for Knowledge-Based Systems*. 1995, 1–24 (see page 32).

[ZZ08] T. Zeugmann and S. Zilles. **Learning recursive functions: A survey**. *Theoretical Computer Science* 397 (2008), 4–56 (see page 60).

List of Publications

Journal Versions of Articles

[1] **Learning from Informants: Relations between Learning Success Criteria**. *arXiv preprint arXiv:1801.10502* (2018). Joint work with M. Aschenbach and T. Kötzing.

[2] **Random Subgroups of Rationals**. *arXiv preprint arXiv:1901.04743* (2019). Joint work with Ziyuan Gao, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, Alexander Melnikov, and Frank Stephan.

[3] **Learning Half-Spaces and other Concept Classes in the Limit with Iterative Learners**. *arXiv preprint arXiv:2010.03227* (2020). Joint work with Ardalan Khazraei and Timo Kötzing.

[4] **Learning Languages in the Limit from Positive Information with Finitely Many Memory Changes**. *arXiv preprint arXiv:2010.04782* (2020). Joint work with Timo Kötzing.

Conference Versions of Articles

[5] **Random Subgroups of Rationals**. In: *44th International Symposium on Mathematical Foundations of Computer Science (MFCS 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2019. Joint work with Ziyuan Gao, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, Alexander Melnikov, and Frank Stephan.

[6] **Automatic Learning from Repetitive Texts**. In: *Proc. of Algorithmic Learning Theory*. 2017, 129–150. Joint work with R. Hölzl, S. Jain, P. Schlicht, and F. Stephan.

[7] **Learning Languages in the Limit from Positive Information with Finitely Many Memory Changes**. In: *Conference on Computability in Europe*. 2021, 318–329. Joint work with T. Kötzing.

[8] **Towards a Map for Incremental Learning in the Limit from Positive and Negative Information**. In: *Conference on Computability in Europe*. 2021, 273–284. Joint work with T. Kötzing.

[9] **Normal Forms in Semantic Language Identification**. In: *Proc. of Algorithmic Learning Theory*. 2017, 493–516. Joint work with T. Kötzing and M. Schirneck.