Probabilistic Learning of Indexed Families
under Monotonicity Constraints

Hierarchy Results and
Complexity Aspects

vorgelegt von

Diplom Mathematikerin

Lea Meyer

aus Kehl

Dissertation zur Erlangung des Doktorgrades der Fakultät für Angewandte
Wissenschaften der Albert-Ludwigs-Universität Freiburg im Breisgau
Abstract

We are concerned with probabilistic identification of indexed families of uniformly recursive languages from positive data under monotonicity constraints. Thereby, we consider conservative, strong-monotonic and monotonic probabilistic learning of indexed families with respect to class comprising, class preserving and proper hypothesis spaces, and investigate the probabilistic hierarchies in these learning models.

In the setting of learning indexed families, probabilistic learning under monotonicity constraints is more powerful than deterministic learning under monotonicity constraints, even if the probability is close to 1, provided the learning machines are restricted to proper or class preserving hypothesis spaces. In the class comprising case, each of the investigated probabilistic hierarchies has a threshold. In particular, we can show for class comprising conservative learning as well as for learning without additional constraints that probabilistic identification and team identification are equivalent. This yields discrete probabilistic hierarchies in these cases.

In the second part of our work, we investigate the relation between probabilistic learning and oracle identification under monotonicity constraints. We deal with the question how much additional information provided by oracles is sufficient and necessary for compensating the additional power of probabilistic learning machines. In this context, we introduce a complexity measure in order to measure the additional power of the probabilistic machines in qualitative terms. One main result is that for each oracle $A \leq_T K$, there exists an indexed family $L_A$ which is properly conservatively identifiable with $p = 1/2$, and which exactly reflects the Turing degree of $A$, i.e., $L_A$ is properly conservatively identifiable by an oracle machine $M[B]$ iff $A \leq_T B$. However, not every indexed family which is conservatively identifiable with probability $p = 1/2$ reflects the Turing degree of an oracle. Hence, the conservative probabilistic learning classes are higher structured than the Turing degrees below $K$. Finally, we prove that there exist learning problems which are conservatively (monotonically) identifiable with probability $p = 1/2$ ($p = 2/3$), but conservatively (monotonically) identifiable only by oracle machines having access to $\text{TOT}$. 

Dekan: Prof. Dr. Gerald Urban
Vorsitzender der Prüfungskommission: Prof. Dr. Thomas Ottmann
Erstreferent: Prof. Dr. Britta Schinzel
Zweitreferent: Prof. Dr. Thomas Zeugmann (Uni Lübeck)
Datum der Disputation: 31. Mai 2001
Acknowledgements

The author wishes to thank Professor Thomas Zeugmann for suggesting the topic of this work, for support and helpful discussion, and Britta Schinzel for encouragement, support and helpful comments.

Freiburg, August 2000
## Contents

1 Introduction, Preliminaries and Results ................................................. 6
   1.1 Introduction ................................................................................. 6
   1.1.1 Motivation ............................................................................. 6
   1.1.2 Inductive Inference ................................................................. 7
   1.1.3 Monotonicity Constraints ....................................................... 9
   1.1.4 Probabilistic Inference ............................................................. 11
   1.1.5 Complexity aspects of Probabilistic Inductive Inference .......... 14
   1.1.6 Oracle Complexity ................................................................ 16
   1.2 Preliminaries .............................................................................. 18
   1.2.1 Standard definitions .............................................................. 18
   1.2.2 Definitions and basic results from Inductive Inference ........... 20
   1.2.3 Oracle Identification ............................................................... 26
   1.2.4 Probabilistic Inductive Inference ............................................. 27
   1.2.5 The key definitions ............................................................... 31
   1.3 Results ..................................................................................... 33

2 Probabilistic Hierarchies ........................................................................... 38
   2.1 Proper Probabilistic Learning under Monotonicity Constraints .... 39
      2.1.1 Conservative and Strong-monotonic Probabilistic Learn-    ing ................................................................. 39
      2.1.2 Monotonic Probabilistic Learning ................................. 49
      2.1.3 Proper Probabilistic Learning with probability \( p = 1 \) .... 54
   2.2 Class Preserving Learning under Monotonicity Constraints ....... 60
      2.2.1 Conservative Probabilistic Learning with probability \( p \geq \frac{1}{3} \) ................................................................. 60
      2.2.2 Monotonic Probabilistic Learning with probability \( p \geq \frac{2}{3} \) ................................................................. 67
      2.2.3 Strong-monotonic Probabilistic Learning ......................... 72
2.3 Class Comprising Probabilistic Learning 76
   2.3.1 Probabilistic Learning of indexed families without additional constraints 76
   2.3.2 Conservative and Strong-monotonic Probabilistic Learning 81
   2.3.3 Monotonic Probabilistic Learning with probability $p \geq 2/3$ 85

2.4 Conclusion 90

3 Complexity aspects of probabilistic learning 92
   3.1 Comparing the power of Probabilistic Learning and Oracle Identification 93
      3.1.1 The power of the oracle $\mathcal{K}$ 93
      3.1.2 When $\mathcal{K}$ is not sufficient 105
      3.1.3 The power of Peano-complete oracles 108
      3.1.4 Incomparability of probabilistic learning classes and oracle learning classes 112
   3.2 Oracle Complexity 115
      3.2.1 Characterization of oracles below $\mathcal{K}$ 115
      3.2.2 Maximal complicated problems 123
      3.2.3 Characterization of various sets 127
      3.2.4 Indexed families with high oracle-complexity 130
      3.2.5 Indexed families with maximal oracle-complexity 138
Chapter 1

Introduction, Preliminaries and Results

1.1 Introduction

1.1.1 Motivation

Many fields in Machine Learning are concerned with investigating and formalizing human inference processes in order to utilize the results in designing computer programs. In our work, we deal with a formalization of *inductive inference*. We investigate a class of probabilistic inductive inference models, namely *probabilistic inductive inference of indexed families under monotonicity constraints*. All these inference models are based on the inductive inference model introduced by Gold [48]. We discuss in detail how powerful these probabilistic learning models are in contrast to their deterministic counterparts and introduce a complexity measure in order to measure the additional power of the probabilistic machines in qualitative terms.

In the following, we give an informal description of the inference model we investigate. We start with the basic inductive inference of Gold. After that, we describe the modifications of Gold’s model we consider. Detailed definitions are given in Section 1.2. For a survey on the theoretical and application oriented results in the field of inductive inference, we refer the reader to [9].
1.1.2 Inductive Inference

Inductive inference is the process of determining a general rule from a set of examples of the rule. A well-known example for inductive inference is the classical method of the natural sciences, where a scientist uses the outcome of a finite number of observations or experiments for constituting a predictive theory of a phenomenon. Another example is childrens ability to learn their mother tongue on the basis of finite, incomplete and ambiguous information.

Inductive inference has applications in many sciences as, for example, pattern recognition, cryptography or linguistic. Recently, inductive inference finds applications in the rapidly growing field of data mining, i.e., the analysis of huge sets of data with respect to correlations between the data instances (cf. [85] for an overview).

One of the first theoretical studies of inductive inference was performed by Gold (cf. [48]), who introduced a theory of formal language learning. The general situation investigated in language learning in the limit can be described as follows.

An inductive inference machine is an algorithmic device which is fed more and more information about a language to be inferred. This information can consist of positive and negative examples or only positive ones. We consider the case of learning from text. Thereby, a text for a language $L$ is an infinite sequence of strings that eventually contains all strings of $L$. When fed a text for a language $L$, the inductive inference machine has to produce hypotheses about $L$. The hypotheses, the learner produces, have to be members of an admissible set of hypotheses. Every such admissible set is called hypothesis space. The hypothesis space may be a set of grammars or a set of decision procedures for the languages to be learned. Finally, the sequence of hypotheses has to converge to a hypothesis correctly describing the language $L$ to be learned. If the learner converges for every positive presentation for $L$ to a correct description of $L$, then it is said to identify the language in the limit from text. A learner identifies a collection of languages in the limit from text if and only if it identifies each member of this collection in the limit from text.

In our work, we consider a variant of the basic model. We exclusively deal with identification of collections of recursive languages. Since we are interested in potential applications of our work, we do not consider arbitrary collections of recursive languages, but restrict ourselves to indexed families of uniformly recursive languages, i.e., families $L = (L_i)_{i \in \mathbb{N}}$ of recursive lan-
guages with uniformly decidable membership problem. Learning of indexed families was first studied by Angluin (cf. [4]), and further investigated by various authors (cf., e.g., [57, 70, 76, 91, 98, 108, 109], and the references therein).

Next, we specify which hypothesis space the inference machines may use. As mentioned above, the hypothesis space may be a set of grammars or a set of decision procedures for the target languages. In general, learning machines synthesizing grammars are more powerful than learning machines which have to produce decision procedures for the languages to be inferred (cf. [24, 48, 103]). Lange and Zeugmann [68] showed that in the case of identification of indexed families, there is no loss of learning power when the machines are required to output decision procedures for the target languages. Therefore, we may assume, without losing learning power, that the learning machines output grammars as hypotheses. However, we do not allow arbitrary sets of grammars as hypothesis spaces, but only enumerable families of grammars with uniformly decidable membership (cf., e.g., [109]). Obviously, the hypothesis space for an indexed family \( L \) has to contain at least one description for each \( L \in \text{range}(\mathcal{L}) \). Hence, the learning machines may use the indexed family \( \mathcal{L} \) itself as hypothesis space (cf., e.g., [4, 57, 76, 91]). If \( \mathcal{L} \) can be learned with respect to \( \mathcal{L} \) itself, then we call \( \mathcal{L} \) properly identifiable.\(^1\) However, there are many results showing that the requirement to learn properly may lead to a decrease of the learning power (cf., e.g., [68, 70, 109]). Therefore, we investigate not only proper learning, but additionally class preserving learning. In the case of class preserving learning, the learner is allowed to choose a possibly different enumeration of \( \mathcal{L} \) and possibly different descriptions of the target languages. More formally, \( \mathcal{L} \) is identifiable with respect to a class preserving hypothesis space if and only if there are an inductive inference machine \( M \) and a hypothesis space \( G \) which has a range equal to \( \text{range}(\mathcal{L}) \) such that \( M \) learns \( \mathcal{L} \) and only chooses hypotheses from \( G \).

Until now, we have required that the hypothesis space excludes grammars describing languages which are not contained in the range of the indexed family to be learned. This seems to be a natural requirement, since such hypotheses are not correct. However, it may be appropriate to allow

\(^1\)Initially, learning with respect to \( \mathcal{L} \) itself was denoted by exact learning (cf., e.g., [109]). However, the term “exact learning” is also used to denote another learning model, where the inductive learning machine is claimed to learn the indexed family \( \mathcal{L} \), but no proper superset of \( \mathcal{L} \) (cf., e.g., [78]). Therefore, we use the term “proper learning”.

8
the learning machine to construct new hypotheses during the learning process. These constructed hypotheses may describe languages which are not contained in the range of the target family. Consequently, we consider the case of class comprising learning. Thereby, an indexed family $\mathcal{L}$ is identifiable with respect to a class comprising hypothesis space if and only if there are an inductive inference machine $M$ and a hypothesis space $\mathcal{G}$ which has a range comprising $\text{range}(\mathcal{L})$ such that $M$ learns $\mathcal{L}$ and only chooses hypotheses from $\mathcal{G}$. There are lots of results showing that an enlargement of the hypothesis space not only affects the learning power, but also the efficiency of learning, see, for example, [67, 68, 69, 70]. For an overview, we refer the reader to [109].

1.1.3 Monotonicity Constraints

In the last decades, Gold’s learning model [48] has been refined and modified in various ways (cf., e.g., [9, 25, 79] for an overview). One of the most important modifications concerned a major problem arising when learning languages from text, namely to avoid or detect overgeneralizations, i.e., hypotheses that describe proper supersets of the language to be inferred. This problem is also called the subset problem. In order to handle the subset problem, several authors proposed the so-called subset principle (cf., e.g., [17, 101]). Informally, the subset principle can be described as follows. The learner has to guess the “least” language from the hypothesis space with respect to set inclusion that fits the data the learner has seen so far. The notion of conservative learning, introduced by Angluin [4], is one possible formalization of this requirement. Thereby, a learner is called conservative (cf. Definition 1.2.2) if and only if it may change its hypothesis only if it receives an input string which is not contained in the language generated by the current hypothesis. Thus, the learner does not adopt a new hypothesis as long as the actual hypothesis fits the experiences made so far. Hence, the learner behaves conservatively in the sense that it prefers a well-established explanation for a phenomenon instead of risking to reject a possibly good hypothesis. For more information about conservative learning, we refer the reader to [4, 109].

There is yet another interpretation of conservative learning. When observing human inference processes, we notice that people often use learning strategies to improve their hypotheses during the learning process, i.e., they reject a hypothesis only if they are convinced that the new hypothesis is
better than the previous conjecture with respect to the given learning problem. Generalization strategies as well as specialization strategies belong to the most important learning heuristics that are used to guarantee the improvement of the hypotheses during the learning process. In our work, we deal with several formalizations of generalization strategies, i.e., we consider learning algorithms that start by hypothesizing a grammar for a language “smaller” than the language $L$ to be learned, and refine this hypothesis gradually, until a correct hypothesis for $L$ is found.

Jantke [56] interpreted generalization in its strongest sense by defining the notion of strong-monotonicity (cf. Definition 1.2.2). Thereby, the learner, when successively fed a text for the target language $L$, has to produce a chain of hypotheses such that $L_i \subseteq L_j$, if $j$ is guessed later than $i$. Since strong-monotonicity is a very restrictive constraint on the behavior of an inductive inference machine (cf. [66]), it seems only natural to introduce weaker formalizations of the generalization principle. One of these notions is due to Wiehagen [105], namely monotonicity. Informally, the learner, when successively fed a text for the target language $L$, learns monotonically (cf. Definition 1.2.2), if it produces a chain of hypotheses such that, for any two hypotheses, the hypothesis produced later is as least as good as the earlier one with respect to $L$. More precisely, we require that $L_i \cap L \subseteq L_j \cap L$, if $j$ is conjectured after $i$. Besides monotonic learning, we consider another weakening of strong-monotonicity which is called weak-monotonicity (cf. [56]). Weak-monotonicity can be described as follows (cf. Definition 1.2.2). Assume that the learner conjectures $j$ after $i$. Then the following condition holds. If the set of strings seen by the learner, when $j$ is guessed, is a subset of $L_i$, then $L_i \subseteq L_j$. For more information about monotonic learning of recursive or recursively enumerable languages, we refer the reader to [52, 56, 57, 105, 109].

Recently, the impact of monotonicity constraints on the power of inductive inference machines learning indexed families has been investigated intensively. In particular, it was shown that the learning power of an inductive inference machine strongly depends on the hypothesis space the learner may use, see, for example, Lange and Zeugmann [65, 66, 67, 68, 109], and Lange, Zeugmann and Kapur [70].

Furthermore, Lange and Zeugmann [65] showed that weak-monotonic learning and conservative learning of indexed families are equivalent. Thus, conservative learning can be viewed as a generalization strategy.
1.1.4 Probabilistic Inference

In the inference models described above, the learning machines are required to learn \textit{deterministically}, i.e., the learner solves a given problem successfully with respect to a given learning model if and only if it produces a correct solution, and adheres to the constraints required by the learning model. However, when observing the human ability to learn with high probabilities, we notice that people often enhance their learning power by allowing that their learning processes fail with a small probability. In particular, the learning power \textit{strictly} decreases with increasing probability. Hence, the question arises whether the probabilistic learning models studied in the last decades reflect this ability.

In the following, we describe two well-known formal models of \textit{probabilistic learning}. After that, we give a review on previous results in the field of probabilistic inductive inference which are related to our work.

In Learning Theory, there are two major branches of probabilistic learning, namely \textit{PAC-Learning}, introduced by Valiant [99], and \textit{probabilistic inductive inference} (cf. [38]).

In the PAC-Learning model, the learner is fed examples of a \textit{concept}. Thereby, the examples are provided by a random source and distributed according to some unknown, but predetermined probability distribution. A class of concepts is PAC-identifiable if there is an algorithm which, in polynomial time, produces approximately correct hypotheses with a high probability. Thus, randomness appears three times - the examples are produced by a random source, the algorithm has to be successful with a high probability, and the hypotheses may differ from the target concept on a set of examples with a small probability according to the distribution. For more information on PAC-Learning see, for example, [21, 82, 99], and the references therein.

Probabilistic inductive inference of recursive functions was first studied by Freivalds (cf. [38]). Thereby, the underlying deterministic inference model was \textit{function EX-identification} (cf. [25, 48]). Probabilistic \textit{EX}-identification has attracted considerable attention during the last years. In particular, the research focused on \textit{finite probabilistic learning} and \textit{popperian finite probabilistic learning}, see for example Ambainis [2], Daley \textit{et al.} [29, 30, 31], and Wiehagen \textit{et al.} [106]. Probabilistic inductive inference of recursive functions was also intensively studied by Pitt [80], Pitt and Smith [81], Wiehagen \textit{et al.} [106, 107], and Kinber and Zeugmann [63]. Finally, Wiehagen \textit{et al.} [106, 107] and Pitt [80] proved some results on \textit{behaviorally correct identification} (ab-
Probabilistic language learning from text or informant was investigated by Pitt [80], and Jain and Sharma [51, 55]. Another approach to probabilistic language learning, namely language learning from stochastic examples, is due to Angluin [6], and was further investigated by Kapur and Bilardi [59], and Gavalda [47].

Some important results in the field of probabilistic learning concern function identification with bounded mind changes. Identification with bounded mind changes was introduced by Barzdin and Freivalds [11], and further investigated, for example, by Wiehagen, Freivalds and Kinber [106], Case and Smith [25], Lange and Zeugmann [67], and Mukouchi [76]. Informally, an inductive inference machine $M$ $EX$-identifies a function $f$ with mind changes $\leq n$ if $M$ $EX$-identifies $f$, and changes the hypothesis at most $n$ times.

Many interesting results in the field of probabilistic learning concern the relation between probabilistic $EX$-identification and team $EX$-identification (cf., e.g., [2, 29, 31, 51, 80]). The notion of team learning was introduced by Smith [92], and further investigated, for example, by Daley [27], and Jain and Sharma [54]. The introduction of team learning can be motivated by the observation that, in general, scientific discovery is the result of the effort of a scientific community. Informally, a team $M_1, \ldots, M_n$ is a collection of inductive inference machines. $M_1, \ldots, M_n$ $EX$-identifies a function $f$ if there exists at least one machine in the team which $EX$-identifies $f$. Jain and Sharma [51] transferred the notion of team learning to language identification. More information concerning team learning of recursive languages or team learning of indexed families can be found in [54, 71, 93, 98].

As already mentioned, we are mainly interested in probabilistic learning models which reflect the human ability to learn with high probabilities. Obviously, the PAC-learning model coincides with our intuition of probabilistic learning, since the learners are claimed to produce approximately correct hypotheses with a high probability. In the case of probabilistic inductive inference, the situation is different. Pitt [80] investigated the relation between probabilistic learning and team inference of recursive functions, and showed that identification by a single probabilistic machine is equivalent to team learning. In particular, each collection of recursive functions, which $EX$-identifiable with a probability $p > 1/2$, is deterministically $EX$-identifiable (cf. [80, 106]). A further result of Pitt (cf. [80]) concerns behaviorally correct identification. Pitt proved an equivalence between probabilistic $BC$-
identification and the corresponding notion of team $BC$-identification. Thus, each collection of recursive functions, which is $BC$-identifiable with a probability $p > 1/2$, is already deterministically $BC$-identifiable. For the case of finite probabilistic identification, Freivalds [38] showed that the corresponding probabilistic hierarchy has a threshold beginning with $2/3$. Daley and Kalyanasundaram [28] showed an analogous result for posperian finite probabilistic learning.

Considering language learning from text, each collection of recursive languages, which is identifiable from text with probability $p > 2/3$, is deterministically identifiable (cf. [80]). Consequently, in the case of language learning as well, we lose too much certainty in order to gain learning power, although probabilistic identification was shown to be strictly more powerful than team identification (cf. [51]).

The following results show that randomization not only enhances the learning power of the machines, but also has an effect on the efficiency of learning. Barzdin and Freivalds [11], and Podniek [84] showed that randomization is useful in order to reduce the number of mind changes required by an inductive inference machine. In the learning model investigated in their work, a probabilistic machine is allowed to flip a coin in order to reduce the number of hypothesis changes during the learning process on the average. Thereby, the probabilistic machine is claimed to identify a set of recursive functions with probability $p = 1$. It turns out that the final hypothesis has to be a member of a particular Gödel numbering of the partial recursive functions. In [106], Wiehagen, Freivalds and Kinber proved that, for all $n \geq 2$, there is a collection of recursive functions $\mathcal{F}$ which can be $EX$-identified with arbitrary high probability making at most $n$ mind changes. However, every deterministic $EX$-learner which $EX$-identifies $\mathcal{F}$ strictly exceeds the bound of $n$ mind changes. Thus, when restricting the number of mind changes, probabilistic learners are more powerful than deterministic ones even if the probability is claimed to be close to one. However, their result does not yield a probabilistic hierarchy that strictly decreases with increasing probability, i.e., we obtain no information about the relations between the probabilistic learning classes. Moreover, Wiehagen et al. [107] constructed a nonstandard hypothesis space $\mathcal{G}$ such that each infinite set of recursive functions, which is $EX$-identifiable with respect to some acceptable Gödel numbering, is $EX$-identifiable with arbitrary high probability with respect to $\mathcal{G}$, but not deterministically $EX$-identifiable with respect to $\mathcal{G}$. For $BC$-identification, they showed that there exists a nonstandard hypothesis space $\mathcal{G}'$ such that each infinite set of re-
cursive functions, which is $BC$-identifiable with respect to some acceptable Gödel numbering, is $BC$-identifiable with probability 1 with respect to $\mathcal{G}'$, but not deterministically $BC$-identifiable with respect to $\mathcal{G}'$. However, these hypothesis spaces do not induce natural deterministic learning classes, since no infinite set of recursive functions is $EX$-identifiable with respect to $\mathcal{G}$, and $BC$-identifiable with respect to $\mathcal{G}'$, respectively. Thus, the difference of learning power between probabilistic and deterministic learning is too large.

These results could tempt us to conclude that, in general, probabilistic inductive inference does not reflect the human ability to learn with probability less than or equal to one. However, the picture changes when investigating proper and class preserving probabilistic learning of indexed families under monotonicity constraints (cf. Definition 1.2.8). *Probabilistic learning under monotonicity constraints is more powerful than deterministic learning under monotonicity constraints even if the probability is close to 1 provided the learning machines are restricted to proper or class preserving hypothesis spaces. Moreover, the learning power strictly decreases with increasing probability* (cf. Section 1.3).

### 1.1.5 Complexity aspects of Probabilistic Inductive Inference

In the second part of our work, we investigate how powerful the probabilistic learning machines are in contrast to their deterministic counterparts. We address the question whether it is possible to measure the additional power, the probabilistic machines have, in quantitative or qualitative terms. Furthermore, we introduce a complexity measure in order to express the complexity of the learning problems which are identifiable with a probability close to 1, but not deterministically identifiable.

Before we describe our approach, let us give a short review on previous approaches and results. In the setting of algorithmic learning theory, various complexity measures were introduced and investigated. In the PAC-Learning-Scenario, the complexity of a learning algorithm can be measured in terms of *sample size*, i.e., the number of examples which are necessary in order to learn the problem with arbitrary high probability. Blumer *et al.* [21] showed that a combinatorial parameter of a concept class, the so-called Vapnik-Chervonenkis dimension (cf. [100]), relates to a bound on the sample size needed for identifying the class within the PAC-learning model.
In inductive inference, there is a wide range of work addressing questions of complexity. Blum and Blum [20], Wiehagen [104], and Zeugmann [110] characterized identifiable classes of functions as operator complexity classes. In Angluin [3] and Gold [49], it was shown that certain inference problems are members of well-known complexity classes such as $\mathcal{P}$ or $\mathcal{NP}$. Barzdin and Freivalds [11], and Freivalds [37] use the number of distinct hypotheses made by an inference strategy as a measure for the complexity of the inference process. Another well-studied complexity measure is the number of mind changes performed by an inductive inference machine during the learning process (cf., e.g., [25]). As already mentioned, Wiehagen et al. [106] showed that randomness can be used in order to reduce the number of hypothesis changes during the learning process. Hence, when allowing the machines to fail with a small probability, it is possible to identify a learning problem with less mind changes than in the deterministic case. For further information on complexity within this learning model, see [90], and the references therein.

Daley and Smith [32] developed an axiomatization of the complexity of inductive inference which parallels the approach to machine independent complexity of recursive functions given by Blum [18]. In their work, the amount of computation resources needed up to the point of convergence of an inductive inference machine is used as a complexity measure. Further axiomatic approaches to the complexity of inductive inference have been investigated by Schäfer-Richter (cf. [89], and the references therein).

In analogy to classical complexity theory (cf., e.g., [50]), where the resources time and space play an important role, several authors introduced and investigated formal models of speed and memory limited learning (cf., e.g., [12, 40, 89, 102]). Wiehagen [102] introduced a formalization of memory limited learning called iterative inductive learning. In this approach, a learner is not allowed to store all the data seen so far, but only its last hypothesis and a set of examples, where the maximal number of examples is bounded by a natural number $n$ which is given a priori. For more information about iterative inductive learning and its variants, see, for example, [23, 76], and the references therein.

The intrinsic complexity of learning problems was investigated by Jain and Sharma [53], and Freivalds, Smith and Kinber [39]. They introduced a notion of reducibility which relates to the inherent structure of the problems to be learned, but not to the algorithms considered. Other notions of reducibility were studied in [45, 44, 46, 83].

Recently, the field of average-case analysis of learning algorithms at-
tracted a lot of attention. Zeugmann [112] investigated Lange and Wiehagen’s pattern language learning algorithm, Reischuk and Zeugmann were concerned with learning pattern languages in linear average time, and the average-case analysis of learning monomials (cf. [86, 87]).

1.1.6 Oracle Complexity

In our work, we deal with an information oriented approach to complexity which was developed within the setting of learning with additional information. In this learning model, the machines are allowed to ask questions to an external information source, for example a teacher or an oracle. The information the learner has access to may consist of information about the problem to be learned or information which is not linked to the learning problem, e.g., context information or known learning strategies for other learning problems. Given the kind of information source, the learner has access to, and the kind of questions, the learner is allowed to ask, there are essentially two possibilities for measuring the complexity of a given learning problem. First, we may ask how many questions at least a learner must ask in order to identify the target languages. In this case, the complexity of the learning problem is measured in quantitative terms. Secondly, the information content of the weakest information source being sufficient for learning the target languages may be used as a complexity measure.

In the following, we give a review on previous work done in the field of learning with additional information and problems of complexity studied therein.

Lots of work was done in the fields learning via queries (cf., e.g., [5, 33, 44, 45, 46, 62]) and oracle identification (cf., e.g., [35, 60, 61, 94, 96]). Beside these two major branches, there are other approaches to learning with additional information (cf., e.g., [41]).

In the learning model developed by Angluin [5], the collection of objects to be learned is the set of all concepts \( c: \mathcal{X} \rightarrow \{0, 1\} \) which are describable by a monomial over \( \mathcal{X} \). The learner is allowed to ask questions to an oracle which provides information about the target languages. There are various kinds of queries. The simplest kind of query is the so-called membership query, i.e., the query “does \( x \) belong to the target language?”. At a first glance, this learning model is similar to learning by informant, i.e., learning by positive and negative examples. However, the learning machine decides which query it makes whereas in the case of learning by informant, the examples are
Another kind of query is the **equivalence query** which is defined as follows. The learner asks “is the concept $c$ describing the target language?”, and receives an answer “yes” or “no”. If the answer is “no”, then the learner receives a counterexample. In [7], Angluin also permits **subset queries** and **superset queries**, i.e., the learner may ask whether or not a language is a subset or a superset of the target language. For more information about this learning model, see, for example, [6, 7, 8]. A recursion theoretic version of Angluin’s learning model was investigated by Gasarch and Smith [46]. Thereby, they permitted questions with symbols from the alphabet $S := \{+, *, \text{succ}, <\}$, where $\text{succ}$ is a symbol for the successor function. A question is modeled as a sentence in first order logic over $S$, e.g., $\forall x (x > 34 \rightarrow f(x) = f(x) + 17)$, where $f$ is the function to be learned. Similar problems were addressed by Gasarch, Plezkoch and Solovay (cf. [45]). In their work, the set of symbols was restricted to $\{+, <\}$. For questions of complexity within this learning model, we refer the reader to [8], or Gavalda [47], who characterized the power of different learning protocols in terms of complexity classes of oracle machines.

The notion **learning with an oracle** or **oracle identification** is used for a learning model, where the learner has access to information which is not linked to the problems to be learned. More exactly, in the setting of oracle identification, the learning machines are allowed to ask questions of the form “$x \in A$” to an oracle $A \subseteq \mathbb{N}$ (cf. Definition 1.2.5, 1.2.6). For example, the learner may ask questions to the “halting problem” $\mathcal{K}$ or to any recursively enumerable oracle. Oracle identification has been intensively studied by Fortnow et. al. [35], Kummer and Stephan [61], Kinber [60], Slaman and Solovay [94] and Stephan [96]. In Fortnow et al. (cf. [35]), the authors are concerned with the question, how the information content of an oracle, i.e., its Turing degree, relates with the learning power of the oracle machines in dependence on the underlying inference criterion. It was investigated which inference degrees are trivial, i.e., which oracles do not enhance the learning power, and which inference degrees are omniscient, i.e., which oracles allow the identification of the set of the recursive functions. The authors addressed these questions in various learning models, e.g., $EX$- and $BC$-identification of functions. Similar questions have been studied in [61]. For oracle identification of languages, it has been shown that every nonrecursive oracle strictly increases the learning power with respect to informant, but not necessarily with respect to text. In order to measure the complexity of learning problems
within this learning model, several authors use the number of queries needed to identify a function or a language (cf., e.g., Beigel et al. [13, 14, 15, 16], Gasarch [43], and Kummer [64]). Brandt [22] studied qualitative aspects of complexity in inductive inference. In particular, she proved that the problem of identifying the set of partial recursive functions has the same degree of unsolvability as \( \mathcal{K} \).

In our work, we adopt a qualitative approach to complexity. The learning machines have access to additional information provided by oracles. Hence, the information is not linked to the problems to be learned.\(^2\) We investigate how much information content an information source must have such that the oracle machines, having access to this source, achieve at least the learning power of their probabilistic counterparts. We introduce and study a complexity measure on the class of indexed families \( \mathcal{L} \) for which there are an oracle machine \( M[] \), and an oracle \( A \) such that \( M[A] \) identifies \( \mathcal{L} \) with respect to a monotonicity constraint \( \mu \). This complexity measure is called oracle-complexity.\(^3\) For an indexed family \( \mathcal{L} \), the oracle-complexity of \( \mathcal{L} \) with respect to a monotonicity constraint \( \mu \) is defined to be the set of all oracles \( B \) such that an oracle machine \( M[B] \) exists which identifies \( \mathcal{L} \) and fulfills the constraint \( \mu \). If this set of oracles is of the form \( \{ B \mid A \leq_T B \} \) for an oracle \( A \), then the complexity of \( \mathcal{L} \) can be identified with the Turing degree of \( A \). Hence, the oracle-complexity of an indexed family within a learning model is defined to be the information content of the weakest information source which is sufficient for learning the indexed family within this model.

1.2 Preliminaries

1.2.1 Standard definitions

We denote the natural numbers by \( \mathbb{N} = \{0, 1, 2, \ldots \} \). The set of positive natural numbers is denoted by \( \mathbb{N}^+ \). Let \( M_0, M_1, \ldots \) be a standard list of all Turing machines, and let \( M_1[], M_2[] \ldots \) be the standard list of all oracle Turing machines. Let \( \varphi_0, \varphi_1, \ldots \) be the acceptable programming system, obtained by defining \( \varphi_i \) to be the partial recursive function computed by \( M_i \).

\(^2\)However, most of our examples are indexed families which encode an oracle. In these cases, there is no significant difference between the two approaches.

\(^3\)Notice that we use the term "complexity measure" in an intuitive way, i.e., we do not give an axiomatic approach.
Let $\Phi_0, \Phi_1, \ldots$ be any associated complexity measure (cf. [18]). Without loss of generality, we may assume that $\Phi_k(x) \in \mathbb{N}^+$ for all $k, x \in \mathbb{N}$. Furthermore, let $k, x \in \mathbb{N}$. If $\varphi_k(x)$ is defined, we say that $\varphi_k(x)$ converges and write $\varphi_k(x) \downarrow$. Otherwise $\varphi_k(x)$ diverges and we write $\varphi_k(x) \uparrow$. We denote the set of all recursive sets by $\mathcal{REQ}$, and the set of all recursively enumerable sets by $\mathcal{RE}$. By $\mathcal{K}$, we denote the set $\{ k \mid \varphi_k(k) \downarrow \}$. The set $\{ k \mid \varphi_k \text{ is total} \}$ is denoted by $\mathcal{TOT}$.

Let $A, B \subseteq \mathbb{N}$. For the complement of $A$ in $\mathbb{N}$, we write $\overline{A}$. By $A'$, we denote the halting problem relative to $A$. Let $A, B \subseteq \mathbb{N}$. $A$ is Turing reducible to $B$ ($A \leq_T B$) if the characteristic function $\chi_A$ of $A$ can be computed by an oracle Turing machine $M[\ ]$ which has access to $B$. More exactly, $M[\ ]$ has access to an infinite database which returns, for each $x \in \mathbb{N}$, whether or not $x \in B$. Such a database is called an oracle. The sets $A$ and $B$ are of the same degree of unsolvability ($A \equiv_T B$) if $A \leq_T B$ and $B \leq_T A$. The class $\{ A \mid A \equiv_T B \}$ is called the Turing degree of $B$. We denote the Turing degree of $B$ by $\text{deg}_T(B)$.

We mainly deal with oracles $A$ with $A \leq_T \mathcal{K}$, and the oracle $\mathcal{TOT}$. Moreover, we consider Peano-complete oracles. An oracle $A$ is said to be Peano-complete, if every pair of disjoint, recursively enumerable sets can be separated by a $A$-recursive function. Peano-complete oracles were introduced in the context of an extended theory of recursively enumerable sets (cf. [77]), where pairs of recursively enumerable sets instead of single recursively enumerable sets are investigated. A Peano-complete set is as powerful with respect to pairs of recursively enumerable oracles as $\mathcal{K}$ with respect to the recursively enumerable sets. However, there are Peano-complete oracles $A$ which are low, i.e., whose halting problem $A'$ is Turing equivalent to $\mathcal{K}$ (cf. [95]).

Next, we define the modulus of convergence for a sequence of recursive sets. Let $A$ be a nonrecursive oracle with $A \leq_T \mathcal{K}$. By the Limit Lemma [95], there exists a sequence of recursive sets $(A_i)_{i \in \mathbb{N}}$ such that $\lim_{i \to \infty} \chi_{A_i}(x) = \chi_A(x)$. Following Soare [95], we define a modulus of convergence for $(\chi_{A_i})_{i \in \mathbb{N}}$ to be a function $m^A$ with to following property:

$$\forall x \in \mathbb{N} \ (s \geq m^A(x) \rightarrow \chi_{A_s}(x) = \chi_A(x))$$

The least modulus of convergence for $(\chi_{A_i})_{i \in \mathbb{N}}$ is defined as follows:

$$m^A_{\text{min}}(x) = \mu s(\forall t \geq s \chi_{A_t}(x) = \chi_A(x)).$$
Thus, the modulus of convergence marks the converging point of the sequence \((\chi_{A_i}(x))_{i \in \mathbb{N}}\) for all \(x \in \mathbb{N}\).

As figured out in [95], an oracle \(A\) is recursively enumerable if and only if \(\chi_A\) is the limit of a recursive sequence \((\chi_{A_i})_{i \in \mathbb{N}}\) which has an \(A\)-recursive modulus of convergence. Thus, if \(A <_T K\), \(A\) not recursively enumerable, then the modulus of convergence of the sequence \((\chi_{A_i})_{i \in \mathbb{N}}\) is not \(A\)-recursive.

For more details about sets and Turing reducibility, we refer the reader to Odifreddi or Soare (cf. [77, 95]).

In the sequel, we assume familiarity with formal language theory (cf. [50]). Let \(\Sigma\) be any fixed finite alphabet of symbols containing the set of symbols \(\{a, b\}\), and let \(\Sigma^*\) be the free monoid over \(\Sigma\). Let \(\sigma = w_0, w_1, \ldots\) be a finite or infinite sequence of strings from \(\Sigma^*\). Define \(\text{range}(\sigma) := \{w_k \mid k \in \mathbb{N}\}\). Any recursively enumerable subset \(L \subseteq \Sigma^*\) is called a language. Let \(L\) be a language. An infinite sequence \(\tau = w_0, w_1, \ldots\) of strings from \(\Sigma^*\) with \(\text{range}(\tau) = L\) is called a text for \(L\). By \(\text{text}(L)\), we denote the set of all texts for \(L\). Let \(\tau\) be a text, let \(x \in \mathbb{N}\), and let \(\sigma\) be a finite sequence of strings from \(\Sigma^*\). By \(\tau_x\), we denote the initial segment of \(\tau\) of length \(x + 1\). Furthermore, we write \(\sigma \subseteq \tau\) if there is an \(n \in \mathbb{N}\) with \(\sigma = \tau_n\).

Next we define the notion of the canonical text for a nonempty recursive language \(L\) (cf. [65]). Let \(w_0, w_1, \ldots\) be the lexicographically ordered text of \(\Sigma^*\). Test whether \(w_z \in L\) for \(z = 0, 1, 2, \ldots\) until the first \(z \in \mathbb{N}\) is found such that \(w_z \in L\). Since \(L \neq \emptyset\), there must be at least one such \(z \in \mathbb{N}\). Set \(\tau_0^L = s_z\). For all \(x \in \mathbb{N}\) define:

\[
\tau_{x+1}^L = \begin{cases} 
\tau_x^L \cdot s_{z+x+1}, & \text{if } s_{z+x+1} \in L, \\
\tau_x^L \cdot s, & \text{otherwise, where } s \text{ is the last string in } \tau_x^L.
\end{cases}
\]

### 1.2.2 Definitions and basic results from Inductive Inference

In the sequel, we exclusively deal with the learnability of indexed families of uniformly recursive languages defined as follows (cf. [4]). A sequence \(L = (L_j)_{j \in \mathbb{N}}\) is said to be an indexed family of uniformly recursive languages provided \(L_j \neq \emptyset\) for all \(j \in \mathbb{N}\), and there is a recursive function \(F\) such that for all \(j \in \mathbb{N}\) and all \(w \in \Sigma^*\):

\[
F(j, w) := \begin{cases} 
1, & \text{if } w \in L_j, \\
0, & \text{otherwise.}
\end{cases}
\]
In the following, we refer to indexed families of uniformly recursive languages as *indexed families* for short. Let $\mathcal{L}$ be an indexed family. Remark that we use the term indexed family not only to denote the sequence of languages $\mathcal{L} = (L_j)_{j \in \mathbb{N}}$, but also to denote the recursive function enumerating $\mathcal{L}$. By $\text{range}(\mathcal{L})$, we denote $\{L_j \mid j \in \mathbb{N}\}$.

In many of our proofs, we need a special set of recursive languages which encodes the halting problem defined as follows (cf. [68]). Let $k$ be a natural number. Set $L_k := \{a^k b^m \mid m \in \mathbb{N}\}$. Moreover, set $L'_k := \begin{cases} L_k, & \text{if } \varphi_k(k) \uparrow; \\ \{a^k b^m \mid m \leq \Phi_k(k)\}, & \text{if } \varphi_k(k) \downarrow. \end{cases}$

Notice that $L'_k$ is recursive for all $k \in \mathbb{N}$, since $a^k b^m \in L'_k$ if and only if $\Phi_k(k) \geq n$. It is easy to see that $(L'_k)_{k \in \mathbb{N}}$ is an indexed family. We denote the canonical text for $L_k$ by $\tau_k$. Without loss of generality, we may assume that $\tau_k x = (a^k b^m)_{m \leq x}$.

Next, we define for every recursively enumerable set $A \notin \mathcal{REC}$ an encoding indexed family $(L^A_k)_{k \in \mathbb{N}}$. Let $A$ be recursively enumerable, $A$ not recursive. Let $E_A$ be an algorithm enumerating $A$ without repetitions. For $n \in \mathbb{N}$, $E_A(n)$ is the $n$-th element in the list enumerated by $E_A$. For convenience, we assume that $E_A(0)$ is not defined. Let $k \in A$ be a natural number. We use $E^{-1}_A(k)$ to denote the natural number $n$ with $E_A(n) = k$. If $k \notin A$ then $E^{-1}_A(k)$ is not defined. By convention, $E^{-1}_A(k) \geq 1$ for all $k \in \mathbb{N}$. Now define for $k \in \mathbb{N}$:

$$L^A_k := \begin{cases} L_k, & \text{if } k \notin A; \\ \{a^k b^m \mid m \leq E^{-1}_A(k)\}, & \text{if } k \in A. \end{cases}$$

Note that the predicate “$m \leq E^{-1}_A(k)$” is uniformly decidable for all $k, m \in \mathbb{N}$, since $m \leq E^{-1}_A(k)$ if and only if $k \notin \{E_A(1), \ldots, E_A(m)\}$ or $E_A(m) = k$. Thus, $(L^A_k)_{k \in \mathbb{N}}$ is an indexed family.

As in Gold (cf. [48]), we define an *inductive inference machine* (abbr. IIM) to be an algorithmic device working as follows. An IIM $M$ takes as its input larger and larger initial segments of a text $\tau$ and it either takes the next input string, or it first outputs a hypothesis, i.e., a number encoding a certain computer program, and then requests the next input string.\footnote{For a recursion theoretic approach to language learning in the limit, we refer the}
all admissible hypotheses is called hypothesis space. As mentioned in the
introduction, we do not allow every set of hypotheses as a hypothesis space
but only enumerable families of grammars \(G_0, G_1, G_2, \ldots\) over the terminal
alphabet \(\Sigma\) such that \(\text{range}(\mathcal{L}) \subseteq \{L(G_j) \mid j \in \mathbb{N}\}\), and membership in
\(L(G_j)\) is uniformly decidable for all \(j \in \mathbb{N}\) and all strings \(w \in \Sigma^*\). If an
IIM \(M\) outputs a number \(j\), then we are interpreting this number to be the
index of the grammar \(G_j\), i.e., \(M\) guesses the language \(L(G_j)\). Let \(\sigma\) be a
finite sequence of strings from \(\Sigma^*\), and let \(j \in \mathbb{N}\) be a hypothesis. Then \(j\) is
said to be consistent with \(\sigma\) iff \(\text{range}(\sigma) \subseteq L(G_j)\). For a hypothesis space
\(\mathcal{G} = (L(G_j))_{j \in \mathbb{N}}\), we use \(\text{range}(\mathcal{G})\) to denote \(\{L(G_j) \mid j \in \mathbb{N}\}\).

Let \(\tau\) be a text for a recursive language \(L\) and let \(x \in \mathbb{N}\). By \(M(\tau_x)\), we denote
the last hypothesis \(M\) outputs when fed \(\tau_x\). If there is no such hypothesis,
then \(M(\tau_x)\) is said to be \(\bot\). \(M(\tau_x)\) is said to be refutable for \(L\) iff there exists
an \(w \in \Sigma^*\) with \(w \in L \setminus L(G_{M(\tau_x)})\), and \(M(\tau_x)\) is said to be correct for \(L\) iff
\(L = L(G_{M(\tau_x)})\). If \(L \subset L(G_{M(\tau_x)})\), we call \(L(G_{M(\tau_x)})\) overgeneralization of \(L\).

The sequence \((M(\tau_x))_{x \in \mathbb{N}}\) is said to converge to the number \(j\) iff either
there exists some \(n \in \mathbb{N}\) with \(M(\tau_x) = j\) for all \(x \geq n\), or \((M(\tau_x))_{x \in \mathbb{N}}\) is
finite and its last member is \(j\). Let \(\mathcal{G} = (G_j)_{j \in \mathbb{N}}\) be a hypothesis space. \(M\)
is said to converge correctly on \(\tau\) with respect to \(\mathcal{G}\) iff \((M(\tau_x))_{x \in \mathbb{N}}\) converges
to \(j\) with \(L(G_j) = L\). Now we define learning in the limit (cf. [48]).

**Definition 1.2.1.** (Gold [48]). Let \(\mathcal{L}\) be an indexed family, let \(L \in \text{range}(\mathcal{L})\),
and let \(\mathcal{G} = (G_j)_{j \in \mathbb{N}}\) be a hypothesis space. An IIM \(M\) CLIM-identifies \(L\)
from text with respect to \(\mathcal{G}\) iff \(M\) converges correctly with respect to \(\mathcal{G}\) on
every text \(\tau\) for \(L\).

\(M\) CLIM-identifies \(\mathcal{L}\) with respect to \(\mathcal{G}\) iff, for each \(L \in \text{range}(\mathcal{L})\), \(M\) CLIM-
identifies \(L\) from text with respect to \(\mathcal{G}\).

Let CLIM denote the collection of all indexed families for which there are an
IIM \(M\) and a hypothesis space \(\mathcal{G}\) such that \(M\) CLIM-identifies \(\mathcal{L}\) with respect
to \(\mathcal{G}\).

The prefix \(C\) in CLIM is used to denote class comprising learning, i.e., \(\mathcal{L}\)
reader to [79]. Thereby, every partial recursive function \(f: \mathbb{N} \to \mathbb{N}\) is considered as a
learning machine. A learning machine \(f\) takes as its input natural numbers which are code
numbers of finite sequences of strings, i.e., a natural number \(n\) encodes a finite sequence
\(\sigma = s_0, \ldots, s_m, s_i \in \Sigma^*\). More exactly, there is an effective encoding \(\text{CODE}\) which maps
the set of all finite sequences of strings from \(\Sigma^*\) to \(\mathbb{N}\). In this approach, \(M(\sigma)\) stands for
\(M(\text{CODE}(\sigma))\).
can be learned with respect to some hypothesis space $\mathcal{G}$ with $\text{range}(\mathcal{L}) \subseteq \text{range}(\mathcal{G})$. By $\text{LIM}$, we denote the collection of all indexed families $\mathcal{L}$ that can be learned in the limit with respect to a class preserving hypothesis space $\mathcal{G}$, i.e., $\text{range}(\mathcal{L}) = \text{range}(\mathcal{G})$. The empty prefix for $\text{LIM}$ is denoted by $\varepsilon$. If an indexed family $\mathcal{L}$ has to be inferred with respect to $\mathcal{L}$ itself, then we replace the prefix $C$ by $E$, i.e., $\text{ELIM}$ is the collection of indexed families that can be learned properly in the limit. We adopt this distinction for all the learning types defined below.

Next we define conservative (cf. [4]), weak-monotonic, monotonic, and strong-monotonic inference (cf. [56], [105]).

**Definition 1.2.2.** (Angluin [4], Jantke [56], Wiehagen [105]). Let $\mathcal{L}$ be an indexed family, let $L \in \text{range}(\mathcal{L})$, and let $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$ be a hypothesis space. An IIM $M$ is said to identify $L$ from text with respect to $\mathcal{G}$

(A) conservatively

(B) strong-monotonically

(C) monotonically

(D) weak-monotonically

iff $M$ $\text{CLIM}$-identifies $L$ from text with respect to $\mathcal{G}$, and, for every text $\tau$ for $L$ as well as for all $x,k \in \mathbb{N}$, $k \in \mathbb{N}^+$, with $M(\tau_x) \neq \bot$, the corresponding condition is satisfied:

(A) if $M(\tau_x) \neq M(\tau_{x+k})$, then $\text{range}(\tau_{x+k}) \not\subseteq \text{L}(G_M(\tau_x))$

(B) $\text{L}(G_M(\tau_x)) \subseteq \text{L}(G_M(\tau_{x+k}))$

(C) $\text{L}(G_M(\tau_x)) \cap \text{L} \subseteq \text{L}(G_M(\tau_{x+k})) \cap \text{L}$

(D) if $\text{range}(\tau_{x+k}) \subseteq \text{L}(G_M(\tau_x))$, then $\text{L}(G_M(\tau_x)) \subseteq \text{L}(G_M(\tau_{x+k}))$

$M$ identifies $\mathcal{L}$ with respect to $\mathcal{G}$ conservatively, strong-monotonically, monotonically, and weak-monotonically, respectively, iff. for each $L \in \text{range}(\mathcal{L})$, $M$ identifies $L$ from text with respect to $\mathcal{G}$ conservatively, strong-monotonically, monotonically, and weak-monotonically, respectively.

By $\text{CCOV}$, $\text{CSMON}$, $\text{CMON}$, and $\text{CWMON}$, we denote the collection of all indexed families $\mathcal{L}$ for which there are an IIM $M$ and a hypothesis space $\mathcal{G}$
such that $M$ identifies $L$ with respect to $G$ conservatively, strong-monotonically, monotonically and weak-monotonically, respectively.

In the following, we define some often used abbreviations. Let $L$ be an indexed family, let $L \in \text{range}(L)$, and let $\tau \in \text{text}(L)$. Let $G$ be a hypothesis space, and let $\mu \in \{COV, SMON, MON, WMON\}$ be a monotonicity constraint.

$M$ is said to $C\mu$-converge correctly on $\tau$ with respect to $G$ iff $M$ converges correctly on $\tau$ with respect to $G$, and $(M(\tau_x))_{x \in \mathbb{N}}$ satisfies the condition $\mu$. Furthermore, if $M$ identifies $L$ from text with respect to $G$ conservatively, strong-monotonically, monotonically and weak-monotonically, respectively, then $M$ is said to $C\mu$-identify $L$ from text with respect to $G$. Analogously, we define the corresponding abbreviations for proper and class preserving probabilistic learning.

Next we define team learning of indexed families (cf. [51]). Team learning of recursive functions was first investigated by Smith [92]. For an overview in this field, we refer the reader to Smith [93]. Recall that a team of IIMs is defined to be a multiset of IIMs.

**Definition 1.2.3.** (Jain and Sharma [51]). Let $L$ be an indexed family, let $L \in \text{range}(L)$, let $G = (G_j)_{j \in \mathbb{N}}$ be a hypothesis space, and let $n \in \mathbb{N}$. A team of IIMs $M_1, \ldots, M_n$ $\text{CLIM}_{\text{team}}(n)$-identifies $L$ from text with respect to $G$ iff, for every $\tau \in \text{text}(L)$, there exists a $j \in \{1, \ldots, n\}$ such that $M_j$ converges correctly on $\tau$ with respect to $G$.

$M_1, \ldots, M_n$ $\text{CLIM}_{\text{team}}(n)$-identifies $L$ iff $M_1, \ldots, M_n$ $\text{CLIM}_{\text{team}}(n)$-identifies each $L \in \text{range}(L)$ from text with respect to $G$.

Let $\text{CLIM}_{\text{team}}(n)$ denote the collection of all indexed families $L$ for which there are a team $M_1, \ldots, M_n$ and a hypothesis space $G$ such that $M_1, \ldots, M_n$ $\text{CLIM}_{\text{team}}(n)$-identifies $L$ with respect to $G$.

In Definition 1.2.3, we require that for every text $\tau$ for a language $L$ there is a team member identifying this text, i.e., the identifying machine depends on the presentation of $L$. However, Jain and Sharma [51] showed that for each collection of recursive languages $L$ identifiable by a team of $n$ inductive inference machines, there is a team of $n$ machines identifying $L$ such that for every $L$ in $\text{range}(L)$ there is a member of the team that identifies every text for $L$.

Lange and Zeugmann [68] showed that the learning power in the LIM-case
does not depend on the hypothesis space the learner may use, i.e., \( CLIM = ELIM \). Similarly, we can prove the following theorem.

**Theorem 1.2.1.** \( CLIM_{\text{team}}(n) = ELIM_{\text{team}}(n) \) for all \( n \in \mathbb{N}^+ \).

There are two natural ways to define monotonic notions of team learning (cf. [55]). We can require that every IIM in a team fulfills the monotonicity requirement, or we can restrict this demand to the correctly converging team members.

**Definition 1.2.4.** (Meyer [71], Jain and Sharma [55]). Let \( \mathcal{L} \) be an indexed family, let \( L \in \text{range}(\mathcal{L}) \), and let \( \mathcal{G} = (G_j)_{j \in \mathbb{N}} \) be a hypothesis space. Let \( \mu \in \{\text{COV}, \text{SMON}, \text{MON}, \text{WMON}\} \) be a monotonicity constraint. Let \( n \in \mathbb{N}^+ \), and let \( M_1, \ldots, M_n \) be a team of IIMs.

(A) \( M_1, \ldots, M_n C_{\mu_{\text{team}}(n)} \)-identifies \( L \) from text with respect to \( \mathcal{G} \) iff, for every text \( \tau \) for \( L \), there exists a \( j \in \{1, \ldots, n\} \) such that \( M_j C_{\mu} \)-converges correctly on \( \tau \) with respect to \( \mathcal{G} \).

\[ M_1, \ldots, M_n C_{\mu_{\text{team}}(n)} \]-identifies \( L \) with respect to \( \mathcal{G} \) iff \( M_1, \ldots, M_n C_{\mu_{\text{team}}(n)} \)-identifies each \( L \in \text{range}(\mathcal{L}) \) from text with respect to \( \mathcal{G} \).

(B) \( M_1, \ldots, M_n C_{\mu_{\text{st}}_{\text{team}}(n)} \)-identifies \( L \) from text with respect to \( \mathcal{G} \) iff \( M_1, \ldots, M_n CLIM_{\text{team}}(n) \)-identifies \( L \) from text with respect to \( \tau \), and for every text \( \tau \) for \( L \) and for all \( i \in \{1, \ldots, n\} \), \( M_i \) satisfies the condition \( \mu \) on \( \tau \) with respect to \( \mathcal{G} \).

\[ M_1, \ldots, M_n C_{\mu_{\text{st}}_{\text{team}}(n)} \]-identifies \( L \) with respect to \( \mathcal{G} \) iff \( M_1, \ldots, M_n C_{\mu_{\text{st}}_{\text{team}}(n)} \)-identifies each \( L \in \text{range}(\mathcal{L}) \) from text with respect to \( \mathcal{G} \).

The collections \( C_{\mu_{\text{team}}(n)} \) and \( C_{\mu_{\text{st}}_{\text{team}}(n)} \) are defined as usual.

Lange and Zeugmann [65] showed that weak-monotonic learning and conservative learning of indexed families are equivalent. With the same argument, the following theorem can be proved.

**Theorem 1.2.2.** Let \( n \in \mathbb{N}^+ \), and let \( \lambda \in \{E, \varepsilon, C\} \). Then

(a) \( \lambda WMON_{\text{team}}(n) = \lambda COV_{\text{team}}(n) \),

(b) \( \lambda WMON_{\text{st}}_{\text{team}}(n) = \lambda COV_{\text{st}}_{\text{team}}(n) \).

Thus, for \( \lambda \in \{E, \varepsilon, C\} \), it suffices to deal with the notions \( \lambda COV_{\text{team}}(p) \) and \( \lambda COV_{\text{st}}_{\text{team}}(p) \), respectively.
1.2.3 Oracle Identification

Let $M[\ ]$ be an oracle Turing machine. If $M[\ ]$ has access to $A$, we denote the resulting machine by $M[A]$. As in [35], we define $M[\ ]$ to be categorically total if and only if $M[A]$ is total for every oracle $A$. An oracle inductive inference machine (abbr. OIM) is a categorically total oracle Turing machine. Hence, $M[A]$ can be interpreted as an IIM which has access to an infinite database $A$, i.e., $M[A]$ may ask questions of the form “$x \in A$?”. For more details about the definition of OIMs, we refer the reader to [35].

**Definition 1.2.5.** (Fortnow et al. [35]). Let $L$ be an indexed family, let $L \in \text{range}(L)$, and let $G = (G_j)_{j \in \mathbb{N}}$ be a hypothesis space. Let $A$ be a set of natural numbers. An OIM $M[A]$ CLIM-identifies $L$ from text with respect to $G$ iff $M[A]$ converges correctly with respect to $G$ on every text $\tau$ for $L$.

$M[A]$ CLIM-identifies $L$ with respect to $G$ iff, for each $L \in \text{range}(L)$, $M[A]$ CLIM-identifies $L$ from text with respect to $G$.

Let $\text{CLIM}[A]$ denote the collection of all indexed families for which there are an OIM $M[A]$ and a hypothesis space $G$ such that $M[A]$ CLIM-identifies $L$ with respect to $G$.

This definition can easily be extended to inductive learning under monotonicity constraints.

**Definition 1.2.6.** (Meyer [74], Stephan [96]).

Let $\mu \in \{\text{COV}, \text{SMON}, \text{MON}, \text{WMON}\}$. Let $L$ be an indexed family, let $L$ be a language, and let $G = (G_j)_{j \in \mathbb{N}}$ be a hypothesis space. Let $A$ be an oracle, and let $M[A]$ be an OIM. Then $M[A]$ $C_\mu$-identifies $L$ from text with respect to $G$ if, for every text $\tau$ for $L$, $M[A]$ $C_\mu$-converges correctly on $\tau$ with respect to $G$.

$M[A]$ $C_\mu$-identifies $L$ with respect to $G$ iff $M[A]$ $C_\mu$-identifies each $L \in \text{range}(L)$.

Let $\text{CLIM}[A]$ denote the collection of all indexed families for which there are an OIM $M[A]$ and a hypothesis space $G$ such that $M[A]$ $C_\mu$-identifies $L$ with respect to $G$.

As in the case of team learning, we can show that weak-monotonic and conservative learning in the setting of indexed families are equivalent.
Theorem 1.2.3. Let $\lambda \in \{E, \epsilon, C\}$. Let $A$ be an oracle. Then

$$\lambda \text{WMON}[A] = \lambda \text{COV}[A].$$

Thus, it suffices to deal with the notions $\lambda \text{COV}[A]$ for $\lambda \in \{E, \epsilon, C\}$.

1.2.4 Probabilistic Inductive Inference

The next definition concerns probabilistic inductive inference machines (abbr. PIM). A PIM is an IIM which has the possibility to flip a $t$-sided coin and to base the choice of the next hypothesis not only on the text but additionally on the outcome of the coin flip. Analogously, probabilistic oracle machines can be defined. Thereby, a probabilistic oracle inductive inference machine (abbr. POIM) is an OIM equipped with a $t$-sided coin. The hypotheses produced by an POIM, when fed a text $\tau$, depend on the text, on the answers to the queries, and on the outcome of the coin flips.

More formally, a PIM $P$ is an IIM equipped with a $t$-sided coin. A coin-oracle $C$ is an infinite sequence $c_0, c_1, \ldots$ where $c_i \in \{0, \ldots, t-1\}$. By $C^n$, we denote the initial segment $c_0, \ldots, c_n$ of $C$ for all $n \in \mathbb{N}$. Instead of $C^0$, we write $c_0$. Let $C$ be a coin-oracle. We denote the deterministic IIM defined by running $P$ with coin-oracle $C$ by $P^C$. Intuitively, $P^C$, fed a text $\tau$ for the target language, outputs the hypothesis $j$ on $\tau_x$ if $P$ outputs $j$ on $\tau_x$ under the condition that the first $x+1$ flips of the $t$-sided coin were $C^x$. Instead of $P^C(\tau_x)$, we write $P^{C^x}(\tau_x)$.

Let $P$ be a PIM. We assume $P$ to behave well, i.e., $P$ flips the coin each time it requests a new input string. In ongoing time, $P$ is fed more and more information about a language to be inferred and has to produce hypotheses about this language, i.e. a grammar for the language to be learned. In the case $d = 1$, the notion of a PIM is the same as the notion of an inductive inference machine. A PIM $P$ is said to identify a text $\tau$ for a language $L$ with probability $p$, if the probability taken over all infinite sequences $c_0, c_1, \ldots$ of coin flips such that the sequence of hypotheses guessed by $P$, when fed $\tau$ and $c_0, c_1, \ldots$, is converging to a hypothesis correctly describing $L$, is greater than or equal to $p$. If $P$ identifies every positive presentation for the target language with probability $p$, then it is said to identify the language from text with probability $p$. A PIM identifies a collection of languages from text with probability $p$ if and only if it identifies each member of this collection from text with probability $p$. 

27
For the sake of readability, we now define the notion of an infinite computation tree (cf. [80]). For a PIM $P$ equipped with a $t$-sided coin, and a text $\tau$ for a recursive language $L$, we define $T_{P,\tau}$ to be the $t$-ary tree representing all possible outputs of $P$ when fed $\tau$. Each node of $T_{P,\tau}$ can be identified with a member of the set $\bigcup_{n \in \mathbb{N}} \{0, \ldots, t - 1\}^n$ and corresponds to a hypothesis produced by $P$ when fed $\tau$.\footnote{Thereby, $\{0, \ldots, t - 1\}^n$ is the set of all finite sequences $(c_0, \ldots, c_{n-1})$ of length $n$ with $c_i \in \{0, \ldots, t - 1\}$ for $i \leq n - 1.$} Thus, the paths of $T_{P,\tau}$ correspond to the infinite sequences $(P^{c_i}(\tau_x))_{x \in \mathbb{N}}$.

Let $o$ be a node in $T_{P,\tau}$, and let $C^m \in \{0, \ldots, t - 1\}^{m+1}$ be the corresponding finite sequence for an $m \in \mathbb{N}$. The hypothesis $P^{C^m}(\tau_m)$ is denoted by $\text{ind}(o)$. Let $o'$ be any other node in $T_{P,\tau}$. $o'$ is said to be a successor of $o$, and $o$ is called a predecessor of $o'$ iff there exist a coin-oracle $C'$, and an $n \in \mathbb{N}$, $n > m$, such that $C^m = (C')^m$ and $(C')^n$ corresponds to $o'$.

Next we define the notion of a level in $T_{P,\tau}$. Let $x$ be a natural number. The level 0 is defined to be $\emptyset$. For $x \in \mathbb{N}^+$, we define the level $x$ of the tree $T_{P,\tau}$ to be the set of all nodes $o$ such that $o$ corresponds to a finite sequence $s \in \{0, \ldots, t - 1\}^x$. Furthermore, we define $T_{P,\tau_x}$ to be the finite subtree of $T_{P,\tau}$ consisting of all nodes $o \in$ level $y$, $y \leq x + 1$.

Finally, let $T$ be an arbitrary binary tree. We define an enumeration of the nodes of $T$. The root of $T$ has the number 1, the nodes on level 1 have the numbers 1 and 2, and so on. On each level, we count from the left to the right side. Thus, the nodes on level $x$ on $T$ are the nodes $o_{2x}, \ldots, o_{2x+1}$.

Let $C \in \{0, \ldots, t - 1\}^\infty$ be a coin-oracle. A path $(P^{c_i}(\tau_x))_{x \in \mathbb{N}}$ is said to pass through a node in $T_{P,\tau}$ if and only if there is an $m \in \mathbb{N}$ such that $o$ corresponds to $C^m$. $(P^{c_i}(\tau_x))_{x \in \mathbb{N}}$ is said to contain a $j \in \mathbb{N}$ if and only if $(P^{c_i}(\tau_x))_{x \in \mathbb{N}}$ passes through a node $o$ with $\text{ind}(o) = j$.

Let $P$ be a PIM equipped with a $t$-sided coin, let $L$ be a recursive language, let $\tau \in \text{text}(L)$, and let $O$ be a set of nodes in $T_{P,\tau}$. Then we define the weight of $O$ (abbr. $w(O)$) to be the probability of the set of all coin-oracles $C \in \{0, \ldots, t - 1\}^\infty$ such that $(P^{c_i}(\tau_x))_{x \in \mathbb{N}}$ passes through a node $o \in O$. It is easy to see that $w(O)$ is invariant against the adding of successors, i.e., if $o$ is a node, $o \notin O$, and there exists a node $o' \in O$ such that $o$ is a successor of $o'$, then $w(O) = w(O \cup \{o\})$.

Now let $Pr$ denote the canonical Borel-measure on the Borel-$\sigma$-algebra on $\{0, \ldots, t - 1\}^\infty$. For more details about probabilistic IMIs, measurability and infinite computation trees we refer the reader to Pitt (cf. [80]).
**Definition 1.2.7.** (Wiehagen et al. [106], Pitt [80]). Let \( \mathcal{L} \) be an indexed family, let \( L \in \text{range}(\mathcal{L}) \), and let \( \mathcal{G} = (G_j)_{j \in \mathbb{N}} \) be a hypothesis space. Let \( p \in [0, 1] \). A PIM \( \mathcal{P} \) \( \text{CLIM}_{\text{prob}}(p) \)-identifies \( L \) from text with respect to \( \mathcal{G} \) iff the following holds for every text \( \tau \) for \( L \):

\[
\Pr\{ \mathcal{C} \mid \mathcal{P}^C \text{ converges correctly on } \tau \text{ w.r.t. } \mathcal{G} \} \geq p.
\]

\( \mathcal{P} \) \( \text{CLIM}_{\text{prob}}(p) \)-identifies \( \mathcal{L} \) from text with respect to \( \mathcal{G} \) iff \( \mathcal{P} \) \( \text{CLIM}_{\text{prob}}(p) \)-identifies each \( L \in \text{range}(\mathcal{L}) \) from text with respect to \( \mathcal{G} \).

Let \( \mathcal{CLIM}_{\text{prob}}(p) \) denote the collection of all indexed families for which there are a PIM \( \mathcal{P} \) and a hypothesis space \( \mathcal{G} \) such that \( \mathcal{P} \) \( \text{CLIM}_{\text{prob}}(p) \)-identifies \( L \) with respect to \( \mathcal{G} \).

As in Definition 1.2.4, we now define monotonic probabilistic learning (cf. [55]). It turns out that the requirement that the probabilistic machine has to fulfill the monotonicity constraint on every path can restrict the learning power (cf. Corollary 2.2.7).

**Definition 1.2.8.** (Meyer [71], Jain and Sharma [55]). Let \( \mathcal{L} \) be an indexed family, let \( L \in \text{range}(\mathcal{L}) \), and let \( \mathcal{G} = (G_j)_{j \in \mathbb{N}} \) be a hypothesis space. Let \( \mu \in \{\text{COV, SMON, MON, WMON}\} \) be a monotonicity constraint. Let \( p \in [0, 1] \), and let \( \mathcal{P} \) be a PIM equipped with a \( t \)-sided coin.

(A) \( \mathcal{P} \) \( \text{C}_{\mu, \text{prob}}(p) \)-identifies \( L \) from text with respect to \( \mathcal{G} \) iff the following condition holds for every text \( \tau \) for \( L \):

\[
\Pr\{ \mathcal{C} \mid \mathcal{P}^C \mu - \text{converges correctly on } \tau \text{ w.r.t. } \mathcal{G} \} \geq p.
\]

\( \mathcal{P} \) \( \text{C}_{\mu, \text{prob}}(p) \)-identifies \( \mathcal{L} \) with respect to \( \mathcal{G} \) iff \( \mathcal{P} \) \( \text{C}_{\mu, \text{prob}}(p) \)-identifies each \( L \in \text{range}(\mathcal{L}) \) with respect to \( \mathcal{G} \).

(B) \( \mathcal{P} \) \( \text{C}_{\mu, \text{st}}(p) \)-identifies \( L \) from text with respect to \( \mathcal{G} \) if and only if \( \mathcal{P} \) \( \text{CLIM}_{\text{prob}}(p) \)-identifies \( L \) from text with respect to \( \mathcal{G} \), and, for every text \( \tau \) for \( L \) and every coin-oracle \( \mathcal{C} \), \( \mathcal{P}^C \mu \) satisfies the condition \( \mu \) on \( \tau \) with respect to \( \mathcal{G} \). \( \mathcal{P} \) \( \text{C}_{\mu, \text{st}}(p) \)-identifies \( \mathcal{L} \) with respect to \( \mathcal{G} \) iff \( \mathcal{P} \) \( \text{C}_{\mu, \text{st}}(p) \)-identifies each \( L \in \text{range}(\mathcal{L}) \) with respect to \( \mathcal{G} \).

The collections \( \text{C}_{\mu, \text{prob}}(p) \) and \( \text{C}_{\mu, \text{st}}(p) \) are defined as usual.
Let $P$ be a PIM equipped with a $t$-sided coin, let $L$ be a recursive language, and let $\tau \in \text{text}(L)$. Let $p \in [0, 1]$. Assume that $P \ C_{\mu_{\text{prob}}(p)}$-identifies $L$ from text with respect to a hypothesis space $G$. Let $C \in \{0, \ldots, t-1\}^\infty$ be a coin-oracle which does not contribute to the learning success of $P$ when fed $\tau$. Then the definition implies that $P$ diverges on that run or converges and injures the monotonicity constraint.

Pitt [80] showed that every PIM $P$ equipped with a $t$-sided coin can be simulated by a PIM $P'$ equipped with a two-sided coin such that every language identifiable from $P$ with a probability $\geq p$ is identifiable from $P'$ with a probability $\geq p$. The same result holds for probabilistic inference machines fulfilling monotonicity constraints. Unless otherwise specified, we assume that every PIM considered is equipped with a two-sided coin.

It is easy to see that Theorem 1.2.2 holds for conservative and weak-monotonic probabilistic learning too.

**Theorem 1.2.4.** Let $p \in [0, 1]$, and let $\lambda \in \{E, \varepsilon, C\}$. Then

(a) $\lambda W_{\text{prob}}(p) = \lambda COV_{\text{prob}}(p)$,

(b) $\lambda W_{\text{prob}}^{st}(p) = \lambda COV_{\text{prob}}^{st}(p)$.

Thus, it suffices to deal with the notions $\lambda COV_{\text{prob}}(p)$ and $\lambda COV_{\text{prob}}^{st}(p)$, respectively. Moreover, it is easy to see that strong-monotonic probabilistic learning is weaker than conservative and monotonic probabilistic learning.

**Theorem 1.2.5.** Let $p \in [0, 1]$, and let $\lambda \in \{E, \varepsilon, C\}$. Then

(a) $\lambda SM_{\text{prob}}(p) \subseteq \lambda COV_{\text{prob}}(p)$,

(b) $\lambda SM_{\text{prob}}(p) \subseteq \lambda MON_{\text{prob}}(p)$.

An analogous result holds for the probabilistic learning classes $\lambda \mu_{\text{prob}}^{st}(p)$, $\mu \in \{COV, SMON, MON\}$, $\lambda \in \{E, \varepsilon, C\}$.

Following Lange and Zeugmann [65], we can assume that every inductive inference machine considered in this paper is consistent. Thereby, an IIM $M$ is consistent on a text $\tau$ for a language $L$ with respect to a hypothesis space $G$, if $\text{range}(\tau_x) \subseteq L(G_M(\tau_x))$ for all $x \in \mathbb{N}$ with $M(\tau_x) \neq \bot$. For more information about consistency, we refer the reader to [108]. For probabilistic inductive inference machines, we can adapt this result when dealing with class comprising hypothesis spaces.
Theorem 1.2.6. Let $\mu \in \{\text{LIM}, \text{COV}, \text{SMON}, \text{MON}\}$, let $\mathcal{L}$ be an indexed family, and let $\mathcal{G}$ be a hypothesis space. Let $P$ be an PIM which $C_{\mu_{\text{prob}}}(p)$-identifies $\mathcal{L}$ with respect to $\mathcal{G}$. Then there exist a PIM $P'$ and a hypothesis space $\mathcal{G}'$ with the following properties.

(a) $P'$ $C_{\mu_{\text{prob}}}(p)$-identifies $\mathcal{L}$ with respect to $\mathcal{G}'$.

(b) Let $L \in \text{range}({\mathcal{L}})$, let $\tau \in \text{text}(L)$, let $\mathcal{C}$ be a coin-oracle, and let $x$ be a natural number. Then $\text{range}(\tau^x) \subseteq L(G'_{C^{x^2}(\tau^x)})$.

This result can be easily shown by adding the set $\mathcal{E}$ of all finite subsets of $\Sigma^*$ and the set of all unions of languages in $\mathcal{G}$ and $\mathcal{E}$ to the hypothesis space $\mathcal{G}$. In particular, we can construct $P'$ in a way such that it never outputs $\bot$.

The same result holds for every other notion of class comprising probabilistic learning and class comprising team learning defined in Definition 1.2.3 and Definition 1.2.4. Unless otherwise specified, we assume that every PIM, which may use a class comprising hypothesis space, is consistent.

1.2.5 The key definitions

In this last subsection of this part, we note the definitions which are central for our work. First, we define the term probabilistic hierarchy. Let $\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$. Since $\lambda_{\mu_{\text{prob}}}(p)$ is a subset of $\lambda_{\mu_{\text{prob}}}(q)$ for all $p, q \in [0, 1]$, $p \leq q$, the collection $\{\lambda_{\mu_{\text{prob}}}(p) \mid p \in [0, 1]\}$ can be considered as a hierarchy of probabilistic learning classes. We call this collection probabilistic hierarchy for $\lambda \mu_{\text{prob}}$, and write $\langle \lambda_{\mu_{\text{prob}}}(p) \rangle_{p \in [0,1]}$. As the following definition shows, a probabilistic hierarchy can be structured in many different ways.

Definition 1.2.9.

Let $\mu \in \{\text{LIM}, \text{COV}, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$

(A) Let $(p_n)_{n \in \mathbb{N}}$ be a sequence of real numbers with $p_n \in [0, 1]$, $p_{n+1} > p_n$ for all $n \in \mathbb{N}$, and $\lim_{n \to \infty} p_n = 1$. $\langle \lambda_{\mu_{\text{prob}}}(p) \rangle_{p \in [0,1]}$ is said to be strictly decreasing at points $(p_n)_{n \in \mathbb{N}}$ iff $\lambda_{\mu_{\text{prob}}}(p_{n+1}) \subset \lambda_{\mu_{\text{prob}}}(p_n)$ for all $n \in \mathbb{N}$.

(B) Let $x, y \in \mathbb{N}$. $\langle \lambda_{\mu_{\text{prob}}}(p) \rangle_{p \in [0,1]}$ is said to be dense in the interval $[x, y]$ iff, for each $p, q \in [x, y]$, $p < q$, there exists an $r \in (x, y)$, $p < r < q$, with $\lambda_{\mu_{\text{prob}}}(p) \subset \lambda_{\mu_{\text{prob}}}(r) \subset \lambda_{\mu_{\text{prob}}}(q)$.

31
Let $p \in [0, 1)$. Then $\langle \lambda \mu_{\text{prob}}(p) \rangle_{p \in [0, 1]}$ has a threshold beginning with $p$ if there exists an indexed family $\mathcal{L}$ with $\mathcal{L} \in \lambda \mu_{\text{prob}}(p) \setminus \bigcup_{p,q \leq 1} \lambda \mu_{\text{prob}}(q)$, and $\lambda \mu_{\text{prob}}(q) = \lambda \mu$ for all $q \in (p, 1]$.

Let $(p_n)_{n \in \mathbb{N}}$ be a sequence of real numbers with $p_n \in [0, 1]$ and $p_{n+1} < p_n$ for all $n \in \mathbb{N}$. The probabilistic hierarchy $\langle \lambda \mu_{\text{prob}}(p) \rangle_{p \in [0, 1]}$ is said to be discrete with breakpoints at $(p_n)_{n \in \mathbb{N}}$ iff, for all $n \in \mathbb{N}$, and all $p \in (p_{n+1}, p_n]$, $\lambda \mu_{\text{prob}}(p_{n+1}) \setminus \lambda \mu_{\text{prob}}(p_n) \neq \emptyset$ and $\lambda \mu_{\text{prob}}(p) = \lambda \mu_{\text{prob}}(p_n)$.

Next, we give the precise definition of the notions sufficient and necessary in the context of probabilistic learning of indexed families under monotonicity constraints.

**Definition 1.2.10.** Let $\lambda \in \{E, \epsilon, C\}$, $\mu \in \{COV, SMON, MON\}$, and $p \in [0, 1]$. Let $A$ be an oracle.

(A) An oracle $A$ is called sufficient for compensating the power of $\lambda \mu_{\text{prob}}(p)$-learning if and only if $\lambda \mu_{\text{prob}}(p) \subseteq \lambda \mu[A]$.

(B) An oracle $A$ is called adequate for compensating the power of $\lambda \mu_{\text{prob}}(p)$-learning, if and only if the following condition is fulfilled. For all oracles $B$, $\lambda \mu_{\text{prob}}(p) \subseteq \lambda \mu[A]$ if and only if $A \leq_T B$.

If an oracle $A$ is adequate for compensating the power of $\lambda \mu_{\text{prob}}(p)$-learning, then $\lambda \mu_{\text{prob}}(p) \subseteq \lambda \mu[B]$ if and only if $B \geq_T A$. Notice that $A$ is not required to characterize the probabilistic learning class in the sense that $\lambda \mu_{\text{prob}}(p) = \lambda \mu[A]$. However, if an oracle $A$ is adequate for compensating the power of $\lambda \mu_{\text{prob}}(p)$-learning, then the Turing degree of $A$ may be considered as a measure for the difficulty of the learning problems in $\lambda \mu_{\text{prob}}(p)$.

**Definition 1.2.11.** Let $\mu \in \{COV, SMON, MON\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $A$ be an oracle which is adequate for compensating the power of $\lambda \mu_{\text{prob}}(p)$-learning. Then the oracle-complexity $O(\lambda \mu_{\text{prob}}(p))$ of $\lambda \mu_{\text{prob}}(p)$ is defined to be the Turing degree of $A$.

Finally, we define oracle-complexity of indexed families. Remark, that this complexity measure depends on the learning model considered.

**Definition 1.2.12.** Let $\mu \in \{COV, SMON, MON\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $\mathcal{L}$ be an indexed family. Assume that there exists an oracle $A$ such that $\mathcal{L} \in \lambda \mu[A]$. 

32
(A) The oracle-complexity $O_{\lambda\mu}(\mathcal{L})$ of $\mathcal{L}$ with respect to $\lambda\mu$, is defined to be the set of all oracles $\{A \subseteq \mathbb{N} \mid \mathcal{L} \in \lambda\mu[A]\}$.

(B) If there exists an oracle $A$ such that $O_{\lambda\mu}(\mathcal{L}) = \{B \subseteq \mathbb{N} \mid A \leq_T B\}$, then the oracle-complexity of $\mathcal{L}$ is said to be $\lambda\mu$-simple.

(C) If the oracle-complexity $O_{\lambda\mu}(\mathcal{L})$ of an indexed family $\mathcal{L}$ is equal to $\{B \subseteq \mathbb{N} \mid A \leq_T B\}$ for an oracle $A \notin \mathcal{R}E\mathcal{C}$, then $\mathcal{L}$ is said to be $A$-difficult with respect to $\lambda\mu$.

Remark 1.2.7. Let $\mathcal{L}$ be an indexed family with oracle-complexity $O_{\lambda\mu}(\mathcal{L}) = \{A \mid A \leq_T B\}$. In this case, we may identify the oracle-complexity of $\mathcal{L}$ with the Turing degree of $A$.

### 1.3 Results

In the first part of our work, we show by using a new separation technique that the probabilistic hierarchies in the case of proper and class preserving probabilistic learning under monotonicity constraints are highly structured.

(a) The probabilistic hierarchies in the case of proper conservative probabilistic learning and proper strong-monotonic probabilistic learning are dense in the interval $[0, 1]$. For class preserving conservative probabilistic learning, we show the probabilistic hierarchy to be dense in the interval $(1/2, 1]$.

(b) In the case of proper monotonic probabilistic learning, we are able to show that the probabilistic hierarchy is dense in $(4/5, 1]$. For class preserving monotonic probabilistic learning, we receive a strictly decreasing probabilistic hierarchy.

This can be regarded as a completely new phenomenon in the field of probabilistic identification.

In the case of class preserving strong-monotonic probabilistic learning, the situation turns out to be different. We show that there exists an indexed family which is strong-monotonically identifiable with probability $p = 2/3$ with respect to a class preserving hypothesis space, but not strong-monotonically identifiable with a larger probability with respect to any class preserving hypothesis space. However, this turns out to be the best result possible in
this case, since each indexed family strong-monotonically identifiable with a probability $p > 2/3$ with respect to a class preserving hypothesis space is deterministically strong-monotonically identifiable with respect to a class preserving hypothesis space.

Furthermore, we investigate the case, where we allow the inductive inference machines to choose their hypotheses from arbitrary complex hypothesis spaces, i.e., class comprising probabilistic learning. It is shown that the additional power of probabilistic learning machines can be compensated by allowing their deterministic counterparts to choose their hypotheses from class comprising hypothesis spaces.

For unconstrained probabilistic identification as well as for class comprising conservative probabilistic identification, we show that probabilistic identification and team identification are equivalent. This yields discrete probabilistic hierarchies in these cases. In particular, each indexed family conservatively identifiable with probability $p > 1/2$ is deterministically conservatively identifiable with respect to an appropriate class comprising hypothesis space. Also in the case of class comprising strong-monotonic probabilistic learning, we can show the probabilistic hierarchy to have a threshold beginning with $1/2$, i.e., each indexed family strongly-monotonically identifiable with probability $p > 1/2$ is deterministically strongly-monotonically identifiable with respect to an appropriate class comprising hypothesis space. For class comprising monotonic probabilistic learning, the probabilistic hierarchy has a threshold beginning with $2/3$. Thus, we can conclude that it is possible to compensate the additional power of probabilistic learning machines without losing monotonicity by choosing appropriate hypothesis spaces.

The separation results are proved by using the technique introduced in the first part of our work. Furthermore, we use some sophisticated variations of the amalgamation technique developed in [25]. For more information about this technique see [80].

Finally, we studied the relations between the three learning models considered. It is easy to see that strong-monotonic learning is weaker than conservative and monotonic learning (cf. [111]). However, conservative and monotonic probabilistic learning are not comparable with respect to set inclusion.

The second part of our work is due to questions of complexity. We start by investigating the power of the learning models considered, i.e., we compare probabilistic learning under monotonicity constraints with oracle identification under monotonicity constraints. It turns out that conservative prob-
Probabilistic learning is less powerful than conservative learning with $K$-oracle if the probability $p$ is greater than $1/2$. The same result holds for strong-monotonic learning. For monotonic probabilistic learning, we show $K$ to be sufficient for compensating the power of probabilistic learning machines provided the probability is greater than $2/3$. For strong-monotonic probabilistic learning the situation is different. If $p > 2/3$, then the power of strong-monotonic probabilistic learning can be compensated by using an arbitrary Peano-complete oracle. Hence, strong-monotonic probabilistic learning is weaker than conservative and monotonic probabilistic learning in the sense that there exists a low oracle which is sufficient for compensating the power of strong-monotonic probabilistic learning with $p > 2/3$.

These bounds are strict, i.e., for conservative probabilistic learning with $p = 1/2$, strong-monotonic probabilistic learning with $p = 1/2$, as well as for monotonic probabilistic learning with $p = 2/3$, $K$ is not sufficient for compensating the power of probabilistic learning machines. For strong-monotonic learning with $1/2 < p \leq 2/3$, the Peano-complete oracles are not sufficient for compensating the power of the probabilistic machines. Furthermore, there exists an indexed family which is strong-monotonically identifiable with probability $1/2$, but not conservatively identifiable by an oracle machine with respect to any class comprising hypothesis space. An analogous result holds for monotonic probabilistic learning with $p = 2/3$. Moreover, we deal with the question whether $K$ is adequate for compensating the power of the probabilistic machines. We show that, for each $p \in [0, 1]$, there exists a learning problem which is properly conservatively identifiable with probability $p$, and properly conservatively identifiable by an oracle machine having access to $K$, but not properly conservatively by an oracle machine which has access to a weaker oracle $A \preceq_T K$. Hence, $K$ is adequate. The same result holds for monotonic probabilistic learning with $p \in [0, 1]$, and strong-monotonic learning with probability $p \leq 2/3$. In particular, these results yield that probabilistic learning under monotonicity constraints cannot be characterized in terms of oracle identification.

Further results concern the oracle-complexity of the learning problems which are properly conservatively identifiable with probability $p = 1/2$. First, we ask whether there are learning problems $\mathcal{L}$ such that $\mathcal{O}_{ECOV}(\mathcal{L}) = \{B \mid A \preceq_T B\}$ for an oracle $A$. Indeed, we are able to show that for each oracle $A \preceq_T K$, $A$ not recursive, there exists an indexed family $\mathcal{L}_A$ such that

\[ (+) \quad \mathcal{L}_A \text{ exactly reflects the Turing complexity of } A, \text{ i.e., } \mathcal{L}_A \text{ is properly} \]
conservatively identifiable by an oracle machine $M[B]$ if and only if $A \leq_T B$,

$(\+)
\quad \mathcal{L}_A$ is properly conservatively identifiable with probability $1/2$.

The proof of this result even shows that every oracle machine $M[B]$ which conservatively identifies $\mathcal{L}_A$ can be transformed into a decision procedure for $A$. Thus, for each nonrecursive $A \leq_T \mathcal{K}$, there is an indexed family having the same degree of unsolvability as $A$ with respect to proper conservative learning.

Furthermore, our result gives us a complete picture of the impact of oracles on the power of conservative inductive learning machines.

$(\+)
\quad$ When dealing with proper conservative learning, $A \leq_T \mathcal{K}$ enhances the learning power of the inductive inference machines if and only if $A$ is not recursive.

$(\+)
\quad$ If $A, B \leq_T \mathcal{K}$ are nonrecursive oracles, then $A \leq_T B$ if and only if every indexed family which is properly conservatively identifiable with oracle $A$ is properly conservatively identifiable with oracle $B$.

However, not every indexed family which is conservatively identifiable with $p = 1/2$ characterizes an oracle, i.e., there is an indexed family $\mathcal{L}$ with the following properties:

$(\+)
\quad \mathcal{L}$ is properly conservatively identifiable with $p = 1/2$, and

$(\+)
\quad$ there exists a minimal pair of sets $(A, B)$ (cf. [77] for the definition of minimal pairs) such that there exists an oracle machine $M[A]$, and an oracle machine $M'[B]$ which properly conservatively identify $\mathcal{L}$.

Thus, we may conclude that the complexity of the learning problems which are properly conservatively identifiable with $p = 1/2$ cannot be described completely in terms of Turing complexity, or, in other words, Turing complexity is not rich enough to measure the complexity of the learning problems in $ECOV_{prob}(1/2)$.

Finally, we show that it is possible to find problems with high oracle-complexity in every probabilistic learning class, i.e., there exist learning problems which are properly conservatively identifiable with probability $p = 1/2$, but only properly conservatively identifiable by oracle machines having access
to $\TOT$. Furthermore, we construct a learning problem which is properly monotonically identifiable with $p = 2/3$, but only properly monotonically identifiable by oracle machines having access to $\TOT$. However, there exists no indexed family characterizing $\TOT$ within strong-monotonic learning, since every indexed family which is strong-monotonically identifiable by an oracle machine is properly strong-monotonically identifiable by an oracle machine having access to $\mathcal{K}$. 
Chapter 2

Probabilistic Hierarchies

As figured out in the introduction, a probabilistic learning model should fulfill two requirements.

1. The probabilistic learners should be more powerful than the deterministic learners even if the probability is close or equal to 1.

2. The probabilistic learning classes should be different for any two probabilities \( p, q \in [0, 1] \), i.e., the corresponding probabilistic hierarchy should be dense. It is sufficient when the probabilistic learning classes fulfill this requirement in an interval \((r, 1]\) where \( r > 1 \), since we are mainly interested in learning problems which are identifiable with a probability near or equal to 1.

In terms of definition 1.2.9, our aim of research in this chapter can be formulated as follows.

Question

Which tuples \((\lambda, \mu)\) and \((x, y) \in [0, 1], x < y\), induce a strictly decreasing or a dense probabilistic hierarchy \(\langle \lambda \mu_{\text{prob}}(p) \rangle_{p \in [x,y]}\)?
2.1 Proper Probabilistic Learning under Monotonicity Constraints

2.1.1 Conservative and Strong-monotonic Probabilistic Learning

In this subsection, we show that for every \( p \in [0, 1] \), there exists an indexed family which is strong-monotonically identifiable with probability \( p \) but not conservatively identifiable with a probability \( q > p \). Hence, the probabilistic hierarchies in the case of strong-monotonic and conservative learning are dense in the interval \([0, 1]\).

Intuitively, the desired separations can be achieved as follows. Let \( L \) be an indexed family which contains, for every \( k \in \mathbb{N} \), the infinite language \( L_k \), and the possibly finite language \( L'_k \). When fed the canonical text for \( L_k \), a PIM \( P \) has the choice between a finite number of indices. If it is not decidable whether an index is an index for \( L'_k \) or not, then \( L'_k \) cannot be inferred deterministically without injuring the monotonicity constraint. In the proof of the following theorem, we make this intuition precise.

**Theorem 2.1.1.** Let \( c, d \in \mathbb{N} \) such that \( 1 > \frac{c}{d} > \frac{1}{2} \) and \( \gcd(c, d) = 1 \). Then

\[
\text{ESMON}^{st}_{\text{prob}}\left(\frac{c}{d}\right) \setminus \bigcup_{0 < \epsilon \leq 1 - \frac{c}{d}} \text{ECOV}_{\text{prob}}\left(\frac{c}{d} + \epsilon\right) \neq \emptyset.
\]

**PROOF.** Let \( c, d \in \mathbb{N} \) such that \( 1/2 < c/d < 1 \) and \( \gcd(c, d) = 1 \). The key idea of the proof can be described as follows. We define an indexed family \( \mathcal{L}^{c/d} = (L_{\langle k, j \rangle})_{k, j \in \mathbb{N}, j \leq c-1} \) such that the following holds.

1. If \( \varphi_k(k) \uparrow \), then all the languages \( L_{\langle k, j \rangle} \) for \( j \in \{0, \ldots, c-1\} \) are infinite and equal to \( L_k \).

2. If \( \varphi_k(k) \downarrow \), then exactly \( 2c - d \) of the languages are finite and equal to \( L'_k \). The other \( d - c \) languages are infinite and equal to \( L_k \). The indices \( \{\langle k, j_1 \rangle, \ldots, \langle k, j_{2c-d} \rangle\} \subset \{\langle k, 0 \rangle, \ldots, \langle k, c-1 \rangle\} \) for the finite languages depend only on the value of \( \varphi_k(k) \).

Obviously, \( \mathcal{L}^{c/d} \) is strong-monotonically identifiable with probability \( (2c - d)/c \), since we can guess each \( \langle k, j \rangle, j \leq c-1 \), with probability \( 1/c \). However, the empty hypothesis \( \perp \) can play the role of an index for a finite language as
long as we have no information about $\varphi_k(k)$, and thus we can prove $L^{c/d}$ to be strong-monotonically identifiable with probability at least $c/d$. Intuitively, the identifying PIM guesses each $(k, j)$ with probability $1/d$ and $\perp$ with probability $(d - c)/d$.

In the following, we precise how to choose the indices for the finite languages. For that purpose, we define a surjective, total recursive function $\text{mod}_{2c-d}^c$. Set

$$M_{2c-d}^c = \{S | S \subseteq \{0, \ldots, c - 1\}, |S| = 2c - d\}.$$ 

Let $\text{cod}_{2c-d}^c : M_{2c-d}^c \to \{0, \ldots, (\frac{c}{2c-d}) - 1\}$ be an effective encoding of $M_{2c-d}^c$. Then define $\text{mod}_{2c-d}^c : \mathbb{N} \to \{0, \ldots, (\frac{c}{2c-d}) - 1\}$ by setting

$$\text{mod}_{2c-d}^c(y) := x \text{ iff } x \in \{0, \ldots, \left(\frac{c}{2c-d}\right) - 1\} \land y \equiv x \text{ mod } \left(\frac{c}{2c-d}\right)$$

for all $y \in \mathbb{N}$. Obviously $\text{mod}_{2c-d}^c$ is total recursive, surjective, and $\text{mod}_{2c-d}^c(y)$ encodes a subset of $\{0, \ldots, c - 1\}$ of cardinality $2c - d$ for each $y \in \mathbb{N}$.

Now we are ready to define $L^{c/d}$ more formally. Let $(\cdot, \cdot) : \mathbb{N} \times \{0, \ldots, c - 1\} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \{0, \ldots, c - 1\}$. For $k, j \in \mathbb{N}$, $j \leq c - 1$ define $L_{(k,j)} \subseteq L_k$ as follows. Let $k, j \in \mathbb{N}$, $j \leq c - 1$.

$$L_{(k,j)} := \left\{ \begin{array}{ll} L'_k, & \text{if } \varphi_k(k) \downarrow \land j \in (\text{cod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))), \\ L_k, & \text{if } \varphi_k(k) \downarrow \land j \notin (\text{cod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))), \\ L_k, & \text{if } \varphi_k(k) \uparrow. \end{array} \right.$$ 

The languages $L_{(k,j)}$ can be defined alternatively in the following way. Let $k, j \in \mathbb{N}$, $j \leq c - 1$.

1. If $\Phi_k(k) > n - 1$, then

$$a^k b^n \in L_{(k,j)} \text{ for all } j \leq c - 1,$$

2. if $\Phi_k(k) \leq n - 1$, then

$$a^k b^n \in L_{(k,j)} \text{ iff } j \notin (\text{cod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))).$$

Thus, $L^{c/d} = (L_{(k,j)})_{k,j \in \mathbb{N}, j \leq c - 1}$ is an indexed family, since “$\Phi_k(k) \leq n - 1$” is uniformly decidable for all $k, n \in \mathbb{N}$.

Obviously, $L^{c/d}$ has the following properties. Let $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$. Then $L_{(k,j)} = L'_{k}$ if and only if $j \in (\text{cod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k)))$. Otherwise $L_{(k,j)} = \perp$. 

40
Notice that it is not decidable whether a language \( L \in \text{range}(\mathcal{L}^{c/d}) \) is finite or not. We have to show now that \( \mathcal{L}^{c/d} \) witnesses the desired separation.

In order to prove \( \mathcal{L}^{c/d} \in \text{ESMON}_{\text{prob}}(c/d) \), we define a PIM \( P \) equipped with a \( d \)-sided coin. Let \( L \in \text{range}(\mathcal{L}^{c/d}) \), let \( \tau \) be a text for \( L \), and let \( k \in \mathbb{N} \) with \( \text{range}(\tau) \cap L_k \neq \emptyset \). Let \( x \) be a natural number. Let \( m_x \) be the highest natural number with \( a^k b^{m_x} \in \text{range}(\tau_x) \). Let \( C \in \{0, \ldots, d-1\}^\infty \) be a coin-oracle.

**PIM \( P \):** On input \( \tau_x \), \( P^C \) works as follows.

(A) If \( c_0 = j \) with \( j \in \{0, \ldots, c-1\} \), then distinguish the following cases.

(A1) If \( \langle k, l \rangle \) is consistent with \( \tau_x \) for all \( l \leq c-1 \), then set

\[
P^{C_x}(\tau_x) = \langle k, j \rangle.
\]

(A2) If \( \langle k, l \rangle \) is not consistent with \( \tau_x \) for some \( l \leq c-1 \), then output \( \langle k, i \rangle \) with \( i \) being the least natural number \( \leq c-1 \) such that \( \langle k, i \rangle \) is consistent with \( \tau_x \).

(B) If \( c_0 = j \) with \( j \in \{c, \ldots, d-1\} \), then test whether \( \Phi_k(k) \leq m_x \). If it is, then output \( \langle k, i \rangle \) with \( i \) being the least natural number in \( (\text{cod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\psi_k(k))) \). Otherwise output \( \bot \).

end

Obviously, \( P^C \) is strong-monotonic on \( \tau \), since \( P^C \) never changes from an index for an infinite language to an index for a finite language. Furthermore, the probability of the set of all coin-oracles \( C \) such that \( P^C \) converges correctly on \( \tau \) is \( c/d \). Thus, \( \mathcal{L}^{c/d} \in \text{ESMON}_{\text{prob}}(c/d) \).

It remains to show that \( \mathcal{L}^{c/d} \notin \text{ECOV}_{\text{prob}}(p) \) for all \( p > c/d \). Assume the converse, and let \( \epsilon > 0 \) with \( \mathcal{L}^{c/d} \in \text{ECOV}_{\text{prob}}((c/d) + \epsilon) \). Let \( P \) be the identifying PIM. We define a recursive procedure \( \mathcal{I} \) which computes for each \( k \in \mathbb{N} \) a natural number \( j \in \{0, \ldots, (\frac{c}{2c-d})-1\} \) such that \( j \neq \text{mod}_{2c-d}^c(\phi_k(k)) \). Let \( k \) be a natural number. Recall that \( L_k \in \mathcal{L}^{c/d} \).

In Stage \( n \), \( \mathcal{I} \) works as follows:

If \( \Phi_k(k) \leq n - 1 \), then the procedure halts. Define \( \mathcal{I}(k) \) to be the least
ℓ ∈ \{0, \ldots, (2c - d) - 1\} with ℓ ≠ mod_2c-d(ϕ_k(k)).

Otherwise compute T_{P;\tau^n} and define for all j ∈ N, j ≤ c - 1:

\[ C^j_n = \{ C | P^{c\ell}(\tau^n) = \langle k, j \rangle \text{ and } \forall x < n : P^{c\ell}(\tau^n) \in \{ \langle k, j \rangle, \bot \} \}. \]

Set

p^j_n = Pr(C^j_n).

Test if \( \sum_{j=0}^{c-1} p^j_n > c/d \). If not, then go to stage n + 1. Otherwise there exists a z > 0 with \( \sum_{j=0}^{c-1} p^j_n = (c/d) + z \). Let \( S = \{ j_1, \ldots, j_{2c-d} \} \) be a subset of \( \{0, \ldots, c - 1\} \) of cardinality 2c - d. Set

\[ m_S = \sum_{s=1}^{2c-d} p^n_{j_s} + \left(1 - \frac{c}{d} - z\right). \]

Then define

\[ \mathcal{I}(k) := \text{the least } \ell \in \{0, \ldots, \left(\frac{c}{2c - d}\right) - 1\} \text{ with } m_{(\text{mod}_{2c-d})^{-1}(\ell)} < \frac{c}{d}. \]

end

Now we have to prove that the procedure \( \mathcal{I} \) fulfills the desired conditions.

Claim 1 The procedure \( \mathcal{I} \) terminates for each \( k ∈ \mathbb{N} \).

Proof (of the claim). Let \( k \) be a natural number.

1. \( p^n_j \) is computable for all \( j, n ∈ \mathbb{N}, j ≤ c - 1 \), and there exists an \( n ∈ \mathbb{N} \) with \( \sum_{j=0}^{c-1} p^n_j > c/d \), since \( L^{c/d} \) is assumed to be in \( \text{ECOV}_{\text{prob}}((c/d) + \varepsilon) \).

2. Let \( n ∈ \mathbb{N} \) be the smallest number such that \( \sum_{j=0}^{c-1} p^n_j > c/d \). If \( \Phi_k(k) ≤ n - 1 \), then \( \mathcal{I} \) halts in one of the steps 0, \ldots, n.

If \( \Phi_k(k) > n \), then we show that there exists a subset \( S = \{ j_1, \ldots, j_{2c-d} \} \) of \( \{0, \ldots, c - 1\} \) with \( m_S < c/d \). Assume the converse, i.e., \( m_S ≥ c/d \) for each subset \( S \) of \( \{0, \ldots, c - 1\} \) of cardinality \( 2c - d \). Then

\[ \sum_{S⊂\{0, \ldots, c - 1\}, |S|=2c-d} m_S ≥ \left(\frac{c}{2c - d}\right) \cdot \frac{c}{d}. \]
From the definition of $m_S$ and the equation $\sum_{j=0}^{c-1} p_j^n = (c/d) + z$, we obtain
\[
\left(\frac{c-1}{2c-d-1}\right) \cdot \left(\sum_{j=0}^{c-1} p_j^n\right) + \left(\frac{c}{2c-d}\right) \cdot \left(1 - \frac{c}{d} - z\right) \geq \left(\frac{c}{2c-d}\right) \cdot \frac{c}{d},
\]
and hence,
\[
\frac{c}{d} + z - \frac{d}{c} \cdot z \geq \frac{c}{d},
\]
a contradiction, since $d/c > 1$.

Consequently, $I$ is effective and outputs the code number of a subset of \{0,\ldots,c-1\} of cardinality $2c - d$ for all $k \in \mathbb{N}$. end (of Proof).

Claim 2 For every $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$, the following holds:
\[
I(k) \neq \text{mod}_{2c-d}(\varphi_k(k)).
\]

Proof (of the claim). In order to prove this claim, let $k$ be a natural number with $\varphi_k(k) \downarrow$. Let \{L_{(k,j)1},\ldots,L_{(k,j_{2c-d})}\} be the set of finite languages, i.e., $L_{(k,j_i)} = L'_k$ for all $i \in \{1,\ldots,2c-d\}$. Set $S_k = \{j_1,\ldots,j_{2c-d}\}$. Let $n \in \mathbb{N}$ such that $I$ stops in Stage $n$. Then either $\Phi_k(k) \leq n - 1$, and thus $I(k) \neq \text{mod}_{2c-d}(\varphi_k(k))$ by definition of $I$, or $\Phi_k(k) > n - 1$. In the latter case,
\[
\tau_k^n = (\tau^{L_k})_n \quad \text{and} \quad I(k) = \ell \quad \text{with} \quad m_{(\text{mod}_{2c-d})^{-1}(\ell)} < \frac{c}{d}.
\]
Obviously, every path not summed up in $m_{S_k}$ does not converge conservatively on $\tau^{L_k}$, since these paths contain at least one index for $L_k$ in the subtree induced by $(\tau^{L_k})_n$. Since $L'_k$ is assumed to be in $ECOV_{\text{prob}}((c/d) + \epsilon)$, the finite language $L'_k$ has to be conservatively identified with a probability $p \geq (c/d) + \epsilon$. Consequently, $m_{S_k}$ has to be greater or equal to $(c/d) + \epsilon$, and we can conclude that $I(k) \neq \text{mod}_{2c-d}(\varphi_k(k))$. end (of Proof).

With these two claims, we are able to prove the desired separation. $I$ is total and hence equal to a total recursive function $\varphi_I = \varphi_{k_0}$ for a $k_0 \in \mathbb{N}$. Thus, $\varphi_{k_0}(k_0)$ equals $\text{mod}_{2c-d}(\varphi_{k_0}(k_0))$. However, by Claim 2, $\text{mod}_{2c-d}(\varphi_{k_0}(k_0)) \neq I(k_0)$, a contradiction. It follows that $L^{c/d} \notin ECOV_{\text{prob}}((c/d) + \epsilon)$. ♯

It is easy to see that the indexed family $L^{c/d}$ defined in Theorem 2.1.1 is properly monotonically identifiable with probability $2c/(c+d)$. Thus, in contrast to the result in the deterministic case, proper monotonic probabilistic learning is not weaker than proper conservative probabilistic learning.

43
**Theorem 2.1.2.** Let $c, d \in \mathbb{N}$ such that $1 > \frac{c}{d} > \frac{1}{2}$ and $\gcd(c, d) = 1$. Then there exists an indexed family $\mathcal{L}^{c,d}$ which is properly monotonically identifiable with probability $p = \frac{2c}{c+d}$ but not properly conservatively identifiable with probability $p > \frac{c}{d}$.

**PROOF.** Define $\mathcal{L}_{c/d}$ as in the proof of Theorem 2.1.1. We only have to show that $\mathcal{L}_{c/d} \in \text{EMON}_{\text{prob}}(2c/c + d)$. Define a PIM $P$ equipped with a $(c + d)$-sided coin as follows. Let $L \in \text{range}(\mathcal{L}^{c/d})$, let $\tau$ be a text for $L$, and let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $x$ be a natural number. Let $m_x$ be the highest natural number with $a^kb^{m_x} \in \text{range}(\tau_x)$. Let $C \in \{0, \ldots, c + d - 1\}^\infty$ be a coin-oracle.

**PIM $P$:** On input $\tau_x$, $P^c$ works as follows.

(A) If $c_0 \in \{0, \ldots, c - 1\}$, then distinguish the following cases.

(A1) If $\langle k, l \rangle$ is consistent with $\tau_x$ for all $l \leq c - 1$, then set

$$P^{c_x}(\tau_x) = \langle k, c_0 \rangle.$$

(A2) If $\langle k, l \rangle$ is not consistent with $\tau_x$ for some $l \leq c - 1$, then output $\langle k, i \rangle$ with $i$ being the least natural number $\leq c - 1$ such that $\langle k, i \rangle$ is consistent with $\tau_x$.

(B) If $c_0 \in \{c, \ldots, 2c - 1\}$, then test whether $\Phi_k(k) \leq m_x$, and distinguish the following cases.

(B1) If $\Phi_k(k) \leq m_x$, and if $\langle k, l \rangle$ is consistent with $\tau_x$ for all $l \leq c - 1$, then set

$$P^{c_x}(\tau_x) = \langle k, i \rangle,$$

where $i$ is the least natural number in $(\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))$.

(B2) If $\Phi_k(k) \leq m_x$, and if $\langle k, l \rangle$ is not consistent with $\tau_x$ for some $l \leq c - 1$, then output $\langle k, i \rangle$ with $i$ being the least natural number $\leq c - 1$ such that $L_{\langle k, i \rangle}$ is consistent with $\tau_x$.

(B3) Otherwise output $\langle k, c_0 \rangle$.

(C) If $c_0 = j$ with $j \in \{2c, \ldots, c + d\}$, then test whether $\Phi_k(k) \leq m_x$.

If not, then output $\bot$. If it is, then output $\langle k, i \rangle$ with $i$ being the least natural number in $(\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))$ provided $\langle k, i \rangle$ is consistent with $\tau_x$. Otherwise output $\langle k, l \rangle$ with $l$ being the least natural number $\leq c - 1$ such that $\langle k, l \rangle$ is consistent with $\tau_x$.
Let $k \in \mathbb{N}$ with $\varphi_k(k) \uparrow$. Then all languages are equal to $L_k$, and $P$ monotonically identifies $L_k$ with probability $p = 2c/(c + d)$. Let $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$. In step (B1), the algorithm $P$ guesses a language $L'_k$ after guessing a language $L_k$. Hence, the total weight of all paths which are monotonic and convergent for $L'_k$ and $L_k$, respectively, is equal to $(c + 2c - d + d - c)/(c + d) = 2c/(c + d)$. Thus, $P$ EMON$_{\text{prob}}(2c/(c + d))$-identifies $L_{c,d}$.

In the following, we investigate the structure of the probabilistic hierarchies $\langle \text{ESMON}_{\text{st}} \text{prob}(p) \rangle_{p \in [0,1]}$, $\langle \text{ESMON}_{\text{prob}}(p) \rangle_{p \in [0,1]}$, and $\langle \text{ECOV}_{\text{prob}}(p) \rangle_{p \in [0,1]}$ in the interval $[0,1/2]$.

**Theorem 2.1.3.** Let $c, d \in \mathbb{N}$ such that $1 > c/d > 1/2$ and $\gcd(c, d) = 1$. Let $n \in \mathbb{N}$, $n \geq 2$. Then

$$\text{ESMON}_{\text{st}} \text{prob}\left(\frac{c}{d \cdot n}\right) \setminus \bigcup_{0 < \epsilon \leq 1 - \frac{c}{2n}} \text{ESMON}_{\text{prob}}\left(\frac{c}{d \cdot n} + \epsilon\right) \neq \emptyset,$$

and

$$\text{ECOV}_{\text{prob}}\left(\frac{c}{c(n - 1) + d}\right) \setminus \bigcup_{0 < \epsilon \leq 1 - \frac{c}{c(n - 1) + d}} \text{ECOV}_{\text{prob}}\left(\frac{c}{c(n - 1) + d} + \epsilon\right) \neq \emptyset.$$

**Proof.** Let $n \in \mathbb{N}$, $n \geq 2$, and let $c, d \in \mathbb{N}$ with $c/d > 1/2$ and $\gcd(c, d) = 1$. Let $\mathcal{L}^{c/d} = (L_{(i,j)})_{i,j \in \mathbb{N}, j \leq c - 1}$ be the indexed family defined in Theorem 2.1.1, and let $\mathcal{L}^{n-1} = (L^Q_i)_{i \in \mathbb{N}}$ be the indexed family defined in the proof of Theorem 2.3.4. Then $\mathcal{L}^{c/d}$ is ESMON$_{\text{prob}}(c/d)$-identifiable, but not conservatively identifiable with a higher probability, and $\mathcal{L}^{n-1}$ is ESMON$_{\text{prob}}(1/n)$-identifiable, but not CLIM$_{\text{prob}}(p)$-identifiable with a higher probability. We define the indexed family witnessing the desired separations by combining these two indexed families.

Let $\Sigma = \{1, a, b\}$. Let $\langle , , \rangle : \mathbb{N} \times \mathbb{N} \times \{0, \ldots, c - 1\} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N} \times \{0, \ldots, c - 1\}$. Set

$$L_{<i,k,j>} := L^Q_i \cup L_{(k,j)}$$

for all $i, k, j \in \mathbb{N}$, $j \leq c - 1$. Set $\mathcal{L}^{c/d}_{n-1} = (L_{<i,k,j>})_{i,j,k \in \mathbb{N}, j \leq c - 1}$. Obviously, $\mathcal{L}^{c/d}_{n-1}$ is an indexed family. Now we have to show that $\mathcal{L}^{c/d}_{n-1}$ establishes the wanted separation. For the sake of readability, we restrict ourselves to the
case \(n = 2\). Recall that \(\mathcal{L}^1 = (L^Q_i)_{i \in \mathbb{N}}\) where \(L^Q_0 = \mathbb{Q}^+\), and \(L^Q_m = \{(x, y) \mid (x, y) \leq (m, 1)\}\) for all \(m \in \mathbb{N}^+\) (cf. Theorem 2.3.4).

First of all, we prove that \(L^{c/d}_1 \in \text{ESMON}^\text{st}_{\text{prob}}(c/2d)\). Define a PIM \(P\) equipped with a 2\(d\)-sided coin as follows. Let \(L \in \text{range}(\mathcal{L}^1_{c/d})\), let \(\tau\) be a text for \(L\), and let \(x\) be a natural number. Let \(k \in \mathbb{N}\) with \(\text{range}(\tau) \cap L_k \neq \emptyset\). We may assume that \(\text{range}(\tau_x)\) is the union of a finite set of rational numbers and the range of an initial segment of \(\tau^k\) for a \(k \in \mathbb{N}\). Let \(m\) be the least natural number with \(\text{range}(\tau_x) \cap \mathbb{Q}^+ \subset [0, m]\). Furthermore, let \(m_x\) be the highest natural number with \(a^k b^{m_x} \in \text{range}(\tau_x)\). Let \(C \in \{0, \ldots, 2d - 1\}^{\infty}\) be a coin-oracle.

**PIM \(P\):** On input \(\tau_x\), \(P^c\) works as follows.

(A) If \(c_0 = j\) with \(j \in \{0, \ldots, c - 1\}\) then distinguish the following cases.

(A1) If \(\langle k, l \rangle\) is consistent with \(\tau_x \cap L_k\) for all \(l \leq c - 1\), then set \(P^c_0(\tau_x) = \langle m, k, j \rangle\).

(A2) If \(\langle k, l \rangle\) is not consistent with \(\tau_x \cap L_k\) for some \(l \leq c - 1\), then output \(\langle m, k, i \rangle\) with \(i\) being the least natural number \(\leq c - 1\) such that \(\langle k, i \rangle\) is consistent with \(\tau_x \cap L_k\).

(B) If \(c_0 = j\) with \(j \in \{c, \ldots, d - 1\}\), then test whether \(\Phi_k(k) \leq m_x\). If it is, then output \(\langle m, k, i \rangle\) with \(i\) being the least natural number in \((\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))\) and request the next input. Otherwise output \(\bot\).

(C) If \(c_0 = j\) with \(j \in \{d, \ldots, d + c - 1\}\), then distinguish the following cases.

(C1) If \(\langle k, l \rangle\) is consistent with \(\tau_x \cap L_k\) for all \(l \leq c - 1\), then set \(P^c_0(\tau_x) = \langle 0, k, j \rangle\).

(C2) If \(\langle k, l \rangle\) is not consistent with \(\tau_x \cap L_k\) for some \(l \leq c - 1\), then output \(\langle 0, k, i \rangle\) with \(i\) being the least natural number \(\leq c - 1\) such that \(\langle k, i \rangle\) is consistent with \(\tau_x \cap L_k\).

(D) If \(c_0 = j\) with \(j \in \{d + c, \ldots, 2d - 1\}\), then test whether \(\Phi_k(k) \leq m_x\). If it is, then output \(\langle 0, k, i \rangle\) with \(i\) being the least natural number in \((\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))\). Otherwise output \(\bot\).
Obviously, $P \text{ESMON}_{prob}^{st}(c/2d)$-identifies $\mathcal{L}_1^{c/d}$.

Next we show that $\mathcal{L}_1^{c/d} \in \text{ECOV}_{prob}(c/(c+d))$. For that purpose, we define a PIM $P$ equipped with a $(c+d)$-sided coin. Let $L \in \text{range}(\mathcal{L}_1^{c/d})$, let $\tau$ be a text for $L$, and let $x$ be a natural number. Let $k \in \mathbb{N}$ with $\text{range} \tau \cap L_k \neq \emptyset$. We may assume that $\text{range} \tau$ is the union of a finite set of rational numbers and the range of an initial segment of $\tau$ for a $k \in \mathbb{N}$. Let $m$ be the least natural number with $\text{range} \tau \cap \mathbb{Q}^+ \subset [0, m]$, and let $m_x$ be the highest natural number with $a^kb^m \in \text{range} \tau$. Let $\mathcal{C} \in \{0, \ldots, c + d - 1\}^\infty$ be a coin-oracle. Notice that $c + d = 2(2c - d) + 3(d - c)$.

**PIM $P$:** On input $\tau_x$, $P^c$ works as follows.

(A) If $c_0 = j$ with $j \in \{1, \ldots, c - 1\}$, then distinguish the following cases.

(A1) If $\langle 0, k, l \rangle$ is consistent with $\tau_x$ for all $l \leq c - 1$, then set $P^c(\tau_x) = \langle 0, k, j \rangle$.

(A2) If $\langle 0, k, l \rangle$ is not consistent with $\tau_x$ for some $l \leq c - 1$, then output $\langle 0, k, i \rangle$ with $i$ being the least natural number $\leq c - 1$ such that $\langle 0, k, i \rangle$ is consistent with $\tau_x$.

(B) If $c_0 = j$ with $j \in \{c, \ldots, 2c - 1\}$, then test whether $\Phi_k(k) \leq m_x$ and distinguish the following cases.

(B1) If $\Phi_k(k) > m_x$, then output $\langle m, k, j \rangle$.

(B2) Assume $\Phi_k(k) \leq m_x$. If $P^{c-1}(\tau_{x-1})$ is consistent with $\tau_x$, then output $P^{c-1}(\tau_{x-1})$ and request the next input. If $P^{c-1}(\tau_{x-1})$ is not consistent with $\tau_x$, then distinguish the following cases. 

- Assume $j \notin (\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))$.
  - If $\Phi_k(k) < m_x$, then output $\langle m, k, j \rangle$. If $\Phi_k(k) = m_x$, then let $i$ be the least natural number in $(\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))$ and output $\langle 0, k, i \rangle$ with.
  - If $j \in (\text{cod}_{2c-d})^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))$, and $\langle m, k, j \rangle$ is consistent with $\tau_x$, then output $\langle m, k, j \rangle$; if $\langle m, k, j \rangle$ is not consistent with $\tau_x$, then output $\langle m, k, i \rangle$ with $i$ being the least natural number $\leq c - 1$ such that $\langle m, k, i \rangle$ is consistent with $\tau_x$. 

47
If $c_0 = j$ with $j \in \{2c, \ldots, d + c - 1\}$, then test whether $\Phi_k(k) \leq m_x$.

If it is, then output $\langle m, k, i \rangle$ with $i$ being the least natural number in $\langle \text{cod}_{2c-d}^{-1}(\text{mod}_{2c-d}(\varphi_k(k))) \rangle$ provided $\Phi_k(k) = m_x$. Otherwise output $\langle m, k, i \rangle$ with $i$ being the least natural number $\leq c - 1$ such that $\langle m, k, i \rangle$ is consistent with $\tau_x$. If $\Phi_k(k) > m_x$, then output $\bot$.

Then $P$ properly conservatively identifies $L_{1}^{c/d}$ with probability $c/(c + d))$.

It remains to show that $L_{1}^{c/d}$ establishes the wanted separations. First we show that $L_{1}^{c/d} \notin \text{ESMON}_{\text{prob}}((c/2d) + \epsilon)$ for all $\epsilon \in (0, 1 - (c/2d))$. Assume the converse and let $P$ be a PIM which $\text{ESMON}$-identifies $L_{1}^{c/d}$ with a probability $p > c/2d$. We first notice that every path in the infinite computation tree induced by $P$ and $\tau$ which $\text{ESMON}$-converges for a language in $L_{1}^{c/d}$ has to be strong-monotonic in both components, i.e., whenever a hypothesis $\langle u', k', j' \rangle$ is produced after a hypothesis $\langle u, k, j \rangle$ for $u, k, j \in \mathbb{N}$, $j \leq c - 1$, then $L_{u}' \subseteq L_{u}$ and $L_{(k,j)}` \subseteq L_{(k,j')}$. In particular, a path which $\text{ESMON}$-converges for $L_{m}^{Q} \cup L_{k}^{L}$, $m > 0$, may not contain an index of the form $\langle 0, k, j \rangle$. By considering these facts, we can construct a PIM $P'$ which $\text{ESMON}$-identifies $L_{c/d}$ with a probability $p > c/d$, a contradiction to Theorem 2.1.1.

Next we show that $L_{1}^{c/d} \notin \text{ECOV}_{\text{prob}}(c/(c + d) + \epsilon)$ for all $\epsilon \in (0, d/(c + d))$. Assume the converse and let $P$ be a PIM which $\text{ECOV}_{\text{prob}}(p)$-identifies $L_{1}^{c/d}$ with a probability $p > c/(c + d)$. As in the proof of Theorem 2.1.1, we can define a recursive procedure $I$ computing a natural number $j \in \{0, \ldots, (c_{2c-d}) - 1\}$ such that $j \neq \text{mod}_{2c-d}(\varphi_k(k))$ for all $k \in \mathbb{N}$. Since the technique is known from Theorem 2.1.1, we only give a sketch of the proof.

Let $k$ be a natural number. Let $\tau^{Q,k}$ be the canonical text for $Q \cup L_k$. Let $T_{P,\tau^{Q,k}}$ be the corresponding infinite computation tree. Let $n, j \in \mathbb{N}$, $j \leq c - 1$, and define

$$p_{j,n}^{Q} = \text{Pr}(\{ c \mid P_{\text{can}}(\tau^{Q,k})_n = \langle 0, k, j \rangle \}).$$

By assumption, $Q \cup L_k$ is conservatively identifiable with $p > c/(c + d)$, and thus, there is an $n_0 \in \mathbb{N}$ such that

$$\sum_{j=0}^{c-1} p_{j,n_0}^{Q} = \frac{c}{c + d} + z_Q$$

48
for a $z_Q > 0$. Let $m_0$ be the least natural number with $\text{range}((\tau^{Q,k}_{n_0}) \cap \mathbb{Q}^+ \subseteq \{0, m_0\}$. Let $\tau$ be a text for $L^Q_{m_0} \cup L_{(k,j)}$ with $(\tau^{Q,k}_{n_0}) \subseteq \tau$. Define for $n \in \mathbb{N}$, $n \geq n_0$,

$$p_{j,n}^{n_0} = \Pr(\{ C \mid \forall x \leq n_0 \ P^C_{x}(\tau_x) \neq \langle 0, k, j \rangle, \text{ and } P^C_n(\tau_n) = \langle m_0, k, j \rangle \})$$. 

Obviously, every path $(P^C_{x}(\tau_x))_{x \in \mathbb{N}}$ in $T_{P,\tau}$ with $P^C_{n_0}(\tau_{n_0}) = P^C_{n_0}((\tau^{Q,k}_{n_0}) = \langle 0, k, j \rangle$ is not conservative for $L^Q_{m_0} \cup L_{(k,j)}$. Thus, there exists an $n_1 \in \mathbb{N}$, $n_1 > n_0$, with

$$\sum_{j=0}^{c-1} p_{j,n_1}^{n_0} = \frac{c}{c+d} + z_{n_0}$$

for a $z_{n_0} > 0$. Let $S = \{j_1, \ldots, j_{2c-d}\}$ be a subset of $\{0, \ldots, c-1\}$ of cardinality $2c-d$, and set

$$m_{S}^{n_0} = \sum_{s=1}^{2c-d} p_{j,s,n_1}^{n_0} + \left(\frac{d-c}{c+d} - z_Q - z_{n_0}\right)$$

Now we can show by using the same counting argument as in the first part of the proof that there exists at least one subset $S = \{j_1, \ldots, j_{2c-d}\}$ of $\{0, \ldots, c-1\}$ such that $m_{S}^{n_0} < c/(c+d)$. Assume $\varphi_k(k) \downarrow$. Then the set $\{\langle m_0, k, j_i \rangle \mid j_i \in S\}$ contains at least one index for $L^Q_{m_0} \cup L_k$, since $L^Q_{m_0} \cup L_k$ is assumed to be conservatively identifiable with $p > c/(c+d)$. Hence, we can conclude that $L_{1}^{c/d} \notin \text{ECOV}_p$ for all $p > c/(c+d)$. This completes the proof of the theorem. ♦

Theorem 2.1.1 and Theorem 2.1.3 directly yield the following consequences.

**Corollary 2.1.4.** $\langle \text{ESMON}^\ast_{\text{prob}}(p) \rangle_{p \in [0,1]}$ and $\langle \text{ESMON}_{\text{prob}}(p) \rangle_{p \in [0,1]}$ as well as $\langle \text{ECOV}_{\text{prob}}(p) \rangle_{p \in [0,1]}$ are dense in $[0, 1]$.

### 2.1.2 Monotonic Probabilistic Learning

Next, we prove two theorems similar to Theorem 2.1.1 for the case of proper monotonic probabilistic learning. First we investigate the probabilistic hierarchy $\langle \text{EMON}^\ast_{\text{prob}}(p) \rangle_{p \in [0,1]}$. In the following, we show a theorem establishing a useful connection between the probabilistic learning classes $\lambda\text{MON}^\ast_{\text{prob}}(p)$ and $\lambda\text{COV}_{\text{prob}}(p)$ for $\lambda \in \{E, \varepsilon\}$. 
Theorem 2.1.5. Let \( p \in [0, 1] \), and let \( \lambda \in \{ E, \, \varepsilon \} \). Then \( \lambda \text{MON}_{\text{prob}}^{st}(p) \subseteq \lambda \text{COV}_{\text{prob}}(p) \).

PROOF. Let \( p \in [0, 1] \), let \( \lambda \in \{ E, \, \varepsilon \} \), and let \( \mathcal{G} \) be a hypothesis space. Let \( \mathcal{L} \) be an indexed family with \( \mathcal{L} \in \lambda \text{MON}_{\text{prob}}^{st}(p) \). Let \( P \) be any identifying PIM, let \( L \in \text{range}(\mathcal{L}) \), and let \( \tau \) be a text for \( L \). It suffices to show that a path in \( \mathcal{T}_{P,\tau} \) which is monotonic and converges correctly on \( \tau \) with respect to \( \mathcal{G} \) may not contain an index for an overgeneralization of \( L \). Assume the converse. Then there exist a coin-oracle \( \mathcal{C} \), an \( m_0 \in \mathbb{N} \) and an overgeneralization \( L' \) for \( L \) such that there are \( x, \, y \in \mathbb{N} \), \( x < y \leq m_0 \), with \( L' = L(G_{\text{pro}}(\tau_x)) \) and \( L = L(G_{\text{pro}}(\tau_y)) \). Consequently, \( (P^{\text{pro}}(\tau_x'))_{x \in \mathbb{N}} \) is not monotonically for \( L' \) for every text \( \tau' \in \text{text}(L') \) with \( \tau_{m_0} \subsetneq \tau' \), a contradiction to our assumption. Hence, we can easily define a PIM which conservatively identifies \( \mathcal{L} \) with probability \( p \).

From Theorem 2.1.1 and Theorem 2.1.5 follows immediately that the probabilistic hierarchy \( \langle \text{EMON}_{\text{prob}}^{st}(p) \rangle_{p \in [0, 1]} \) is dense in the interval \([1/2, 1]\).

Corollary 2.1.6. \( \langle \text{EMON}_{\text{prob}}^{st}(p) \rangle_{p \in [0, 1]} \) is dense in \([1/2, 1]\).

In the case of \( \langle \text{EMON}_{\text{prob}}(p) \rangle_{p \in [0, 1]} \), we are able to show that the probabilistic hierarchy is dense in a sufficiently large neighborhood of 1. The idea of the proof is analogous to the idea used in the proof of Theorem 2.1.1. However, the technical realization in this case is more sophisticated than in the case of strong-monotonic and conservative probabilistic learning, since the probabilistic machines may choose overgeneralizations of the language to be learned on the converging paths. Thus, we give the precise proof of the following theorem in order to show how to deal with the difficulties arising in the case of monotonic probabilistic learning.

Theorem 2.1.7. Let \( c, \, d \in \mathbb{N} \) such that \( \frac{2c}{d} > \frac{2}{3} \), and \( \gcd(c, d) = 1 \). Then

\[
\text{EMON}_{\text{prob}}\left(\frac{2c}{d}\right) \setminus \bigcup_{0 < \epsilon \leq 1 - \frac{2c}{d}} \text{EMON}_{\text{prob}}\left(\frac{2c}{d} + \epsilon\right) \neq \emptyset.
\]

PROOF. Let \( c, \, d \in \mathbb{N} \) such that \( 2c/d > 2/3 \) and \( \gcd(c, d) = 1 \). Let \( \langle \; \rangle : \mathbb{N} \times \{0, \ldots, c-1\} \to \mathbb{N} \) be an effective encoding of \( \mathbb{N} \times \{0, \ldots, c-1\} \). For \( k, \, j \in \mathbb{N} \), \( j \leq c-1 \), define \( L_{(k, j)} \subseteq L_k \) as follows. Let \( k, \, j \in \mathbb{N} \).

\[
L_{(k, j)} := \begin{cases} 
L'_k, & \text{if } \varphi_k(k) \downarrow \land j \in (\text{cod}_{3c-d})^{-1}(\text{mod}_{3c-d}(\varphi_k(k))), \\
L_k, & \text{if } \varphi_k(k) \downarrow \land j \notin (\text{cod}_{3c-d})^{-1}(\text{mod}_{3c-d}(\varphi_k(k))), \\
L_k, & \text{if } \varphi_k(k) \uparrow .
\end{cases}
\]
Set $\mathcal{L}^{2c/d} := (L_{i,j})_{j \in \mathbb{N}}$, $j \leq c-1$. Obviously, $\mathcal{L}^{2c/d}$ is an indexed family. In order to prove $\mathcal{L}^{2c/d} \in \text{ESMON}_{\text{prob}}(2c/d)$, we define a PIM $P$ equipped with a $d$-sided coin. Let $L \in \text{range}(\mathcal{L}^{2c/d})$, let $\tau \in \text{text}(L)$, and let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $x$ be a natural number. Let $m_x$ be the highest natural number with $a^{k}b^{m_x} \in \text{range}(\tau)$. Let $\mathcal{C} \in \{0, \ldots, d-1\}^\infty$ be a coin-oracle.

**PIM $P$:** On input $\tau_x$, $P^c$ works as follows.

(A) If $c_0 \in \{0, \ldots, c-1\}$, then distinguish the following cases.

(A1) If $\langle k, l \rangle$ is consistent with $\tau_x$ for all $l \leq c-1$, then set

$$P^{c_0}(\tau_x) = \langle k, c_0 \rangle.$$ 

(A2) If $\langle k, l \rangle$ is not consistent with $\tau_x$ for some $l \leq c-1$, then output $\langle k, i \rangle$ with $i$ being the least natural number $\leq c-1$ such that $\langle k, i \rangle$ is consistent with $\tau_x$.

(B) If $c_0 \in \{c, \ldots, 2c-1\}$, then test whether $\Phi_k(k) \leq m_x$. If it is not, then set $P^{c_0}(\tau_x) = \langle k, c_0 \rangle$. If it is, then distinguish the following cases.

(B1) If $c_0 \in (\text{cod}_{3c-d}^c)^{-1}(\text{mod}_{3c-d}^c(\varphi_k(k)))$, and if $\langle k, c_0 \rangle$ is consistent with $\tau_x$, then output $\langle k, c_0 \rangle$. Otherwise output $\langle k, i \rangle$ with $i$ being the least natural number $\leq c-1$ such that $\langle k, i \rangle$ is consistent with $\tau_x$.

(B2) If $c_0 \notin (\text{cod}_{3c-d}^c)^{-1}(\text{mod}_{3c-d}^c(\varphi_k(k)))$, then output $\langle k, i \rangle$ where $i$ is the least natural number $\leq c-1$ with $i \in (\text{cod}_{3c-d}^c)^{-1}(\text{mod}_{3c-d}^c(\varphi_k(k)))$.

(C) If $c_0 \in \{2c, \ldots, d-1\}$, then test whether $\Phi_k(k) \leq m_x$.

If not, then output $\bot$. If it is, then output $\langle k, j \rangle$ where $j$ is the least natural number in $(\text{cod}_{3c-d}^c)^{-1}(\text{mod}_{3c-d}^c(\varphi_k(k)))$ provided $\langle k, j \rangle$ is consistent with $\tau_x$. If $\langle k, j \rangle$ is not consistent with $\tau_x$, then output $\langle k, i \rangle$ where $i$ is the least natural number $\leq c-1$ with $i \notin (\text{cod}_{3c-d}^c)^{-1}(\text{mod}_{3c-d}^c(\varphi_k(k)))$.

end

Obviously, $P \ \text{EMON}_{\text{prob}}(2c/d)$-identifies $\mathcal{L}^{2c/d}$.

It remains to show that $\mathcal{L}^{2c/d} \notin \text{EMON}_{\text{prob}}(p)$ for all $p > 2c/d$. Assume the converse. Let $\epsilon > 0$ with $\mathcal{L}^{2c/d} \in \text{EMON}_{\text{prob}}((2c/d) + \epsilon)$. Let $P$ be the
identifying PIM. We define a recursive procedure \( I \) computing for each \( k \in \mathbb{N} \) a natural number \( j \in \{0, \ldots, (\frac{c}{3c-d}) - 1\} \) such that \( j \neq \text{mod}_{3c-d}(\varphi_k(k)) \). Let \( k \) be a natural number. Recall that \( L_k \in \mathcal{L}^{2c/d} \).

In Stage \( n \), \( I \) works as follows:

If \( \Phi_k(k) \leq n - 1 \), then define \( I(k) \) to be the least \( \ell \in \{0, \ldots, (\frac{c}{3c-d}) - 1\} \) with \( \ell \neq \text{mod}_{3c-d}(\varphi_k(k)) \). Otherwise compute \( T_{P, \tau_k} \) and define for all \( j \in \mathbb{N} \), \( j \leq c - 1 \),

\[
p^n_j = \text{Pr}\{ C \mid P^{c^n}(\tau^n_k) = \langle k, j \rangle \}.
\]

Test if \( \sum_{j=0}^{c-1} p^n_j > 2c/d \). If not, then request another input and go to stage \( n + 1 \). If it is, then let \( S = \{j_1, \ldots, j_{3c-d}\} \) be a subset of \( \{0, \ldots, c - 1\} \) of cardinality \( 3c - d \), and set

\[
m_S = \sum_{s=1}^{3c-d} p^n_{j_s}.
\]

Define \( S_j := (\text{cod}_{2c-d})^{-1}(j) \) for \( j \in \{0, \ldots, (\frac{c}{2c-d}) - 1\} \), and

\[
I(k) := \text{the least } \ell \in \{0, \ldots, \left(\frac{c}{2c-d}\right) - 1\} \text{ with } m_{S_\ell} < \frac{2(3c - d)}{d}.
\]

end

Now we have to prove that the procedure \( I \) fulfills the desired conditions.

Claim 1 The procedure \( I \) terminates for each \( k \in \mathbb{N} \).

Proof (of the claim). Let \( k \) be a natural number.

1. \( p^n_j \) is computable for all \( j, n \in \mathbb{N}, \ j \leq c - 1 \), and there exists an \( n \in \mathbb{N} \) with \( \sum_{j=0}^{c-1} p^n_j > 2c/d \), since \( \mathcal{L}^{2c/d} \) is assumed to be in \( \text{EMON}_{\text{prob}}((2c/d) + \epsilon) \).

2. Let \( n \in \mathbb{N} \) with \( \sum_{j=0}^{c-1} p^n_j > 2c/d \). If \( \Phi_k(k) \leq n \), then \( I \) stops and outputs a natural number. Otherwise \( \Phi_k(k) \leq n \). In this case, we can show by using a combinatorical argument (cf. Theorem 2.1.1) that there exists a subset \( S = \{j_1, \ldots, j_{3c-d}\} \) of \( \{0, \ldots, c - 1\} \) with \( m_S < 2(3c - d)/d \).
Consequently, \( \mathcal{I} \) is effective and outputs the code number of a subset of \( \{0, \ldots, c-1\} \) of cardinality \( 3c - d \) for all \( k \in \mathbb{N} \). \( \text{end} \) (of Proof).

Claim 2: Let \( k \in \mathbb{N} \) with \( \varphi_k(k) \downarrow \). Then

\[ \mathcal{I}(k) \neq \mod^c_{3c-d}(\varphi_k(k)). \]

Proof (of the claim). Let \( k \in \mathbb{N} \) with \( \varphi_k(k) \downarrow \). Let \( n \in \mathbb{N} \) such that \( \mathcal{I} \) stops in Stage \( n \). Then either \( \Phi_k(k) \leq n - 1 \), and thus \( \mathcal{I}(k) \neq \mod^c_{3c-d}(\varphi_k(k)) \) by definition of \( \mathcal{I} \), or \( \Phi_k(k) > n - 1 \). In the latter case,

\[ \tau^k_n = (\tau^L_k)_n \quad \text{and} \quad \mathcal{I}(k) = \ell \text{ with } m_{(\text{cod}^L_{3c-d})^{-1}(\ell)} < \frac{2(3c-d)}{d}. \]

Let \( \{L_{(k,j_1)}, \ldots, L_{(k,j_{3c-d})}\} \) be the set of the finite languages, i.e., \( L_{(k,j_i)} = L'_k \) for all \( i \in \{1, \ldots, 3c - d\} \). Set \( S = \{j_1, \ldots, j_{3c-d}\} \).

The first approximation to the probability of the set of all coin-oracles \( C \) such that \( P^c \) potentially \( \text{EMON} \)-converges correctly on the canonical text for \( L'_k \) is given by

\[ m_S + 1 - \frac{2c}{d} - z. \]

Since \( L'_k \) is assumed to be monotonically identifiable with a probability \( p \geq (2c/d) + \epsilon \), the probability of the set of all coin-oracles \( C \) such that \( \text{PC}^\mathbb{N}((\tau^L_k)_n) \) is an index for \( L_k \) and \( \text{PC}^\mathbb{N}((\tau^L_k)_x) \), \( x \in \mathbb{N} \), \( \text{EMON} \)-converges to an index for \( L'_k \) must be greater than \( (2c/d) - (m_S + 1 - (2c/d) - z) \), i.e., we need an additional probability \( > (2c/d) - (m_S + 1 - (2c/d) - z) \).

However, these paths are “lost” for \( L_k \), since monotonicity does not allow a change from an index for \( L_k \) to an index for \( L'_k \) and back to an index for \( L_k \). Let \( \max(L_k) \) be the probability of the set of all coin-oracles \( C \) such that \( P^c \) \( \text{EMON} \)-converges correctly on \( \tau^k \). Then

\[ \max(L_k) < 1 - \left( \frac{2c}{d} - (m_S + 1 - \frac{2c}{d} - z) \right). \]

Since \( L_k \) is assumed to be monotonically identifiable with a probability \( p > 2c/d \), we can conclude that

\[ m_S > \frac{2(3c-d)}{d}. \]

Consequently, \( \mathcal{I}(k) \neq \mod^c_{3c-d}(\varphi_k(k)) \). \( \text{end} \) (of Proof).

However, there is a \( k_0 \in \mathbb{N} \) with \( \mathcal{I}(k_0) = \mod^c_{3c-d}(\varphi_{k_0}(k_0)) \), a contradiction. \( \diamond \)

53
2.1.3 Proper Probabilistic Learning with probability $p = 1$

When observing previous results in the field of probabilistic function or language identification, we notice that in all but one of the investigated learning models, namely $BC$-identification with respect to nonstandard hypothesis spaces (cf. [107]), the learning power is not increased when the machines are required to learn with probability $p = 1$. However, it seems to be a reasonable property of a learning model that there exist learning problems which strictly separate deterministic learning from probabilistic learning. It is easy to see that proper strong-monotonic and proper conservative learning do not fulfill this requirement, i.e., $ESMON_{\text{prob}}(1) = ESMON$, and $ECOV_{\text{prob}}(1) = ECOV$. In the case of conservative probabilistic learning with $p = 1$, this result follows from the fact that an infinite computation tree induced by a PIM and a text for a language $L$ may not contain an overgeneralization $L'$ of $L$, since otherwise $L'$ would not be conservatively identifiable with probability $p = 1$. In the case of strong-monotonic probabilistic learning with $p = 1$, all paths in an infinite computation tree are $SMON$-converging correctly. For monotonic probabilistic learning, the situation is different. In this case, we are able to show that $EMON_{\text{prob}}(1) \neq EMON$.

Separation results in the field of learning of indexed families are usually proved by constructing indexed families encoding the halting problem in an appropriate way (cf., e.g., [65, 109] for an overview). For probabilistic learning under monotonicity constraints, we showed in the previous subsections that there exist indexed families separating the probabilistic learning classes in the interval $(1/2, 1]$ which encode problems being weaker than the halting problem. In the case of monotonic probabilistic learning with $p = 1$, we use a sophisticated version of this technique.

**Theorem 2.1.8.** $EMON_{\text{prob}}(1) \setminus EMON \neq \emptyset$

**Proof.** Let $\Sigma := \{a, b, d\}$. Let $\langle \cdot, \cdot \rangle : \mathbb{N} \times \mathbb{N} \times \{0, 1\} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N} \times \{0, 1\}$. We define the following abbreviations. Let $m \in \mathbb{N}$ be a number.

If $\varphi_k(i) \downarrow$ for all $i \leq m$, then set

$$N_m^{-1} := \{ a^k b^n | n \leq \sum_{i=0}^{m-1} (\Phi_k(i) + 2) + \Phi_k(m) - 1 \}. $$

54
For $s \in \mathbb{N}$, set

$$N^s_m := \{a^kb^n|n \leq \sum_{i=0}^{m-1} (\Phi_k(i) + 2) + \Phi_k(m) + s\}.$$ 

Furthermore set $h_m := \sum_{i=0}^{m-1} (\Phi_k(i) + 2) + \Phi_k(m)$.

If $\varphi_k(i) \uparrow$ for an $i \leq m$, then set $N^{-1}_m := L_k$, and $N^s_m := L_k$ for all $s \in \mathbb{N}$.

The indexed family $\mathcal{L}^1$ to be defined has the following properties. Let $k$ be a natural number. For every $j \in \mathbb{N}$, $\mathcal{L}^1$ contains a pair of languages $(L_{(\lfloor k,0,j \rfloor)}, L_{(\lfloor k,1,j \rfloor)})$. If $\varphi_k(k)$ is total, then there exists a $j_0 \in \mathbb{N}$ such that, for all $j \geq j_0$, exactly one language $L_{(\lfloor k,v,j \rfloor)}$ in the pair $(L_{(\lfloor k,0,j \rfloor)}, L_{(\lfloor k,1,j \rfloor)})$ is equal to $L_k$. The other language is the union of a finite subset of $L_k$ and a string of the form $d^m$. If $\varphi_k(k)$ is not total, then there exists a $j_0 \in \mathbb{N}$ such that $L_{(\lfloor k,v,j \rfloor)} = L_k$ for all $j \geq j_0$ and $v \in \{0,1\}$. In particular, $\mathcal{L}^1$ contains $L_k$ for all $k \in \mathbb{N}$. Furthermore, the following conditions are fulfilled.

1. Let $m \in \mathbb{N}$. Assume that $L_{(\lfloor k,v,j \rfloor)}, j \in \mathbb{N}, v \in \{0,1\}$, is not consistent with $a^kb^m$. Let $j' > j$ be the smallest natural number with $a^kb^m \in L_{(\lfloor k,0,j' \rfloor)} \cap L_{(\lfloor k,1,j' \rfloor)}$. Then $L_{(\lfloor k,v,j \rfloor)} \cap L_k \subseteq L_{(\lfloor k,v,j' \rfloor)}$ for every $v \in \{0,1\}$.

2. For $j \in \mathbb{N}$, it is not decidable which language in $(L_{(\lfloor k,0,j \rfloor)}, L_{(\lfloor k,1,j \rfloor)})$ is equal to $L_k$.

3. There may be a pair $(L_{(\lfloor k,0,j \rfloor)}, L_{(\lfloor k,1,j \rfloor)})$ in $\mathcal{L}^1$ such that $L_{(\lfloor k,0,j \rfloor)}$ and $L_{(\lfloor k,1,j \rfloor)}$ are not comparable with respect to $L_k$, i.e., there exist $w_1, w_2 \in L_k$ with $w_1 \in L_{(\lfloor k,0,j \rfloor)} \setminus L_{(\lfloor k,1,j \rfloor)}$, and $w_2 \in L_{(\lfloor k,1,j \rfloor)} \setminus L_{(\lfloor k,0,j \rfloor)}$. It is not decidable whether such a pair of languages does exist or not.

The condition 1. guarantees monotonicity on every text $\tau$ for a language $L$ in $\mathcal{L}^1$. The conditions 2. and 3. guarantee that, for every every HIM $M$, there exists a $k_0 \in \mathbb{N}$ such that $M$ diverges on the canonical text for $L_{k_0}$.

Now we are ready to define $\mathcal{L}^1$ more formally. Let $k$ be a natural number. For convenience, we assume that for all $j \in \mathbb{N}$, $\Phi_k(j) \geq 3$. This assumption is needed to simplify the construction and may be eliminated easily.

1. If $\varphi_k(0) \uparrow$, then set $L_{(\lfloor k,v,j \rfloor)} := L_k$ for all $j \in \mathbb{N}$, and $v \in \{0,1\}$. 

55
(2) Assume $\varphi_k(0) \downarrow$. Distinguish the following cases.

(A) There exists an $l \in \mathbb{N}$ such that

(A1) $\varphi_k(j) \downarrow$ for all $j \leq l$,

(A2) $\varphi_k(l) < l$.

(B) There exists no such $l \in \mathbb{N}$.

If (B) holds, then define the languages $L_{(k,v,j)}$, $j \in \mathbb{N}$, $v \in \{0,1\}$, as follows. Let $j$ be a natural number.

If $\varphi_k(m) \uparrow$ for an $m \leq j$, then set $L_{(k,v,j)} := L_k$ for every $v \in \{0,1\}$. If $\varphi_k(m) \downarrow$ for all $m \leq j$, then distinguish the following cases.

If $\varphi_k(j) \equiv 0 \mod 2$, then set

- $L_{(k,0,j)} := N_j^\neg \cup \{d^{(j,0,h_j)}\}$, and
- $L_{(k,1,j)} := L_k$.

If $\varphi_k(j) \equiv 1 \mod 2$, then set

- $L_{(k,0,j)} := L_k$.
- $L_{(k,1,j)} := N_j^\neg \cup \{d^{(j,1,h_j)}\}$.

If (A) holds, then let $i_0$ be the smallest natural number with the properties (A1) and (A2). Define $L_{(k,v,j)}$, $v \in \{0,1\}$, as follows. Let $j$ be a natural number.

(2.1) If $j < \varphi_k(i_0)$, then distinguish the following cases.

If $\varphi_k(j) \equiv 0 \mod 2$, then set

- $L_{(k,0,j)} := N_j^\neg \cup \{d^{(j,0,h_j)}\}$, and
- $L_{(k,1,j)} := N_{i_0}^\neg \cup \{d^{(j,1,h_{i_0})}\}$.

If $\varphi_k(j) \equiv 1 \mod 2$, then set

- $L_{(k,0,j)} := N_{i_0}^\neg \cup \{d^{(j,0,h_{i_0})}\}$,
- and $L_{(k,1,j)} := N_j^\neg \cup \{d^{(j,1,h_j)}\}$.

(2.2) If $j = \varphi_k(i_0)$, then distinguish the following cases. Remark that $\varphi_k(i_0) < i_0$

If $\varphi_k(j) \equiv 0 \mod 2$, then set
C∈{number. Let

We define a 2-sided PIM as follows. Let

Proof (of the claim).

Claim 1

In order to prove that

Set

τ

Let

If

If

If

If

If

If

If

Set

\( L^1 = (L_{(k,v,j)})_{k,v,j \in \mathbb{N}, v \in \{0,1\}} \). Clearly, \( L^1 \) is an indexed family.

In order to prove that \( L^1 \) fulfills the desired separation, we prove the following claims.

Claim 1: \( L^1 \) is properly monotonically identifiable with probability \( p = 1 \).

**Proof (of the claim).** We define a 2-sided PIM as follows. Let \( L \in \text{range}(\mathcal{L}) \), let \( \tau \in \text{text}(L) \), and let \( k \in \mathbb{N} \) with \( \text{range}(\tau) \cap L_k \neq \emptyset \). Let \( x \) be a natural number. Let \( C \in \{0,1\}^\infty \) be a coin-oracle.

PIM P: On input \( \tau_x \), \( P^C \) works as follows.

(A) If \( x = 0 \), then set \( P^{c_0}(\tau_0) := \langle k, c_0, 0 \rangle \).

(B) Let \( \tau_x = (\tau_{x-1}, d^{(j,v,z)}) \) for some \( j, v, z \in \mathbb{N}, v \in \{0,1\} \). If \( P^{c_{x-1}}(\tau_{x-1}) \) is consistent with \( \tau_x \), then output \( P^{c_{x-1}}(\tau_{x-1}) \). Otherwise output \( L_{(k,v,j)} \).

(C) Let \( \tau_x = (\tau_{x-1}, a^kb^n) \) for an \( n \in \mathbb{N} \). If \( P^{c_{x-1}}(\tau_{x-1}) \) is consistent with \( a^kb^n \), then output \( P^{c_{x-1}}(\tau_{x-1}) \). Otherwise let \( j_{x-1} \in \mathbb{N} \) with \( P^{c_{x-1}}(\tau_{x-1}) \in \{\langle k, j_{x-1}, 0 \rangle, \langle k, j_{x-1}, 1 \rangle\} \).

Otherwise there exists a number \( m \in \mathbb{N} \) with \( h_m \leq n \).\(^1\) Let \( r \) be maximal with this property. Distinguish the following cases.

\(^1\)If \( P^{c_{x-1}}(\tau_{x-1}) \) is not consistent with \( a^kb^n \), then \( L_{P^{c_{x-1}}(\tau_{x-1})} \) is a finite language, and the strings not contained in \( L_{P^{c_{x-1}}(\tau_{x-1})} \) are of the form \( a^lb^l \) where \( l \geq h_m \) for an \( m \in \mathbb{N} \).
If \( b^{h+1}, b^{h+2} \notin L_{pcx-1}(\tau_{x-1}) \), then search for the least \( j > j_{x-1} \) such that \( \text{range}(\tau_x) \subseteq L_{(k,0,j)} \cap L_{(k,1,j)} \), and set \( P_{Cx}^{\tau_x} := \langle k, c_x, j \rangle \).

(C2) Otherwise, either \( b^{h+1} \in L_{pcx-1}(\tau_{x-1}) \) or \( b^{h+2} \in L_{pcx-1}(\tau_{x-1}) \).
Then, by construction, \( r \) fulfills the conditions (A1) and (A2). Request new inputs until the actual text either contains a string of the form \( d^m \) or the set \( N_r \). In the first case output the unique language containing \( d^m \). In the second case search for the least \( j > j_{x-1} \) such that \( \text{range}(\tau_x) \subseteq L_{(k,0,j)} \cap L_{(k,1,j)} \), and set \( P_{Cx}^{\tau_x} := \langle k, c_x, j \rangle \).

Let \( L \in \mathbb{N} \), let \( \tau \) be a text for \( L \), let \( C \) be a coin-oracle, and let \( k \in \mathbb{N} \) with \( L \cap L_k \neq \emptyset \).

(+): If \( L = L_k \), then every path is monotonic, since \( P \) always chooses the new hypothesis among the indices containing more elements of \( L_k \). Remark that case (2.2) in the construction of \( L^1 \) does not disturb monotonicity because of the conditions (C1) and (C2) in the algorithm.

(+): If \( L \neq L_k \), then \( P \) eventually chooses a hypothesis \( \langle k, j_2, v_2 \rangle \) after a hypothesis \( \langle k, j_1, v_1 \rangle \) with \( j_2 \leq j_1 \). However, this happens if and only if the last element of the text seen so far is a string of the form \( d^m \). It is easy to see that this choice does not disturb monotonicity provided \( P \) never changes the hypothesis on this path again. But since the elements of the form \( d^m \) are not ambiguous, \( P \) identifies as soon as a such a string appears in the text.

Consequently, \( P \) works monotonically on every text. In order to prove that \( P \) identifies \( L^1 \) with probability \( p = 1 \), we only have to consider the languages \( L_k \) for \( k \in \mathbb{N} \), since every other language \( L \) is marked by a string \( d^m \) for an \( m \in \mathbb{N} \) which determines an index for \( L \), and we already argued that in this case every path converges. Considering \( L_k \) for \( k \in \mathbb{N} \), we have to distinguish the following two cases.

(+): First, let \( k \in \mathbb{N} \) with \( \varphi_k(i) \uparrow \) for an \( i \in \mathbb{N} \). Then every path in the corresponding computation tree converges correctly, since there is a \( j_0 \in \mathbb{N} \) such that \( L_{(k,v,j)} = L_k \) for all \( j \geq j_0, v \in \{0,1\} \).
(+ ) Assume $\varphi_k(i) \downarrow$ for all $i \in \mathbb{N}$. Obviously, the language $L_k$ appears in \{ $L_{i,j,0}$, $L_{i,j,1}$ \} for almost all $j \in \mathbb{N}$. Consequently, for every $L \in \mathcal{L}^1$, and every text $\tau$ for $L$, there is a level $n_{\tau,0}$ of $T_{P,\tau}$ such that the weight of all paths containing an index for $L_k$ up to level $n_{\tau,0}$ is at least $1/2$. For the remaining paths, we can conclude that there is a level $n_{\tau,1} \geq n_{\tau,0}$ such that the weight of all paths containing an index for $L_k$ up to level $n_{\tau,1}$ is at least $3/4$ and so on. Consequently, $P$ identifies $L_k$ with probability $p = 1$.

end (of Proof).

It remains to show that $\mathcal{L}^1$ is not properly monotonically identifiable. Assume the converse, and let $M$ be an IIM which $EMON$-identifies $\mathcal{L}^1$. We can prove the following claim.

**Claim 2**

There exists a $k_0 \in \mathbb{N}$ such that $M$ diverges on the canonical text for $L_{k_0}$.

**Proof (of the claim).** Intuitively, the claim follows from the fact that every deterministic machine $M$ has to choose its hypotheses among hypotheses in tuples of the form $(\langle k, 0, j \rangle, \langle k, 1, j \rangle)$, but $M$ is not allowed to change from a hypothesis $\langle k, 0, j \rangle$ to a hypothesis $\langle k, 1, j \rangle$ or vice versa if no string $d^m$ occurs, because if it would change in this way, the path possibly would not be monotonic for $L_k$, namely in the case where there exists an $l \in \mathbb{N}$, $l > j$ with $\varphi_k(l) \leq j$.

More exactly, we define a recursive function $F : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ as follows. Let $k \in \mathbb{N}$, and let $\tau^k$ be the canonical text for $L_k$. When fed $\tau^k$, $M$ outputs a sequence of hypotheses $\perp, \perp, \ldots, \perp, \langle k, v_0, j_0 \rangle, \langle k, v_1, j_1 \rangle, \ldots$ Remark, that $L_k$ is in $\mathcal{L}^1$. Thus, $M(\tau^k) \neq \perp$ for almost all $x \in \mathbb{N}$. Moreover, we may assume without loss of generality that the sequence is infinite.

1. Let $j \in \{0, \ldots, j_0\}$.
   - If $v_0 = 0$, then set $F(k, j) := 2^{j_0+1}$, and
   - if $v_0 = 1$, then set $F(k, j) := 3^{j_0+1}$.

2. Assume that $F$ is defined for all $j \in \{0, \ldots, j_x\}$ for an $x \geq 0$. If $j_{x+1} = j_x$ and $v_{x+1} = v_x$, then do nothing. If $j_{x+1} > j_x$, then define $F$ for $j \in \{j_x + 1, \ldots, j_{x+1}\}$ like follows.

59
If $v_{x+1} = 0$, then $F(k, j) := 2^{j_{x+1}+1}$.

If $v_{x+1} = 1$, then $F(k, j) := 3^{j_{x+1}+1}$.

If $j_{x+1} < j_x$ or $j_{x+1} = j_x$ and $v_{x+1} \neq v_x$, then define $F$ for $j \geq j_x + 1$ as follows.

$$F(k, j_{x+1}) := j_{x+1}, \text{ and } F(k, j) = 0 \text{ for all } j > j_x + 1.$$ 

It is easy to see that $F$ is recursive, since $M$ is recursive. By applying the Recursion Theorem, we obtain a $k_0 \in \mathbb{N}$ with $\varphi_{k_0}(i) = F(k_0, i)$ for all $i \in \mathbb{N}$. Remark that $\varphi_{k_0}$ is total. We show that $k_0$ fulfills the requirements from the claim. Distinguish the following cases.

(+) There exists an $i \in \mathbb{N}$ with $\varphi_{k_0}(i) < i$. By construction, this happens if and only if there exists an $x \in \mathbb{N}$ such that $M$ outputs a hypothesis $\langle k_0, v_{x+1}, j_{x+1} \rangle$ after a hypothesis $\langle k_0, v_x, j_x \rangle$ with $j_{x+1} < j_x$ or $j_{x+1} = j_x$ and $v_{x+1} \neq v_x$. In the first case, $\langle k_0, v_{x+1}, j_{x+1} \rangle$ describes a language which is either a proper subset or incompatible with the language described by $\langle k_0, v_x, j_x \rangle$ with respect to $L_{k_0}$. Hence, $M$ does not work monotonically on $(\alpha^{k_0}b^m)_{m \in \mathbb{N}}$. In the second case, $M$ does not work monotonically on $\tau^{k_0}$, since it changes between two hypotheses in a pair of languages $(L_{\langle k_0,0,j_x \rangle}, L_{\langle k_0,1,j_x \rangle})$ which are not compatible with respect to $L_{k_0}$.

(+) Suppose that, for all $i \in \mathbb{N}$, $\varphi_{k_0}(i) \geq i$. Then $M$ outputs a sequence of hypotheses $(\langle k_0, v_i, j_i \rangle)_{i \in \mathbb{N}}$ with $L_{k_0} \neq L_{\langle k_0,v_i,j_i \rangle}$ for all $i \in \mathbb{N}$.

Thus, $M$ does not EMON-identify $\tau$. end (of Proof).

This contradiction completes the proof of Theorem 2.1.8. ◇
monotonic learning with probability \( p > 1/2 \), respectively. An indexed family witnessing this claim can be defined as follows. Set \( L_0 := \mathbb{N} \), and set \( L_i := \{ n \mid n \leq i \} \). Then \( \mathcal{L} := (L_i)_{i \in \mathbb{N}} \) is properly strong-monotonically identifiable with \( p = 1/2 \), but not conservatively or monotonically identifiable with a probability \( p > 1/2 \) with respect to any class comprising hypothesis space. However, \( \mathcal{L} \) is not \( \text{LIM}-\text{identifiable} \).

In the following theorem, we show that there exists an indexed family being \( \text{LIM}-\text{identifiable} \) which witnesses the separation described above. In order to prove the result, we use the separation tool developed in Lange and Zeugmann (cf. [69]) can also be used to separate these learning classes.

**Theorem 2.2.1.** Let \( \mu \in \{ \text{COV, MON} \} \). Then \( \text{ESMON}_{\text{prob}}(1/2) \setminus \bigcup_{0 < \epsilon \leq 1/2} C\mu_{\text{prob}}(1/2 + \epsilon) \neq \emptyset \).

**PROOF.** Define \( \mathcal{L} = (L_{(k,j)})_{k,j \in \mathbb{N}} \) witnessing the desired separation as follows (cf. [69]). For \( k \in \mathbb{N} \), set \( L_{(k,0)} := L_k \). For \( j > 0 \) set

\[
L_{(k,j)} := \begin{cases} 
L_k & \text{if } \Phi_k(k) > j, \\
\{ a^k b^m \mid m \leq d \} & \text{if } \Phi_k(k) \leq j, \text{ where } d = 2\Phi_k(k) - j, \\
\{ a^k b^0 \} & \text{if } j > 2\Phi_k(k).
\end{cases}
\]

Obviously, \( \mathcal{L} \) is an indexed family with \( \mathcal{L} \in \text{ESMON}_{\text{prob}}(1/2) \).

First, let \( \mu = \text{COV} \). With an argument due to Lange and Zeugmann (cf. [69]), we show that \( \mathcal{L} \notin \text{CCOV}_{\text{prob}}((1/2) + \epsilon) \) for all \( \epsilon \in (0,1/2] \). Assume the converse, i.e., let \( P \) a PIM, and let \( \mathcal{G} \) be a hypothesis space such that \( P \text{CCOV}_{\text{prob}}(p) \)-identifies \( \mathcal{L} \) with a probability \( p > 1/2 \) with respect to \( \mathcal{G} \). Let \( \tau^k \) be the canonical text for \( L_k \), and let \( T_{P,\tau^k} \) be the corresponding infinite computation tree. Since \( L_{(k,0)} \) is assumed to be conservatively identifiable with probability \( p > 1/2 \), there exists an \( m_0 \in \mathbb{N} \) such that

\[
Pr(\{ \mathcal{C} \in \{0,1\}^\infty \mid \text{range}(\tau_{m_0+1}^k) \subseteq L(G_{Pc^{m_0}(\tau_{m_0}^k)}) \} > \frac{1}{2}.
\]

Then \( \varphi_k(k) \uparrow \) or \( \Phi_k(k) \leq m_0 \), since otherwise \( P \) could not identify the language \( \text{range}(\tau_{m_0}^k) \) conservatively. Thus, we could solve the halting problem. Consequently, \( \mathcal{L} \notin \bigcup_{0 < \epsilon \leq 1/2} \text{CCOV}_{\text{prob}}((1/2) + \epsilon) \).

For the monotonic case, let \( P \) be a PIM, and let \( \mathcal{G} \) be a hypothesis space such that \( P \text{CMON}_{\text{prob}}(p) \)-identifies \( \mathcal{L} \) with a probability \( p = 1/2 + \epsilon, \epsilon > 0 \),
with respect to $G$. Let $\tau^k$ be the canonical text for $L_k$, and let $T_{P,\tau^k}$ be the corresponding infinite computation tree. Let $k \in \mathbb{N}$ with $L \subseteq L_k$. Define for $m \in \mathbb{N}$:

$$C_m := \{C \in \{0, 1\}^\infty \mid \text{range}(\tau_m^k) \subseteq L(G_{pcm}^m(\rho_m^k))\}.$$ 

Since $L_{(k,0)} = L_k$ is assumed to be conservatively identifiable with probability $p = 1/2 + \epsilon$, there exists an $m_0 \in \mathbb{N}$ with

$$Pr(C_{m_0}) \geq \frac{1}{2} + \epsilon.$$ 

We show the following claim: if $\varphi_k(k) \downarrow$, then we can compute an $m_\epsilon$ such that $\Phi_k(k) \leq m_\epsilon$.

Assume that $\varphi_k(k) \downarrow$ and $\Phi_k(k) \geq m_0 + 1$. Then there exists a finite language in $\mathcal{L}$ containing $\text{range}(\tau_m^k)$ which is a proper subset of $L_k$. By assumption, $P$ monotonically identifies this language with $p = 1/2 + \epsilon$.

Since $\text{range}(\tau_m^k)$ belongs to $\mathcal{L}$, there exists a text $\rho$ for $\text{range}(\tau_m^k)$ with $\tau_m^k \sqsubseteq \rho$, and a level $m'_0 > m_0$ such that

$$Pr\{(C \in \{0, 1\}^\infty \mid \text{range}(\tau_m^k) = L(G_{pc}^{m'_0}(\rho_{m'_0}))\} \geq \frac{1}{2} + \epsilon.$$ 

Set

$$C_{m'_0} := \{C \in \{0, 1\}^\infty \mid \text{range}(\tau_m^k) = L(G_{pc}^{m'_0}(\rho_{m'_0}))\}.$$ 

Obviously, $Pr(C_{m_0} \cap C_{m'_0}) \geq \epsilon/2$. Let $\tau'$ be a text for $L_k$ with $\rho_{m'_0} \sqsubseteq \tau'$. Obviously, for every $C \in C_{m_0} \cap C_{m'_0}$, the path $(P^{C_x}(\tau'_x))_{x \in \mathbb{N}}$ is not monotonic for every language $L \in \mathcal{L}$ with $\text{range}(\tau_m^k) \subseteq L$, and hence not monotonic for $L_k$.

This construction can be repeated. Define $C_m$ as follows. Let $\tau^1$ be a text for $L_k$ with $\rho_{m_0} \sqsubseteq \tau^1$. Define for $m \in \mathbb{N}$:

$$C_m := \{C \in \{0, 1\}^\infty \mid \text{range}(\tau_m^1) \subseteq L(G_{pc}^{m}(\tau_m^1))\}.$$ 

Since $L_{(k,0)} = L_k$ is assumed to be conservatively identifiable with probability $p = 1/2 + \epsilon$, there exists an $m_1 > m_0 + 1 \in \mathbb{N}$ with

$$Pr(C_{m_1}) \geq \frac{1}{2} + \epsilon.$$ 


Assume that \( \varphi_k(k) \downarrow \) and \( \Phi_k(k) \geq m_1 + 1 \). Then there exists a finite language in \( \mathcal{L} \) containing \( \text{range}(\tau_{m_1}^1) \) which is a proper subset of \( L_k \). By assumption, \( P \) monotonically identifies this language with \( p = 1/2 + \epsilon \).

Let \( \rho \) be a text for \( \text{range}(\tau_{m_1}^1) \) with \( \tau_{m_1}^1 \subseteq \rho \). For \( s \in \mathbb{N} \), \( s \geq m_1 \), define \( C_s \) as follows. Let \( C \) be a coin-oracle. Then \( C \in C_s \) if an only if

1. \( \text{range}(\tau_{m_1}^1) = L_G(C_p\rho) \), and
2. \( (P^c\rho)_{x \leq s} \) is monotonic for \( \text{range}(\tau_{m_1}^1) \).

Since \( \text{range}(\tau_{m_1}^1) \) belongs to \( \mathcal{L} \), there exists a text \( \rho \) for \( \text{range}(\tau_{m_1}^1) \) with \( \tau_{m_1}^1 \subseteq \rho \), and a level \( m'_1 > m_1 \) with \( \Pr(C_{m'_1}) > 1/2 \). Then \( \Pr(C_{m_1} \cap C_{m'_1}) \geq \epsilon/2 \).

Let \( \tau' \) be a text for \( L_k \) with \( \rho \subseteq \tau_{m'_1}^1 \). Obviously, for every \( C \in C_{m_1} \cap C_{m'_1} \), the path \( (P^c\rho)_{x \in \mathbb{N}} \) is not monotonic for every language \( L \in \mathcal{L} \) with \( \text{range}(\tau_{m_1}^1) \subseteq L \), and hence not monotonic for \( L_k \).

Obviously, \( (C_{m_0} \cap C_{m'_0}) \cap (C_{m_1} \cap C_{m'_1}) \) is empty, since all oracles in \( C_{m_0} \cap C_{m'_0} \) induce paths which are not monotonic for all proper superset from \( \text{range}(\tau_{m_1}^1) \), and hence not monotonic for \( \text{range}(\tau_{m_1}^1) \).

Now it is obvious how to complete the proof. We repeat the construction above \( \lfloor \epsilon^{-1} \rfloor \)-times. At the end of the construction, we have constructed an \( m'_i \), a text \( \tau' \) for \( L_k \), and sets \( (C_{m_i} \cap C_{m'_i})_{i \leq \lfloor \epsilon^{-1} \rfloor} \) such that the following holds:

1. \( \Pr(C_{m_i} \cap C_{m'_i}) \leq \frac{\epsilon}{2} \).
2. \( (C_{m_i} \cap C_{m'_i}) \) and \( (C_{m_j} \cap C_{m'_j}) \) are disjoint for all \( i, j \leq \lfloor \epsilon^{-1} \rfloor \), \( i \neq j \).
3. For every \( i \leq \lfloor \epsilon^{-1} \rfloor \), and for every \( C \in C_{m_i} \cap C_{m'_i} \), the path \( (P^c\rho)_{x \in \mathbb{N}} \) is not monotonic for \( L_k \).

Set \( C_\epsilon := \bigcup_{i \leq \lfloor \epsilon^{-1} \rfloor} (C_{m_i} \cap C_{m'_i}) \). Then \( \Pr(C_\epsilon) \geq 1/2 \). Hence, from (3) follows that \( L_k \) is not monotonically identifiable with a probability \( p > 1/2 \), a contradiction. Thus \( \varphi_k(k) \uparrow \) or \( \varphi_k(k) \downarrow \) and \( \varphi_k(k) < m_\epsilon \). This completes the proof of the theorem. \( \diamond \)

By using a refined version of this separation technique, we are able to show that the probabilistic hierarchy in the case of class preserving conservative probabilistic learning is dense in the interval \( (1/2, 1] \).

**Theorem 2.2.2.** Let \( c, d \in \mathbb{N} \), \( 1 \leq c < d \). Then

\[
ECOV_{\text{prob}}^c \left( \frac{d}{d + c} \right) \bigcup_{0 < \epsilon \leq 1 - \frac{d}{d + c}} \text{COV}_{\text{prob}}^c \left( \frac{d}{d + c} + \epsilon \right) \neq \emptyset.
\]
PROOF. Let $\Sigma := \{a, b\}$. Let $c, d \in \mathbb{N}$ with $c < d$. We define an indexed family $\mathcal{L}^{c,d} \in ECOV^{st}_{\text{prob}}(\frac{d}{d+c})$ such that $\mathcal{L}^{c,d} \notin COV_{\text{prob}}(q)$ for all $q > d/(d+c)$. Let $\langle \cdot, \cdot \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$. Let $(D_i)_{i \in \mathbb{N}, i \leq \binom{d}{c}}$ be an enumeration of all subsets of $\{0, \ldots, d-1\}$ of cardinality $c$.

Let $k$ be natural number. If $\varphi_k(k) \downarrow$, then set

$$g_k := \Phi_k(k) + \varphi_k(k) + 1.$$  

Now let $j \in \mathbb{N}$ be a number, and let $z, r \in \mathbb{N}$, $r \leq \binom{d}{c} - 1$, with $j = \binom{d}{c} z + r$.

(1) If $\varphi_k(k) \downarrow$, and $z \leq \Phi_k(k) - 1$, then set

$$L_{\langle k, j \rangle} := \{a^k b^n | n \leq g_k\} \cup \{a^k b^{g_k+1} | v \in D_r\},$$

(2) if $\varphi_k(k) \downarrow$, $\Phi_k(k) \leq z < 2\Phi_k(k)$, and $r \leq d - 1$, then set

$$L_{\langle k, j \rangle} := \{a^k b^m | m \leq \varphi_k(k) + 1\} \cup \{a^k b^{g_k+r+1}\},$$

(3) If $\varphi_k(k) \downarrow$, and $2\Phi_k(k) \leq z$, or if $\varphi_k(k) \uparrow$, then set $L_{\langle k, j \rangle} = L_k$.

Then $\mathcal{L}^{c,d} = (L_{\langle k, j \rangle})_{k,j \in \mathbb{N}}$ is an indexed family which is properly conservatively identifiable with probability $d/(d+c)$. Define a $(2\binom{d}{c} - \binom{d-1}{c-1})$-sided PIM $P$ as follows. Let $L \in \text{range}(\mathcal{L}^{c,d})$ be a language, and let $\tau \in \text{text}(L)$. Let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $\mathcal{C}$ be a coin-oracle. Let $x$ be a natural number.

PIM $P$: On input $\tau_x$, $P^c$ works as follows.

If $x = 0$, then distinguish the following cases.

(A1) If $c_0 \leq \binom{d}{c} - 1$, then set $P^{c_0}(\tau_0) := \langle k, c_0 \rangle$.

(A2) If $c_0 > \binom{d}{c} - 1$, then set $P^{c_0}(\tau_0) := \bot$.

If $x > 0$, then test whether $\text{range}(\tau_x) \subseteq L'_k$. If yes, then $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$. Otherwise distinguish the following cases.

(B1) If $P^{c_{x-1}}(\tau_{x-1})$ is consistent with $\tau_x$, then set $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$.  

64
(B2) If $P^{c,i}(\tau_{x-1})$ is not consistent with $\tau_x$, then $P^c$ tests whether there are numbers $z, r \in \mathbb{N}$, $\Phi_k(k) \leq z < 2\Phi_k(k)$, $r \leq d - 1$, with \( \text{range}(\tau_x) \subseteq L_{(k_i^j)^zd+r} \). If yes, then set $P^{c,i}(\tau_x) := (k, (d^j)z + r)$. If there are no such numbers $r$ and $z$, then set $P^{c,i}(\tau_x) := (k, j)$ where $j$ is the smallest natural number such that \( \text{range}(\tau_x) \subseteq L_{(k,j)} \).

Request a new input string.

end

It remains to show that $P$ ECOV$_{prob}(d/(d + c))$-identifies $L^{c,d}$. Let $k$ be natural number. If $\varphi(k) \uparrow$, then all languages $L_{(k,j)}$, $j \in \mathbb{N}$, in $L^{c,d}$ are equal to $L_k$, and $P$ identifies $L_k$ with probability $p = \binom{d}{c}/(2\binom{d}{c} - \binom{d}{c-1}) = d/(d + c)$. If $\varphi(k) \downarrow$, then, for every $z, z', r \in \mathbb{N}$, $z' < \Phi_k(k)$, $\Phi_k(k) \leq z < 2\Phi_k(k)$, and $r \leq n - 1$, the language $L_{(k,(d^j)z+r)}$ has exactly $\binom{d}{c} - \binom{d}{c-1}$ overgeneralizations in $\{L_{(k,(d^j)z')}, \ldots, L_{(k,(d^j)z+r-1)}\}$. Thus, the weight the set of all conservative paths for the languages $L_{(k,j)} = \{a^kb^m | m \leq \varphi_k(k) + 1\} \cup \{a^{d^j}b^{\varphi_k(k)+1}\}$ is exactly $d/(d + c)$. Obviously, the languages $L_{(k,j)} = \{a^kb^m | m \leq g_k\} \cup \{a^k(b^\alpha v) | v \in D_r\}$, $j = \binom{d}{c}z + r, z \leq \Phi_k(k) - 1, r \leq \binom{d}{c} - 1$, are conservatively identifiable with probability $p = d/(d + c)$. Thus, $L^{c,d}$ is properly conservatively identifiable with probability $p = d/(d + c)$.

In order to prove that $L^{c,d}$ is not conservatively identifiable with a probability higher than $d/(d + c)$, we use the separation technique sketched in Theorem 2.2.1 and the proof technique developed in Theorem 2.1.1. Let $P$ be a two-sided PIM, and let $G$ be a class preserving hypothesis space such that $P$ identifies $L^{c,d}$ conservatively with a probability $p > d/(d + c)$ with respect to $G$. We show the following claim.

Claim

There is an total recursive function $I$ such that $I(k) \neq \varphi_k(k)$ for all $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$.

Proof (of the claim). For $k, m \in \mathbb{N}$ set

$$C^m := \{C \in \{0, 1\}^\infty | \text{range}(\tau_{m+2}^k) \subseteq L(G_{P^c(m+2)}(r_k))\}.$$ 

Since $L_k \in L^{c,d}$, there exists a least natural number $m$ such that

$$Pr(C^m) = \frac{d}{d + c} + \epsilon,$$ 

where

$$\epsilon = \min_{t > 1} \left(\frac{1}{2} - \frac{1}{2t}\right).$$
for an $\epsilon \in (0, 1 - d/(d + c)]$. Define
\[ \mathcal{I}(k) := m_k. \]
Suppose there is a $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$, and $m_k = \mathcal{I}(k) = \varphi_k(k)$. Then for every $r \in \{0, \ldots, d - 1\}$, there exist languages $L_{(k,j,r)} \in \mathcal{L}_{c,d}$ with
\[ L_{(k,j,r)} = \{ a^k b^m \mid m \leq m_k + 1 \} \cup \{ a^k b^{g_k+r+1} \}. \]
It is easy to compute an index for the language $L_{(k,j,r)}$. However, for the sake of readability, we write $L_{(k,j,r)}$ throughout the ongoing proof.

Let $\mathcal{C} \in \mathcal{C}^{m_k}$ be a coin-oracle. Set $h_{\mathcal{C},m_k} := P_{\mathcal{C}^{m_k}}(\tau_{m_k}^k)$. Then $L(G_{h_{\mathcal{C},m_k}})$ is a member of $\text{range}(\mathcal{L})$, since $\mathcal{G}$ is assumed to be class preserving. Moreover,
\[ L(G_{h_{\mathcal{C},m_k}}) \neq L_{(k,j,r)} \quad \text{for all} \quad r \leq d - 1, \]
since $\text{range}(\tau_{m_k}^k) \subseteq L(G_{h_{\mathcal{C},m_k}})$, and $a^k b^{m_k+2} \notin L_{(k,j,r)}$ for all $r \leq d - 1.\footnote{This follows from the fact that $g_k+r+1 = \Phi_k(k) + \varphi_k(k)+1+r+1 > \varphi_k(k)+2 = m_k+2$ for all $r \leq d - 1.$}$
Hence, $L(G_{h_{\mathcal{C},m_k}})$ is of the form $\{ a^k b^n \mid n \leq g_k \} \cup \{ a^k b^{g_k+v} \mid v \in D_r \}$ for an $r \leq {d \choose c} - 1$. Now define for every $r \leq {d \choose c} - 1$:
\[ C_{D_r} := \{ \mathcal{C} \in \mathcal{C}^{m_k} \mid L(G_{h_{\mathcal{C},m_k}}) = \{ a^k b^m \mid m \leq g_k \} \cup \{ a^k b^{g_k+r+1} \mid r \in D_r \} \} \]
Moreover set
\[ C_d := \{ \mathcal{C} \in \mathcal{C}^{m_0} \mid L(G_{h_{\mathcal{C},m_k}}) = L_k \}. \]
Clearly, the sets $C_{D_r}, r \leq {d \choose c} - 1$, and $C_d$ are pairwise disjoint. Furthermore, their union is equal to $\mathcal{C}^{m_k}$.

Now we compute the weight of the paths which are conservative for the languages $L_{(k,j,r)}, r \leq d - 1$. Let $r \leq d - 1$ be a natural number. Let $\mathcal{C}$ be a coin-oracle. Then $L(G_{h_{\mathcal{C},m_k}})$ is an overgeneralization of $L_{(k,j,r)}$ if and only if $\mathcal{C} \in C_{D_i}$ with $r \in D_i$, or $\mathcal{C} \in C_d$. In these cases, the finite sequence $(P_{\mathcal{C}^r}(\tau_{m_k}^k))_{x \leq m_k}$ cannot be extended to a path which $\text{ECOV}$-converges to an index for $L_{(k,j,r)}$. Hence, for every $r \leq d - 1$, the weight of conservative paths in $T_{P_{r_{m_k}}}$ for $L_{(k,j,r)}$ is at most
\[ \sum_{i \leq {d \choose c} - 1, r \notin D_i} Pr(C_{D_i}) + \left( 1 - \frac{d}{d+c} - \epsilon \right). \]
By assumption, \( L_{(k,j,r)} \) is conservatively identifiable with probability \( p > d/(d + c) \). Thus, it follows that

\[
\sum_{i \leq \binom{d}{c} - 1, r \in D_i} Pr(C_{D_i}) + \left(1 - \frac{d}{d + c} - \epsilon\right) > \frac{d}{d + c}.
\]

Since every set \( D_i, i \leq \binom{d}{c} - 1, \) has cardinality \( c \), the paths \( \langle PC^x(\tau_k^x)\rangle_{x \leq m_k}, C \in C_{D_i} \), are conservative for exactly \( (d - c) \) many languages \( L_{(k,j,r)} \), \( r \leq d - 1 \), is at most \( (d - c)(\sum_{r \leq \binom{d}{c} - 1} Pr(C_{D_r})) + d(1 - (d/(d + c) - v)). \) Hence,

\[
(d - c)(\sum_{r \leq \binom{d}{c} - 1} Pr(C_{D_r})) + d(1 - \frac{d}{d + c} - v) > d\left(\frac{d}{d + c}\right),
\]

a contradiction. Hence, our claim is proved. This completes the proof of the theorem. \( \ast \)

Since the set \( D = \{m \in [1/2, 1] \mid \exists c, d \in \mathbb{N}, 1 \leq c \leq d, \text{ with } m = d/(d + c) \} \) is dense in the interval \([1/2, 1]\), it follows from Theorem 2.2.2 that the probabilistic hierarchy in the case of class preserving conservative probabilistic learning with \( p \geq 1/2 \) is dense.

**Corollary 2.2.3.** \( \langle \text{COV}_{\text{prob}}(p) \rangle_{p \in [0, 1]} \) and \( \langle \text{COV}_{\text{st}}^{\text{prob}}(p) \rangle_{p \in [0, 1]} \) are dense in the interval \([1/2, 1]\).

### 2.2.2 Monotonic Probabilistic Learning with probability \( p \geq \frac{2}{3} \)

In the following subsection, we are concerned with class preserving monotonic probabilistic learning. Let \( D = \{p \in [0, 1] \mid p = 2d/(2d + c), 2d/(2d + 1) \geq 4/5\} \). We show that, for every \( p \in D \), the probabilistic hierarchy \( \langle \text{MON}_{\text{prob}}(p) \rangle_{p \in [0, 1]} \) contains a strictly decreasing sequence of probabilistic learning classes converging to \( \text{EMON}_{\text{prob}}(p) \). The same holds for \( D' = \{p \in [0, 1] \mid p = 2d/(2d + c), 2d/(2d + 1) \geq 2/3\} \) and \( \langle \text{MON}_{\text{st}}^{\text{prob}}(p) \rangle_{p \in [0, 1]} \) in the interval \([2/3, 1]\).
Theorem 2.2.4. Let $c,d \in \mathbb{N}$, $1 \leq c < d$ such that $c + 1$ is a factor of $d$, and $\frac{2d}{2d+c} \geq 4/5$. Then

$$\text{EMON}_{\text{prob}} \left( \frac{2d}{2d+c} \right) \cup \bigcup_{0 < c \leq 1 - \frac{2d}{2d+c}} \text{MON}_{\text{prob}} \left( \frac{2d}{2d+c} + \epsilon \right) \neq \emptyset.$$  

PROOF. Let $\Sigma := \{a, b\}$. Let $z \in \mathbb{N}$ with $d = z(c + 1)$. Notice that $c \leq z$, since $2d/(2d+c) \geq 4/5$.

Let

$$D^r = \{m \in \mathbb{N} \mid r(c + 1) \leq m < (r + 1)(c + 1)\}.$$  

Let $(D^r_i)_{i \leq \frac{c+1}{2} - 1}$ be an effective enumeration of all subsets of $D_r$ with cardinality 2. Let $\langle \ , \ \rangle : \mathbb{N} \times \{0, \ldots, d + z\left(\frac{c+1}{2}\right)\} \rightarrow \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \{0, \ldots, d + z\left(\frac{c+1}{2}\right)\}$. Let $k, j \in \mathbb{N}$, $j \leq (d + z\left(\frac{c+1}{2}\right)).$

1. If $\varphi_k(k) \uparrow$, then $L_{(k,j)} := L_k$ for all $j \leq d + z\left(\frac{c+1}{2}\right)$.

2. If $\varphi_k(k) \downarrow$, and $j \leq d - 1$, then $L_{(k,j)} := L'_k \cup \{a^k b^{\Phi_k(k) + (j+1)}\}.$

3. If $\varphi_k(k) \downarrow$, and $d \leq j \leq d + z\left(\frac{c+1}{2}\right) - 1$, then let $r \in \mathbb{N}$, $0 \leq r \leq z - 1$ with $d + r\left(\frac{c+1}{2}\right) \leq j < d + (r + 1)\left(\frac{c+1}{2}\right)$. Then set

$$L_{(k,j)} := L'_k \cup \{a^k b^{\Phi_k(k) + (v+1)} \mid v \in D^r_{j-(d+r\left(\frac{c+1}{2}\right))}\}.$$  

4. If $\varphi_k(k) \downarrow$, and $j = (d + z\left(\frac{c+1}{2}\right))$, then set $L_{(k,j)} := L_k$.

It follows that $\mathcal{L}^{c,d} := (L_{(k,j)})_{k,j \in \mathbb{N}, j \leq (d + z\left(\frac{c+1}{2}\right)) - 1}$ is an indexed family witnessing the desired separation. We start by showing that $\mathcal{L}^{c,d}$ is properly monotonically identifiable with probability $p = 2d/(2d+c)$. Define a $(2d+c)$-sided PIM $P$ as follows. Let $L \in \text{range}(\mathcal{L}^{c,d})$, and let $\tau \in \text{text}(L)$. Let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $\mathcal{C}$ be a coin-oracle. Let $x$ be a natural number.

PIM $P$: On input $\tau_x$, $P^c$ works as follows.

If $x = 0$, and if $c_0 \leq d - 1$, then set $P^{c_0}(\tau_0) := \langle k, c_0 \rangle$. If $d \leq c_0 \leq 2d - 1$, then set $P^{c_0}(\tau_0) := \langle k, c_0 - d \rangle$. If $c_0 \geq 2d$, then set $P^{c_0}(\tau_0) := \perp$.

If $x > 0$, then test whether $\text{range}(\tau_x) \subseteq L'_k$. In case it is, set $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$. Otherwise, the actual text $\tau_x$ contradicts at least $d - 2$ of the languages $L_{(k,j)}$, $j \leq d - 1$. Distinguish the following cases.

68
(A) If \( P^{c-1}(\tau_{x-1}) \) is consistent with \( \tau_x \), then set \( P^c(\tau_x) := P^{c-1}(\tau_{x-1}) \).

(B) If \( \tau_x \) contradicts every language \( L_{(k,j)} \), \( j \leq d-1 \), then let \( l \geq d \) be the smallest natural number such that \( range(\tau_x) \subseteq L_{(k,l)} \). Set \( P^c(\tau_x) := \langle langlek,l \rangle \).

(C) Assume that there exists a \( j \leq d-1 \) such that \( \langle k,j \rangle \) is consistent with \( \tau_x \). Let \( r \leq z-1 \) with \( r(c+1) \leq j < (r+1)(c+1) \). Distinguish the following cases.

(C1) If \( c_0 \geq d \), then set \( P^c(\tau_x) := \langle k,j \rangle \).

(C2) Assume \( c_0 \leq d-1 \). If \( c_0 = j \), or \( c_0 < r(c+1) \), or \( c_0 \geq (r+1)(c+1) \), then set \( P^c(\tau_x) := \langle k,j \rangle \).

If \( r(c+1) \leq c_0 < (r+1)(c+1) \), and \( c_0 \neq j \), then set \( P^c(\tau_x) := \langle k,c_0 \rangle \). Notice that in this case \( P^c(\tau_x) \neq \langle k,j \rangle \).

end

We have to show that \( P \) properly monotonically identifies \( \mathcal{L}^{c,d} \) with probability \( p = 2d/(2d + c) \). Let \( k \) be a natural number. If \( \varphi_k(k) \uparrow \), then all languages \( L_{(k,j)} \), \( j \leq d + z^{c+1} \), in \( \mathcal{L}^{c,d} \) are equal to \( L_k \), and \( P \) identifies \( L_k \) with probability \( p = 2d/(2d + c) \). Assume \( \varphi_k(k) \downarrow \). Then it follows from Condition (C2) that, for every \( j \leq d-1 \), there are exactly \( c \) paths which are not convergent on the canonical text for a language \( L_{(k,j)} \). Thus \( P \) monotonically identifies every language \( L_{(k,j)} \), \( j \leq d-1 \), with probability \( p = 2d/(2d + c) \). For \( j \geq d \), it follows from Condition (C1) that there are exactly \( c \) paths which are not convergent on the canonical text for \( L_{(k,j)} \). Thus, \( P \) \( EMN \)-identifies every language \( L_{(k,j)} \), \( j \geq d \), with probability \( p = 2d/(2d + d) \).

Finally, let \( P \) be a PIM which \( \text{MON} \)-identifies \( \mathcal{L}^{c,d} \) with a probability \( p > 2d/(d + c) \) with respect to a class preserving hypothesis space \( \mathcal{G} \). Let \( k \) be a natural number. Let \( \tau^k \) be the canonical text for \( L_k \). Define

\[
\mathcal{C}^m := \{ C \in \{0,1\}^\infty \mid range(\tau^k_{m+2}) \subseteq L(G^{pc^m(\tau^k)}) \}.
\]

Since \( L_k \in \mathcal{L}^{c,d} \), there exists a least natural number \( m_k \) such that

\[
P_F(\mathcal{C}^{m_k}) = \frac{2d}{2d + c} + \epsilon
\]

for an \( \epsilon > 0 \). In the following, we show that \( \Phi_k(k) \leq m_k \) or \( \varphi_k(k) \uparrow \). Assume that \( \varphi_k(k) \downarrow \) and \( \Phi_k(k) \geq m_k + 1 \). Then \( range(\tau^k_{m_k}) \subseteq L_k \). In the following,
we compute for every language $L$ in $\mathcal{L}^{c,d}$ the weight of the set of paths which are potentially monotonic and converging for $L$.

Let $j \leq d - 1$. Let $C_j$ be the set of coin-oracles defined as follows. Let $C$ be a coin-oracle. Then

$$C \in C_j \text{ if and only if } a^k b^{\Phi_k(k) + j + 1} \in L(G_{PC^x(r^k_m)}) .$$

Remind that $D^r = \{ m \in \mathbb{N} \mid r(c + 1) \leq m < (r + 1)(c + 1) \}$. Moreover, $D_r^i, i \leq (c+1) - 1$, was defined to be the sequence of all subsets of $D_r$ with cardinality 2.

Let $i, j \in \mathbb{N}, i, j \in D_r$ for an $r \in \mathbb{N}, i \neq j$. Let $C \in C_i$. Then the finite path $(PC^x(\tau^k_x))_{x \leq m_k}$ can either be extended to a monotonic and convergent path for $L(k,j)$ or to a monotonic and convergent path for a language $L(G_{i,j})$ which contains $a^k b^{\Phi_k(k) + i + 1}$ and $a^k b^{\Phi_k(k) + j + 1}$. Remark that we could give an index for the language $L(G_{i,j})$. However, for the sake of readability, we write $L(G_{i,j})$ throughout the ongoing proof.

Now let $j \leq d - 1$. In this case, we do not compute the probability of all paths which can be extended to a monotonic and converging path for $L(k,j)$, because, for every path which can be extended to monotonic path for $L(k,j)$, a PIM $P$ has to decide whether the path will be extended to a monotonic path for $L(k,j)$ or to a monotonic path for $L(G_{i,j})$. Consequently, the probability of the set of all coin-oracles $C$ such that $L(G_{PC^x(r^k_m)})$ will be extended to a monotonic and converging path for $L(k,j)$ can be computed as follows.

$$P_j := 1 - \sum_{i \in D_r, i \neq j} k_i \Pr(C_i) .$$

Thereby, $k_i \Pr(C_i)$ is the probability of all paths corresponding to an oracle $C \in C_i$ which will be extended to a monotonic and converging path for $L(G_{i,j})$. Notice that the probability $P_j$ is not computable. However, the probability exists and may be used for calculations.

Furthermore, $(1 - k_i) \Pr(C_i)$ is the probability of all paths corresponding to an oracle $C \in C_i$ which will be extended to a monotonic and converging path for $L(k,j)$. Thus, we can compute the probability of the set of all coin-oracles $C$ such that $L(G_{PC^x(r^k_m)})$ will be extended to a monotonic and converging path for $L(G_{i,j})$ as follows.

$$P_{i,j} := 1 - \sum_{i \in D_r, i \neq j} (1 - k_i) \Pr(C_i) .$$
Since the probabilities $P_j$ and $P_{i,j}$ must be greater than $2d/(2d + c)$, we can conclude that
\[
\sum_{r \leq z-1} \sum_{i,j \in D_r, i \neq j} (P_i + P_{i,j}) =
\sum_{r \leq z-1} \sum_{i,j \in D_r, i \neq j} \left(2 - \sum_{i \in D_r, i \neq j} Pr(C_i)\right) > 2d \left(\frac{2d}{2d + c}\right).
\]

We do not need to know the exact value of the parameters $k_i$, since the sum on the left side of the equation is equal to $2cd - zc \sum_{i=0}^{d-1} Pr(C_i)$. Thus
\[
2cd - zc \sum_{i=0}^{d-1} Pr(C_i) > 2cd \left(\frac{2d}{2d + c}\right),
\]
and hence,
\[
2d - z \sum_{i=0}^{d-1} Pr(C_i) > 2d \left(\frac{2d}{2d + c}\right),
\]
a contradiction, since $c \leq z$. \(\qed\)

Similarly, we can show the corresponding result for $\langle \text{MON}^{st}_{\text{prob}}(p) \rangle_{p \in [0,1]}$.

**Theorem 2.2.5.** Let $c, d \in \mathbb{N}$, $1 \leq c < d$ such that $c + 1$ is a factor of $d$. Then
\[
\text{EMON}^{st}_{\text{prob}} \left(\frac{2d}{2d + c}\right) \setminus \bigcup_{0 < \epsilon \leq 1 - \frac{2d}{2d + c}} \text{MON}^{st}_{\text{prob}} \left(\frac{2d}{2d + c} + \epsilon\right) \neq \emptyset.
\]

**PROOF.** Let $c, d \in \mathbb{N}$, $1 \leq c < d$ such that $c + 1$ is a factor of $d$. Define the indexed family $L^{c,d}$ as in the proof of Theorem 2.2.4. The proof of the theorem is a slight variant of the proof of Theorem 2.2.4. Thereby, we have to take into account that a PIM which $\text{MON}^{st}$-identifies $L^{c,d}$ may not change from a hypothesis of a language $L_{k,j}, j \leq D_i^r, r \leq z - 1, i \leq \frac{(c+1)}{2} - 1$, to a language $L_{k,j'}, j' \leq D_i^r, j' \neq j$. The separation result is proved as usual. \(\qed\)

From Theorem 2.2.4, we can conclude that the probabilistic hierarchy in the case of class preserving conservative probabilistic learning with $p \geq 1/2$ is strictly decreasing to 1.

**Corollary 2.2.6.**
(a) \( \langle MON_{\text{prob}}(p) \rangle_{p \in [0,1]} \) is strictly decreasing in the interval \( [\frac{4}{5}, 1] \).

(b) \( \langle MON_{\text{st}}^{\text{prob}}(p) \rangle_{p \in [0,1]} \) is strictly decreasing in the interval \( [\frac{2}{3}, 1] \).

By applying Theorem 2.2.4 and Theorem 2.2.5, we can derive the following corollary.

**Corollary 2.2.7.** Let \( n \in \mathbb{N}^+ \). Then

\[
MON_{\text{prob}} \left( \frac{4n}{4n+1} \right) \setminus MON_{\text{st}}^{\text{prob}}(p) \neq \emptyset \text{ for all } p \in \left( \frac{2n}{2n+1}, 1 \right).
\]

Finally, we note two results concerning the relation of class preserving monotonic and conservative probabilistic learning.

**Theorem 2.2.8.** There exists an indexed family \( \mathcal{L} \) which is properly conservatively identifiable but not monotonically identifiable with \( p > \frac{4}{5} \) with respect to any class preserving hypothesis space.

**PROOF.** The indexed family \( \mathcal{L}^{1,2} \) defined in Theorem 2.2.4 fulfills the requirement. \( \diamond \)

### 2.2.3 Strong-monotonic Probabilistic Learning

In this subsection, we show that \( \langle SMON_{\text{prob}}(p) \rangle_{p \in [0,1]} \) has a threshold beginning with \( \frac{2}{3} \). Intuitively, this result follows from the fact that strong-monotonicity puts some severe restrictions on the hypotheses produced by a PIM when fed a text for a language to be inferred.

**Theorem 2.2.9.** \( \langle SMON_{\text{prob}}(p) \rangle_{0 \leq p \leq 1} \) and \( \langle SMON_{\text{st}}^{\text{prob}}(p) \rangle_{0 \leq p \leq 1} \) have a threshold beginning with \( \frac{2}{3} \).

**PROOF.** Let \( \mathcal{L} \) be an indexed family such that \( \mathcal{L} \in \cup_{2/3 < p < 1} SMON_{\text{prob}}(p) \).

Let \( \mathcal{G} \) be a hypothesis space, and let \( P \) be a PIM such that \( P \ SMON_{\text{prob}}(p) \)-identifies \( \mathcal{L} \) with a probability \( p > \frac{2}{3} \) with respect to \( \mathcal{G} \). We now define an appropriate hypothesis space \( \mathcal{H} \). Let \( (s_j)_{j \in \mathbb{N}} \) be an effective enumeration of all finite sequences of strings \( s \) with range(\( s \)) \( \subset L \) for some \( L \in \text{range}(\mathcal{L}) \).

Let \( k \) be natural number. Let \( \mathcal{T}_{P,s_k} \) be the finite computation tree induced by \( P \) and \( s_k \), let \( l_k \) be the highest level of \( \mathcal{T}_{P,s_k} \), and set

\[
\mathcal{N}_k := \{ O \subseteq \text{level } l_k \mid \text{ind}(o) \neq \bot \text{ for all } o \in O, \text{ and } w(O) > \frac{2}{3} \}.
\]
Let \((O_j)_{j \in \mathbb{N}}\) be an effective enumeration of \(\bigcup_{k \in \mathbb{N}} \mathcal{N}_k\). Let \(j \in \mathbb{N}\) and define

\[ L_{O_j} := \cap \{ L(G_{\text{ind}(o)}) \mid o \in O_j \}. \]

Let \(\mathcal{H}\) be an uniformly decidable hypothesis space containing an index for every language \(L_{O_j}, j \in \mathbb{N}\). It is easy to see that such a hypothesis space exists. Without loss of generality, we may assume that the index for \(L_{O_j}\) in \(\mathcal{H}\) is \(j\) for all \(j \in \mathbb{N}\).

From the fact that \(\mathcal{L}\) is strong-monotonically identifiable with a probability \(p > 2/3\) follows that \(\text{range}(\mathcal{L}) \subseteq \text{range}(\mathcal{H})\). It remains to show that \(\text{range}(\mathcal{H})\) contains not more languages than \(\text{range}(\mathcal{L})\).

Let \(k \in \mathbb{N}\), let \(O \in \mathcal{N}_k\), and let \(T_{P,s_k}\) be the finite computation tree induced by \(P\) and \(s_k\). Furthermore, let \(O = \{o_{j_0}, \ldots, o_{j_m}\}\), and let \(\mathcal{F} = \{L(G_{\text{ind}(o_{j_0})}), \ldots, L(G_{\text{ind}(o_{j_m})})\}\) be the set of languages corresponding to the nodes in \(O\). Let \(L, L' \in \mathcal{F}\) such that neither \(L \subseteq L'\) nor \(L' \subseteq L\). Since both languages are members of \(\mathcal{L}\), and therefore by assumption strong-monotonically identifiable with a probability \(p > 2/3\), we can show by the following counting argument that there exists a coin-oracle \(\mathcal{C}\) such that the finite path \((P_{\mathcal{C}}((s_k)_x))_{x \leq l_k}\) can be extended to a strong-monotonically converging path for \(L\) and \(L'\). Assume the converse, i.e., there is no such coin-oracle. Then

\[ \{o \in O \mid L(G_{\text{ind}(o)}) \subseteq L\} \cap \{o \in O \mid L(G_{\text{ind}(o)}) \subseteq L'\} = \emptyset. \]

By assumption, \(w(O) = 2/3 + z\) for a \(z > 0\). Since \(L\) and \(L'\) are strong-monotonically identifiable with a probability \(p > 2/3\), we can conclude that

\[ \max_L := w(\{o \in O \mid L(G_{\text{ind}(o)}) \subseteq L\}) + \frac{1}{3} - z > \frac{2}{3}, \]

and

\[ \max_{L'} := w(\{o \in O \mid L(G_{\text{ind}(o)}) \subseteq L'\}) + \frac{1}{3} - z > \frac{2}{3}. \]

Thus,

\[ \frac{2}{3} + z + 2(\frac{1}{3} - z) \geq \max_L + \max_{L'} > \frac{4}{3}, \]

a contradiction.

Consequently, there exists a coin-oracle \(\mathcal{C}\) with

\[ P_{\mathcal{C}}(s_k) \neq \perp \quad \text{and} \quad L(P_{\mathcal{C}}(s_k)) \subseteq L \cap L'. \]
This can be performed for any two languages in \( \mathcal{F} \). Hence, we can conclude that the intersection of all languages in \( \mathcal{F} \) is a member of \( \mathcal{F} \) and therefore lying in \( \text{range}(\mathcal{L}) \). Notice that the fact that \( \mathcal{L} \) is strongly-monotonically identifiable with a probability strictly higher than \( 2/3 \), is necessary for our argumentation, since the counting argument mentioned above can only be used under this assumption.

Now we are ready to define the desired IIM \( M \). Let \( \ell \) and \( \text{int}_{\text{act}} \) be variables. Let \( L \in \text{range}(\mathcal{L}) \), let \( \tau \in \text{text}(L) \), and let \( T_{P, \tau} \) be the infinite computation tree induced by \( P \) when fed \( \tau \). Let \( y \in \mathbb{N} \), and let \( \mathcal{O} = \{ o \in \text{level } y \mid \text{ind}(o) \neq \perp \} \). Then set

\[
\text{int}_y := \begin{cases} 
\text{an index for } \bigcap_{o \in \mathcal{O}} L(G_{\text{ind}(o)}), & \text{if } w(\mathcal{O}) > \frac{2}{3}, \\
\perp, & \text{otherwise.}
\end{cases}
\]

Note that it is easy to compute \( \text{int}_y \), since \( \mathcal{O} = \mathcal{O}_j \) for a \( j \in \mathbb{N} \), and thus, \( \text{int}_y = j \). Let \( s \) be the least natural number such that \( \text{int}_s \) is consistent with \( \tau_0 \). Such an \( s \) exists, since \( \mathcal{L} \) is assumed to be strongly-monotonically identifiable with \( p > 2/3 \). Set \( \text{int}_{\text{act}} = \text{int}_s \) and \( \ell = s \). Notice that, for every \( y \in \mathbb{N} \), there exists a \( z \in \mathbb{N} \) such that \( \tau_y \) is consistent with \( \text{int}_z \). Let \( x \) be a natural number.

IIM \( M \): On input \( \tau_x \), \( M \) works as follows.

If \( x = 0 \), then set \( M(\tau_x) = \text{int}_s \). If \( x > 0 \), then distinguish the following cases. Assume \( \ell = y \). Let \( \mathcal{O} \) be the set of nodes such that \( \text{int}_{\text{act}} \) is an index for \( \bigcap\{ L(G_{\text{ind}(o)}) \mid o \in \mathcal{O} \} \). If \( \text{int}_{\text{act}} \) is consistent with \( \tau_x \), then output \( \text{int}_{\text{act}} \).

Otherwise compute

\[
\mathcal{O}' := \{ o \in \mathcal{O} \mid \text{ind}(o) \text{ inconsistent with } \tau_x \}.
\]

(A) If \( w(\mathcal{O} \setminus \mathcal{O}') > 2/3 \), then let \( \text{int}_{\text{act}} \) be an index for the intersection of \( \{ L(G_{\text{ind}(o)}) \mid o \in \mathcal{O} \setminus \mathcal{O}' \} \), and output \( \text{int}_{\text{act}} \).

(B) If \( w(\mathcal{O} \setminus \mathcal{O}') \leq 2/3 \), then set \( \text{int}_{\text{act}} = \text{int}_z \), where \( z \) is the least natural number \( > \ell \) such that \( \text{int}_z \) is consistent with \( \tau_{\max\{x,y\}} \), output \( \text{int}_z \), and set \( \ell := z \).

end

It remains to show that \( M \) identifies \( \mathcal{L} \) strongly-monotonically with respect to \( \mathcal{H} \). Let \( L \in \text{range}(\mathcal{L}) \), and let \( \tau \in \text{text}(L) \). With an argument similar to that
in the proof of Theorem 2.3.12, we can show that $M$ converges correctly on $\tau$ with respect to $\mathcal{H}$. It remains to show that $M$ is strong-monotonic on $\tau$ with respect to $\mathcal{H}$. Let $x \in \mathbb{N}$. Let $y$ be the actual level, i.e., $\ell = y$, and let $\tau_x$ be inconsistent with $h_1 := \text{int}_{act}$. If $M$ must not change the level, then obviously the new hypothesis is an index for a superset of $L(H_{h_1})$. Otherwise notice that $L(H_{h_1})$ is the intersection of $\{L(G_{\text{ind}(o)})| o \in \mathcal{O}\}$ for a set of nodes $\mathcal{O}$ with $w(\mathcal{O}) > 2/3$, i.e., $L(H_{h_1})$ is a subset of $L(G_{\text{ind}(o)})$ for each $o \in \mathcal{O}$. Let $z > y$ be the next considered level, and let $h_2 := \text{int}_z$ be an index for the next intersection $M$ outputs. As shown above, $L(H_{h_2}) \in \text{range}(\mathcal{L})$. Thus, $L(H_{h_2})$ has to be inferred with a probability $p > 2/3$. By definition of $M$, $h_2$ is consistent with $\tau_y$. Consequently, there has to be a path strong-monotonically converging on $\tau$ to an index for $L(H_{h_2})$ which passes through a node $o$ on level $y$ such that $L(G_{\text{ind}(o)})$ contains $L(H_{h_1})$. Hence, $L(H_{h_1}) \subseteq L(G_{\text{ind}(o)}) \subseteq L(H_{h_2})$. Thus, the first part of the theorem is proved.

The first part of this theorem simplifies the second, since it suffices to define an indexed family which is strong-monotonically identifiable with probability $p = 2/3$, but not deterministically strong-monotonically identifiable with respect to any class preserving hypothesis space. Let $(\cdot, \cdot) : \mathbb{N} \times \{0, 1\} \to \mathbb{N}$ be an effective encoding, and let $k, j \in \mathbb{N}, j \leq 1$. Set

$$L_{(k,j)} := \begin{cases} L_k' \cup \{a^kb^{\Phi_k(k)+1}\}, & \text{if } j = 0, \\ L_k' \cup \{a^kb^{\Phi_k(k)+2}\}, & \text{if } j = 1. \end{cases}$$

Obviously, $\mathcal{L} = (L_{(k,j)})_{k,j \in \mathbb{N}, j \leq 1}$ is an indexed family. Let $k$ be a natural number. If $\varphi_k(k) \uparrow$, then $L_{(k,0)} = L_{(k,1)} = L_k$. If $\varphi_k(k) \downarrow$, then both languages are finite, and their intersection is equal to $L_k'$, i.e., does not lie in $\mathcal{L}$.

In order to prove $\mathcal{L} \in \text{ESMON}_{\text{prob}}(2/3)$, we define a PIM $P$ equipped with a 3-sided coin. Let $L \in \text{range}(\mathcal{L})$, let $\tau$ be a text for $L$. Let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $x$ be a natural number, and let $m_x$ be the highest natural number with $a^kb^{m_x} \in \text{range}(\tau_x)$. Let $C \in \{0, 1, 2\}^\infty$ be a coin-oracle.

**PIM $P$:** On input $\tau_x$, $P^c$ works as follows.

(A) If $c_0 \in \{0, 1\}$, then set $P^{c}^x(\tau_x) = \langle k, j \rangle$.

(B) If $c_0 = 2$, then test whether $\Phi_k(k) \leq m_x$. If not, then output $\bot$ and request the next input. If it is, then compute both finite languages and request new inputs until the range of the actual text is equal to
the range of one of the finite languages. Output the index for that language.

end

Obviously, $P_{ESMON}^{prob}(2/3)$-identifies $L$. Suppose now that an IIM $M$ and a class preserving hypothesis space $\mathcal{G}$ exist such that $M$ identifies $L$ strongly-monotonically with respect to $\mathcal{G}$. Let $L \in \text{range}(\mathcal{L})$. Then there exists an $m_0 \in \mathbb{N}$ such that $M(\tau^L_{m_0}) \neq \bot$. Assume that $\varphi_k(k) \downarrow$ and $\Phi_k(k) > m_0$. Then $\text{range}(\tau^L_{m_0}) = \{a^k b^m \mid m \leq m_0\} \subset L_{(k,0)} \cap L_{(k,1)}$. Consequently, $L(G_M(\tau_{m_0}))$ has to be a subset of $L_{(k,0)}$ and $L_{(k,1)}$, since $M$ is assumed to identify $L$ strongly-monotonically with respect to $\mathcal{G}$. However, $\text{range}(\mathcal{G})$ does not contain subsets of $L_{(k,0)}$ and $L_{(k,1)}$. Thus, our assumption must be false, and we can conclude that $\varphi_k(k) \uparrow$ or $\Phi_k(k) \leq m_0$, a contradiction to the unsolvability of the halting problem. Hence, $L \notin SMON$.

2.3 Class Comprising Probabilistic Learning

In the last section of this chapter, we show that it is possible to compensate the power of probabilistic inference machines by enlarging the range of the given hypothesis space. More precisely, we show that a given probabilistic machine $P$ identifying an indexed family may be replaced by a deterministic machine provided this machine is allowed to construct new hypotheses which are not members of the given hypothesis space by amalgamating the hypotheses given in a finite computation tree which is induced by $P$ and a finite text segment. Thus, we can conclude that restrictions on the choice of the hypothesis space have a strong effect on the probabilistic hierarchy when dealing with probabilistic learning under monotonicity constraints.

2.3.1 Probabilistic Learning of indexed families without additional constraints

Jain and Sharma [51] showed that probabilistic identification is strictly more powerful than team identification when dealing with identification of recursively enumerable languages from text. However, in the case of learning from informant, where the learner receives positive and negative information about the language to be inferred, probabilistic identification and team identification are equivalent. Consequently, the nonequivalence of probabilistic iden-
tification and team identification when learning from text can be attributed to the lack of negative information about the languages to be learned (cf. [51]). In the case of identification of indexed families, however, the lack of negative information can be compensated by the function which uniformly decides membership for all languages belonging to the indexed family to be learned. Hence, we are able to show for learning of indexed families without additional constraints that probabilistic learning is equivalent to team learning. We prove this result by using an argument due to Pitt [80].

The equivalence proof of Pitt [80] for probabilistic function EX-identification can be divided into two parts. First he showed by collecting information about potentially converging paths in a given infinite computation tree that for each collection of recursive functions $\mathcal{F}$ identifiable with a probability $p > 1/(n + 1)$, there exists a team $M_1, \ldots, M_n$ of IIMs with the following property. For each function $f \in \mathcal{F}$ to be inferred, there exists a $j \in \{1, \ldots, n\}$ such that $M_j$, when successively fed the graph for $f$, outputs an infinite sequence of finite lists of hypotheses, and converges to a list containing at least one hypothesis which is correct for $f$. Pitt constructed this team $M_1, \ldots, M_n$ by extracting information from the finite subtrees of the considered infinite computation tree. Considering language learning from text, there is no difficulty in adapting this part of the proof. In the second part of the proof, Pitt uses an amalgamation argument, introduced by Case and Smith [25], for constructing single hypotheses from the hypotheses in the lists the team members $M_1, \ldots, M_n$ output. As we show in the next theorem, this part too can be adapted to text identification of indexed families. Consequently, the probabilistic hierarchy is discrete with breakpoints at $1/n$, $n \in \mathbb{N}^+$. In particular, the probabilistic hierarchy has a threshold beginning with $1/2$.

**Theorem 2.3.1.** Let $n \in \mathbb{N}^+$, and let $\frac{1}{n+1} < p \leq \frac{1}{n}$. Then

$$CLIM_{\text{prob}}(p) = ELIM_{\text{team}}(n).$$

**PROOF.** Let $n \in \mathbb{N}^+$, and let $p \in [0,1]$ with $1/n + 1 < p \leq 1/n$. It is obvious that $ELIM_{\text{team}}(n) \subseteq CLIM_{\text{prob}}(p)$ (cf. [80]). Let $\mathcal{L}$ be in $CLIM_{\text{prob}}(p)$. We can assume that there exists a team $M_1, \ldots, M_n$ of IIMs with the following property. For each $L \in \text{range}(\mathcal{L})$ and every $\tau \in \text{text}(L)$, there exists a $j \in \{1, \ldots, n\}$ such that $M_j$, when fed $\tau$, outputs an infinite sequence of finite lists of hypotheses and converges to a list which contains at least one correct hypothesis for $L$ with respect to $\mathcal{G}$. Notice that all indices in the lists
are interpreted as indices of languages in range(\mathcal{G}). Now we define a team M'_1, \ldots, M'_n of IIMs such that M'_1, \ldots, M'_n CLIM_{team}(n)-identifies \mathcal{L} with respect to \mathcal{G}. Let j \in \{1, \ldots, n\}, let L \in range(\mathcal{L}), let \tau be a text for L, and let x be a natural number. Without loss of generality, we may assume that the hypotheses in the list M_j(\tau_{x-1}) are contained in the list M_j(\tau_x).

IIM M'_j: On input \tau_x, M'_j works as follows.

If x = 0, then set M_j(\tau_0) = \bot. Let x \in \mathbb{N}^+. Simulate M_j on \tau_x. If M_j outputs \bot, then output \bot. If M_j, when fed \tau_x, outputs the list \{j_0, \ldots, j_m\}, then delete all hypotheses \{j_i_0, \ldots, j_i_s\} from the list which are not consistent with \tau_x. Compare the remaining hypotheses \{j_k_0, \ldots, j_k_m\} as follows. If there exists some i \in \mathbb{N}, i \leq m - s, with L(G_{j_i}) \cap \{0, \ldots, x\} \subseteq L(G_{j_k}) \cap \{0, \ldots, x\} for all \ell \in \mathbb{N}, \ell \leq m - s, then output j_k, where i is the least natural number with this property. If there is no such i \leq m - s, then output M'_j(\tau_{x-1}).

end

M'_j is recursive, since \mathcal{G} is uniformly decidable, and M'_j converges correctly on \tau with respect to \mathcal{G} if and only if M_j converges to a finite list of hypotheses containing at least one correct hypothesis for L. Consequently, M'_1, \ldots, M'_n CLIM_{team}(n)-identifies \mathcal{L} with respect to the original hypothesis space \mathcal{G}, and thus \mathcal{L} \in ELIM_{team}(n), since, by Theorem 1.2.1, CLIM_{team}(n) = ELIM_{team}(n) for all n \in \mathbb{N}^+. ⋄

Notice that Theorem 2.3.1 has been proved independently by Jain and Sharma (cf. [55]).

Theorem 2.3.1 directly yields the following consequences.

Corollary 2.3.2.

(a) CLIM_{prob}(p) = CLIM for all p \in (\frac{1}{2}, 1],

(b) CLIM_{prob}(p) = ELIM_{prob}(p) for all p \in [0, 1].

PROOF. The proof of the first part is straightforward. To prove the second part, let p \in [0, 1]. Then p \in (\frac{1}{n+1}, \frac{2}{n}] for an n \in \mathbb{N}^+. From Theorem 2.3.1 it follows that CLIM_{prob}(p) = ELIM_{team}(n). Thus, we can conclude that CLIM_{prob}(p) = ELIM_{team}(n) \subseteq ELIM_{prob}(p) \subseteq CLIM_{prob}(p). ⋄

Another corollary, which can be drawn from Theorem 2.3.1, concerns the results from Rolf Wiehagen et al. [107]. In their work, it was shown that a
nonstandard hypothesis space $G$ exists such that each infinite set of recursive functions which is identifiable with respect to some acceptable Gödel-numbering, is identifiable with arbitrary high probability with respect to $G$ but not deterministically identifiable with respect to $G$. In the proof of Theorem 2.3.1, we noticed that every indexed family which is $\text{CLIM}_\text{prop}(p)$-identifiable with a probability $p > 1/2$ with respect to a hypothesis space $G$, is deterministically identifiable with respect to the same hypothesis space $G$. Thus, we can conclude that the result of Wiehagen et al. is not transferable to probabilistic learning of indexed families.

**Corollary 2.3.3.** Let $\mathcal{L}$ be an indexed family, and let $G$ be a hypothesis space. If $\mathcal{L}$ is $\text{CLIM}_\text{prob}(p)$-identifiable with a probability $p > 1/2$ with respect to $G$, then $\mathcal{L}$ is deterministically identifiable with respect to $G$.

In order to show that the probabilistic hierarchy has breakpoints at $p = 1/n$, $n \in \mathbb{N}^+$, we show that $\text{ESMON}^\text{st}_{\text{team}}(n + 1) \setminus \text{ELIM}_{\text{team}}(n) \neq \emptyset$ for all $n \in \mathbb{N}^+$. Before starting the proof of this theorem, we recall a well known result in inductive inference according to which each family $\mathcal{L}$ of recursive languages which contains an infinite language $L$ and all finite subsets of $L$ is not identifiable in the limit from text (cf. for example [48]). We construct the indexed families witnessing the desired separations by generalizing this idea.

**Theorem 2.3.4.** Let $n \in \mathbb{N}^+$. Then $\text{ESMON}^\text{st}_{\text{team}}(n + 1) \setminus \text{ELIM}_{\text{team}}(n) \neq \emptyset$.

**PROOF.** Let $(\ , \ ) : \mathbb{N} \times \mathbb{N}^+ \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}^+$, i.e., the set of all ordered tuples with second component $\neq 0$. Since we can identify $\mathbb{N} \times \mathbb{N}^+$ with the set of the positive rational numbers $\mathbb{Q}^+$ by interpreting $(a, b) = a/b$, we can define a natural ordering on the tuples by setting $(a, b) \leq (c, d)$ if and only if $ad \leq cb$.

Now we define the indexed families witnessing the desired separation. For the sake of readability, we restrict ourselves to the cases $n = 1, 2$. First, let $n = 1$. Then define $L_0 = \mathbb{Q}^+$ and $L_k = \{ (a, b) \mid (a, b) \leq (k, 1) \}$ for all $k \in \mathbb{N}^+$. Let $\mathcal{L}^1 = (L_k)_{k \in \mathbb{N}}$. In the case of $n = 2$, the definition is slightly more complicated. Let $(\ , \ ) : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$. Let $k, j \in \mathbb{N}$. Define

$$L_{(0,k)} = L_{(k,0)} := \mathbb{Q}^+, \quad \text{and}$$

$$L_{(k,j)} := \begin{cases} \{ (a, b) \mid (a, b) \leq (k, 1) \}, & \text{if } j = 1, \\ \{ (a, b) \mid (a, b) \leq (kj - 1, j) \}, & \text{if } j > 1. \end{cases}$$

79
Let \( k, j \) be natural numbers. The language \( L_{(k,1)} \) corresponds to the set of all positive rational numbers which are less than or equal to \( k \), and the language \( L_{(k,j)}, j \geq 2 \), corresponds to the set of positive rational numbers less than or equal to \( (kj - 1)/j = k - 1/j \). Define \( \mathcal{L}^2 = (L_{(k,j)})_{k,j \in \mathbb{N}} \). It is easy to see, that \( \mathcal{L}^1 \) and \( \mathcal{L}^2 \) are indexed families, since \( (a,b) \leq (c,d) \) is uniformly decidable for all \( a,b,c,d \in \mathbb{N}, b,d \neq 0 \). Furthermore, one easily verifies that \( \mathcal{L}^1 \) establishes the wanted separation for \( n = 1 \).

In order to prove that \( \mathcal{L}^2 \) is strong-monotonically identifiable by a team of three IIMs, we define IIMs \( M_0, M_1, M_2 \) such that \( M_0 \) chooses its hypotheses out of the set \( \{\langle 0, k \rangle | k \in \mathbb{N}\} \), \( M_1 \) out of \( \{\langle k, 1 \rangle | k \in \mathbb{N}\} \), and \( M_2 \) out of \( \{\langle k,j \rangle | k,j \in \mathbb{N}, j \geq 2\} \). Each machine works with identification by enumeration with respect to the allowed hypotheses and guesses the minimal language which is consistent with the data seen so far. It is easy to see that the team \( M_0, M_1, M_2 \) identifies \( \mathcal{L}^2 \). Moreover, for each \( L \in \mathcal{L}^2 \), every \( \tau \in \text{text}(L) \), and all \( i \in \{0, 1, 2\} \), \( M_i \) is strong-monotonic on \( \tau \). Thus, \( \mathcal{L}^2 \in \text{CSMON}_{\text{team}}(3) \). It remains to show that \( \mathcal{L}^2 \notin \text{CLIM}_{\text{team}}(2) \). Assume the converse, and let \( M_1, M_2 \) be a team which identifies \( \mathcal{L}^2 \). By using a known diagonalization argument (cf., e.g., [48]), we can show that there is a text for the language \( \mathcal{Q}^+ \) which is neither identifiable by \( M_1 \) nor by \( M_2 \).

An analogous construction can be performed for arbitrary \( n \in \mathbb{N}^+ \), since \( \mathcal{Q} \) is dense. Consequently, we can define \( \mathcal{L}^n \) witnessing the desired separation for all \( n \in \mathbb{N}^+ \).

For further results about team hierarchies in the setting of learning indexed families, we refer the reader to Tabe and Zeugmann [98].

From Theorem 2.3.1 and Theorem 2.3.4, we immediately get the following corollary which tells us that the probabilistic hierarchy in the case of unconstrained probabilistic learning is discrete.

**Corollary 2.3.5.** \( \langle \text{ELIM}_{\text{prob}}(p) \rangle_{p \in [0,1]} \) is discrete with breakpoints at \( \frac{1}{n} \), \( n \in \mathbb{N}^+ \).

**PROOF.** Theorem 2.3.1 yields that \( \text{ELIM}_{\text{prob}}(p) = \text{ELIM}_{\text{team}}(n) \) for all \( p \in (\frac{1}{n+1}, \frac{1}{n}] \). Now, by applying Theorem 2.3.4, we can conclude that the collection \( \langle \text{ELIM}_{\text{prob}}(p) \rangle_{p \in [0,1]} \) is discrete.

Furthermore, we can conclude that every considered probabilistic hierarchy at least has breakpoints at \( 1/n \) for \( n \in \mathbb{N}^+ \).
Corollary 2.3.6. Let $\lambda \in \{E, \varepsilon, C\}$ and $\mu \in \{COV, SMON, MON\}$. Then
\[
\lambda \mu_{prob}\left(\frac{1}{n+1}\right) \setminus \bigcup_{\frac{1}{n+1} < p \leq \frac{1}{n}} \lambda \mu_{prob}(p) \neq \emptyset \quad \text{for all } n \in \mathbb{N}^+.
\]

2.3.2 Conservative and Strong-monotonic Probabilistic Learning

For class comprising conservative probabilistic learning, we are also able to show an equivalence between probabilistic identification and team identification. Before proving the main theorem of this subsection, we note that the probabilistic learning classes $\lambda COV^s_{prob}(p)$ are equal to the classes $\lambda COV_{prob}(p)$ for all $p \in [0, 1]$, and all $\lambda \in \{E, \varepsilon, C\}$. The same result holds for conservative team learning.

Theorem 2.3.7. Let $p \in [0, 1]$, and let $\lambda \in \{E, \varepsilon, C\}$. Then $\lambda COV^s_{prob}(p) = \lambda COV_{prob}(p)$.

PROOF. Let $p \in [0, 1]$, and let $L \in CCOV_{prob}(p)$. Let $P$ be a PIM which $CCOV_{prob}(p)$-identifies $L$ with respect to a hypothesis space $G$. Define a PIM $P'$ as follows. Let $C$ be an coin-oracle. Let $\tau$ be a text for a language $L \in \text{range}(L)$. Let $x \in \mathbb{N}$. If $x = 0$, then set $(P')^c_0(\tau_0) := P^c_0(\tau_0)$. If $x \in \mathbb{N}^+$, then distinguish the following cases. If $\text{range}(\tau_{x+1}) \subseteq L(G(P')^{c^x}(\tau_x))$, then set $(P')^{c^x+1}(\tau_{x+1}) = (P')^{c^x}(\tau_x)$. Otherwise set $(P')^{c^x+1}(\tau_{x+1}) = P^{c^x+1}(\tau_{x+1})$.

Obviously, $P^C$ is conservative on $\tau$ with respect to $G$ for every coin-oracle $C$. Let $L$ be a coin-oracle which induces a conservative path. Then it holds by definition of $P'$: $(P')^{c^x}(\tau_x) = P^{c^x}(\tau_x)$ for all $x \in \mathbb{N}$. Thus, the probability of the set of all coin-oracles $\mathcal{C}$ such that $P^\mathcal{C} CCOV$-converges correctly on $\tau$ with respect to $G$, is greater than or equal to $p$. Hence, $P' CCOV_{prob}(p)$-identifies $L$ with respect to $G$. Notice that we did not change the hypothesis space, i.e., the result holds for all $\lambda \in \{E, \varepsilon, C\}$. $\diamond$

In the following, we prove that $CCOV_{prob}(p) = CCOV_{team}(n)$ for all $p \in (1/(n+1), 1/n]$ and all $n \in \mathbb{N}^+$. As in the proof of Theorem 2.3.1, we need an amalgamating argument in order to construct a single hypothesis from a finite number of hypotheses in a given infinite computation tree. We already saw that it is possible to perform amalgamating algorithms when dealing with indexed families, but in the case of probabilistic learning under monotonicity constraints, we risk loosing monotonicity during the amalgamation process.
However, the problem of amalgamating the hypotheses in a way such that the property to be conservative is preserved can be solved by constructing an appropriate hypothesis space for the indexed family to be learned.

**Theorem 2.3.8.** Let $n \in \mathbb{N}^+$, and let $\frac{1}{n+1} < p \leq \frac{1}{n}$. Then

$$CCOV_{\text{prob}}(p) = CCOV_{\text{team}}(n).$$

**Proof.** By using the same argument as Pitt [80], we can easily show that $CCOV_{\text{team}}(n) \subseteq CCOV_{\text{prob}}(1/n)$ for all $n \in \mathbb{N}^+$. In order to prove the other inclusion, let $p \in (1/(n+1), 1/n]$, and let $\mathcal{L} \in CCOV_{\text{prob}}(p)$. By applying Theorem 2.3.7, we may assume that $\mathcal{L} \in CCOV_{\text{prob}}^{\text{st}}(p)$. Let $P$ be a PIM which $CCOV_{\text{prob}}(p)$-identifies $\mathcal{L}$ with respect to a hypothesis space $\mathcal{G} = \{G_i\}_{i \in \mathbb{N}}$. Let $\tau$ be a text for a language $L \in \text{range} (\mathcal{L})$, and let $T_{P,\tau}$ be the corresponding infinite computation tree. Let $k \in \mathbb{N}$. Define

$$OG_k := w(\{ \text{o \in level k} | L \subset L(G_{\text{ind}(o)}) \}).$$

In an analogous manner define $F_k$ to be the weight of all nodes $o$ on level $k$ such that $\text{ind}(o)$ is refutable for $L$. Notice that there exists some $\epsilon > 0$ such that

$$OG_k < \frac{n}{n+1} - \epsilon \quad \text{for all} \ k \in \mathbb{N},$$

since $\mathcal{L}$ is conservatively identifiable with probability $p > \frac{1}{n+1}$. Moreover, $OG_k \leq OG_\ell$ for all $k, \ell \in \mathbb{N}$, $k < \ell$, since $\mathcal{L} \in CCOV_{\text{prob}}^{\text{st}}(p)$. Consequently, the sequence $(OG_k)_{k \in \mathbb{N}}$ is monotonically increasing, bounded, and thus converging to some $s < n/(n+1)$.

Let $\mathcal{N}$ be the set of all finite subsets of $\mathbb{N}$, and let $(N_i)_{i \in \mathbb{N}}$ be an effective enumeration of $\mathcal{N}$. Let $j \in \mathbb{N}$. Then let $L'_j$ be the intersection of $\{L(G_i) | i \in N_j\}$. Notice that $L'_j$ may be empty. Now let $\mathcal{G}_I$ be an uniformly decidable hypothesis space containing an index for every language $L'_j$, $j \in \mathbb{N}$. It is easy to see that such a hypothesis space exists. Moreover, it is obvious, that $\text{range}(\mathcal{L}) \subseteq \text{range}(\mathcal{G}_I)$.

Now we define the wanted team $M_1, \ldots, M_n$ of IIMs such that $M_1, \ldots, M_n$ $CCOV_{\text{team}}(n)$-identifies $\mathcal{L}$ with respect to $\mathcal{G}_I$. For $j \in \{1, \ldots, n\}$, $M_j$ guesses that

$$\lim_{k \to \infty} OG_k \in \left[ \frac{j-1}{n+1}, \frac{j}{n+1} \right].$$

82
i.e., there exists some $\epsilon > 0$, such that $OG_k < \frac{j}{n+1} - \epsilon$ for all $k \in \mathbb{N}$. Intuitively, we modeled a kind of belief, i.e., $M_j$ believes that $OG_k < \frac{j}{n+1} - \epsilon$ for an $\epsilon > 0$.

Let $m_j$, $\ell_j$ and $int_{act}^j$ be variables for $j \in \{1, \ldots, n\}$. Let $L \in \text{range}(L)$, let $\tau \in \text{text}(L)$, and let $j \in \{1, \ldots, n\}$. Set $m_j = 1$, $\ell_j = 1$. Furthermore, set

$$int_{act}^j := \text{an index for } L(G_{j_1}) \cap L(G_{j_2})$$

where $j_1, j_2$ are the indices corresponding to the nodes in $T_{P_{\tau_0}}$. Let $x$ be a natural number.

IIM $M_j$: On input $\tau_x$, $M_j$ works as follows.

If $x = 0$, then output $int_1$. If $x \in \mathbb{N}^+$, then distinguish the following cases. If $int_{act}^j$ is consistent with $\tau_x$, then output $int_{act}^j$. Otherwise assume that $int_{act}^j$ is an index for $\cap\{L(G_{j_1}), \ldots, L(G_{j_l})\}$, and find the languages $\{L(G_{j_1}), \ldots, L(G_{j_l})\}$ which are inconsistent with $\tau_x$. Let $s := |\{o \in \text{level } \ell_j \mid \text{ind}(o) \in \{j_1, \ldots, j_k\}\}|$. Notice that $s \geq k$ since some indices could appear several times on level $\ell_j$. Set

$$m_j := m_j - \frac{s}{2^\ell_j}.$$

(A) If $m_j \geq j/(n+1)$, then let $int_{act}^j$ be an index for $\cap\{L(G_{j_1}), \ldots, L(G_{j_l})\}\setminus \{L(G_{j_1}), \ldots, L(G_{j_k})\}$, and output $int_{act}^j$.

(B) Assume $m_j < j/(n+1)$. Let $h_{\ell_j+1}$ be an index for the intersection of the languages corresponding to the nodes in level $(\ell_j+1)$. Set $int_{act}^j = h_{\ell_j+1}$;

$\ell_j := \ell_j + 1$, and $m_j := 1$. Output nothing.

end

Now we have to prove that $L \in CCOV_{team}(n)$. Let $L \in \text{range}(L)$, and let $\tau \in \text{text}(L)$. Let $j \in \{1, \ldots, n\}$. Suppose that $M_j$ is the machine correctly guessing $\lim_{k \to \infty} OG_k$. Then $M_j$ converges correctly on $\tau$ with respect to $G_I$, since there exists a $k_0 \in \mathbb{N}$ such that $F_k < 1 - j/(n+1)$ for all $k \geq k_0$. Furthermore, $M_j$ never outputs an overgeneralization of $L$, since it changes the level as soon as it would have to delete too many indices from the actual intersection-index. Consequently, $L \in CCOV_{team}(n)$. ⋄

Thus, we can conclude that the probabilistic hierarchy in the case of class comprising conservative learning is discrete.
Corollary 2.3.9. \( \langle CCOV_{\text{prob}}(p) \rangle_{p \in [0,1]} \) is discrete with breakpoints at \( \frac{1}{n} \), \( n \in \mathbb{N}^+ \).

In the case of class comprising strong-monotonic probabilistic learning, it seems to be more difficult to preserve the monotonicity constraint, but it turns out that there is no gain of learning power when the probability ranges over the interval \( (\frac{1}{2}, 1] \). The structure of \( \langle CSMON_{\text{prob}}(p) \rangle_{p \in [0,1]} \) in the interval \( [0, \frac{1}{2}) \) is not yet investigated.

Theorem 2.3.10. \( \langle CSMON_{\text{prob}}(p) \rangle_{p \in [0,1]} \) has a threshold beginning with \( \frac{1}{2} \).

PROOF. As mentioned in Corollary 2.3.6, there exists an indexed family strong-monotonically identifiable with probability \( \frac{1}{2} \), but not identifiable with a higher probability. It remains to show that

\[ CSMON_{\text{prob}}(p) = CSMON \quad \text{for all} \quad \frac{1}{2} < p \leq 1. \]

Let \( \mathcal{L} \) an indexed family with \( \mathcal{L} \in \bigcup_{\frac{1}{2} < p \leq 1} CSMON_{\text{prob}}(p) \), and let \( P \) be the identifying PIM. Assume \( P \) to be equipped with a two-sided coin. Let \( \mathcal{G}_{IU} \) be a hypothesis space extending the hypothesis space \( \mathcal{G}_I \) defined in the proof of Theorem 2.3.8 which contains for each finite set \( \{j_0, \ldots, j_n\} \) in \( \mathcal{G}_I \) an index \( \cup \{j_0, \ldots, j_n\} \) for the finite union of \( L(G_{j_0}), \ldots, L(G_{j_n}) \). Obviously, such a hypothesis space exists. Now we define an IIM \( M \) such that \( M \) CSMON-identifies \( \mathcal{L} \) with respect to \( \mathcal{G}_{IU} \). Let \( lev, \cap_{act,k}, \) and \( m \) be variables for \( k \in \mathbb{N} \) and set

\[ lev := 0, \quad m := 1 \quad \text{and} \quad \cap_{act,k} := \cap_k, \]

where \( \cap_k \) is an index for the intersection of all languages on level \( k \). Let \( L \in \mathcal{L} \), let \( \tau \in \text{text}(L) \), and let \( x \) be a natural number.

IIM \( M \): On input \( \tau_x \), \( M \) works as follows.

If \( x = 0 \), then output \( \cap_0 \). If \( x \neq 0 \), then assume \( lev = z \) for some \( z \in \mathbb{N}, z \leq x \). If \( M(\tau_{x-1}) \) is consistent with \( \tau_x \), then output \( M(\tau_{x-1}) \). Otherwise compute the languages \( \{L(G_{j_{i_0}}), \ldots, L(G_{j_{i_k}})\} \) in \( \cap_{act,z} = \{L(G_{j_{i_0}}), \ldots, L(G_{j_{i_k}})\} \) which are inconsistent with \( \tau_x \), and define

\[ s := |\{o \in \text{level} z \mid \text{ind}(o) \in \{j_{i_0}, \ldots, j_{i_k}\}\}, \quad m := m - \frac{s}{2^z}. \]

(A) If \( m > \frac{1}{2} \), then set \( \cap_{act,z} := \cap \{\{i_0, \ldots, i_k\} \setminus \{j_{i_0}, \ldots, j_{i_k}\}\} \), then output

\[ \cup_{k=0}^{z} \cap_{act,k} \cdot \]

84
(B) Otherwise output

\[ \bigcup_{k=0}^{z} \cap \text{act},k \bigcup \cap_{z+1} : \]

Set \( \text{lev} := \text{lev} + 1 \), and set \( m := 1 \)

end

It is easy to see that \( M \) works strong-monotonically on \( \tau \), since an intersection-index \( \cap_{\text{act},k} \) appearing in the actual union-index can be enlarged but will never be deleted. Moreover, every actual intersection-index is an index for a subset of \( L \), since

\[ (+) \ \text{weight}(\{o|o \in \text{level} k|L \subseteq L(G_{\text{ind}}(o))\}) > \frac{1}{2} \text{ for all } k \in \mathbb{N}, \text{ and} \]

\[ (+) \ M \ \text{never deletes more than } \frac{1}{2}-\text{many indices from } \cap_{\text{act},k} \text{ for all } k \leq z \]

where \( z \) is the actual level \( M \) works on.

Furthermore, since \( L \in \text{CSMON}_{\text{prob}}(p) \) with a probability \( p \in (\frac{1}{2},1] \), there exists a least level \( x_0 \) such that \( \text{weight}(\{o|o \in \text{level} x_0|L \subseteq L(G_{\text{ind}}(o))\}) > \frac{1}{2} \). Obviously, there exists at least one node \( o \) on level \( x_0 \) with \( L_{\text{ind}}(o) = L \), since \( L \) is \( \text{SMON} \)-identifiable with a probability \( p > 1/2 \). For every level \( x < x_0 \), \( \text{weight}(\{o|o \in \text{level} x_0|L \subseteq L(G_{\text{ind}}(o))\}) \leq \frac{1}{2} \), so we can conclude that \( M \) reaches level \( x_0 \) and outputs - after deleting enough refutable indices on level \( x_0 \) - a correct index for \( L \) which will never be changed. Consequently, \( M \) converges correctly and strong-monotonically on \( \tau \) with respect to \( \mathcal{G}_{\text{IU}} \). ◦

From Theorem 2.3.10, we can derive the following corollary.

**Corollary 2.3.11.** Let \( p > \frac{1}{2} \). Then \( \text{CSMON}_{\text{prob}}^*(p) = \text{CSMON}_{\text{prob}}(p) = \text{CSMON} \).

### 2.3.3 Monotonic Probabilistic Learning with probability \( p \geq 2/3 \)

In the following, we show that \( \langle \text{CMON}_{\text{prob}}(p) \rangle_{0 \leq p \leq 1} \) has a threshold beginning with \( 2/3 \). In order to prove this result, we use a version of the argument used in Theorem 2.2.2.

**Theorem 2.3.12.** The probabilistic hierarchy \( \langle \text{CMON}_{\text{prob}}(p) \rangle_{0 \leq p \leq 1} \) has a threshold beginning with \( \frac{2}{3} \).
PROOF. We start by showing that $\text{CMON}_{\text{prob}}(p) = \text{CMON}$ for every $p > 2/3$. Let $\mathcal{L}$ be an indexed family such that $\mathcal{L} \in \cup_{2/3 < p \leq 1} \text{CMON}_{\text{prob}}(p)$, let $\mathcal{G}$ be a hypothesis space, and let $P$ be a PIM such that $P$ $\text{CMON}_{\text{prob}}(p)$-identifies $\mathcal{L}$ with a probability $p > 2/3$ with respect to $\mathcal{G}$. Let $\mathcal{H}$ be a hypothesis space extending $\mathcal{G}$ defined as follows.

Let $(s_j)_{j \in \mathbb{N}}$ be an effective enumeration of all finite sequences of strings $s$ with $\text{range}(s) \subset L$ for some $L \in \text{range}(\mathcal{L})$. Let $k \in \mathbb{N}$, let $T_{P,s_k}$ be the finite computation tree induced by $P$ and $s_k$, and let $l_k$ be the highest level in $T_{P,s_k}$. Let $x \in \mathbb{N}$, $x \leq l_k$. Let $O \subseteq \text{level}_l$ be a set of nodes with $w(O) > \frac{2}{3}$. Define

$$\text{succ}_{s_k}(O) := \{ o \in \text{level}_l \mid \exists \text{ a node } o' \in O, \text{ o' predecessor of } o \}.$$ 

Then define the following languages:

$$L^O := \cap \{ L(G_{\text{ind}(o)}) \mid o \in O \},$$

and

$$L^O_{s_k} := \begin{cases} L^O, & \text{if } x = l_k, \\ \cap_{O' \subseteq \text{succ}_{s_k}(O), w(O') > \frac{1}{3}} \bigcup_{o \in O'} (L(G_{\text{ind}(o)}) \cap L^O), & \text{if } x < l_k. \end{cases}$$

Finally, set

$$L_{s_k} := \bigcup_{x=1}^{l_k} \left( \bigcup_{O \subseteq \text{level}_x, w(O) > \frac{2}{3}} L^O_{s_k} \right).$$

Now let $\mathcal{H}$ be an uniformly decidable hypothesis space which contains an index for each language $L_{s_k}$, $k \in \mathbb{N}$. Denote the index for $L_{s_k}$ by $\alpha_{s_k}$ for all $k \in \mathbb{N}$.

Let $L \in \text{range}(\mathcal{L})$, and let $\tau$ be a text for $L$. Let $y \in \mathbb{N}$. Then there exists a $k \in \mathbb{N}$ with $s_k = \tau_y$. For the sake of readability, we write $\alpha_{\tau_y}$ end $\text{succ}_{\tau_y}(O)$ instead of $\alpha_{s_k}$ and $\text{succ}_{s_k}(O)$. Note that the highest level $l_k$ in $T_{P,\tau_y}$ is $y + 1$. In the following, we show the following claim which implies the first part of the proof of Theorem 2.3.12.

Claim

(1) There exists an $x_0 \in \mathbb{N}$ such that $L(H_{\alpha_{\tau_y}}) = L$ for all $y \geq x_0$.

(2) $L(H_{\alpha_{\tau_y}}) \cap L \subseteq L(H_{\alpha_{\tau_z}}) \cap L$ for all $x, y \in \mathbb{N}$ with $x \leq y + 1$. 

86
Proof (of the claim). Since we assumed \( \mathcal{L} \) to be monotonically identifiable with a probability \( p > 2/3 \), there exist a natural number \( x_0 \in \mathbb{N} \) and an \( \mathcal{O}_0 \subseteq \text{level } x_0 \) with \( w(\mathcal{O}_0) > 2/3 \) such that \( L(G_{\text{ind}(o)}) = L \) for every node \( o \in \mathcal{O}_0 \), and \( L(G_{\text{ind}(o')}) = L \) for every node \( o' \in \mathcal{T}_{\mathcal{P},\tau} \) being a successor of a node in \( \mathcal{O}_0 \). Let \( x, y \in \mathbb{N} \) with \( y \geq x_0 \), \( x \leq y + 1 \). Then

\(+\) \( L_{\tau y}^{\mathcal{O}_0} = L \), and

\(+\) \( L_{\tau y}^{\mathcal{O}} \subseteq L \) for every \( \mathcal{O} \subseteq \text{level } x \) with \( w(\mathcal{O}) > \frac{2}{3} \).

Obviously, \( L_{\tau y}^{\mathcal{O}_0} = L \). Let \( \mathcal{O} \subseteq \text{level } x \) with \( w(\mathcal{O}) > 2/3 \). Then there exists a set of nodes \( \mathcal{O}' \subset \text{succ}_{\tau y}(\mathcal{O}), w(\mathcal{O}') > 1/3 \), with \( L(G_{\text{ind}(o)}) = L \) for all \( o \in \mathcal{O}' \). Thus, \( L_{\tau y}^{\mathcal{O}} \subseteq L \). Consequently, \( \alpha_{\tau y} \) is a correct index for \( L \) with respect to \( \mathcal{H} \) for all \( y \in \mathbb{N}, y \geq x_0 \).

Let \( x, y \in \mathbb{N}, x \leq y + 1 \). Let \( \mathcal{O} \subseteq \text{level } x \) with \( w(\mathcal{O}) > 2/3 \). In order to prove the second part of our claim, we show that \( L^{\mathcal{O}} \cap L = L_{\tau y}^{\mathcal{O}} \cap L \). If \( x = y + 1 \), then by definition \( L_{\tau y}^{\mathcal{O}} = L^{\mathcal{O}} \). Let \( x < y + 1 \), then let \( \mathcal{O}' \subset \text{succ}_{\tau y}(\mathcal{O}) \) with \( w(\mathcal{O}') > 1/3 \). Then there exist a node \( o \in \mathcal{O} \) and a node \( o' \in \mathcal{O}' \) such that \( o' \) is a successor of \( o \) and there exists a monotonically converging path passing through \( o \) and \( o' \). From this and from the fact that \( L^{\mathcal{O}} \) is the intersection of the languages given by the nodes in \( \mathcal{O} \), we can see that

\[ L^{\mathcal{O}} \cap L \subseteq L(G_{\text{ind}(o')}) \cap L. \]

Thus, we can follow from the definition of \( L_{\tau y}^{\mathcal{O}} \) that

\[ L^{\mathcal{O}} \cap L = L_{\tau y}^{\mathcal{O}} \cap L. \]

Consequently,

\[ L(H_{\alpha_{\tau y}}) \cap L = \bigcup_{x=1}^{y+1} \left( \bigcup_{\mathcal{O} \subseteq \text{level } x, w(\mathcal{O}) > \frac{2}{3}} (L^{\mathcal{O}} \cap L) \right) \text{ for all } y \in \mathbb{N}. \]

This completes the proof of the claim. end (of Proof).

Now the definition of an IIM \( M \) with the desired properties is straightforward. \( M \) chooses its hypotheses from the infinite sequence \( (\alpha_{\tau y})_{y \in \mathbb{N}} \) by respecting
the order of the sequence and it changes the hypothesis if and only if an inconsistency with the actual text appears. Thus, $M$ identifies $\mathcal{L}$ monotonically with respect to $\mathcal{H}$. In particular, this implies that $\text{range}(\mathcal{L}) \subseteq \text{range}(\mathcal{H})$.

In order to prove the second part of the theorem, let $\langle, \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$, and let $k \in \mathbb{N}$. Set $L_{\langle k, 0 \rangle} := L_k$. Let $j \in \mathbb{N}$, and define

$$L_{\langle k, j \rangle} := \begin{cases} L_k, & \text{if } \Phi_k(k) \neq j, \\ \{a^k b^m \mid m \leq \varphi_k(k) + 1\}, & \text{if } \Phi_k(k) = j. \end{cases}$$

Let $\mathcal{L} = (L_{\langle k, j \rangle})_{k, j \in \mathbb{N}}$. In order to prove that $\mathcal{L} \in \text{EMON}_{\text{prob}}(2/3)$, we define a PIM $P$ equipped with a 3-sided coin. Let $L \in \text{range}(\mathcal{L})$. Let $\tau \in \text{text}(L)$. Let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $x$ be a natural number, and let $m_x$ be the highest natural number with $a^k b^{m_x} \in \text{range}(\tau_x)$. Let $C \in \{0, 1, 2\}^\infty$ be a coin-oracle.

**PIM $P$:** On input $\tau_x$, $P^c$ works as follows.

(A) If $\Phi_k(k) > m_x$, then distinguish the following cases.

(A1) If $c_0 \in \{0, 1\}$, then set $P^{c_0}(\tau_x) = \langle k, 0 \rangle$.

(A2) If $c_0 = 2$, then output $\bot$.

(B) If $\Phi_k(k) \leq m_x$, then distinguish the following cases.

(B1) If $c_0 = 0$, then set $P^{c_0}(\tau_x) = \langle k, 0 \rangle$.

(B2) If $c_0 = 1, 2$, then output $\langle k, \Phi_k(k) \rangle$ provided $\langle k, \Phi_k(k) \rangle$ is consistent with $\tau_x$. Otherwise output $\langle k, 0 \rangle$.

end

Clearly, $P$ identifies $\mathcal{L}$ monotonically with probability $2/3$. Hence, it suffices to show, that $\mathcal{L}$ is not monotonically identifiable with respect to any hypothesis space comprising $\text{range}(\mathcal{L})$. Suppose, by way of contradiction, that there exist an IIM $M$ and a hypothesis space $\mathcal{G}$ such that $M$ $\text{CMON}$-identifies $\mathcal{L}$ with respect to $\mathcal{G}$. Let $k$ be natural number. Define the function $\mathcal{I} : \mathbb{N} \to \mathbb{N}$ by setting

$$\mathcal{I}(k) := \text{the least natural number } \ell \text{ with}$$

(1) $M(\tau^k_\ell) \neq \bot$, and
(2) $\tau_{\ell+2}^k \subseteq L(G_{M(\tau_\ell^k)})$.

Then $\mathcal{I}$ is total and recursive, since $M$ is assumed to identify $L_k$ for all $k \in \mathbb{N}$. Moreover, we can conclude that

$$\varphi_k(k) \neq \mathcal{I}(k) \text{ for all } k \in \mathbb{N} \text{ with } \varphi_k(k) \downarrow.$$ 

Otherwise, $L_{(k, \Phi_k(k))} = \{a^k b^m \mid m \leq \mathcal{I}(k) + 1\}$, and hence, $M(\tau_{\mathcal{I}(k)}^k)$ is an index for an overgeneralization of $L_{(k, \Phi_k(k))}$. In this case, $L_k$ is not monotonically identifiable with respect to $G$, since we can construct a text $\rho$ for $L_k$ with $\rho_{\mathcal{I}(k)} = \tau_{\mathcal{I}(k)}^k$, such that there exists $y_k, z_k \in \mathbb{N}$, $\mathcal{I}(k) < y_k < z_k$ with $M(\rho_{y_k})$ being an index for $L_{(k, \Phi_k(k))}$ and $M(\rho_{z_k})$ being an index for $L_k$. Consequently, $\varphi_k(k) \neq \mathcal{I}(k)$ for all $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$. However, there exists a $k_0 \in \mathbb{N}$ with $\varphi_{k_0}(k_0) = \mathcal{I}(k_0)$, a contradiction. Thus, $L \notin \text{CMON}$. \(\diamond\)

Notice that the PIM identifying $L$ with probability $2/3$ is allowed to change from an index for $L_k$ to an index for a finite language in case $\varphi_k(k) \downarrow$, i.e., there may be nonmonotonic paths in an infinite computation tree for $L_k$. From the following theorem, we can derive that every PIM identifying $L$ with a probability $p > 1/2$ has to allow nonmonotonotic paths.

**Theorem 2.3.13.** The probabilistic hierarchy $\langle \text{CMON}^\text{st}_{\text{prob}}(p) \rangle_{0 \leq p \leq 1}$ has a threshold beginning with $1/2$.

**PROOF.** The proof is similar to the proof of the theorems above and therefore omitted. \(\diamond\)

Theorem 2.3.12 and Theorem 2.3.13 allow the following corollary about the relationship between $\langle \text{CMON}_{\text{prob}}(p) \rangle_{0 \leq p \leq 1}$ and $\langle \text{CMON}^\text{st}_{\text{prob}}(p) \rangle_{0 \leq p \leq 1}$ in the interval $(1/2, 2/3]$.

**Corollary 2.3.14.** Let $1/2 < p \leq 2/3$. Then $\text{CMON}_{\text{prob}}(p) \setminus \text{CMON}^\text{st}_{\text{prob}}(p) \neq \emptyset$.

Another corollary which can be drawn from Theorem 2.3.12 concerns the relation between class preserving conservative and monotonic learning.

**Theorem 2.3.15.** There exists an indexed family $\mathcal{L}$ which is properly monotonically identifiable with probability $p = 2/3$ but not conservatively identifiable with probability $p > 1/2$ with respect to any class comprising hypothesis space.

**PROOF.** Define $\mathcal{L}$ as in Theorem 2.3.12. Obviously, $\mathcal{L}$ is conservatively identifiable with $p = 1/2$. Assume that $\mathcal{L}$ is conservatively identifiable with
a probability $p > 1/2$. As in the proof of Theorem 2.2.2, we define a total recursive function $I$ with $I(k) \neq \varphi_k(k)$ for all $k \in \mathbb{N}$ with $\varphi_k(k) \downarrow$. Let $k$ be a natural number. Let $\tau^k$ be the canonical text for $L_k$. Since $L_k \in \mathcal{L}$, there exists an $m_k \in \mathbb{N}$ such that $\tau^k_{m_k+2} \subseteq L(G_M(\tau^k_{m_k}))$. Then $\varphi_k(k) \neq m_k$, since otherwise $L(G_M(\tau^k_{m_k}))$ would be an overgeneralization of $\{a^kb^m \mid m \leq \varphi_k(k) + 1\}$. Set $I(k) := m_k$. Then $I$ has the desired properties.

By defining a slight variant of this indexed family, we can show that there is an indexed family which is properly conservatively identifiable but not monotonically identifiable with $p > 1/2$.

**Theorem 2.3.16.** There exists an indexed family $\mathcal{L}$ which is properly conservatively identifiable but not monotonically identifiable with probability $p > 2/3$ with respect to any class comprising hypothesis space.

**PROOF.** Let $\Sigma = \{a, b, d\}$. Let $\langle , \rangle : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$, and let $k \in \mathbb{N}$. Set $L_{(k,0)} := L_k$. Let $j \in \mathbb{N}$, and define

$$L_{(k,j)} := \begin{cases} L_k, & \text{if } j < \Phi_k(k), \\ \{a^kb^m \mid m \leq \varphi_k(k) + 1\} \cup \{d\}, & \text{if } j = \Phi_k(k), \\ \{a^kb^m \mid m \leq \varphi_k(k) + 2\} \cup \{d\}, & \text{if } j = \Phi_k(k) + 1. \end{cases}$$

Let $\mathcal{L} = (L_{(k,j)})_{k,j \in \mathbb{N}}$. By applying the proof technique used in Theorem 2.3.12, we can easily show that $\mathcal{L}$ witnesses the desired separation.

Finally, we note some results on class preserving probabilistic learning under monotonicity constraints with probability $p = 1$. As in subsection 2.1.3, it can be shown that $\lambda_{SMON_{prob}}(1) = \lambda_{SMON}$, and $\lambda_{COV_{prob}}(1) = \lambda_{COV}$ for $\lambda \in \{C, \varepsilon\}$. In the case of class preserving monotonic learning, we suggest that an analogous result to Theorem 2.1.8 holds.

**Conjecture 2.3.17.** $MON_{prob}(1) \setminus MON \neq \emptyset$

### 2.4 Conclusion

We can draw the following conclusion concerning the probabilistic hierarchies and the relationship between conservative and monotonic probabilistic learning.
1. For proper conservative, monotonic and strong-monotonic probabilistic learning, and for class preserving conservative probabilistic learning, the probabilistic hierarchies are dense in a sufficiently large neighborhood of 1. In the case of class preserving monotonic probabilistic learning, the probabilistic hierarchy is strictly decreasing to 1. Thus, these probabilistic learning models fulfill our intuition of probabilistic learning.

2. From Theorem 2.1.2 and Theorem 2.2.8 follows that proper monotonic probabilistic learning and proper conservative probabilistic learning are not comparable with respect to set inclusion. Theorem 2.3.15 and Theorem 2.3.16 imply the following relations between the probabilistic learning classes.

   (a) $EMON_{prob}(2/3) \nsubseteq CCOV_{prob}(p)$ for all $p > 1/2$.

   (b) $ECOV_{prob}(p) \nsubseteq CMON_{prob}(p)$ for all $p > 2/3$.

   However, we do not know whether $CMON_{prob}(p) \subseteq CCOV_{prob}(p)$ for $p > 2/3$. 
Chapter 3

Complexity aspects of probabilistic learning

We showed in the last chapter that in the case of probabilistic learning under monotonicity constraints, probabilistic learning is stronger than deterministic learning even if the probability is required to be close to 1. In this chapter, we are concerned with the question, how much information is necessary for compensating the additional power of the probabilistic learning machines. In the first part of this chapter, we investigate which oracles are adequate for compensating the power of the probabilistic learning machines. In the second part, we investigate the complexity of learning problems. More precisely, we deal with the following questions. Let $\mu \in \{COV, SMON, MON\}$, let $\lambda \in \{E, \epsilon, C\}$, and let $p \in [0, 1]$.

1. How powerful are the probabilistic learning models considered? Is there an oracle $A$ which is adequate for compensating the power of $\lambda\mu_{\text{prob}}(p)$-learning?

2. Let $L$ be an indexed family which is $\lambda\mu_{\text{prob}}(p)$-identifiable, but not $\lambda\mu$-identifiable. Is $L$ $A$-difficult for an oracle $A$?

3. Conservative and monotonic probabilistic learning are not comparable with respect to set inclusion. Can we compare these probabilistic learning models with respect to oracle-complexity.
3.1 Comparing the power of Probabilistic Learning and Oracle Identification

3.1.1 The power of the oracle $K$

The first results in this subsection concern the power of oracle machines having access to $K$. We start by showing that $K$ is sufficient for compensating the power of probabilistic learning under monotonicity constraints provided the probabilistic learning machines are required to learn with a probability $p > 1/2$ in the case of strong-monotonic and conservative learning, and $p > 2/3$ in the case of monotonic learning.

Theorem 3.1.1. Let $p > 1/2$, let $\mu \in \{\text{SMON}, \text{COV}\}$ be a monotonicity constraint, and let $L$ be an indexed family such that $L$ is $\mu_{\text{prob}}(p)$-identifiable with respect to a class preserving hypothesis space $G$. Then $L$ is $\mu$-identifiable with respect to $G$ by an oracle machine which has access to $K$.

PROOF. Stephan [96] proved that $K$ closes the gap between conservative learning and identification in the limit, i.e., $ECOV[K] = LIM$. From this, and from the fact that every indexed family which is identifiable with a probability $p > 1/2$ is deterministically identifiable (cf. Theorem 2.3.1), it follows that $K$ is sufficient for compensating the power of conservative probabilistic learning if $p > 1/2$.

Let $\mu = \text{SMON}$. Let $p \in [0,1]$, $p > 1/2$, and let $L \in \text{SMON}_{\text{prob}}(p)$ with respect to a class preserving hypothesis space $G$. Let $P$ be a PIM which $\text{SMON}$-identifies $L$ with probability $p$. Let $L \in \text{range}(L)$, let $\tau \in \text{text}(L)$, and let $T_{P,\tau}$ be the infinite computation tree induced by $P$ and $\tau$. Since $P$ $\text{SMON}$-identifies $L$ with probability $p > 1/2$, there exist a sequence $k_1, k_2, \ldots$ of natural numbers with $k_i < k_j$ for all $i, j \in \mathbb{N}$, and, for every $i \in \mathbb{N}$, a set of nodes $O_i \subseteq \text{level } k_i$ such that the sequence $(O_i)_{i \in \mathbb{N}}$ has the following properties:

1. $w(O_i) > 1/2$ for all $i \in \mathbb{N}$,
2. $L(G_{\text{ind}(o')}) = L(G_{\text{ind}(o)})$ for all $o, o' \in O_i$, and for all $i \in \mathbb{N}$, and
3. $L(G_{\text{ind}(o)}) \subseteq L(G_{\text{ind}(o')})$ for all $o \in O_i$, $o' \in O_j$, and for all $i, j \in \mathbb{N}$, $i < j$. 

93
Furthermore, there exists an $i \in \mathbb{N}$ such that for all $j \geq i$, and for all $o \in O_j$, $L(G_{\text{ind}(o)})$ is equal to $L$. Moreover, Condition (3) yields that, for $o_{k_i} \in O_i$, $i \in \mathbb{N}$, the sequence of languages $(L(G_{\text{ind}(o_{k_i})}))_{i \in \mathbb{N}}$ is strong-monotonic.

Obviously, we can enumerate a sequence $(k_i)_{i \in \mathbb{N}}$ fulfilling (1), (2) and (3) by asking questions to $\mathcal{K}$. Hence, we can define an OIM $M[\mathcal{K}]$ which $\text{SMON}$-identifies $\mathcal{L}$ with respect to $\mathcal{G}$ as follows. Let $L \in \text{range}(\mathcal{L})$, let $\tau \in \text{text}(L)$, and let $x \in \mathbb{N}$.

**OIM $M[\mathcal{K}]$:** On input $\tau_x$, $M[\mathcal{K}]$ works as follows.

If $x = 0$, then set $M[\mathcal{K}](\tau_x) := \bot$. If $x > 0$, and if $M[\mathcal{K}](\tau_{x-1})$ is consistent with $\tau_x$, then set $M[\mathcal{K}](\tau_x) := M[\mathcal{K}](\tau_{x-1})$.

Otherwise, use $\mathcal{K}$-oracle to compute the finite sequence $k_0, k_1, \ldots$. If $x < k_0$, then set $M[\mathcal{K}](\tau_x) := \bot$. Otherwise let $m$ be the greatest natural number with $k_m \leq x$. Test whether $\text{range}(\tau_x) \subseteq L(G_{\text{ind}(o)})$ for $o \in O_m$. If yes, then set $M[\mathcal{K}](\tau_x) := \text{ind}(o)$ for a node $o \in O_m$. Otherwise set $M[\mathcal{K}](\tau_x) := M[\mathcal{K}](\tau_{x-1})$.

end

Obviously, $M[\mathcal{K}]$ $\text{SMON}$-identifies $\mathcal{L}$ with respect to $\mathcal{G}$. ◦

In the case of monotonic learning, we can show an analogous result provided the probability $p$ is greater than $2/3$.

**Theorem 3.1.2.** Let $p > 2/3$, and let $\mathcal{L}$ be an indexed family such that $\mathcal{L}$ is $\text{MON}_{\text{prob}}(p)$-identifiable with respect to a class preserving hypothesis space $\mathcal{G}$. Then $\mathcal{L}$ is $\text{MON}$-identifiable with respect to $\mathcal{G}$ by an oracle machine which has access to $\mathcal{K}$.

**Proof.** Let $p > 2/3$, and let $\mathcal{L} \in \text{MON}_{\text{prob}}(p)$ with respect to a class preserving hypothesis space $\mathcal{G}$. Let $P$ be a PIM which $\text{MON}$-identifies $\mathcal{L}$ with probability $p$. Let $L \in \text{range}(\mathcal{L})$, let $\tau$ be a text for $L$, and let $T_{P,\tau}$ be the infinite computation tree induced by $P$ and $\tau$. Since $P$ $\text{MON}$-identifies $\mathcal{L}$ with probability $p > 2/3$, there exists a sequence $k_1, k_2, \ldots$ of natural numbers with $k_i < k_j$ for all $i, j \in \mathbb{N}$, and, for every $i \in \mathbb{N}$, a set of nodes $O_i \subseteq \text{level } k_i$ such that the sequence $(O_i)_{i \in \mathbb{N}}$ has the following properties:

1. $w(O_i) > 2/3$ for all $i \in \mathbb{N}$, and

2. $L(G_{\text{ind}(o)}) = L(G_{\text{ind}(o')})$ for all $o, o' \in O_i$, and for all $i \in \mathbb{N}$.
Clearly, we can enumerate an infinite sequence \((k_i)_{i \in \mathbb{N}}\) fulfilling (1) and (2) by using the \(K\)-oracle. Furthermore, there exists an \(i \in \mathbb{N}\) such that for all \(j \geq i\), and for all \(o \in O_j\), \(L(G_{ind(o)}) = L\). Finally, we have to show that for \(o_{k_i} \in O_i\), \(i \in \mathbb{N}\), the sequence of languages \((L(G_{ind(o_{k_i})}))_{i \in \mathbb{N}}\) is monotonic. Let \(i, j \in \mathbb{N}\), \(i < j\). Let \(\text{succ}_j(O_i)\) be the set of nodes on level \(k_j\) which have a predecessor in \(O_i\). Then the weight of the set \(O_j \cap \text{succ}_j(O_i)\) is greater than \(1/3\). Thus, \(L(G_{ind(o)}) \cap L \subseteq L(G_{ind(o')}) \cap L\) for all \(o \in O_i\), and \(o' \in O_j \cap \text{succ}_j(O_i)\). Consequently, the sequence \((L(G_{ind(o_{k_i})}))_{i \in \mathbb{N}}\) is monotonic for all \(o_{k_i} \in O_i\), and all \(i \in \mathbb{N}\). Hence, an identifying OIM may be constructed as in Theorem 3.1.1.

Since in both proofs the hypothesis space has not been changed, we immediately get the following corollary for proper and class preserving probabilistic learning.

**Corollary 3.1.3.** Let \(\lambda \in \{E, \varepsilon\}\).

(a) \(\lambda \text{SMON}_{prob}(p) \subseteq \lambda \text{SMON}[K]\) for all \(p > 1/2\).

(b) \(\lambda \text{COV}_{prob}(p) \subseteq \lambda \text{COV}[K]\) for all \(p > 1/2\).

(c) \(\lambda \text{MON}_{prob}(p) \subseteq \lambda \text{MON}[K]\) for all \(p > 2/3\).

Obviously, \(K\) is sufficient for compensating the power of class comprising conservative and class comprising strong-monotonic probabilistic learning with probability \(p > 1/2\), and class comprising monotonic probabilistic learning with \(p > 2/3\), since there is no gain of learning power in the case of class comprising learning (cf. Section 2.3). Hence, \(K\) is sufficient for compensating the power of conservative and strong-monotonic probabilistic learning with \(p > 1/2\), and monotonic probabilistic learning with \(p > 2/3\).

By using the proof techniques developed in Section 2.2, we show that \(K\) is adequate for compensating the power of conservative probabilistic learning with probability \(p > 1/2\), monotonic probabilistic learning with probability \(p > 2/3\), and strong-monotonic probabilistic learning with probability \(1/2 < p \leq 2/3\).

**Theorem 3.1.4.** Let \(c, d \in \mathbb{N}\), \(1 \leq c < d\). Then there exists an indexed family \(L_{c,d}^c \subseteq E\text{COV}_{prob}(\frac{d-c}{d+c})\) such that (a) and (b) are satisfied.

(a) For all oracles \(B, \mathbb{K} \leq_T B\), there exists an oracle machine \(M[\ ]\) such that \(M[B]\) COV-identifies \(L_{c,d}^c\).
(b) For every oracle machine $M[\cdot]$, and for every oracle $B$, the following holds: if $M[B]$ \textit{COV}-identifies $\mathcal{L}_c^d$, then $\mathcal{K} \leq_T B$.

\textbf{PROOF.} We define $\mathcal{L}_c^d$ as in the proof of Theorem 2.2.2. For convenience, we repeat the construction. Let $\Sigma := \{a, b\}$. Let $c, d \in \mathbb{N}, c < d$. Let $(\ , \ )$ be an effective encoding. Let $(D_i)_{i \in \mathbb{N}}, i \leq (d)_c$ be an enumeration of all subsets of $\{0, \ldots, d-1\}$ of cardinality $c$. Let $k$ be a natural number. In case $\varphi_k(k) \downarrow$, set $g_k := \Phi_k(k) + \varphi_k(k) + 1$. Define $\mathcal{L}_c^d = (L_{(k,j)})_{j \in \mathbb{N}}$. Let $j \in \mathbb{N}$, and let $z, r \in \mathbb{N}, r \leq (d)_c - 1$ with $j = (d)_c z + r$.

1. If $\varphi_k(k) \downarrow$, and $z \leq \Phi_k(k) - 1$, then set
   \[
   L_{(k,j)} := \{a^k b^n \mid n \leq g_k\} \cup \{a^k b^{g_k+1} \mid v \in D_r\}.
   \]

2. If $\varphi_k(k) \downarrow$, $\Phi_k(k) \leq z < 2\Phi_k(k)$, and $r \leq d - 1$, then set
   \[
   L_{(k,j)} := \{a^k b^m \mid m \leq \varphi_k(k) + 1\} \cup \{a^k b^{\varphi_k(k)+r+1}\}.
   \]

3. If $\varphi_k(k) \downarrow$ and $z \geq 2\Phi_k(k)$, or $\varphi_k(k) \uparrow$, then set $L_{(k,j)} = L_k$.

We already showed in Theorem 2.2.2, that $\mathcal{L}_c^d = (L_{(k,j)})_{k,j \in \mathbb{N}}$ is an indexed family which is properly conservatively identifiable with probability $d/(d+c)$. Since $d/(d+c) > 1/2$, Theorem 3.1.1 yields that $\mathcal{L}_c^d$ is properly conservatively identifiable.

Finally, let $B$ be an oracle, and let $M[B]$ be an oracle machine having access to $B$ which conservatively identifies $\mathcal{L}_c^d$ with respect to a class preserving hypothesis space $G$. Obviously, $L_k \in \mathcal{L}_c^d$. Let $\tau^k$ be the canonical text for $L_k$. Then there must be an $n_0$ such that $\tau^k_{n_0+2} \subseteq L(G_M(\tau^k_{n_0}))$. Notice that $L(G_M(\tau^k_{n_0})) \in \mathcal{L}_c^d$. Hence, $\varphi_k(k) \uparrow$ or $\Phi_k(k) \leq n_0$, since otherwise there exists a language $L$ in $\mathcal{L}_c^d$ containing range($\tau^k_{n_0}$) with $L$ being a proper subset of $L(G_M(\tau^k_{n_0}))$. Thus, $\mathcal{K} \leq_T B$. By applying Theorem 3.1.1 and Theorem 3.1.4, we can conclude that $\mathcal{K}$ is adequate for compensating the power of conservative probabilistic learning with $p > 1/2$.

By generalizing the proof technique used in Theorem 2.2.2, we can prove the following corollary.

\textbf{Corollary 3.1.5.} Let $c, d \in \mathbb{N}, 1 \leq c < d$. Then there exists an indexed family $\mathcal{L}_c^d \in \text{ECOV}_{\text{prob}}(d_{c+d})$ with the following property. Let $P[\cdot]$ be a probabilistic oracle machine, and let $B$ be an oracle. If $P[B]$ conservatively identifies
$\mathcal{L}_{c,d}^e$ with a probability $p > \frac{d}{c+d}$ with respect to a class preserving hypothesis space, then $\mathcal{K} \leq_T B$, i.e., the knowledge of $\mathcal{K}$ is necessary even if we only want to enhance the probability.

The following theorem generalizes Theorem 3.1.4 in another direction.

**Theorem 3.1.6.** Let $A \in \mathcal{RE} \setminus \mathcal{REC}$ be an oracle. Let $n \in \mathbb{N}$, $n \geq 1$. Then there exists an indexed family $\mathcal{L}_A^n \in \text{ECOV}_{\text{prob}}(\frac{n}{n+1})$ such that (a) and (b) are satisfied.

(a) For all oracles $B$, $A \leq_T B$, there exists an oracle machine $M[ ]$ such that $M[B]$ ECOV-identifies $\mathcal{L}_A^n$.

(b) For every oracle machine $M[ ]$, and for every oracle $B$, the following holds: if $M[B]$ COV-identifies $\mathcal{L}_A^n$, then $A \leq_T B$.

**PROOF.** Let $E_A$ be an effective algorithm enumerating $A$. Let $\Sigma := \{a, b\}$. Let $n \in \mathbb{N}$, $n \geq 1$. Let $\langle, \rangle$ be an effective encoding, and let $k, j \in \mathbb{N}$. Let $z, r \in \mathbb{N}$, $r \leq n-1$, with $j = nz + r$. In the following definition of $L_{\langle k, j \rangle}$, we use the languages $L_k$ and $L_A^k$ which were defined in Section 2.2.

1. If $k \in A$ and $z \leq E_A^{-1}(k) - 1$, then set
   \[ L_{\langle k, j \rangle} := L_A^k \cup \{a^k b^{E_A^{-1}(k)+r+1}\}, \]

2. If $k \in A$ and $E_A^{-1}(k) \leq z < 2E_A^{-1}(k)$, then set
   \[ L_{\langle k, j \rangle} := \{a^k b^m \mid m \leq 2E_A^{-1}(k) - z\} \cup \{a^k b^{E_A^{-1}(k)+r+1}\}, \]

3. If $k \in A$ and $z \geq 2E_A^{-1}(k)$, or $k \notin A$, then set $L_{\langle k, j \rangle} := L_k$.

Then $\mathcal{L}_A^n = (L_{\langle k, j \rangle})_{k, j \in \mathbb{N}}$ is an indexed family witnessing the desired separation. We show that $\mathcal{L}_A^n$ is an indexed family which is properly conservatively identifiable by an oracle machine having access to $A$. An identifying oracle machine can be defined as follows. Let $L \in \text{range}(\mathcal{L}_A^n)$ be a language, and let $\tau \in \text{text}(L)$, and let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $x$ be a natural number.

**OIM $M[A]$:** On input $\tau_x$, $M[A]$ works as follows.

If $x = 0$, then $M[A]$ tests whether $x \in A$ or not. If $x \notin A$, then set $M[A](\tau_x) := \langle k, 0 \rangle$. If $x \in A$, then set $M[A](\tau_x) := \bot$.

If $x > 0$, and $x \notin A$, then set $M[A](\tau_x) := \langle k, 0 \rangle$. If $x > 0$, and $x \in A$, then
test whether \( \text{range}(\tau_x) \subseteq I^k \). In case it is, set \( M[A](\tau_x) := \perp \). Otherwise distinguish the following cases.

(A) If \( M[A](\tau_{x-1}) \) is consistent with \( \tau_x \), then set \( M[A](\tau_x) := M[A](\tau_{x-1}) \).

(B) If \( M[A](\tau_{x-1}) \) is not consistent with \( \tau_x \), then \( M[A] \) tests whether there are numbers \( z, r \in \mathbb{N} \), \( E^{-1}_A(k) \leq z < 2E^{-1}_A(k) \), \( r \leq n - 1 \), such that \( L_{(k,nz+r)} \) contains \( \text{range}(\tau_x) \). If yes, then \( M[A] \) searches for the greatest number \( z_{\text{max}} < 2E^{-1}_A(k) \) such that an \( r \leq n - 1 \) exists with \( \text{range}(\tau_x) \subseteq L_{(k,nz_{\text{max}}+r)} \). Set \( M[A](\tau_x) := \langle k, nz_{\text{max}}+r \rangle \). If there are no such numbers \( z \) and \( r \), then \( M[A] \) guesses \( \langle k, j \rangle \) where \( j \) is the smallest natural number such that \( \text{range}(\tau_x) \subseteq L_{(k,j)} \).

\[ \text{end} \]

It is easy to see, that \( M[A] \) properly conservatively identifies \( \mathcal{L}_A^n \). The rest of the proof is analogous to the proof of Theorem 3.1.4 for \( d = n \) and \( c = 1 \).

\[ \diamond \]

For monotonic learning, we can prove an analogous result for every oracle \( A \).

**Theorem 3.1.7.** Let \( A \in \mathcal{RE} \setminus \mathcal{REC} \) be an oracle. Let \( c, d \in \mathbb{N} \) such that \( c+1 \) is a factor of \( d \). Then there exists an indexed family \( \mathcal{L}^{c,d}_A \in \text{EMON}_{\text{prob}}(\frac{2d}{2d+c}) \) such that such that (a) and (b) are satisfied.

(a) For all oracles \( B \), \( A \leq_T B \), there exists an oracle machine \( M[ ] \) such that \( M[B] \) MON-identifies \( \mathcal{L}^{c,d}_A \).

(b) For every oracle machine \( M[ ] \), and for every oracle \( B \), the following holds: if \( M[B] \) MON-identifies \( \mathcal{L}^{c,d}_A \), then \( A \leq_T B \).

**Proof.** We define an indexed family as in Theorem 2.2.4. Let \( \Sigma := \{a, b\} \). Let \( E_A \) be an algorithm enumerating \( A \). Let \( z \in \mathbb{N} \) with \( n = z(c+1) \). Let \( D' = \{m \in \mathbb{N} | r(c+1) \leq m < (r+1)(c+1)\} \).

Let \( (D'_i)_{i \leq \binom{c+1}{2}} \) be an effective enumeration of all subsets of \( D_r \) with cardinality 2. Let \( \langle \ , \rangle : \mathbb{N} \times \{0, \ldots, (n + z(\binom{c+1}{2}) - 1) \} \rightarrow \mathbb{N} \) be an effective encoding. Let \( k, j \in \mathbb{N} \), \( j \leq (d + z(\binom{c+1}{2})) - 1 \).

(1) If \( k \not\in A \), then \( L_{(k,j)} := L_k \) for all \( j \leq (d + z(\binom{c+1}{2})) - 1 \).
(2) If $k \in A$, and $j \leq n - 1$, then $L_{(k,j)} := L_k^A \cup \{a^kb^{E_A^{-1}(k)+(j+1)}\}.$

(3) If $k \in A$, and $j \geq n$, then let $r \in \mathbb{N}$, $0 \leq r \leq z - 1$, with $d + r(c + 1) \leq j < d + (r + 1)(c + 1)$, and set

$$L_{(k,j)} := L_k^A \cup \{a^kb^{E_A^{-1}(k)+(m+1)} \mid m \in D^r_{j-(d+r(c+1))}\}.$$

It is easy to see that $\mathcal{L}_A^{c,d} := (L_{(k,j)})_{k,j \in \mathbb{N}, j \leq n^2(c+1)-1}$ is an indexed family.

As in Theorem 2.2.4, it can be shown that $\mathcal{L}_A^{c,d}$ is properly monotonically identifiable with probability $p = 2d/(2d+c)$. It remains to show that $\mathcal{L}_A^{c,d}$ is monotonically identifiable by an oracle machine having access to $A$. Define an OIM as follows. Let $L \in range(\mathcal{L}_A^{c,d})$ be a language, and let $\tau \in text(L)$. Let $k \in \mathbb{N}$ with $range(\tau_0) \subset L_k$.

**OIM $M[A]$**: On input $\tau_x$, $M[A]$ works as follows.

If $x = 0$, then $M[A]$ tests whether $k \in A$ or not. If $k \not\in A$, then set $M[A](\tau_x) := \langle k, 0 \rangle$. If $k \in A$, then set $M[A](\tau_x) := \perp$.

If $x > 0$ and $k \not\in A$, then set $M[A](\tau_x) := \langle k, 0 \rangle$. If $x > 0$ and $k \in A$, then test whether $range(\tau_x) \subseteq L_k^A$. In case it is, set $M[A](\tau_x) := \perp$. Otherwise distinguish the following cases. If $M[A](\tau_x-1)$ is consistent with $\tau_x$, then set $M[A](\tau_x) := M[A](\tau_x-1)$. If $M[A](\tau_x-1)$ is not consistent with $\tau_x$, then $M[A]$ searches for the smallest natural number $j$ such that $range(\tau_x) \subseteq L_{(k,j)}$. Set $M[A](\tau_x) := \langle k, j \rangle$.

end

Then $M[A]$ monotonically identifies $\mathcal{L}_A^{c,d}$. Now let $B$ be an oracle, and let $M[B]$ be an oracle machine having access to $B$ which monotonically identifies $\mathcal{L}_A^{c,d}$ with respect to a class preserving hypothesis space $G$. Let $L \in range(\mathcal{L}_A^{c,d})$ be a language, and let $k \in \mathbb{N}$, with $L \cap L_k \neq \emptyset$. Let $(\sigma_i)_{i \in \mathbb{N}}$ be the sequence of all finite sequences of strings from $\Sigma^*$ with $range(\sigma_i) \subset L_k$.

Let $x \in \mathbb{N}$ be the smallest natural number with $M[B](\sigma_x) \neq \perp$. Such a natural number exists, since all languages in $\mathcal{L}_A^{c,d}$ are subsets of $L_k$. Furthermore, let $m \in \mathbb{N}$ be the greatest natural number with $a^kb^m \in range(\sigma_x)$. In the following, we show that $E_A^{-1}(k) \leq m$ or $k \not\in A$. Assume that $k \in A$ and $E_A^{-1}(k) \geq m + 1$. Then

$$range(\sigma_x) \subseteq L_k^A.$$
Thus, our assumption leads to a contradiction. It follows that
\[ E \]

By generalizing the proof techniques developed in [78], Section 2.1 and Section 2.2, we can even show the following result.

**Theorem 3.1.8.** Let \( A \in \mathcal{RE} \setminus \mathcal{RNC} \) be an oracle. Let \( c, d \in \mathbb{N} \) such that \( c + 1 \) is a factor of \( n \), and \( \frac{2d}{2d+c} \geq 4/5 \). Then there exists an indexed family \( \mathcal{L}^c_d(A) \in \text{EMON}_{\text{prob}}(\frac{2d}{2d+c}) \) such that the following holds for every probabilistic oracle machine \( P \) and every oracle \( B \). If \( P[B] \) MON-identifies \( \mathcal{L}^c_d(A) \) with a probability \( p > \frac{2d}{2d+c} \) with respect to a class preserving hypothesis space, then \( A \leq_T B \).

From Theorem 3.1.6, and Theorem 3.1.7 immediately follows that every probabilistic learning class \( \lambda_{\text{prob}}(p) \), \( \lambda \in \{ E, \epsilon \} \), \( \mu \in \{ \text{COV}, \text{MON} \} \) contains learning problems which are \( \lambda \mu \)-identifiable by an OIM \( M[B] \) if and only if \( M[B] \) has access to \( K \).

**Corollary 3.1.9.** Let \( A \in \mathcal{RE} \setminus \mathcal{RNC} \) be an oracle. Let \( \mu \in \{ \text{COV}, \text{MON} \} \), and let \( \lambda \in \{ E, \epsilon \} \). Let \( p < 1 \). Then there exists an indexed family \( \mathcal{L}^p(A) \) such that the following holds for every oracle machine \( M[ \ ] \) and every oracle \( B \). If \( M[B] \) \( \lambda \mu \)-identifies \( \mathcal{L}^p(A) \), then \( A \leq_T B \).

In particular, there exists an indexed family \( \mathcal{L}^p(K) \in \lambda_{\text{prob}}(p) \) such that the following holds for every oracle machine \( M[ \ ] \). If \( M[K] \) \( \lambda \mu \)-identifies \( \mathcal{L}^p(K) \), then \( K \leq_T B \).
Consequently, $K$ is adequate for compensating the power of proper and class preserving conservative and monotonic learning, respectively, with probability $p > 1/2$ and $p > 2/3$, respectively. In particular, no weaker oracle is sufficient.

**Corollary 3.1.10.** Let $A$ be an oracle with $A <_T K$, and let $p < 1$. Then

(a) $ECOV_{prob}(p) \setminus ECOV[A] \neq \emptyset$, and

(b) $EMON_{prob}(p) \setminus MON[A] \neq \emptyset$.

Obviously, strong-monotonic probabilistic learning is weaker than conservative and monotonic probabilistic learning, respectively, with respect to set inclusion, i.e., every indexed family, which is strong-monotonically identifiable with probability $p$, is conservatively and monotonically identifiable with probability $p$. In the following, we show that the weakness of strong-monotonic probabilistic learning can be expressed in terms of Turing complexity, i.e., strong-monotonic probabilistic learning is weaker than conservative or monotonic probabilistic learning in the sense that there are low oracles which are sufficient for compensating the power of strong-monotonic probabilistic learning with probability $p > 2/3$. For the definition of low oracles see [95].

In the case of strong-monotonic probabilistic learning with $1/2 < p \leq 2/3$, however, we need $K$ for compensating the power of strong-monotonic probabilistic learning. This follows from the fact that an indexed family which is strong-monotonically identifiable with $p = 2/3$ but not with a higher probability, may contain two languages $L_1, L_2$ such that the question “$L_1 = L_2$?” is Turing equivalent to the halting problem. In particular, the probabilistic learning class $ESMON_{prob}(2/3)$ is able to encode every recursively enumerable oracle.

**Theorem 3.1.11.** Let $A \in \mathcal{RE} \setminus \mathcal{REC}$ be an oracle. Then there exists an indexed family $\mathcal{L}^{2/3}_A \in ESMON_{prob}(2/3)$, $\mathcal{L}^{2/3}_A \in ESMON[A]$, such that the following holds for every oracle machine $M[ ]$, and for every oracle $B$: if $M[B]$ $SMON$-identifies $\mathcal{L}^{2/3}_A$, then $A \leq_T B$.

**PROOF.** Let $A \in \mathcal{RE} \setminus \mathcal{REC}$ be an oracle. Let $E_A$ be an algorithm which enumerates $A$. Let $\langle , \rangle: \mathbb{N} \times \{0, 1\} \rightarrow \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \{0, 1\}$, and let $k, j \in \mathbb{N}, j \in \{0, 1\}$. Set

$$L_{\langle k, j \rangle} := L^A_k \cup \{a^k b^{E_A^{-1}(k) + 1 + j}\}.$$
Obviously, \( L^{2/3}_A = (L_{(k,j)})_{k,j \in \mathbb{N}, j \leq 1} \) is an indexed family. Furthermore, \( L^{2/3}_A \in ESMON_{prob}(2/3) \). Since we proved this part of the proof in Theorem 2.2.9 for \( A = \mathcal{K} \), we omit the proof here. Next we show that \( L^{2/3}_A \) is properly strong-monotonically identifiable by an oracle machine having access to \( A \).

An identifying oracle machine may be defined as follows. Let \( L \in \text{range}(L^{2/3}_A) \) be a language, and let \( \tau \in \text{text}(L) \). Let \( k \in \mathbb{N} \) with \( \text{range}(\tau_0) \subseteq L_k \), and let \( x \in \mathbb{N} \).

OIM \( M[A] \): On input \( \tau_x \), \( M[A] \) works as follows.

If \( k \not\in A \), then set \( M[A](\tau_0) := \langle k, 0 \rangle \). If \( k \in A \), and if \( \text{range}(\tau_x) \neq L_{(k,j)} \) for every \( j \in \{0,1\} \), then set \( M[A](\tau_x) := \bot \). If \( \text{range}(\tau_x) = L_{(k,j)} \) for a \( j \in \{0,1\} \), then set \( M[A](\tau_x) := \langle k,j \rangle \).

Obviously, \( M[A] \) SMON-identifies \( L^{2/3}_A \). Notice that \( M[A] \) is a finite learner, since it only outputs one hypothesis. It remains to show that \( L^{2/3}_A \) characterizes \( A \). Let \( B \) be an oracle, and let \( M[B] \) be an OIM having access to \( B \) which strong-monotonically learns \( L^{2/3}_A \) with respect to a class preserving hypothesis space \( \mathcal{G} \). Let \( \tau \) be a text for a language \( L \in \text{range}(L^{2/3}_A) \). Let \( x \in \mathbb{N} \) be the smallest number with \( M[B](\tau_x) \neq \bot \), and let \( m \in \mathbb{N} \) be the greatest number with \( a^kb^m \in \text{range}(\tau_x) \). Then, \( k \not\in A \) or \( E^{-1}_A(k) \leq m - 1 \), since otherwise there exists a language \( L \in \text{range}(L^{2/3}_A) \) containing \( \text{range}(\tau_x) \) with \( L(G(M(\tau_x))) \not\subseteq L \). Hence, our Theorem is proved.

From the Theorems 3.1.4, 3.1.7, and 3.1.11, we can draw the following corollary which shows that, for every recursively enumerable oracle \( A \), every oracle learning class \( \lambda \mu[A], \mu \in \{COV, MON, SMON\} \), contains an indexed family which is \( \lambda \mu \)-identifiable by an OIM \( M[B] \) if and only if \( A \leq_T B \).

**Corollary 3.1.12.** Let \( A \in \mathcal{RE} \setminus \mathcal{RE}_C \) be an oracle. Let \( \lambda \in \{E, \varepsilon\} \), and let \( \mu \in \{COV, SMON, MON\} \). Then there exists an indexed family \( \mathcal{L}_A \) such that for all oracles \( B \) holds:

\[
\mathcal{L}_A \in \lambda \mu[B] \quad \text{if and only if} \quad A \leq_T B.
\]

In particular, \( \lambda \mu[A] \setminus \lambda \mu \neq \emptyset \).

The next theorem shows that every Peano-complete oracle is sufficient for compensating the additional power of probabilistic learning machines in the
case of proper strong-monotonic probabilistic learning with \( p > 2/3 \). Hence, \( K \) is not adequate for compensating the power of proper strong-monotonic learning with probability \( p > 2/3 \). Remember in the case of class preserving probabilistic learning, \( SMON_{\text{prob}}(p) = SMON \) for all \( p > 2/3 \) (cf. Theorem 2.2.9).

**Theorem 3.1.13.**

Let \( \mathcal{L} \) be an indexed family such that \( \mathcal{L} \) is \( ESMON_{\text{prob}}(p) \)-identifiable with a probability \( p > 2/3 \). Let \( A \) be a Peano-complete oracle. Then \( \mathcal{L} \) is \( ESMON \)-identifiable by an oracle machine which has access to \( A \).

**PROOF.** Let \( \mathcal{L} \) be an indexed family such that \( L \in \bigcup_{2/3 < p < 1} ESMON_{\text{prob}}(p) \). Let \( P \) be a PIM such that \( P \ ESMON_{\text{prob}}(p) \)-identifies \( \mathcal{L} \) with a probability \( p > 2/3 \). Let \( L \in \text{range}(\mathcal{L}) \) be a language, and let \( \tau = (\tau_x)_{x \in \mathbb{N}} \) be a text for \( L \). Let \( x \in \mathbb{N} \) with

\[
    w\{o \in \text{level } x \mid \text{ind}(o) \neq \bot \text{ and range}(\tau_x) \subseteq L(G_{\text{ind}(o)})\} > 2/3.
\]

Notice that the weight of the set \( \{o \in \text{level } y \mid \text{ind}(o) \neq \bot \text{ and range}(\tau_y) \subseteq L(G_{\text{ind}(o)})\} \) is greater than 2/3 for almost all \( y \in \mathbb{N} \), since \( L \) is strongly-monotonically identifiable with probability \( p > 2/3 \).

Since \( L \) is strongly-monotonically identifiable with probability \( p > 2/3 \), there is a minimal language \( L^x_{\text{min}} \) on level \( x \) of \( T_{P,\tau} \) with the following property (cf. Theorem 2.2.9). Let \( o \in \text{level } x \). If \( \text{range}(\tau_x) \subseteq L(G_{\text{ind}(o)}) \), then \( L^x_{\text{min}} \subseteq L(G_{\text{ind}(o)}) \). Moreover, \( L^x_{\text{min}} \subseteq L \), and there are less than \( 1/3 \cdot 2^x \)-many nodes on level \( x \) with \( L_{\text{ind}(o)} \) being a proper superset of \( L^x_{\text{min}} \). We show that it is possible to find a node \( o \in \text{level } x \) with \( L_{\text{ind}(o)} = L^x_{\text{min}} \) by asking questions to \( A \).

Let \((o_h)_{2^x \leq h \leq 2^{x+1} - 1}\) be an effective enumeration of all nodes on level \( x \) of \( T_{P,\tau} \). Set

\[
    N_x := \{h \in \{2^x, \ldots, 2^{x+1} - 1\} \mid \text{ind}(o_h) \neq \bot, \text{range}(\tau_x) \subseteq L(G_{\text{ind}(o_h)})\}.
\]

Let \((w_i)_{i \in \mathbb{N}}\) be an effective enumeration of \( \Sigma^* \), and let \( W_n := \{w_0, \ldots, w_n\} \) be the set containing the first \( n + 1 \) elements of \( \Sigma^* \) in this enumeration. For \( x, t \in \mathbb{N}, x \leq t \), let \( M_{t,x} \) be the set of all pairs of natural numbers \((t, h), h \in N_x\) such that:

1. \( L_{\text{ind}(o_h)} \cap W_{t'} \subseteq L_{\text{ind}(o_l)} \cap W_{t'} \) for all \( t' < t \), and for all \( l \in N_x, l \neq h \), and
(2) there exists an \( l \in N_x \) with \( L_{\text{ind}(o_l)} \cap W_t \setminus \bigcup_{l \text{ empty}} W_t \neq \emptyset \).

Obviously, if \((t,h) \in M_{t,x}\), then \( L_{\text{ind}(o_l)} \) is a proper superset of the minimal language \( L_{\text{min}}^x \). Since there are less than \( 1/3 \cdot 2^x \)-many nodes \( o \) on level \( x \) with \( L_{\text{min}}^x \subset L(G_{\text{ind}(o)}) \), the set

\[
OG_x := \{ h \in N_x \mid \exists t \in \mathbb{N} : (t,h) \in M_{t,x} \}
\]

has cardinality \(< 1/3 \cdot 2^x \), and because of condition (1) - \( \mid \bigcup_{t \in \mathbb{N}} M_{t,x} \mid < 1/3 \cdot 2^x \). Moreover, there exists an effective procedure \( M \) which enumerates the elements of \( \cup_{t \in \mathbb{N}} M_{t,x} \) for every \( x \in \mathbb{N} \) without repetitions. For the \( m \)-th element appearing in the list \( M(x) \), we write \( M(x)_m \). Notice that \( M(x) \) may be empty.

Now we define for every \( x,m \in \mathbb{N} \), \( m \leq (1/3 \cdot 2^x) - 1 \), a recursive function \( F_{x,m} : N_x \rightarrow \{0,1\} \). Let \( m \leq (1/3 \cdot 2^x) - 1 \), and let \( h \in N_x \):

\[
F_{x,m}(h) := \begin{cases} 
1, & \text{if } \exists t \text{ with } (t,h) \in M_{t,x}, (t,h) = (M(x))_m, \\
0, & \text{if } \exists t,h' \text{ with } (t,h') \in M_{t,x}, (t,h') = (M(x))_m, h \neq h', \\
\uparrow, & \text{otherwise.}
\end{cases}
\]

Intuitively, the functions defined above are links to the overgeneralizations of \( L_{\text{min}}^x \). Note that \( F_{x,m}(h) = 0 \) if there is an \( m \)-th element in the list \( M(x) \) which is not equal to \((t,h)\). Obviously, \( F_{x,m} \) is partial recursive for all \( m \leq (1/3 \cdot 2^x) - 1 \). Moreover, \( F_{x,m} \) has the following properties.

(1) Either \( F_{x,m}(h) \uparrow \) for all \( h \in N_x \), or \( F_{x,m}(h) \) is defined for all \( h \in N_x \).

(2) If there exists an \( m \)-th element in \( M_x \), then \( F_{x,m}(h) = 1 \) for exactly one \( h \in N_x \), and \( F_{x,m}(h') = 0 \) for all \( h' \in N_x \), \( h' \neq h \).

(3) If \( L(G_{\text{ind}(o_l)}) \) is an overgeneralization of \( L_{\text{min}}^x \), then there exists an \( m \leq (1/3 \cdot 2^x) - 1 \) with \( F_{x,m}(h) = 1 \), and \( F_{x,m}(h') = 0 \) for all \( h' \in N_x \), \( h' \neq h \).

Let \( m \leq (1/3 \cdot 2^x) - 1 \). Since \( A \) is Peano-complete, there exists a total, \( A \)-recursive extension \( F^{A}_{x,m} \) of \( F_{x,m} \). Then the following holds: Whenever \( h_1,h_2 \in N_x \), \( h_1 \neq h_2 \), and \( F^{A}_{x,m}(h_1) = F^{A}_{x,m}(h_2) = 1 \), then \( \mid \bigcup_{t \in \mathbb{N}} M_{t,x} \mid < m \), and \( F_{x,m}(h) = \infty \) for all \( h \in N_x \), since otherwise \( F^{A}_{x,m}(h) = F_{x,m}(h) < \infty \).
for all $h \in N_x$. However, we cannot conclude that $| \bigcup_{t \in \mathbb{N}} M_{t,x} | \leq m$ in case $F^A_{\tau_x,m}(h) = 1$ for exactly one $h \in N_x$.

Assume that for every function $F^A_{\tau_x,m}$, $m \in \{0, \ldots, 1/3 \cdot 2^x\}$, there are $h_1, h_2 \in N_x$ with $F^A_{\tau_x,m}(h_1) = F^A_{\tau_x,m}(h_2) = 1$. Then, as argued above, $F_{\tau_x,m}(h) = \infty$ for all $m \in \{0, \ldots, 1/3 \cdot 2^x\}$, and $h \in N_x$. Thus, all languages on level $x$ are equal to $L^x_{\text{min}}$. Otherwise let $F^A_{\tau_x,m_1}, \ldots, F^A_{\tau_x,m_r}$, $m_{i_0}, \ldots, m_{i_r} \in \{0, \ldots, 1/3 \cdot 2^x\}$, be the set of functions with $F^A_{\tau_x,m_{i_v}}(h) = 1$ for at most one $h \in N_x$. Then there exists an $h \in \mathbb{N}_x$ with $F^A_{\tau_x,m_{i_v}}(h) = 0$ for all $v \leq r$. It follows that

$$L_{\text{ind}(o_h)} = L^x_{\text{min}}.$$  

Since Theorem 2.2.9 yields that the sequence of minimal languages is strongly-monotonic and converges to $L$, it is easy to see that an OIM $M[A]$ exists which properly strongly-monotonically identifies $L$. $\Diamond$

Hence, every Peano-complete oracle is sufficient for compensating the power of strong-monotonic probabilistic learning with $p > 2/3$. Now the question arises whether the property to be Peano-complete is necessary in order to compensate the power of proper strong-monotonic probabilistic learning. We conjecture that this is not true, since it is not possible to encode the problem $\varphi_k(k) \equiv i \mod 2$, $k \in \mathbb{N}$, $i \in \{0,1\}$, in a learning problem which is strongly-monotonically identifiable with probability $p > 2/3$.

Since there exist low Peano-complete oracles, we draw the following corollary.

**Corollary 3.1.14.** There exists a low oracle which is sufficient for compensating the power of proper strong-monotonic learning with probability $p > 2/3$.

**PROOF.** The corollary follows from Theorem 3.1.13 and the fact that there exist low Peano-complete oracles (cf. [77]). $\Diamond$

### 3.1.2 When $K$ is not sufficient

We already showed that in the case of strong-monotonic probabilistic learning, the probability $p = 2/3$ plays an important role, since, for $p > 2/3$, an arbitrary Peano-complete oracle is sufficient for compensating the power of the probabilistic learning machines, whereas for $p \leq 2/3$, $K$ is sufficient and necessary. In this subsection, we show some bound results for conservative and monotonic probabilistic learning. In particular, we prove that $K$ is not
sufficient for compensating the power of strong-monotonic and conservative probabilistic learning with \( p = 1/2 \), and monotonic probabilistic learning with \( p = 2/3 \). For conservative learning, it turns out that conservative learning which \( \mathcal{K} \)-oracle is incomparable with conservative probabilistic learning with \( p = 1/2 \).

**Theorem 3.1.15.** Let \( \lambda \in \{ E, \varepsilon, C \} \), \( \mu \in \{ COV, SMON, MON \} \).

(a) \( ESMON_{prob}(1/2) \setminus CLIM[\mathcal{K}] \neq \emptyset \).

(b) \( ESMON[\mathcal{K}] \setminus CCOV_{prob}(1/2) \neq \emptyset \).

**PROOF.** The proof of (a) is straightforward. We define an indexed family \( \mathcal{L} \) consisting of an infinite language \( L \) and all finite subsets of \( L \). Obviously, \( \mathcal{L} \) is \( ESMON \)-identifiable with \( p = 1/2 \). In Gold [48], it was shown that \( \mathcal{L} \) is not in \( CLIM[A] \) for any oracle \( A \). The second part of the theorem is a corollary from Theorem 3.1.23 which will be shown in the next subsection.

Next we show that the bound given in Theorem 3.1.2 is strict.

**Theorem 3.1.16.** There exists an indexed family \( \mathcal{L}^{2/3} \in ECOV \) with

(a) \( \mathcal{L}^{2/3} \in EMON_{prob}(2/3) \),

(b) \( \mathcal{L}^{2/3} \not\in CMON[A] \) for all oracles \( A \).

**PROOF.** In order to prove the theorem, we define an indexed family as follows. Let \( \Sigma := \{ a, d \} \).

1. \( L_0 := \{ a^k \mid k \in \mathbb{N} \} \)

2. Let \( j \in \mathbb{N}, j \geq 1 \). Distinguish the following cases.

   (2.1) If \( j \) is odd, then set \( L_j := \{ a^n \mid n \leq j \} \cup \{ d^j \} \).

   (2.2) If \( j \) is even, then set \( L_j := \{ a^n \mid n \leq j \} \cup \{ d^{j-1} \} \).

Set \( \mathcal{L}^{2/3} = (L_{(k,j)})_{k,j \in \mathbb{N}} \). Obviously, \( \mathcal{L}^{2/3} \) is conservatively identifiable. In order to show that \( \mathcal{L}^{2/3} \) is monotonically identifiable with \( p = 2/3 \), define a 3-
sided PIM as follows. Let $L \in \text{range}(L^{2/3})$ be a language, and let $\tau \in \text{text}(L)$. Let $C$ be a coin-oracle, and let $x \in \mathbb{N}$.

PIM $P$: On input $\tau, x$, $P^C$ works as follows.

Assume $x = 0$. Then set $P^{c_0}(\tau_0) := 0$ if and only if $c_0 \leq 1$. If $c_0 = 2$, then set $P^{c_0}(\tau_0) := \bot$.

If $x > 0$, then distinguish the following cases.

(A) If $P^{c_{x-1}}(\tau_{x-1})$ is consistent with $\tau_x$, then set $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$.

(B) If $P^{c_{x-1}}(\tau_{x-1})$ is not consistent with $\tau_x$, then $P^C$ searches for the smallest natural number $j \geq 1$ such that $L_j$ contains the actual text. Distinguish the following cases.

(B1) If $j$ is odd, then set $P^{c_x}(\tau_x) := j$ if and only if $c_0 \in \{0, 2\}$. If $c_0 = 1$, then set $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$.

(B2) If $j$ is even, then set $P^{c_x}(\tau_x) := j$.

end

Then $P$ EMON$_{prob}(2/3)$-identifies $L^{2/3}$. Finally, let $A$ be an oracle, and let $M[A]$ be an OIM which monotonically identifies $L^{2/3}$ with respect to a class comprising hypothesis space $G$. Let $k \in \mathbb{N}$, and let $\tau$ be the canonical text for $L_0$. Since $M[A]$ CMON-identifies $L_0$, there must be an $n \in \mathbb{N}$ such that $L(G_{M[A]}(\tau_n))$ contains $\text{range}(\tau_n)$, and two strings $a^j, a^{j+1}, j \in \mathbb{N}$ odd, which are not contained in $\text{range}(\tau_n)$. Let $\tau'$ be a text for $L_j$ with $\tau_n \subsetneq \tau'$. Then there must be an $n' > n$ with $M[A](\tau_n') = j$. Now it is easy to see that $M[A]$ cannot identify $L_{j+1}$ monotonically. ⋄

We can summarize our results in the following corollary.

**Corollary 3.1.17.** Let $\lambda \in \{E, \varepsilon\}$.

(a) $O(\lambda COV_{prob}(p)) = \deg_T(K)$ for all $p > 1/2$.

(b) $O(\lambda MON_{prob}(p)) = \deg_T(K)$ for all $p > 2/3$.

(c) $O(\lambda SMON_{prob}(p)) < \deg_T(K)$ for all $p > 2/3$.

(d) For $1/2 < p \leq 2/3$, $O(\lambda SMON_{prob}(p)) = \deg_T(K)$.
3.1.3 The power of Peano-complete oracles

In the setting of learning under monotonicity constraints, every recursively enumerable oracle strictly enhances the learning power (cf. Theorem 3.1.9). Now we show that similar results hold for Peano-complete oracles. First we show that the probabilistic learning class \( ECOV_{\text{prob}}(p) \) contains a learning problem which separates \( ECOV_{\text{prob}}(p) \) from \( ECOV_{\text{prob}}(q) \), \( q > p \), and which is properly conservatively identifiable by an oracle machine having access to a Peano-complete oracle. The same result holds for proper monotonic probabilistic learning.

**Theorem 3.1.18.**

(a) Let \( c, d \in \mathbb{N} \) such that \( 1 > \frac{c}{d} > \frac{1}{2} \), and \( \gcd(c, d) = 1 \). Then there exists an indexed family \( L_{c,d} \in ESMON_{\text{prob}}(\frac{c}{d}) \setminus \bigcup_{q > \frac{c}{d}} ECOV_{\text{prob}}(q) \), and \( L_{c,d} \in ESMON[B] \) for all oracles \( B \), \( B \) Peano-complete.

(b) Let \( c, d \in \mathbb{N} \) such that \( \frac{2c}{d} > \frac{2}{3} \), and \( \gcd(c, d) = 1 \). Then there exists an indexed family \( L'_{c,d} \in ESMON_{\text{prob}}(\frac{2c}{d}) \setminus \bigcup_{q > \frac{2c}{d}} ESMON_{\text{prob}}(q) \), and \( L'_{c,d} \in EMON[B] \) for all oracles \( B \), \( B \) Peano-complete.

**PROOF.** We define an indexed family \( L_{c/d} = (L_{k,j})_{k,j \in \mathbb{N}, j \leq c-1} \) witnessing part (a) as in the proof of Theorem 2.1.1. Define the functions \( mod_{2c-d}^c \) and \( cod_{2c-d}^c \) as in the proof of Theorem 2.1.1. Let \( \langle \ , \rangle : \mathbb{N} \times \{0, \ldots, c-1\} \rightarrow \mathbb{N} \) be an effective encoding of \( \mathbb{N} \times \{0, \ldots, c-1\} \). For \( k, j \in \mathbb{N}, j \leq c-1 \), define \( L_{(k,j)} \) as follows. Let \( k, j \in \mathbb{N}, j \leq c-1 \).

\[
L_{(k,j)} := \begin{cases} 
L'_{k}, & \text{if } \varphi_k(k) \downarrow \wedge j \in (\text{mod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))), \\
L_k, & \text{if } \varphi_k(k) \downarrow \wedge j \notin (\text{mod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))), \\
L_k, & \text{if } \varphi_k(k) \uparrow .
\end{cases}
\]

We already proved that \( L_{c,d} \in ESMON_{\text{prob}}(\frac{c}{d}) \setminus \bigcup_{q > \frac{c}{d}} ECOV_{\text{prob}}(q) \). Moreover, \( L_{c,d} \) can be strong-monotonically identified by an oracle machine having access to a Peano-complete oracle \( B \).

For \( k \in \mathbb{N} \) define \( F_k : \{0, \ldots, c-1\} \rightarrow \mathbb{N} \) as follows. Let \( j \in \{0, \ldots, c-1\} \).

\[
F_k(j) := \begin{cases} 
0, & \text{if } \varphi_k(k) \downarrow \wedge j \in (\text{mod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))), \\
1, & \text{if } \varphi_k(k) \downarrow \wedge j \notin (\text{mod}_{2c-d}^c)^{-1}(\text{mod}_{2c-d}^c(\varphi_k(k))), \\
\infty, & \text{if } \varphi_k(k) \uparrow .
\end{cases}
\]

Let \( B \) be a Peano-complete oracle. Then \( F_k \) can be extended to a total, \( B \)-recursive function \( F_k^B \) which has the following properties.
(A) If $F_B^k(j) \neq 0$ for all $j \in \{0, \ldots, c - 1\}$, then $\varphi_k(k) \uparrow$, since otherwise there exists an $j \in \{0, \ldots, c - 1\}$ with $F_k(j) = F_B^k(j) = 0$. In this case, every language $L_{(k,j)} = L_k$.

(B) If there exists a $j \in \{0, \ldots, c - 1\}$ with $F_B^k(j) = 0$, then either $\varphi_k(k) \uparrow$, or $\varphi_k(k) \downarrow$ and $j \in (\text{cod}_{2c-d}^2)^{-1}(\text{mod}_{2c-d}(\varphi_k(k)))$.

Thus, we can define an identifying OIM $M[B]$ as follows. Let $L \in \text{range}(L_{c,d})$ be a language, and let $\tau \in \text{text}(L)$. Let $k \in \mathbb{N}$ with $\text{range}(\tau_0) \subset L_k$. Let $x \in \mathbb{N}$.

OIM $M[B]$: On input $\tau_x$, $M[B]$ works as follows.

If $x = 0$, then test whether or not there exists a $j \in \{0, \ldots, c - 1\}$ with $F_B^k(j) = 0$. If yes, then set $M[B](\tau_x) := (k, j)$. Otherwise, set $M[B](\tau_x) := (k, 0)$.

If $x > 0$, and if $M[B](\tau_{x-1})$ is consistent with $\tau_x$, then set $M[B](\tau_x) := M[B](\tau_{x-1})$. Otherwise set $M[B](\tau_x) := (k, j)$ where $j$ is the smallest natural number such that $(k, j)$ is consistent with $\tau_x$.

end

An indexed family witnessing the second separation can be defined as in the proof of Theorem 2.1.7. ✓

By applying Theorem 3.1.18, we can derive the corollary that every Peano-complete oracle enhances the learning power in the case of proper strong-monotonic, conservative and monotonic learning.

Corollary 3.1.19. Let $A$ be a Peano-complete oracle. Then $E_{\mu}[A] \setminus E_{\mu} \neq \emptyset$ for all $\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}$.

In the following, we show that this result also holds for class preserving conservative and monotonic learning.

Theorem 3.1.20. Let $A$ be a Peano-complete oracle. Then there exists an indexed family $\mathcal{L}$ with the following properties:

(a) $\mathcal{L} \in \text{ECOV}[A]$,

(b) $\mathcal{L} \in \text{ECOV}_{\text{prob}}(2/3) \setminus \cup_{p>2/3} \text{COV}_{\text{prob}}(p)$,
PROOF. Let $\Sigma := \{a, b, d\}$. Let $(\cdot, \cdot, \cdot) : \mathbb{N} \times \mathbb{N} \times \{0, 1\} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N} \times \{0, 1\}$. Let $k, j \in \mathbb{N}$. Define $L_{(k,j)}$ as follows.

1. Set
   
   $$L_{(k,0)} := L_k \cup \{a^kd\};$$

   and set
   
   $$L_{(k,1)} := L_k \cup \{a^kd^2\}.$$

2. If $1 < j < \Phi_k(k)$, then distinguish the following cases. If $j$ is even, then set
   
   $$L_{(k,j)} := L_k \cup \{a^kd\}.$$  

   If $j$ is odd, then set
   
   $$L_{(k,j)} := L_k \cup \{a^kd^2\}.$$

   If $j \geq \Phi_k(k)$, then distinguish the following cases.

   (2.1) If $\varphi_k(k) \equiv 0 \mod 2$, then set
   
   $$L_{(k,j)} := \{a^kb^n \mid n \leq \varphi_k(k)\} \cup \{a^kd\}.$$

   (2.2) If $\varphi_k(k) \equiv 1 \mod 2$, then set
   
   $$L_{(k,j)} := \{a^kb^n \mid n \leq \varphi_k(k)\} \cup \{a^kd^2\}.$$

Then $L = (L_{(k,j,v)})_{k,j,v \in \mathbb{N}, v \in \{0,1\}}$ fulfills the requirement. The proof is similar to the proofs in Section 2.2. Thus, we only give a sketch of the proof. Clearly, $L$ is $ECOV$-identifiable by any OIM $M[A]$, where $A$ is Peano-complete.

For proving the remaining part of the theorem, we restrict ourselves to the deterministic case, i.e., we show that $L \notin COV$. The proof for probabilistic machines is analogous. Now let $M$ be an IIM which $COV$-identifies $L$. We define a total recursive function $I$ with $I(k) \neq \varphi_k(k)$ for all $k \in \mathbb{N}$. $I$ is defined as follows. For $i \in \{0, 1\}$, let $\tau_i = a^kb^i, a^kb^0, a^kb^1, a^kb^2, \ldots$ be a text for $L_{(k,i)}$. Since $M COV$-identifies $L$, there is an $n_0^i \in \mathbb{N}$ with $range(\tau_{n_0^i+3}) \subseteq L(G_M(\tau_{n_0^i}))$. Then the following holds:

1. Assume that $\varphi_k(k) \downarrow$, and $\varphi_k(k)$ is even. Then either $n_0^0 + 1$ or $n_0^0 + 2$ are even. Without loss of generality, we may assume that $n_0^0 + 1$ is even. Then $\varphi_k(k) \neq n_0^0 + 1$, since otherwise, there exists a language in $L$ which is a proper subset of $L(G_M(\tau_{n_0^0}))$ and contains $range(\tau_{n_0^0})$.

2. Assume that $\varphi_k(k) \downarrow$, and $\varphi_k(k)$ is odd. Then either $n_0^1 + 1$ or $n_0^1 + 2$ are odd. Without loss of generality, we may assume that $n_0^1 + 1$ is odd. Then $\varphi_k(k) \neq n_0^1 + 1$, since otherwise, there exists a language in $L$ which is a proper subset of $L(G_M(\tau_{n_0^1}))$, and contains $range(\tau_{n_0^1})$.

Now it is easy to see how to define $I$. ◇

For monotonic learning, an analogous result holds.
Theorem 3.1.21. Let $A$ be a Peano-complete oracle. Then there exists an indexed family $\mathcal{L}$ with the following properties:

(a) $\mathcal{L} \in \text{EMON}[A],$

(b) $\mathcal{L} \in \text{EMON}_{\text{prob}}(4/5) \setminus \cup_{p>4/5} \text{MON}_{\text{prob}}(p).

PROOF. Let $\Sigma := \{a, b, d\}$. Let $\langle , , , \rangle : \mathbb{N} \times \mathbb{N} \times \{0,1\} \to \mathbb{N}$ be an effective encoding. Let $k, j \in \mathbb{N}$. Define $L_{(k,j)}$ as follows.

(1) Set

$$L_{(k,0)} := L'_k \cup \{a^kb^{\Phi_k(k)+1}, a^kb^{\Phi_k(k)+2}\} \cup \{d\},$$

and set

$$L_{(k,1)} := L'_k \cup \{a^kb^{\Phi_k(k)+3}, a^kb^{\Phi_k(k)+4}\} \cup \{d^2\}.$$

(2) If $1 < j < \Phi_k(k)$, then distinguish the following cases. If $j$ is even, then set $L_{(k,j)} := L_{(k,0)}$. If $j$ is odd, then set $L_{(k,j)} := L_{(k,1)}$.

(3) If $j = \Phi_k(k)$, then distinguish the following cases.

(3.1) If $\varphi_k(k) \equiv 0 \mod 2$, then set $L_{(k,j)} := L'_k \cup \{a^kb^{\Phi_k(k)+1}\} \cup \{d\}$, and $L_{(k,j+1)} := L'_k \cup \{a^kb^{\Phi_k(k)+2}\} \cup \{d\}.

(3.2) If $\varphi_k(k) \equiv 1 \mod 2$, then set $L_{(k,j)} := L'_k \cup \{a^kb^{\Phi_k(k)+3}\} \cup \{d^2\}$, and $L_{(k,j+1)} := L'_k \cup \{a^kb^{\Phi_k(k)+4}\} \cup \{d^2\}.$

(4) If $j > \Phi_k(k) + 1$, then distinguish the following cases. If $j$ is even, then set $L_{(k,j)} := L_{(k,\Phi_k(k))}$. If $j$ is odd, then set $L_{(k,j)} := L_{(k,\Phi_k(k)+1)}$.

Then $\mathcal{L} = (L_{(k,j,v)})_{k,j,v \in \mathbb{N}, v \in \{0,1\}}$ fulfills the requirements. The proof is similar to the proof of Theorem 2.2.4. Clearly, $\mathcal{L}$ is $\text{EMON}$-identifiable by any OIM $M[A]$, where $A$ is Peano-complete. For the remaining part of the proof, we only give a sketch of the deterministic case. Let $M$ be an IIM which $\text{MON}$-identifies $\mathcal{L}$. Let $i \in \mathbb{N}$, and let $\tau^i = d^i, a^kb^ia^kb^2, \ldots$ be a text for $L_{(k,i)}$. Then there exist $n_1, n_2 \in \mathbb{N}$ such that $M(\tau^i_{n_1}) \not\equiv \bot$. Since $M$ works monotonically on $\tau^i$, $i \in \{0,1\}$, we can conclude that, for all $k \in \mathbb{N}$, the following holds:

- If $\varphi_k(k) \downarrow$, and $\varphi_k(k) \equiv 0 \mod 2$, then $\Phi_k(k) \leq n_0 - 1$.
- If $\varphi_k(k) \downarrow$, and $\varphi_k(k) \equiv 1 \mod 2$, then $\Phi_k(k) \leq n_1 - 1$.  

111
Hence, if $\varphi_k(k) \downarrow$, then $\Phi_k(k) \leq \max(n_0 - 1, n_1 - 1)$, a contradiction. ⊁

Hence, Peano-complete oracles enhance the learning power in the case of class preserving conservative and monotonic learning.

**Corollary 3.1.22.** Let $A$ be a Peano-complete oracle. Then $\mu[A] \setminus \mu \neq \emptyset$ for $\mu \in \{\text{COV}, \text{MON}\}$.

### 3.1.4 Incomparability of probabilistic learning classes and oracle learning classes

In the last part of this section, we investigate whether there is an equivalence between an oracle learning class $\lambda \mu[A]$ and a probabilistic learning class $\lambda \mu_{\text{prob}}(p)$ for $p > 1/2$ ($p > 2/3$ in the monotonic case). Such an equivalence holds for example in the case of noisy learning and finite learning with $K$-oracle (cf. [96]). More exactly, Stephan showed that, for learning with informant, noisy learning is equivalent to finite learning with $K$-oracle.

From Corollary 3.1.9, we already can conclude that there is no oracle below $K$ which characterizes a monotonic or conservative probabilistic learning class, or a strong-monotonic learning class $\lambda \text{SMON}_{\text{prob}}(p)$ for $p \leq 2/3$. Thus, $K$ is the only possible candidate for a characterization. However, it turns out that an indexed family exists, which is $\text{ESMON}$-identifiable with $K$-oracle but not identifiable by any probabilistic machine fulfilling a monotonicity constraint.

In order to illustrate the power of the oracle machines, we show in the next theorem that, for every recursively enumerable oracle $A$, there exists an indexed family $\mathcal{L}_A$ which is strong-monotonically identifiable by an oracle machine having access to $A$, but not conservatively or monotonically identifiable with any probability $p > 0$ with respect to any class comprising hypothesis space $G$.

**Theorem 3.1.23.** Let $A \in \mathcal{RE} \setminus \mathcal{REC}$ be an oracle. There exists an indexed family $\mathcal{L}_A$ with

(a) $\mathcal{L}_A \in \text{ESMON}[A]$,
(b) $\mathcal{L}_A \notin \text{CCOV}_{\text{prob}}(p)$ for all $p > 0$,
(c) $\mathcal{L}_A \notin \text{CMON}_{\text{prob}}(p)$ for all $p > 0$.  

112
Before proving Theorem 3.1.23, we note a technical result.

**Lemma 3.1.24.** Let $\mu \in \{\text{SMON}, \text{MON}, \text{COV}\}$. Let $\mathcal{L}$ be an indexed family, let $A$ be an oracle, and let $\mathcal{L} \in \text{ESMON}[A] \setminus C_{\mu_{\text{prob}}}(p)$ for some probability $p < 1$. Then there exists an indexed family $\mathcal{L}'$ such that $\mathcal{L}' \in \text{ESMON}[A] \setminus C_{\mu}(q)$ for every $q \in [0, 1]$, $q > 0$.

**Proof.** Let $\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}$. Let $\mathcal{L} = (L_i)_{i \in \mathbb{N}}$ be an indexed family, and let $A$ be an oracle. Suppose $\mathcal{L} \in \text{ESMON}[A] \setminus C_{\mu_{\text{prob}}}(p)$ for some probability $p < 1$. Let $M[A]$ be an OIM which properly strong-monotonically identifies $\mathcal{L}$.

Let $(E_n)_{n \in \mathbb{N}}$ be an effective enumeration of the set of all nonempty finite subsets of $\mathbb{N}$. Let $\langle \ , \ \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding. Let $n \in \mathbb{N}$. Then define

$$\mathcal{T}_n := \{\langle i, x \rangle \mid i \in E_n, x \in L_i\}.$$ 

Hence, $\mathcal{T}_n$ is the finite join of the languages $\{L_i \mid i \in E_n\}$.

Then $\mathcal{L} = (\mathcal{T}_n)_{n \in \mathbb{N}}$ is an indexed family witnessing the desired separation. In order to prove that $\mathcal{L}$ is properly strong-monotonically identifiable with oracle $A$, we define an OIM $M[A]$ which uses the original algorithm $M[A]$ for each component language. More exactly, let $\mathcal{L} \in \mathcal{L}$, and let $\tau = (\langle k_i, x_i \rangle)_{i \in \mathbb{N}}$ be a text for $\mathcal{L}$. Let $k \in \mathbb{N}$ such that an $x \in \mathbb{N}$ exists with $\langle k, x \rangle \in \text{range}(\tau)$. For $n \in \mathbb{N}$, set $\sigma^n_k := (\langle k_i, x_i \rangle)_{i \leq n, k_i = k}$.

Obviously, $\sigma^n_k$ converges to a text for a language $L$ in $\mathcal{L}$. Since $M[A]$ ESMON-identifies $\mathcal{L}$, the sequence $(M[A](\sigma^n_k))_{n \in \mathbb{N}}$ converges to a correct hypothesis for $L$. Since $\mathcal{L}$ has only finitely many components, $M[A]$ identifies $\mathcal{L}$.

Finally, we have to show that $\mathcal{L}$ is not $C_{\mu_{\text{prob}}}(p)$-identifiable with any probability $q > 0$. This is due to the fact that the number of components of a language $\mathcal{L} \in \mathcal{L}$ is finite but arbitrary large. Every component may be viewed as a copy of $\mathcal{L}$. Since $\mathcal{L}$ is not $C_{\mu}$-identifiable with probability $p$, the join of $n$ copies of $\mathcal{L}$ is not identifiable with probability $r > p^n$. Hence, $\mathcal{L}$ is not $C_{\mu_{\text{prob}}}$-identifiable with any probability $q > 0$. $\diamond$

**Proof.** (of Theorem 3.1.23)

It is sufficient to prove part (b) of the theorem for $p > 1/2$, and part (c) for $p > 2/3$, since Lemma 3.1.24 yields the result for arbitrary $p \in [0, 1]$.

---

1Thereby, $\langle \ , \ \rangle : \mathbb{N} \to \mathbb{N}$ is defined to be the projection on the second component of $\langle \ , \ \rangle$. 

113
Let \( A \in \mathcal{RE} \setminus \mathcal{REC} \). Let \( E_A \) be an algorithm enumerating \( A \). By \( E_A(n) \), we denote the \( n \)-th element of \( A \) generated by \( E_A \). Define an indexed family \( L_A \) as follows. Let \( \langle \cdot, \cdot \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N} \) be an effective encoding. Then set

1. \( L_{\langle k,j \rangle} := L_k \) if and only if \( k \notin \{ E_A(1), \ldots, E_A(j) \} \).

2. In case \( E_A(m) = k \) for an \( m \in \mathbb{N} \), add all subsets of \( \{ a^i b^j \mid i \leq m + 1 \} \) to \( L_A \). More formally, let \( \{ L'_n \mid n \leq 2^{m+1} - 2 \} \) be the set of all nonempty finite subsets of \( \{ a^i b^j \mid i \leq m + 1 \} \). Then set \( L_{\langle k,j \rangle} := L'_n \) for \( j = m + n, n \leq 2^{m+1} - 2 \). If \( j \geq m + 2^{m+1} - 1 \), then set \( L_{\langle k,j \rangle} := \{ a^k b^0 \} \).\(^2\)

Obviously, \( L_A := \langle L_{\langle k,j \rangle} \rangle_{k,j \in \mathbb{N}} \) is an indexed family. Moreover, \( L_A \) is strongly-monotonically identifiable by an OIM having access to \( A \). Now suppose that \( L_A \) is conservatively identifiable by a PIM \( P \) with \( p > 1/2 \) with respect to a class comprising hypothesis space \( \mathcal{G} \). Let \( \tau^k \) be the canonical text for \( L_k \). Then there is an \( n_0 \in \mathbb{N} \) such that \( P \) guesses an overgeneralization of \( \text{range}(\tau^k_{n_0}) \) with probability \( p > 1/2 \). Consequently, \( E_A^{-1}(k) \leq n_0 \) or \( k \notin A \), since otherwise \( P \) could not identify the language \( \text{range}(\tau^k_{n_0}) \) with \( p > 1/2 \).

In the monotonic case, the algorithm may overgeneralize on a certain number of paths. However, every path which contains an overgeneralization for the language to be learned is not monotonic for the infinite language \( L_k \). Thus, we cannot identify \( L_A \) monotonically with probability \( p > 2/3 \). \( \diamond \)

Note that the indexed family defined in the proof of Theorem 3.1.23 is in every oracle learning class \( \lambda \mu[A], \lambda \in \{ E, \epsilon, C \}, \mu \in \{ COV, SMON, MON \} \), but not in the probabilistic learning classes \( \lambda \mu_{\text{prob}}(p), \lambda \in \{ E, \epsilon, C \}, \mu \in \{ COV, SMON, MON \} \). Hence, we can conclude from Theorem 3.1.23, and Corollary 3.1.9 that conservative probabilistic learning, strong-monotonic probabilistic learning, and monotonic probabilistic learning cannot be characterized in terms of oracle identification.

**Corollary 3.1.25.** Let \( p < 1 \).

(a) Let \( \lambda \in \{ E, \epsilon \} \), let \( \mu \in \{ COV, MON \} \), and let \( A \in \mathcal{RE} \setminus \mathcal{REC} \) be an oracle with \( A <_{T} \mathcal{K} \). Then \( \lambda \mu_{\text{prob}}(p) \) and \( \lambda \mu[A] \) are incomparable.

(b) Let \( \lambda, \lambda' \in \{ E, \epsilon, C \} \), and let \( \mu, \mu' \in \{ COV, SMON, MON \} \). Then \( \lambda \mu_{\text{prob}}(p) \neq \lambda' \mu'[\mathcal{K}] \).

\(^2\)For a similar construction for \( A = \mathcal{K} \), we refer the reader to [68].
In particular, we can draw the following corollary concerning class comprising oracle identification under monotonicity constraints.

**Corollary 3.1.26.** Let \( \mu \in \{\text{COV}, \text{SMON}, \text{MON}\} \). Let \( A \in \mathcal{RE} \setminus \mathcal{REC} \) be an oracle. Then \( C\mu[A] \setminus C\mu \neq \emptyset \).

Since the indexed families constructed in Theorem 3.1.23 are \( C\mu \)-identifiable by an OIM \( M[B] \) if and only if \( A \leq_T B \), we can conclude an analogous result to Corollary 3.1.12.

**Corollary 3.1.27.**
Let \( A \in \mathcal{RE} \setminus \mathcal{REC} \) be an oracle. Let \( \mu \in \{\text{COV}, \text{SMON}, \text{MON}\} \). Then there exists an indexed family \( \mathcal{L}_A \) such that the following holds for all oracles \( B \). \( \mathcal{L}_A \in C\mu[B] \) if and only if \( A \leq_T B \).

**Proof.** The proof is similar to the proof of Theorem 3.1.23, and therefore omitted. \( \diamond \)

In [96], Stephan proved that \( \text{LIM}[A] = \text{COV}[\mathcal{K}] \) for every low recursively enumerable oracle. Thus, \( \text{LIM}[A] = \text{LIM}[B] \) for any two low recursively enumerable oracles \( A, B \). In the case of inductive inference under monotonicity constraints, we can conclude that \( \mu[A] \neq \mu[B] \) for all recursively enumerable oracles \( A, B \) with \( A \not\equiv_T B \) (cf. Theorem 3.1.9). Furthermore, we can draw the following corollary.

**Corollary 3.1.28.** Let \( \mu \in \{\text{COV}, \text{SMON}, \text{MON}\} \). Let \( A, B \) be oracles. If \( A \) is recursively enumerable, then \( \mu[A] \subseteq \mu[B] \) if and only if \( A \leq_T B \).

### 3.2 Oracle Complexity

#### 3.2.1 Characterization of oracles below \( \mathcal{K} \)

In the last section, it has been shown that, for every \( \lambda \in \{E, \varepsilon\} \), and every \( p < 1 \), the probabilistic learning class \( \lambda \text{COV}_{\text{prob}}(p) \) contains a learning problem \( \mathcal{L}_\mathcal{K} \) which is \( \mathcal{K} \)-difficult in the sense that a conservative learning algorithm needs a database of power \( \mathcal{K} \) in order to \( \lambda \text{COV} \)-identify \( \mathcal{L}_\mathcal{K} \). The same holds for \( \lambda \text{MON} \)-learning with \( p < 1 \), and strong-monotonic learning with \( p \leq 2/3 \). In particular, every oracle machine \( M[B] \) which \( \lambda \mu \)-identifies \( \mathcal{L}_\mathcal{K} \) can be transformed into a decision procedure for \( \mathcal{K} \) (cf. Definition 3.2.1).
Analogous results were proved for every recursive enumerable oracle (cf. Theorem 3.1.4, 3.1.7 and 3.1.11).

Let $\mu \in \{COV, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $p \in [0, 1]$. We investigate whether analogous results hold for every oracle $A \leq_T K$, i.e., whether there exists an indexed family which is $A$-difficult with respect to $\lambda \mu$ (cf. Definition 1.2.12). On the other hand, we address the question, whether every indexed family in $\lambda \mu_{\text{prob}}(p)$, $\mu \in \{COV, \text{SMON}, \text{MON}\}$, $\lambda \in \{E, \epsilon\}$, characterizes an oracle, or, in other words, whether it is possible to express the complexity of every indexed family in a probabilistic learning class $\lambda \mu_{\text{prob}}(p)$ in terms of Turing complexity. More formally, we deal with the following questions. Let $\mu \in \{COV, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $p \in [0, 1]$.

(1) Let $A \leq_T K$, $A$ not recursive. Is there an indexed family $L \in \lambda \mu_{\text{prob}}(p)$ with $O_{\lambda \mu}(L_A) = \{B \mid A \leq_T B\}$?

(2) Is $O_{\lambda \mu}(L)$ $\lambda \mu$-simple for every indexed family $L \in \lambda \mu_{\text{prob}}(p)$?

For recursively enumerable oracles, we already answered the first question positively, since the Theorems 3.1.4, 3.1.7, and 3.1.11 yield the following corollary.

**Corollary 3.2.1.**

Let $A \in \mathcal{RE} \setminus \mathcal{REC}$ be an oracle. Let $\mu \in \{COV, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$. Then $\lambda \mu[A]$ contains an indexed family $\mathcal{L}$ with oracle-complexity $O_{\lambda \mu} = \{B \mid A \leq_T B\}$.

In the following, we will see that in the case of conservative learning, this result can be extended to arbitrary oracles.

**Theorem 3.2.2.**

Let $A \notin \mathcal{REC}$ be an oracle with $A \leq_T K$. Then there exists an indexed family $L_A \in \text{ECOV}_{\text{prob}}(\frac{1}{2})$ which is $A$-difficult with respect to ECOV.

**Proof.** Let $A \notin \mathcal{REC}$ be an oracle with $A \leq_T K$. For $A \equiv_T K$, the result follows from Theorem 3.1.4. Assume $A <_T K$. By the Limit Lemma (cf. [95]), there exists a sequence of recursive sets $(A_i)_{i \in \mathbb{N}}$ such that for all $k \in \mathbb{N}$:

$$\lim_{i \to \infty} \chi_{A_i}(k) = \chi_A(k).$$
In Soare [95] it is shown that this sequence is uniformly recursive, i.e., there exists a recursive function \( F : \mathbb{N} \times \mathbb{N} \to \mathbb{N} \) such that \( \lim_{i \to \infty} F(k, i) = \chi_A(k) \) for all \( k \in \mathbb{N} \). Hence, the sequence may be used to define an indexed family.

In the following, we assume that \( A_0 = A_1 = \mathbb{N} \), and, for all \( n \geq 1 \), \( A_j = A_{2^n-1} \) for all \( j \in \{2^{n-1}+1, \ldots, 2^n-1\} \), i.e., \( A_2 = A_3, A_4 = \ldots = A_7, A_8 = \ldots = A_{15} \) and so on.

Since the construction is very technical, we first give an informal description of the indexed family \( \mathcal{L}_A \) fulfilling the requirement.

1. For all \( k \in \mathbb{N} \), \( \mathcal{L}_A \) contains a sequence of languages \( (L_{(k,0,z)})_{z \in \mathbb{N}} \), and a sequence of languages \( (L_{(k,1,z)})_{z \in \mathbb{N}} \) such that the following holds. If \( k \in A \), then for every language \( L_{(k,0,z)} \), there exists a language \( L_{(k,1,z')} \) which is a proper subset of \( L_{(k,0,z)} \). If \( k \notin A \), then for every language \( L_{(k,1,z)} \), there exists a language \( L_{(k,0,z')} \) which is a proper subset of \( L_{(k,1,z)} \).

2. Let \( M[B] \) be an oracle machine identifying \( \mathcal{L}_A \) conservatively. Then, for all \( k \in \mathbb{N} \), \( M[B] \) can compute a finite sequence of strings \( \sigma^k \) with

- \( M[B](\sigma^k) \neq \bot \), and
- \( k \in A \) if and only if \( M[B](\sigma^k) = (k, 1, z) \) for a \( z \in \mathbb{N} \).

Before defining an indexed family with these properties, we will introduce some notations. For \( k \in \mathbb{N} \) let \( W_k : \mathbb{N} \to \mathbb{N} \) be the recursive function which counts the number of changes between 0 and 1 in the finite sequence \( (\chi_{A_i}(k))_{i \leq n} \). More exactly, set \( W_k(0) := 0 \). For \( n \in \mathbb{N}, n > 1 \), set \( W_k(n) := W_k(n-1) \) if and only if \( \chi_{A_n}(x) = \chi_{A_{n-1}}(x) \), and \( W_k(n) := W_k(n-1) + 1 \) otherwise. Obviously, \( k \in A \) if and only if the number of changes in \( (\chi_{A_i}(k))_{i \in \mathbb{N}} \) is even or zero, since \( A_0 = \mathbb{N} \). Next, we define the notion of a changing point in \( (\chi_{A_i}(k))_{i \in \mathbb{N}} \). Let \( w \in \mathbb{N} \). \( w \) is said to be a changing point in \( (\chi_{A_i}(k))_{i \in \mathbb{N}} \) if and only if \( W_k(w) := W_k(w-1) + 1 \). Since \( (\chi_{A_i}(k))_{i \in \mathbb{N}} \) converges, there are only finitely many changing points. For \( k \in \mathbb{N} \), let \( w_0 < w_1 < \ldots < w_{r_k} \) be the sequence of changing points in \( (\chi_{A_i}(k))_{i \in \mathbb{N}} \). Hence, \( W_k(i) = W_k(w_{r_k}) \) for all \( i \geq w_{r_k} \). Notice that, for \( n \in \mathbb{N} \), it is decidable whether or not \( n \) is a changing point in \( (\chi_{A_i}(k))_{i \in \mathbb{N}} \), but not decidable whether or not \( n = w_{r_k} \).

\[3\]This assumption is needed for the construction of the indexed family fulfilling the requirement. It is easy to construct a converging sequence fulfilling this assumption.

117
Now we are ready to define $L_A$. Let $\Sigma = \{a,b,d\}$. Let $\langle , , \rangle : \mathbb{N} \times \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$. Moreover, let $\langle , , \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$. Let $k, n \in \mathbb{N}$. Initialize every language $L_{(k,j,z)}, j \in \{0,1\}, z \in \mathbb{N}$, with $\{a^kb^{0,l} | l \in \mathbb{N}\}$.

In the $n$-th step, we define for $k \in \mathbb{N}$ which elements $a^kb^{(n,l)}, n \in \mathbb{N}, l \in \{0,1\}$, and $d^{m}, m \in \mathbb{N}$, belong to the languages $L_{(k,j,z)}, j \in \{0,1\}, z \leq n$.

1. If $W_k(n) = 0$, then add $d^{(n,0,0)}$ to $L_{(k,0,n)}$. Add no other elements to any language $L_{(k,j,n)}, j \in \{0,1\}, z \leq n$.

2. Assume $W_k(n) > 0$.

   2.1 Assume $W_k(n) \neq W_k(n - 1)$, and $W_k(n)$ is odd.

      (i) Add $a^kb^{(n,i)}, i \leq 1$, to every language $L_{(k,1,z)}, z < n$.
      Add $a^kb^{(n,i)}, i \leq 1$, to $L_{(k,1,n)}$, and $a^kb^{(n,0)}$ to $L_{(k,0,n)}$.
      (ii) Assume $z < n$. Add $d^{(n,0,z+4)}$ to $L_{(k,0,z)}$, and $d^{(n,1,z+4)}$ to $L_{(k,1,z)}$.
      (iii) Add no other $a^kb^{(n,l)}$ to any language $L_{(k,j,z)}, j \in \{0,1\}, z \leq n$.

   2.2 Assume $W_k(n) \neq W_k(n - 1)$, and $W_k(n) > 0$ is even.

      (i) Add $a^kb^{(n,i)}, i \leq 1$, to every language $L_{(k,0,z)}, z < n$.
      Add $a^kb^{(n,i)}, i \leq 1$, to $L_{(k,0,n)}$, and $a^kb^{(n,0)}$ to $L_{(k,1,n)}$.
      (ii) Let $z < n$. Add $d^{(n,0,z+4)}$ to $L_{(k,0,z)}$, and $d^{(n,1,z+4)}$ to $L_{(k,1,z)}$.
      (iii) Add no other $a^kb^{(n,l)}$ to any language $L_{(k,j,z)}, j \in \{0,1\}, z \leq n$.

   2.3 Assume $W_k(n) = W_k(n - 1)$. Let $w_0 < w_1 < \ldots < w_r$ be the sequence of changing points in $(\chi_{A_k}))(i \leq n)$. Then by definition $w_r \leq n - 1$, and $W_k(n) = W_k(w_r)$. Let $\mu \in \mathbb{N}$, $\mu \geq 1$, with $n = w_r + \mu$.

      (i) If $W_k(n)$ is odd, then add $a^kb^{(w_r,i)}, i \leq 1$, to $L_{(k,1,n)}$, and $a^kb^{(w_r,0)}$ to $L_{(k,0,n)}$. If $\mu \leq r - 1$, then add $a^kb^{(w_\mu,0)}$ to $L_{(k,j,n)}$, $j \in \{0,1\}$.
      (ii) If $W_k(n)$ is even, then add $a^kb^{(w_r,i)}, i \leq 1$, to $L_{(k,0,n)}$, and $a^kb^{(w_r,0)}$ to $L_{(k,1,n)}$. If $\mu \leq r - 1$, then add $a^kb^{(w_\mu,0)}$ to $L_{(k,j,n)}$, $j \in \{0,1\}$.
(iii) Add no other $a^k b^{(w_i,l)}$, $i < r_n$, $l \in \mathbb{N}$, to $L_{(k,j,n)}$, $j \in \{0,1\}$.

Then $L_A = (L_{(k,j,z)})_{k,j,z \in \mathbb{N}, j \leq 1}$ is an indexed family with the following properties.

Let $k$ be a natural number. If $W_k(n) = 0$ for all $n \in \mathbb{N}$, then $L_{(k,1,z)} \subseteq L_{(k,0,z)}$ for all $z \in \mathbb{N}$. Assume $W_k(n) \neq 0$ for an $n \in \mathbb{N}$, i.e., $w_{r_k} > 0$. Then for all $z, z' < w_{r_k}$, and $j, j' \in \{0,1\}$, the languages $L_{(k,j,z)}$ and $L_{(k,j',z')}$ are incomparable. Moreover, for all $z < w_{r_k}$, and $j \in \{0,1\}$, $L_{(k,j,z)}$ contains a string which determines the index $\langle k, j, z \rangle$. Furthermore, the following holds.

1. If $k \in A$, then for all $z \in \mathbb{N}$, there exists a $z' \geq w_{r_k}$ with $L_{(k,1,z')} \subset L_{(k,0,z)}$. Furthermore, the languages $L_{(k,1,z)}$ and $L_{(k,1,z')}$ are incomparable for all $z < w_{r_k}$, and $z' \geq w_{r_k}$.

2. If $k \notin A$, then for all $z \in \mathbb{N}$, there exists a $z' \geq w_{r_k}$ with $L_{(k,0,z')} \subset L_{(k,1,z)}$. Moreover, the languages $L_{(k,0,z)}$ and $L_{(k,0,z')}$ are incomparable for all $z < w_{r_k}$, and $z' \geq w_{r_k}$.

Finally, we note a property of $L_A$ which is important for the verification of the algorithms identifying $L_A$. Let $n$ be a natural number. Suppose, $n$ is a changing point, i.e., $n = w_i$ for an $i \in \mathbb{N}$. Then, depending on $W_k(n)$ being odd or even, the elements $a^k b^{(w_i,i)}$, $i \leq 1$, are added to $L_{(k,1,z)}$ or $L_{(k,0,z)}$, $z < w_i$. Assume that there exists a changing point $w_{i+1} > w_i$. Then $a^k b^{(w_i,1)}$ is not contained in $L_{(k,j,z)}$, $j \in \{0,1\}$, $z \geq w_{i+1}$. However, there exist languages $L_{(k,j,z)}$, $j \in \{0,1\}$, $z > w_{i+1}$, containing $a^k b^{(w_i,0)}$.

Now we have to show that $L_A$ is conservatively identifiable with probability $p = 1/2$, and conservatively identifiable by an OIM $M[B]$ if and only if $A$ is Turing reducible to $B$. First, we define a two-sided probabilistic machine $P$. Let $w_0 < w_1 < \ldots < w_{r_k}$ be the sequence of changing points in $(\lambda A_n(k))_{n \in \mathbb{N}}$. Let $L \in \text{range}(L_A)$ be a language, and let $\tau$ be a text for $L$ with $\text{range}(\tau_0) \subset L_k$. Let $\mathcal{C}$ be a coin-oracle, and let $x$ be a natural number.

**PIM $P$:** On input $\tau_x$, $P^c$ works as follows.

If $x = 0$, then set $P^{c_0}(\tau_0) := \langle k, c_0, 0 \rangle$.

If $x > 0$, then test whether or not $\text{range}(\tau_x) \subseteq L_{\text{P}^{c_{x-1}}(\tau_{x-1})}$. If it is, then set $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$. Otherwise distinguish the following cases.
Let \( M \) be a PIM machine \( P \) defined above. Let \( \mathcal{A} \) be a language, and let \( \tau \in \text{text}(L) \). Let \( k \in \mathbb{N} \) with \( \text{range}(\tau) \cap L_k \neq \emptyset \). Let \( x \) be a natural number. We define \( M[A] \) by using the probabilistic machine \( P \) defined above. Let \( j \in \{0,1\} \), and let \( \mathcal{C}_j = (c_{j,i})_{i \in \mathbb{N}} \) be a coin-oracle with \( c_{j,0} = j \), and \( c_{j,i} = 0 \) for all \( i > 0 \).

**OIM \( M[A] \):** On input \( \tau \), \( M[A] \) works as follows.

Let \( x = 0 \). If \( x \in A \), then set \( M[A](\tau_x) := \langle k, 1, 0 \rangle \). If \( x \not\in A \), then set \( M[A](\tau_x) := \langle k, 0, 0 \rangle \).

Let \( x > 0 \). Distinguish the following cases.

(A) If \( x \in A \), then set \( M[A](\tau_x) := P_{c_1}^{\tau}(\tau_x) \).
(B) If $x \notin A$, then set $M[A](\tau_x) := P^{C_0}(\tau_x)$. 

end

Finally, assume that $\mathcal{L}_A$ is conservatively identifiable by an oracle machine having access to an oracle $B$. Let $k \in \mathbb{N}$, and let $w_0 < w_1 < \ldots < w_{r_k}$ be the sequence of changing points in $(\chi_{A_i}(k))_{i \in \mathbb{N}}$. By construction, there exists a language $L \in \text{range}(\mathcal{L}_A)$ with

$$L = \{a^k b^{(0,l)} \mid l \in \mathbb{N}\} \cup \{a^k b^{(w_{r_k},0)}\}.$$

Set for $j < r_k$:

$$M_j = \{a^k b^{(0,l)} \mid l \in \mathbb{N}\} \cup \{a^k b^{(w_j,0)}\}.$$

Let $(\sigma_i)_{i \in \mathbb{N}}$ be an effective enumeration of the set of all finite sequences of strings from $\Sigma^*$ such that a $j \in \mathbb{N}$ exists with $\text{range}(\sigma) \subset M_j$. Since $L = M_l$ for an $l \in \mathbb{N}$, there exists an $\sigma_x$ such that $M[B](\sigma_x) \neq \bot$. Suppose $M[B](\sigma_x) = \langle k, 0, z \rangle$ for a $z \in \mathbb{N}$. Assume the number of changes in $(\chi_{A_i}(k))_{i \in \mathbb{N}}$ is zero or even. If it is zero, then $\text{range}(\sigma_x)$ is a subset of $L_{\langle k, 1, z \rangle}$, and $L_{\langle k, 1, z \rangle} \subset L_{\langle k, 0, z \rangle}$, a contradiction. If it is even, then by construction, there exists a $z' \in \mathbb{N}$, $z' \geq w_{r_k}$, such that $L_{\langle k, 1, z' \rangle}$ contains $\text{range}(\sigma_x)$, and $L_{\langle k, 1, z' \rangle}$ is a proper subset of $L_{\langle k, 0, z \rangle}$, a contradiction. Thus, the number of changes in $(\chi_{A_i}(k))_{i \in \mathbb{N}}$ is odd. Hence, $x \notin A$. With the same argument it can be shown that $M[B](\sigma_x) = \langle k, 1, z \rangle$ for a $z \in \mathbb{N}$ yields $x \in A$. □

Notice that $\mathcal{L}_A$ is not properly conservatively identifiable with $p > 1/2$ by any probabilistic machine. Thus, we can immediately draw the following corollary.

**Corollary 3.2.3.** Let $A \leq_T K$, $A \notin \mathcal{REC}$, and let $p > 1/2$. Then

$$ECOV[A] \setminus ECOV_{\text{prob}}(p) \neq \emptyset.$$

In Subsection 3.2.4, we proved that $ECOV_{\text{prob}}(p)$, $p > 1/2$, and $ECOV[A]$ are not comparable for all oracles $A \in \mathcal{RE} \setminus \mathcal{REC}$, $A \leq_T K$. The next corollary shows that this holds for arbitrary oracle $A \leq_T K$.

**Corollary 3.2.4.** Let $A \leq_T K$, $A \notin \mathcal{REC}$. For $p > 1/2$, $ECOV_{\text{prob}}(p)$ and $ECOV[A]$ are not comparable.
PROOF. The result follows from Theorem 3.2.2 and the fact that every probabilistic learning class $ECOV_{\text{prob}}(p)$ contains a $K$-difficult problem.

Moreover, we conjecture that the indexed family defined in Theorem 3.2.2 is not conservatively identifiable with respect to any class preserving hypothesis space.

**Conjecture 3.2.5.** Let $A \leq_T K$, $A \notin \mathcal{R}\mathcal{E}\mathcal{C}$. Let $L_A$ be the indexed family defined in Theorem 2.1.1. We propose that $L_A$ is not conservatively identifiable by any OIM having access to an oracle $B$ with $A \not\leq_T B$ with respect to any class preserving hypothesis space.

Next we answer the question whether or not there are learning problems in the probabilistic learning classes which are not characterizing an oracle.

**Theorem 3.2.6.**

1. Let $\mu \in \{COV, SMON, MON\}$, and let $p < 1$. There exists an indexed family $L \in E\mu_{\text{prob}}(p)$ such that $O_{E\mu}(L)$ is not $E\mu$-simple.

2. There exists an indexed family $L \in EMON_{\text{prob}}(1)$ such that $O_{EMON}(L)$ is not $EMON$-simple.

**PROOF.** Let $\mu \in \{COV, SMON, MON\}$. Let $A$ be a Peano-complete oracle. In Theorem 2.1.1 was proved that for all $p > 1/2$, $ESMON_{\text{prob}}(p) \cup q > p ECOV_{\text{prob}}(q) \neq \emptyset$. In particular, the indexed families constructed in the proof of Theorem 2.1.1 are not properly monotonically identifiable. Now let $p > 2/3$, and let $L^p \in ESMON_{\text{prob}}(p)$, $L \notin ECOV(q)$ for $q > p$. For the construction of this language see Theorem 2.1.1. By Theorem 3.1.13, $L \in ESMON[A]$ for every Peano-complete oracle. Now let $A, B$ be a minimal pair of Peano-complete oracles. Then there is no set $\mathcal{C}$, $C \notin \mathcal{R}\mathcal{E}\mathcal{C}$, with $C \leq_T A, C \leq_T B$. Thus, we can conclude that $O_{E\mu}(L)$ is not $E\mu$-simple. In particular, every probabilistic learning class $ESMON_{\text{prob}}(p)$, $ECOV_{\text{prob}}(p)$, and $EMON_{\text{prob}}(p), p \in [0, 1]$, contains an indexed family $L$ such that $O_{E\mu}(L)$ is not $E\mu$-simple.

For proving the second part of the theorem, let $L^1$ be the indexed family defined in Theorem 2.1.8. Obviously, $L^1$ is $EMON$-identifiable by an OIM which has access to an arbitrary Peano-complete oracle. The same argument as in the first part of the proof shows that $L^1$ is not $EMON$-simple.

In the following, we show that Theorem 3.2.2 not only yields the existence of maximal complicated problems in $ECOV[A]$, but also information about
the relations between the oracle learning classes $ECOV[A]$ with $A \leq_T K$. Kummer and Stephan (cf. [61]) showed that in the case of learning recursive enumerable languages from text $TXTEX[A] = TXTEX$ if and only if $A \leq_T K$ and $A$ is 1-generic. In the case of proper conservative learning of indexed families, $A$ enhances the learning power if and only if $A$ is not recursive. Moreover, the inclusion structure of oracle learning classes below $\mathcal{K}$ in the conservative case differs from the general case, since, in contrast to the result of Kummer and Stephan, the power of learning indexed families conservatively directly corresponds to the Turing degree of the oracle. The following theorem provides a complete picture of the impact of oracles below $\mathcal{K}$ on the power of conservative inductive inference machines.

**Corollary 3.2.7.** Let $A, B \leq_T \mathcal{K}$ be oracles.

(a) $ECOV[A] = ECOV$ if and only if $A \in \mathcal{REC}$.

(b) $ECOV[A] \subseteq ECOV[B]$ if and only if $A \leq_T B$.

**PROOF.** Both results follow directly from Theorem 3.2.2. ◇

### 3.2.2 Maximal complicated problems

Let $A \leq_T \mathcal{K}$ be an oracle, $A \notin \mathcal{REC}$, and let $\mathcal{L}_A$ be the indexed family defined in Theorem 3.2.2. We proved $\mathcal{L}_A$ to be $A$-difficult with respect to $ECOV$, i.e., $\mathcal{L}_A$ is $ECOV$-identifiable by an OIM $M[B]$ if and only if $A \leq_T B$. In particular, we proved that every OIM $M[B]$ which $ECOV$-identifies $\mathcal{L}_A$ can be transformed into a decision procedure for $A$ in the following sense.

**Definition 3.2.1.** Let $M[B]$ be an OIM, and let $A$ be an oracle. $M[B]$ can be transformed into a decision procedure for $A$ if and only if there exist a $B$-recursive procedure $\mathcal{E}_A$, and, for every $k \in \mathbb{N}$, a language $L^k \in \text{range}(\mathcal{L})$, and a text $\tau^k$ for $L^k$, with the following properties:

(A) $\mathcal{E}_A$ takes as its input the finite sequences of strings $\tau^k_x$, $x \in \mathbb{N}$, and the corresponding hypotheses $M[B](\tau^k_x)$, $x \in \mathbb{N}$.

(B) After finitely many steps, $\mathcal{E}_A$ stops and outputs $j \in \{0, 1\}$ with $j = \chi_A(k)$. 

123
Hence, $\mathcal{L}_A$ can be considered to be a maximal complicated problem in the oracle learning class $ECOV[A]$. In the following, we give a formal definition of this notion. For other definitions of complexity and reducibility see [39, 44, 45, 46, 53, 83], and the references therein.

**Definition 3.2.2.** Let $\mu \in \{COV, SMON, MON\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $A \notin \mathcal{REC}$ be an oracle. $\mathcal{L}$ is denoted to be maximal complicated in $\lambda \mu[A]$ if and only if $\mathcal{L} \in \lambda \mu[A]$, and the following condition holds for every oracle machine $M[]$, and every oracle $B$.

If $M[B] \lambda \mu$-identifies $\mathcal{L}$, then $M[B]$ can be transformed into a decision procedure for $A$.

Although it is not trivial that, for every oracle $A \leq_T K$, $ECOV[A]$ contains a maximal complicated indexed family, it is not surprising that a maximal complicated indexed family is $ECOV$-identifiable with probability $p = \frac{1}{2}$, since a probabilistic machine, which only has to learn with probability $p = \frac{1}{2}$, can distinguish between $k \in A$ and $k \notin A$ for every $k \in \mathbb{N}$. However, Theorem 3.2.2 can be generalized for arbitrary probabilities $p < 1$.

**Theorem 3.2.8.** Let $A \leq_T K$ be an oracle, $A \notin \mathcal{REC}$. Let $n$ be a natural number. Then $ECOV_{prob}(\frac{2n-1}{2n})$ contains an indexed family $\mathcal{L}^n_A$ which is maximal complicated in $ECOV[A]$.

**Proof.** The theorem may be proved by combining the techniques used in Theorem 2.1.1 and Theorem 3.2.2. We only give a sketch of the proof. Let $n$ be a natural number. Then we define $\mathcal{L}^n_A$ as follows. Instead of sequences of languages $(L_{\langle k,0,l,z \rangle})_{z \in \mathbb{N}}$, and $(L_{\langle k,1,l,z \rangle})_{z \in \mathbb{N}}$, we define sequences of blocks of languages $([L_{\langle k,v,l,z \rangle}]_{0 \leq n-1})_{z \in \mathbb{N}}, v \in \{0,1\}$. The construction of the indexed family fulfilling the requirement is analogous to the construction performed in the proof of Theorem 3.2.2. The indexed family has to fulfill the following two conditions:

1. If $k \in A$, then for every language $L_{\langle k,0,l,z \rangle}$, there exists a $z' \in \mathbb{N}$ such that the language $L_{\langle k,1,l,z' \rangle}$ is a proper subset of $L_{\langle k,0,l,z \rangle}$. The languages $L_{\langle k,0,l',z' \rangle}$, $l \neq l'$, $z' \in \mathbb{N}$, are incomparable with $L_{\langle k,1,l,z \rangle}$ for all $z \in \mathbb{N}, l \leq n-1$. If $k \notin A$, then change the roles of 0 and 1.

2. Let $M[B]$ be an OIM which $ECOV$-identifies $\mathcal{L}_A$. Then, for all $k \in \mathbb{N}$, $M[B]$ computes a finite sequence of strings $\sigma^k$ such that $M[B](\sigma^k) \neq \bot$, and $k \in A$ if and only if $M[B](\sigma^k) = \langle k,1,l,z \rangle$ for a $z \in \mathbb{N}$, and an $l \leq n-1$. 

124
The proof is analogous to the proof of Theorem 3.2.2 and therefore omitted.

\[\Diamond\]

**Remark 3.2.9.** We may interpret Theorem 3.2.2 from a recursion theoretic point of view. Obviously, the result yields an injective operator which embeds the Turing degrees below \(K\) into the probabilistic learning class \(\text{ECOV}_{\text{prob}}(p)\) for every \(p > 1/2\).

Finally, we show that it is not possible to characterize all oracles \(A\) below \(K\) within strong-monotonic and monotonic learning in the strong sense proved in Theorem 3.2.2. Before we give a formal definition of the terminus *strongly characterizing an oracle within a learning model*, we analyse the properties of the indexed family \(L_A\) constructed in Theorem 3.2.2. First, we notice that, for every \(k \in \mathbb{N}\), there are two classes of hypotheses, namely \((\langle k, 0, z \rangle)_{z \in \mathbb{N}}\) and \((\langle k, 1, z \rangle)_{z \in \mathbb{N}}\). Depending on the answers to its queries, the identifying OIM \(M[B]\) computes a sequence of strings \(\sigma_k\) such that \(M[B](\sigma_k) = \langle k, j, z \rangle\) if and only if \(\chi_A(k) = j\). Secondly, for all \(z' \in \mathbb{N}\), the following holds. If \(\text{range}(\sigma_k) \subseteq L_{\langle k, 1 - j, z' \rangle}\) then there exists a \(z'' \in \mathbb{N}\) such that \(L_{\langle k, 1 - j, z'' \rangle}\) is an overgeneralization of \(L_{\langle k, j, z' \rangle}\). Hence, if an OIM guesses a hypothesis \(\langle k, 1 - j, z' \rangle\) on \(\sigma_k\), then the learning process fails. Thus, there is a correspondence between learning success and correct classification.

In the following, we generalize these properties to a notion of strong characterization.

**Definition 3.2.3.** Let \(\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}\). Let \(A\) be an oracle. Let \(\mathcal{L}\) be an indexed family with \(\mathcal{L} \in E\mu[A]\). Then \(\mathcal{L}\) strongly characterizes \(A\) with respect to \(E\mu\) if and only if the following condition is fulfilled. There exist two IIMs \(M_0\) and \(M_1\), and, for every \(k \in \mathbb{N}\), an indexed family \((L_{k,i})_{i \in \mathbb{N}}\), \(L_{k,i} \in \text{range}(\mathcal{L})\) for all \(i \in \mathbb{N}\), such that the following holds.

\[\chi_A(k) = j\] if and only if the following two conditions hold.

(A) For all \(i \in \mathbb{N}\), \(M_j\) \(E\mu\)-correctly converges on every text for \(L_{k,i}\).

(B) There exists a finite sequence of strings \(\sigma\) from \(\Sigma^*\), \(\sigma \in \text{range}(L_{k,i})\) for an \(i \in \mathbb{N}\), such that \(M_{1-j}\) when successively fed \(\sigma\), injures \(\mu\).
The conditions in Definition 3.2.3 can be interpreted as follows. $M_0$ and $M_1$ are learning strategies. Depending from $k$ being in $A$ or not, one of the strategies is successful, whereas the other fails.

In the following, we make precise what the condition (B) in the definition means for $\mu = COV$. In this case, there exists a $\sigma = s_0, \ldots, s_m$ with the following property. Let $n \leq m$. Set $\sigma_n := s_0, \ldots, s_n$. Then there exist an $n, n' \in \mathbb{N}$, $n < n'$, such that $L_{M_2(\sigma_n)}$ is a proper superset of $L_{M_2(\sigma_{n'})}$. For $\mu = \{SMON, MON\}$, the condition can be described analogously (see Definition 1.2.2).

Remark 3.2.10. Let $L$ be $A$-difficult with respect to $\lambda \mu$. In this case, $L \in \lambda \mu[B]$ if and only if $A \leq_T B$. We shall see that we cannot conclude that $L$ strongly characterizes $A$.

For proper conservative learning, Theorem 3.2.2 yields a strongly characterizing indexed family for every oracle $A \leq_T K$.

Corollary 3.2.11. Let $A$ be an oracle, $A \leq_T K$, $A \notin \mathcal{REC}$. Then there exists an indexed family $L_A$ which strongly characterizes $A$ with respect to ECOV.

For proper strong-monotonic and monotonic learning, the analogous result does not hold.

Theorem 3.2.12. Let $A$ be an oracle, $A \leq_T K$. Let $L_A$ be an indexed family which strongly characterizes $A$ with respect to ESMON or EMON. Then $A \in \mathcal{REC}$.

PROOF. Let $\mu \in \{SMON, MON\}$, and let $M$ be an IIM. Then there exists an algorithm which enumerates all finite sequences of strings $\sigma$ such that $M$, when successively fed $\sigma$, injures $\mu$.

Since strong characterization yields the existence of two IIMs such that one of these algorithms injures $\mu$ on at least one finite sequence of strings, we can define an algorithm testing for $k \in \mathbb{N}$ which IIM injures $\mu$. $\Diamond$

Next, we analyze the proofs of the Theorems 3.1.4, 3.1.7, and 3.1.11. In this theorems, we investigated recursively enumerable oracles. In all these proofs, the identifying OIM computes an upper bound for $E_A^{-1}(k)$ for all $k \in \mathbb{N}$, where $E_A$ is an algorithm enumerating $A$. This property can be generalized to the following notion of characterization.
Definition 3.2.4. Let $\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $A$ be an oracle, $A \leq_T K$. Let $(A_i)_{i \in \mathbb{N}}$ be an infinite sequence of recursive functions approximating $A$, and let $\text{mod}_A$ be a modulus function for $(A_i)_{i \in \mathbb{N}}$. Let $\mathcal{L}$ be an indexed family with $\mathcal{L} \in \lambda \mu[A]$. Then $\mathcal{L}$ weakly characterizes $A$ with respect to $\lambda \mu$ if and only if the following condition is fulfilled.

Let $M[B]$ be an OIM which $\lambda \mu$-identifies $\mathcal{L}$. Then there exist a $B$-recursive procedure $E_A : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, and, for every $k \in \mathbb{N}$, a language $L^k \in \text{range}(\mathcal{L})$, and a text $\tau^k$ for $L^k$, with the following property:

$E_A$ takes as its input the finite sequences of strings $\tau_x^k$, $x \in \mathbb{N}$, and the hypotheses $M[B](\tau_x^k)$, $x \in \mathbb{N}$. After finitely many steps, $E_A$ stops, and outputs a natural number $n_k$. Then $\text{mod}_A(k) \leq n_k$ or $k \notin A$.

For weakly characterization, the following theorem can be shown.

Theorem 3.2.13. Let $\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}$, and let $\lambda \in \{E, \epsilon, C\}$. Let $A$ be an oracle, $A \leq_T K$. Let $\mathcal{L}_A$ be an indexed family which weakly characterizes $A$ with respect to $\lambda \mu$-learning. Then $A \in \mathcal{RE}$.

Moreover, we can conclude from Corollary 3.1.9 and Theorem 3.2.12 that there exist indexed families which are $A$-difficult with respect to $\lambda \mu$, but do not strongly characterize $A$.

3.2.3 Characterization of various sets

In the previous subsections, we addressed the problem whether, for a given oracle $A$, there exists an indexed family $\mathcal{L}_A$ which is maximal complicated in $\lambda \mu[A]$. We could also ask whether, for a given class of oracles $\mathcal{A}$, there is an indexed family $\mathcal{L}_A$ such that the oracle-complexity $\mathcal{O}_{\lambda \mu}(\mathcal{L}_A)$ is $\mathcal{A}$. Next, we show that the class of Peano-complete oracles can be characterized.

Theorem 3.2.14. Let $\mu \in \{\text{COV}, \text{SMON}, \text{MON}\}$. There exists an indexed family $\mathcal{L}_{PA} \in \text{ESMON}_{\text{prob}}(\frac{2}{3})$, with $\mathcal{L}_{PA} \in E \mu[A]$ if and only if $A$ Peano-complete.

Proof. Let $(\cdot, \cdot) : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$. First, we define an indexed family encoding the halting problem for every $k, j \in \mathbb{N}$. Let $k, j \in \mathbb{N}$. Set $L_{(k,j)} := \{a^{(k,j)}b^m \mid m \in \mathbb{N}\}$. Moreover, set

$$L'_{(k,j)} := \begin{cases} L_{(k,j)}, & \text{if } \varphi_k(j) \uparrow, \\ \{a^{(k,j)}b^m \mid m \leq \Phi_k(j)\}, & \text{if } \varphi_k(j) \downarrow. \end{cases}$$
Let \( \langle \cdot, \cdot \rangle : \mathbb{N} \times \mathbb{N} \times \{0, 1\} \to \mathbb{N} \) be an effective encoding. Let \( k, j \in \mathbb{N} \). Define \( L_{(k,j,v)} \), \( v \in \{0,1\} \), as follows.

\[
L_{(k,j,0)} := \begin{cases} 
L'_{(k,j)}; & \text{if } \varphi_k(j) \downarrow \text{ and } \varphi_k(j) = 0, \\
L_{(k,j)}; & \text{if } \varphi_k(j) \downarrow \text{ and } \varphi_k(j) \neq 0, \\
L_{(k,j)}; & \text{if } \varphi_k(j) \uparrow.
\end{cases}
\]

\[
L_{(k,j,1)} := \begin{cases} 
L'_{(k,j)}; & \text{if } \varphi_k(j) \downarrow \text{ and } \varphi_k(j) \neq 0, \\
L_{(k,j)}; & \text{if } \varphi_k(j) \downarrow \text{ and } \varphi_k(j) = 0, \\
L_{(k,j)}; & \text{if } \varphi_k(j) \uparrow.
\end{cases}
\]

Then \( \mathcal{L}_{PA} := (L_{(k,j,v)})_{k,j,v \in \mathbb{N}, v \in \{0,1\}} \) is an indexed family. We can prove that \( \mathcal{L}_{PA} \) is strong-monotonically identifiable with probability \( p = 2/3 \) by using the same argument as in Theorem 2.1.1. We have to show that \( \mathcal{L}_{PA} \) is \( \mu \)-identifiable by an OIM \( M[B] \) if and only \( B \) is Peano-complete. We restrict ourselves to the case of \( \mu = COV \). The other two cases can be proved analogously.

First, let \( B \) be a Peano-complete oracle. As in the proof of Theorem 3.1.18, we can show that there exists an OIM \( M[B] \) which \( ECOV \)-identifies \( \mathcal{L}_{PA} \). For the other direction, let \( M[B] \) be an OIM which \( ECOV \)-identifies \( \mathcal{L}_{PA} \). Let \( \tau_{k,j} \) be the canonical text for \( L_{(k,j)} \), and let \( x \in \mathbb{N} \) be the smallest natural number with \( M[B]((\tau_{k,j})_x) \neq \perp \). Then \( M[B]((\tau_{k,j})_x) = \langle k,j,v_0 \rangle \) for a \( v_0 \in \{0,1\} \). Assume that \( \varphi_k(j) \downarrow \), and \( v_0 = 0 \). Furthermore, assume that \( \varphi_k(j) \neq 0 \). Then \( L_{(k,j,v_0)} = L_{(k,j)} \). Thus, \( M[B] \) could not identify \( L_{(k,j,v_1)} = L'_{(k,j)} \) conservatively. Hence, \( \varphi_k(j) = 0 \). Similarly, the following claim can be shown. If \( \varphi_k(j) \downarrow \), and \( v_0 = 1 \), then \( \varphi_k(j) \neq 0 \).

Now we are ready to prove the result. Let \( f : \mathbb{N} \to \{0,1\} \) be a partial recursive function. Then \( f = \varphi_{k_0} \) for a \( k_0 \in \mathbb{N} \). Define \( F_{k_0} : \mathbb{N} \to \mathbb{N} \) as follows. For \( j \in \mathbb{N} \), let \( \langle k_0, j, v_0 \rangle \) be the first hypothesis produced by \( M[B] \) when fed be the canonical text for \( L_{k_0,j} \). Set \( F_{k_0}(j) := v_0 \). Then \( F_{k_0} \) is a \( B \)-recursive extension of \( f \).

In the following theorem, we define an indexed family which characterizes the modulus of convergence for a given sequence \( (\chi_{A_{i}})_{i \in \mathbb{N}} \). Thus, the Turing degree of a modulus of convergence of a nonrecursive oracle can be encoded in an indexed family.

**Theorem 3.2.15.**
Let \( \mu \in \{COV, SMON, MON\} \), and let \( \lambda \in \{E, \varepsilon\} \). Let \( A \) be a nonrecursive oracle with \( A \preceq T K \), and let \( (A_{i})_{i \in \mathbb{N}} \) be an infinite sequence of recursive
sets approximating $A$. Then there exists an indexed family $L_{\text{mod}} \in \lambda (1/2)$, such that every OIM $M[B]$ which $\lambda\mu$-identifies $L_{\text{mod}}$ can be transformed into a procedure computing the least modulus of convergence for $(\chi_{A_i})_{i \in \mathbb{N}}$.

**PROOF.** Let $A$ be a nonrecursive oracle with $A \leq_T K$, and let $(A_i)_{i \in \mathbb{N}}$ be an infinite sequence of recursive sets approximating $A$. Without loss of generality, we may assume that $A_0 := \emptyset$, and $A_1 = \mathbb{N}$. Define $W_k(n)$ for $k, n \in \mathbb{N}$ as in the proof of Theorem 3.2.2. Let $\Sigma = \{a, b\}$. Let $(\tau, \sigma) : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ be an effective encoding of $\mathbb{N} \times \mathbb{N}$. Let $t$ be a variable. In step 0 of the construction, set $L_{(k,0)} := L_k$, and initialize $t := 0$. In step $n$, assume that $L_{(k,j)}, j \leq t$, are already defined. Distinguish the following cases.

1. If $W_k(n) \neq W_k(n-1)$, then add all subsets of $\{a^k b^m \mid m \leq n\}$ to $L$. More formally, let $\{L'_r \mid r \leq 2^{n+1} - 2\}$ be the set of all nonempty finite subsets of $\{a^k b^m \mid m \leq n\}$. Then set $L_{(k,j)} := L'_r$ for $j = t + r + 1$, $r \leq 2^{n+1} - 2$. Set $t := t + 2^{n+1} - 1$, and goto step $n + 1$.

2. If $W_k(n) = W_k(n-1)$, then set $L_{(k,t+1)} := L_{(k,t)}$. Set $t := t + 1$, and goto step $n + 1$.

Then $L_{\text{mod}} = (L_{(k,j)})_{i \in \mathbb{N}}$ is an indexed family with the following property. For all $k \in \mathbb{N}$, $L_{\text{mod}}$ contains $L_k$, a proper subset $L_{\text{max}}$ of $L_k$, and all subsets of $L_{\text{max}}$. Without loss of generality, we may assume that $L_{(k,1)} = \{a^k b^0\}$.

It is easy to see that $L_{\text{mod}}$ is strong-monotonically identifiable by any OIM which can compute the minimal modulus of convergence for $(\chi_{A_i})_{i \in \mathbb{N}}$. Moreover, $L_{\text{mod}} \in ESMON_{\text{prob}}(1/2)$. A two-sided PIM $P$ can be defined as follows. Let $L$ be a language in $\text{range}(L_{\text{mod}})$, let $\tau \in \text{text}(L)$, and let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $C$ be a coin-oracle.

**PIM $P$:** On input $\tau_x$, $P^C$ works as follows.

If $x = 0$, set $P^{c_0}(\tau_0) := \langle k, c_0 \rangle$. If $x > 0$, then distinguish the following cases.

(A) If $P^{c_{x-1}}(\tau_{x-1})$ is consistent with $\tau_x$, then set $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$.

(B) If $P^{c_{x-1}}(\tau_{x-1})$ is not consistent with $\tau_x$, then $P^C$ searches for the smallest natural number $j \geq 1$ such that $L_{(k,j)} = \text{range}(\tau_x)$. If such a language exists, then set $P^{c_x}(\tau_x) := \langle k, j \rangle$.

end

Obviously, $P$ $ESMON_{\text{prob}}(1/2)$-identifies $L_{\text{mod}}$.  

129
Finally, we assume that there is an OIM \( M[\mathcal{B}] \) which conservatively or monotonically identifies \( \mathcal{L}_{\text{mod},A} \) with respect to a class preserving hypothesis space \( \mathcal{G} \). Let \( \tau^k \) be the canonical text for \( L_k \). Then there must be an \( n \in \mathbb{N} \) such that \( L_k \cap L(G_{M[\mathcal{B}](\tau^k)}) \) is a proper superset of \( \text{range}(\tau^k_n) \). Let \( n_0 \in \mathbb{N} \) be the smallest number with this property. Assume \( \text{range}(\tau^k_{n_0}) \subseteq L_{\text{max}k} \). Then there must be an \( n \in \mathbb{N} \) such that \( L_k \cap L(G_{M[\mathcal{B}](\tau^k)}) \) is a proper superset of \( \text{range}(\tau^k_n) \). Let \( n_0 \in \mathbb{N} \) be the smallest number with this property. Assume \( \text{range}(\tau^k_{n_0}) \subseteq \mathcal{L}_{\text{mod},A} \). Then \( M[\mathcal{B}] \) does not \( \text{ECOV} \)-identify \( \text{range}(\tau^k_{n_0}) \) conservatively with respect to \( \mathcal{G} \). In the monotonic case, a similar argument shows that \( M[\mathcal{B}] \) does not \( \text{EMON} \)-identify \( L_k \) monotonically with respect to \( \mathcal{G} \). Thus, \( L_{\text{max}k} \subset \text{range}(\tau^k_{n_0}) \) is the least natural number \( m \) with \( \chi_{A_j}(k) = \chi_{A_{n_0}}(k) \) for all \( j \in \mathbb{N} \), \( m \leq j \leq n_0 \).

As a corollary, we note the promised results.

**Corollary 3.2.16.** Let \( \mu \in \{ \text{COV}, \text{SMON}, \text{MON} \} \) and let \( \lambda \in \{ E, \varepsilon \} \).

(a) Let \( A \) be an oracle, and let \( m^A \) be a modulus of convergence for \( (\chi_{A_i})_{i \in \mathbb{N}} \) where \( (A_i)_{i \in \mathbb{N}} \) is a sequence of recursive sets approximating \( A \). There exists an indexed family \( \mathcal{L}_{\text{mod},A} \) with \( \mathcal{O}_{\lambda}(\mathcal{L}_{\text{mod}}) = \{ B \mid m^A \leq T_B \} \).

(b) There exists an indexed family \( \mathcal{L}_{\text{PA}} \) with \( \mathcal{O}_{\mu}(\mathcal{L}_{\text{PA}}) = \{ B \mid \text{mod}_A \leq T_B \} \).

### 3.2.4 Indexed families with high oracle-complexity

In the following subsection of this chapter, we deal with the question, whether it is possible to construct learning problems which are \( A \)-difficult for an oracle \( A \) with \( K < T_A \). It turns out that the probabilistic learning classes \( \text{ECOV}_{\text{prob}}(1/2) \) and \( \text{EMON}_{\text{prob}}(2/3) \) contain indexed families with oracle-complexity \( \{ B \mid \text{TOT} \leq T_B \} \). In the second part of this subsection, it is shown that there is no analogous result for strong-monotonic learning.

**Theorem 3.2.17.** Let \( A \) be an oracle with \( K \leq T_A \). There exist an indexed family \( \mathcal{L}^{1/2} \) with

(a) \( \mathcal{L}^{1/2} \in \text{ECOV}_{\text{prob}}(1/2) \),

(b) \( \mathcal{L}^{1/2} \in \text{ECOV}[\text{TOT}] \),

(c) \( \mathcal{L}^{1/2} \in \text{ECOV}[A] \) if and only if \( \text{TOT} \leq T_A \).
PROOF. We define an indexed family as follows. Let \( \langle \ , \ \rangle : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N} \) be an effective encoding of \( \mathbb{N} \times \mathbb{N} \). For every \( k \in \mathbb{N} \), we define a chain of languages \( \langle L_{(k,j)} \rangle_{j \in \mathbb{N}} \) in dependence on \( \varphi_k \) being total or not. Thereby, the chain converges to a finite language \( L \subset L_k \) if and only if \( \varphi_k \) is not total. Otherwise the chain does not converge. The languages in the chain are defined in a way such that every OIM which cannot decide whether or not \( \varphi_k \) is total fails to learn the indexed family conservatively.

Let \( \Sigma := \{a, b, d\} \). Define the indexed family \( \langle L_{(k,j)} \rangle_{k,j \in \mathbb{N}} \) as follows. Initialize \( L_{(k,0)} \) with \( \{a^k b^0\} \) for all \( k,j \in \mathbb{N} \). Remember that \( \Phi_k(j) \geq 1 \) for all \( k,j \in \mathbb{N} \). Let \( k \) be a natural number.

(1) \( L_{(k,0)} := L_k \).

(2) If \( \varphi_k(0) \uparrow \), then set \( L_{(k,j)} := L_k \) for all \( j \in \mathbb{N} \).

(3) If \( \varphi_k(0) \downarrow \), and if \( j < \Phi_k(0) \), then set

\[
L_{(k,j)} := \{a^k b^n \mid n \leq \Phi_k(0) - 1\} \cup \{d^{\Phi_k(0)}\}.
\]

If \( \varphi_k(0) \downarrow \), and if \( j \geq \Phi_k(0) \), then compute \( s \in \mathbb{N} \) such that

\[
\sum_{i=0}^{s} \Phi_k(i) \leq j < \sum_{i=0}^{s+1} \Phi_k(i).
\]

Distinguish the following cases.

(3.1) If \( \varphi_k(s+1) \uparrow \), then set \( L_{(k,j)} := \{a^k b^n \mid n \leq \sum_{i=0}^{s} \Phi_k(i)\} \).

(3.2) If \( \varphi_k(s+1) \downarrow \), then set

\[
L_{(k,j)} := \{a^k b^n \mid n \leq \sum_{i=0}^{s} \Phi_k(i)\} \cup \{d^{\sum_{i=0}^{s+1} \Phi_k(i)}\}.
\]

Then \( L_{1/2} = \langle L_{(k,j)} \rangle_{k,j \in \mathbb{N}} \) is an indexed family with the following properties. If \( \varphi_k \) is total, then every language \( L_{(k,j)} \), \( j > 0 \), contains a string \( d^m \). If \( \varphi_k \) is not total, then \( L_{1/2} \) contains a proper subset of \( L_k \), namely \( L_{(k,\sum_{i=0}^{s} \Phi_k(i))} \), where \( s \) is the smallest natural number with \( \varphi_k(s+1) \uparrow \). Furthermore, \( L_{(k,0)} \) equals \( L_k \) independent of \( \varphi_k \) being total or not. However, no other language equals \( L_k \), since in both cases, every language \( L_{(k,j)} \), \( j > 0 \), is finite.
We have to show that $L^{1/2} \in ECOV[A]$ for every oracle $A$ with $TOT \leq_T A$. Define an oracle machine $M[A]$ as follows. Let $L \in \text{range}(L^{1/2})$ be a language, and let $\tau \in \text{text}(L)$. Let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $x$ be a natural number.

OIM $M[A]$: On input $\tau_x$, $M[A]$ works as follows.

(A) If $\varphi_k$ is total, then set $M[A](\tau_x) := \langle k, 0 \rangle$ if and only if $\text{range}(\tau_x)$ does not contain a string of the form $d^m$. Otherwise $M[A]$ outputs $\langle k, j \rangle$ where $j$ is the smallest natural number with $d^m \in L\langle k, j \rangle$.

(B) If $\varphi_k$ is not total, then compute the smallest natural number $s$ with $\varphi_k(s + 1) \uparrow$. If $x = 0$, then set $M[A](\tau_x) := \langle k, \sum_{i=0}^{s} \Phi_k(i) \rangle$. If $x > 0$, and $M[A](\tau_{x-1})$ is consistent with $\tau_x$, then set $M[A](\tau_x) := M[A](\tau_{x-1})$. If $M[A](\tau_{x-1})$ is not consistent with $\tau_x$, and the contradicting string in the text is of the form $d^m$, then $M[A]$ outputs $\langle k, j \rangle$ where $j$ is the smallest natural number such that $L\langle k, j \rangle$ contains $d^m$. If the contradicting string in the text is in $L_k$, then $M[A]$ outputs $\langle k, 0 \rangle$.

end

Then $M[A]$ conservatively identifies $L^{1/2}$, since in case $k \not\in TOT$, every language $L\langle k, j \rangle$, $j \geq 1$, which is not equal to $L\langle k, \sum_{i=0}^{s} \Phi_k(i) \rangle$, contains an element of the form $d^m$. Notice that $M[A]$ performs at most one mind change on every text for a language in $L^{1/2}$.

Moreover, $L^{1/2} \in ECOV_{\text{prob}}(1/2)$. We define a 2-sided PIM as follows. Let $L \in \text{range}(L^{1/2})$ be a language, and let $\tau \in \text{text}(L)$. Let $k \in \mathbb{N}$ with $\text{range}(\tau) \cap L_k \neq \emptyset$. Let $C$ be a coin-oracle, and let $x$ be a natural number.

PIM $P$: On input $\tau_x$, $P^c$ works as follows.

If $x = 0$, then set $P^{c_0}(\tau_0) := \langle k, c_0 \rangle$.

If $x > 0$, then distinguish the following cases.

(A) If $P^{c_{x-1}}(\tau_{x-1})$ is consistent with $\tau_x$, then $P^{c_x}(\tau_x) := P^{c_{x-1}}(\tau_{x-1})$.

(B) Otherwise $P^c$ searches for the smallest natural number $j \geq 1$ such that $L\langle k, j \rangle$ contains the actual text. Set $P^c(\tau_x) := \langle k, j \rangle$. 
Then \( P \text{ECOV}_{\text{prob}}(1/2) \)-identifies \( \mathcal{L}^{1/2} \). It remains to show that \( \Omega_{\text{ECOV}}(\mathcal{L}^{1/2}) = \{ A | \text{TOT} \leq_T A \} \). Let \( A \) be an oracle, \( A \geq_T K \), and let \( M[A] \) be an OIM which conservatively identifies \( \mathcal{L}^{1/2} \). Let \( k \) be a natural number. Let \( \tau^k \) be the canonical text for \( L_{(k,0)} = L_k \). Since \( M[A] \) \text{ECOV}-identifies \( L_{(k,0)} \), there must be an \( n_0 \in \mathbb{N} \) such that \( M[A](\tau^k_{n_0}) = \langle k, 0 \rangle \). Let \( \ell \) be the least natural number such that \( \sum_{i=1}^{\ell} \Phi_k(i) > n_0 \). Then either \( \varphi_k \) is total or \( \min \{ r \in \mathbb{N} | \varphi_k(r) \uparrow \} \leq \ell \). Since \( \mathcal{K} \leq_T A \), it follows that \( \text{TOT} \leq_T A \). \( \diamond \)

For monotonic probabilistic learning, an analogous result for \( p = 2/3 \) holds.

Theorem 3.2.18. Let \( A \) be an oracle with \( \mathcal{K} \leq_T A \). There exists an indexed family \( \mathcal{L}^{2/3} \) with

(a) \( \mathcal{L}^{2/3} \in \text{EMON}_{\text{prob}}(2/3) \),

(b) \( \mathcal{L}^{2/3} \in \text{EMON}[\text{TOT}] \),

(c) \( \mathcal{L}^{2/3} \in \text{EMON}[A] \) if and only if \( \text{TOT} \leq_T A \).

Proof. Let \( \Sigma := \{ a, b, d \} \). Let \( (, ) : \mathbb{N} \times \mathbb{N} \to \mathbb{N} \) be an effective encoding of \( \mathbb{N} \times \mathbb{N} \). We define \( (L_{(k,j)})_{k,j \in \mathbb{N}} \) in dependence of \( \varphi_k(j) \), \( j \in \mathbb{N} \). Initialize \( L_{(k,j)} \) with \( \{ a^k b^0 \} \) for all \( k, j \in \mathbb{N} \). Let \( k \) be a natural number.

1. \( L_{(k,0)} := L_k \).

2. If \( j < \Phi_k(0) \), then set \( L_{(k,j)} := L_k \).

3. If \( j \geq \Phi_k(0) \), then compute \( s \in \mathbb{N} \) such that

\[
\sum_{i=0}^{s-1} (\Phi_k(i) + 1) + \Phi_k(s) \leq j < \sum_{i=0}^{s} (\Phi_k(i) + 1) + \Phi_k(s + 1).
\]

Set \( N_s := \{ a^k b^n \mid n \leq \sum_{i=0}^{s-1} (\Phi_k(i) + 1) + \Phi_k(s) \} \). Distinguish the following cases.

3.1 If \( j = \sum_{i=0}^{s-1} (\Phi_k(i) + 1) + \Phi_k(s) \), and if \( \varphi_k(s + 1) \uparrow \), then set

\[
L_{(k,j)} := N_s \cup \{ d^{\sum_{i=0}^{s} \Phi_k(i), 0} \}.
\]

If \( j = \sum_{i=0}^{s-1} (\Phi_k(i) + 1) + \Phi_k(s) \), and if \( \varphi_k(s + 1) \downarrow \), then set

\[
L_{(k,j)} := N_s \cup \{ d^{\sum_{i=0}^{s} \Phi_k(i), 0}, d^{\sum_{i=0}^{s+1} \Phi_k(i), 1} \}.
\]
(3.2) If \( j = \sum_{i=0}^{s}(\Phi_k(i) + 1) \), and if \( \varphi_k(s + 1) \uparrow \), then set
\[ L_{(k,j)} := N_s \cup \{ a^k b^j, d^{\sum_{i=0}^{s} \Phi_k(i)} \} \]

If \( j = \sum_{i=0}^{s}(\Phi_k(i) + 1) \), and if \( \varphi_k(s + 1) \downarrow \), then set
\[ L_{(k,j)} := N_s \cup \{ a^k b^j, d^{\sum_{i=0}^{s} \Phi_k(i)}, d^{\sum_{i=0}^{s+1} \Phi_k(i)} \} \]

(3.3) If \( j > \sum_{i=0}^{s}(\Phi_k(i) + 1) \), then set \( L_{(k,j)} := L_{(k,\sum_{i=0}^{s}(\Phi_k(i)+1))} \).

Then \( \mathcal{L}^{2/3} := (L_{(k,j)})_{k,j \in \mathbb{N}} \) is an indexed family with the following properties.
Let \( k \) be a natural number. If \( k \in \mathcal{T \mathcal{O} \mathcal{T}} \), then every language \( L_{(k,j)} \), \( j \geq \Phi_k(0) \), is finite, and contains a string of the form \( d^{\sum_{i=0}^{m} \Phi_k(i),h} \) for an \( m \in \mathbb{N} \), and an \( h \in \{1,2\} \). This string determines the index \( \langle k, \sum_{i=0}^{m-1} (\Phi_k(i) + 1) + \Phi_k(m) + h - 1 \rangle \). If \( k \notin \mathcal{T \mathcal{O} \mathcal{T}} \), then every language \( L_{(k,j)} \), which is not equal to \( L_k \), contains a string of the form \( d^{\sum_{i=0}^{m} \Phi_k(i),h} \) for an \( h \in \{0,1,2\} \).

Furthermore, let \( s \) be the least natural number with \( \varphi_k(s + 1) \uparrow \). Then
\[ L_{(k,\sum_{i=0}^{s-1} (\Phi_k(i) + 1) + \Phi_k(s))} \subset L_{(k,\sum_{i=0}^{s} (\Phi_k(i) + 1))} \]
and
\[ L_{(k,\sum_{i=0}^{s} (\Phi_k(i) + 1))} \cap L_{(k,\sum_{i=0}^{s-1} (\Phi_k(i) + 1) + \Phi_k(s))} \subset L_{(k,\sum_{i=0}^{s} (\Phi_k(i) + 1))} \cap L_{(k,0)} \]

In order to show that \( \mathcal{L}^{2/3} \) is properly monotonically identifiable by an OIM having access to \( \mathcal{T \mathcal{O} \mathcal{T}} \), let \( \tau \) be a text for \( L \in \text{range}(\mathcal{L}^{2/3}) \), and let \( k \in \mathbb{N} \) with \( \text{range}(\tau) \cap L_k \neq \emptyset \). Let \( x \) be a natural number.

**OIM \( M[A] \):** On input \( \tau_x \), \( M[A] \) works as follows.

(A) If \( \varphi_k \) is total, then \( M[A](\tau_x) := \langle k,0 \rangle \) if and only if \( \text{range}(\tau_x) \) does not contain a string \( d^{\sum_{i=0}^{m} \Phi_k(i),h} \) for an \( m \in \mathbb{N} \), and an \( h \in \{1,2\} \). If \( \text{range}(\tau_x) \) contains such a string, then set
\[ M[A](\tau_x) := \langle k, \sum_{i=0}^{m-1} (\Phi_k(i) + 1) + \Phi_k(m) + h - 1 \rangle \]

(B) If \( \varphi_k \) is not total, then distinguish the following cases.
(B1) If $x = 0$, then distinguish the following cases. If $\varphi_k(0) \downarrow$, then set $M[A](\tau_x) := \langle k, \Phi_k(0) \rangle$. If $\varphi_k(0) \uparrow$, then set $M[A](\tau_x) := \langle k, 0 \rangle$.

(B2) Assume $x > 0$. If $M[A](\tau_{x-1})$ is consistent with $\tau_x$, then set $M[A](\tau_x) := M[A](\tau_{x-1})$. Otherwise compute the smallest number $s$ with $\varphi_k(s + 1) \uparrow$, and distinguish the following cases.

i. If $\text{range}(\tau_x) \cap L_{\langle k, 0 \rangle}$ is a subset of $L_{\langle k, \sum_{i=0}^{s} \Phi_k(i)+1 \rangle}$, then compute the smallest natural number $j$ such that $\text{range}(\tau_x) \subseteq L_{\langle k, j \rangle}$, and set $M[A](\tau_x) := \langle k, j \rangle$.

ii. If $\text{range}(\tau_x) \cap L_{\langle k, 0 \rangle}$ is not a subset of $L_{\langle k, \sum_{i=0}^{s} \Phi_k(i)+1 \rangle}$, then set $M[A](\tau_x) := \langle k, 0 \rangle$.

end

Then $M[A]$ properly monotonically identifies $L^{2/3}$. Next, we show that $L^{2/3}$ is properly monotonically identifiable with $p = 2/3$. Define a $3$-sided PIM as follows. Let $L \in \text{range}(L^{2/3})$ be a language, and let $\tau$ be a text for $L$. Let $k \in \mathbb{N}$ with $\text{range}(\tau_0) \subseteq L_k$. Let $c$ be a coin-oracle, and let $x$ be a natural number.

PIM $P$: On input $\tau_x$, $P^c$ works as follows.

If $x = 0$ then set $P^c(\tau_0) := \langle k, 0 \rangle$ if and only if $c_0 \leq 1$. If $c_0 = 2$, then set $P^c(\tau_x) := \bot$.

If $x > 0$, then distinguish the following cases.

(A) If $\text{range}(\tau_x)$ does not contain a string $d_{\langle \sum_{i=0}^{m+1} \Phi_k(i), h \rangle}$ for an $m \in \mathbb{N}$, and an $h \in \{0, 1, 2\}$, then set $P^c(\tau_x) := P^c_{x-1}(\tau_{x-1})$.

(B) If $\text{range}(\tau_x)$ contains a string $d_{\langle \sum_{i=0}^{m+1} \Phi_k(i), h \rangle}$ for an $m \in \mathbb{N}$, and an $h \in \{1, 2\}$, then set $P^c(\tau_x) := \langle k, \sum_{i=0}^{m-1} \Phi_k(i) + 1 + \Phi_k(m) + h - 1 \rangle$.

(C) If $\text{range}(\tau_x)$ contains exactly one string of the form $d_{\langle \sum_{i=0}^{m+1} \Phi_k(i), 0 \rangle}$, $m \in \mathbb{N}$, then distinguish the following cases.

(C1) If $P^c_{x-1}(\tau_{x-1})$ is consistent with $\tau_x$, then set $P^c(\tau_x) := P^c_{x-1}(\tau_{x-1})$.

(C2) Assume that $P^c_{x-1}(\tau_{x-1})$ is not consistent with $\tau_x$. Compute the smallest natural number $u$ such that $\text{range}(\tau_x) \subseteq L_{\langle k, u \rangle}$. Then $u = \sum_{i=0}^{m-1} \Phi_k(i) + 1 + \Phi_k(m) + l$ for an $m \in \mathbb{N}$, and an $l \leq 1$. Distinguish the following cases.
i. If \( u = \sum_{i=0}^{m-1} (\Phi_k(i) + 1) + \Phi_k(m) \), then \( P^C_x(\tau_x) := \langle k, u \rangle \) if and only if \( c_0 \in \{0, 2\} \).

ii. If \( u = \sum_{i=0}^{m} (\Phi_k(i) + 1) \), then \( P^C_x(\tau_x) := \langle k, u \rangle \).

Then \( P^EMON_{prob}(2/3) \)-identifies \( L^{2/3} \). Finally, let \( A \) be an oracle, \( A \geq_T K \), and let \( M[A] \) be an OIM which monotonically identifies \( L^{2/3} \). Let \( k \) be a natural number. Let \( \tau^k \) be the canonical text for \( L_{(k,0)} \). Since \( M[A] \) \( EMON \)-identifies \( L_k \), there must be an \( n_0 \in \mathbb{N} \) such that \( M[A](\tau^k_{n_0}) = \langle k, j \rangle \) for a \( j < \Phi_k(0) \). Let \( \ell \) be the least natural number such that \( \sum_{i=1}^{\ell} (\Phi_k(i) + 1) + \Phi_k(\ell) > n_0 \). Then either \( \varphi_k \) is total or \( \min\{ r \in \mathbb{N} | \varphi_k(r) \uparrow \} \leq \ell \). Since \( K \leq_T A \), it follows that \( TOT \leq_T A \).

By combining the indexed families defined in Theorem 3.1.4 and Theorem 3.2.17, and Theorem 3.1.7 and Theorem 3.2.18, respectively, we are able to show that \( ECOV_{prob}(1/2) \) and \( EMON_{prob}(2/3) \) contain indexed families with oracle-complexity \( \{ B | TOT \leq_T B \} \).

**Corollary 3.2.19.**

There exist an indexed family \( L \in \text{LIM} \cap ECOV_{prob}(1/2) \) with \( \mathcal{O}_{ECOV}(L) = \{ A | TOT \leq_T A \} \), and an indexed family \( L' \in \text{LIM} \cap EMON_{prob}(2/3) \) with \( \mathcal{O}_{EMON}(L') = \{ A | TOT \leq_T A \} \).

**Proof.** The indexed family \( L^{1/2} \) defined in Theorem 3.2.17 is in \( \text{LIM} \), but not in \( ECOV[A] \) for any \( K \leq_T A < TOT \). Let \( L_K \) be an indexed family which is \( \text{LIM} \)-identifiable but not conservatively identifiable by any OIM \( M[B] \) where \( K \not<_T B \). The existence of such an indexed family follows from Theorem 3.1.7. Now we can easily define a join of \( L^{1/2} \) and \( L_K \) - for the construction see the proof of Lemma 3.1.24 - which is \( LIM \)-identifiable, but not conservatively identifiable by any OIM having access to an oracle \( A \) with \( TOT \not<_T A \). The same argument can be used in the case of monotonic probabilistic learning.

In the previous subsection, we defined a learning problem \( L \) to be maximal complicated in \( A \) with respect to \( \lambda \mu \) if and only if \( L \in \lambda \mu[A] \), and every OIM \( M[B] \) which \( \lambda \mu \)-identifies \( L \) can be transformed into a decision procedure for \( A \). This definition can be generalized in the following way.

**Definition 3.2.5.** Let \( \mu \in \{ COV, SMON, MON \} \). Let \( L \) be an indexed family. Then \( L \) is called \( \mu \)-complicated in \( E\mu[A] \) if and only if \( L \in E\mu[A] \),
and every OIM \( M[B] \) which \( C\mu \)-identifies \( \mathcal{L} \) can be transformed into a decision procedure for \( A \).

The next theorem shows that there is an indexed family \( \mathcal{L} \) which is \( cc \)-complicated in \( E\mu[A] \) for \( \mu \in \{ COV, MON \} \).

**Theorem 3.2.20.** There exists an indexed family \( \mathcal{L} \) with

(a) \( \mathcal{L} \in ECOV[\text{TOT}] \cap E\mu[\text{TOT}] \),

(b) \( \mathcal{L} \notin CCOV[A] \cup CMON[A] \) for every oracle \( A \) with \( \text{TOT} \nleq_T A \).

**PROOF.** Let \( \Sigma := \{ a, b, d \} \). Let \( \langle \ , \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N} \) an effective encoding of \( \mathbb{N} \times \mathbb{N} \). We define \( \langle L_{\langle k,j \rangle} \rangle_{k,j \in \mathbb{N}} \) in dependence of \( \varphi_k(j), k, j \in \mathbb{N} \). Let \( k \) be a natural number. In step 0 of the construction, we add \( a^k b^0 \) to every language. In the \( n \)-th step of the construction, compute the least \( s \in \mathbb{N}, s \geq 1 \), such that

\[
\sum_{i=0}^{s} \Phi_k(i) \leq n < \sum_{i=0}^{s+1} \Phi_k(i).
\]

Assume that \( L_{\langle k,j \rangle}, j \leq j_n \), are already defined. Distinguish the following cases.

1. If \( n = \sum_{i=0}^{s} \Phi_k(i) \), then add \( a^k b^n \) to every language \( L_{\langle k,j \rangle}, 0 < j \leq j_n \).

2. If \( n > \sum_{i=0}^{s} \Phi_k(i) \), then add every subset of \( \{ a^k b^n \mid \exists y \leq s \text{ with } n = \sum_{j=0}^{y} \Phi_k(j) \} \) to \( \mathcal{L} \).

Then \( \mathcal{L} := (L_{\langle k,j \rangle})_{k,j \in \mathbb{N}} \) fulfills the desired conditions. \( \mathcal{L} \) has the following properties. For every \( k \in \mathbb{N} \), \( k \in \text{TOT} \) if and only if \( L_{\langle k,0 \rangle} \) is infinite, and \( \mathcal{L} \) contains a language \( E \cup \{ d^m \} \) for every finite subset \( E \) of \( L_{\langle k,0 \rangle} \). Thereby, \( m \) is of the form \( \langle j, n \rangle \) for a \( j \in \mathbb{N} \), and an \( n \in \mathbb{N} \). Thus, it determines the index \( \langle k, j \rangle \). Moreover, \( k \notin \text{TOT} \) if and only if \( L_{\langle k,0 \rangle} \) is finite, \( \mathcal{L} \) contains every finite subset of \( L_{\langle k,0 \rangle} \), and finitely many languages \( E \cup \{ d^m \} \) with \( E \subseteq \{ a^k b^n \mid \exists y \leq s \text{ with } n = \sum_{j=0}^{y} \Phi_k(j) \} \), where \( s + 1 \) is the minimal natural number with \( \varphi_k(s + 1) \). Notice that \( m \) is of the form \( \langle j, n \rangle \).

In order to show that \( \mathcal{L} \) is properly conservatively (monotonically) identifiable by an OIM having access to \( \text{TOT} \), let \( \tau \) be a text for \( L \in \text{range}(\mathcal{L}) \), and let \( k \in \mathbb{N} \) with \( \text{range}(\tau_0) \subset L_k \). If \( k \in \text{TOT} \), then \( M[\text{TOT}] \) guesses \( \langle k, 0 \rangle \) as long as no string of the form \( d^m \) appears in the text, and the unique language
containing $d^m$ otherwise. Assume that $k \notin TOT$. If a string of the form $d^m$ appears in the text, then $M[TOT]$ outputs the unique language containing $d^m$. Otherwise $M[TOT]$ computes the minimal $s \in \mathbb{N}$ with $\varphi_k(s+1) \uparrow$, and searches for an index for the minimal finite language which contains the actual text. By construction, this language is equal to the range of the actual text. Hence, $M[TOT]$ monotonically - and thus conservatively - identifies $\mathcal{L}$.

In order to prove the last part of the theorem, let $M[A]$ be an OIM which identifies $\mathcal{L}$ conservatively or monotonically with respect to a class comprising hypothesis space $\mathcal{G}$. Let $\tau$ be the canonical text for $L_{(k,0)}$. In case $k \in TOT$, $L_{(k,0)}$ is infinite. Thus, there exists an $n \in \mathbb{N}$ such that $L(G_{M[A]}(\tau_n)) \cap L_{(k,0)}$ is a proper superset of range$(\tau_n)$. Intuitively, $L(G_{M[A]}(\tau_n))$ is a language containing relevant information about $L_{(k,0)}$ which is not contained in range$(\tau_n)$. Notice that such an $n \in \mathbb{N}$ exists in the case of conservative learning as well as in the case of monotonic learning.

In case $k \notin TOT$, every subset of $L_{(k,0)}$ is in $\mathcal{L}$. Thus, range$(\tau_n)$ is in $\mathcal{L}$ for every $n \in \mathbb{N}$. Assume that there exists an $n \in \mathbb{N}$ such that $L(G_{M[A]}(\tau_n)) \cap L_{(k,0)}$ is a proper superset of range$(\tau_n)$. In the case of conservative learning, $M[A]$ is not allowed to guess an overgeneralization of the text seen so far. In the monotonic case, $M[A]$ is not allowed to guess a language which contains information about $L_{(k,0)}$ which is not contained in range$(\tau_n)$, since in this case, $L_{(k,0)}$ is not monotonically identifiable by $M[A]$. Hence, our assumption leads to a contradiction.

Consequently, $k \in TOT$ if and only if there exists an $n \in \mathbb{N}$ such that $L(G_{M[A]}(\tau_n)) \cap L_{(k,0)}$ is a proper superset of range$(\tau_n)$. Consequently, $TOT$ is recursively enumerable in $A$. Since $K$ and $\overline{K}$ are $m$-reducible to $TOT$, both sets are recursively enumerable in $A$. Hence, $K \leq_T A$. Since $\overline{TOT}$ is recursively enumerable in $K$, it follows that $TOT$ is recursively enumerable in $A$. Hence, $TOT \leq_T A$. ☐

Thus, it immediately follows:

**Corollary 3.2.21.** For $\lambda \in \{E, \epsilon, C\}$, $\mu \in \{COV, MON\}$, there is an indexed family in $\lambda\mu[TOT]$ with oracle-complexity $\{B \mid TOT \leq_T B\}$. In particular, $\mathcal{L}$ is cc-complicated in $E\mu[A]$ for $\mu \in \{COV, MON\}$.

### 3.2.5 Indexed families with maximal oracle-complexity

We showed that in the case of conservative and monotonic learning, there are indexed families with oracle-complexity $\{B \mid TOT \leq_T B\}$. When dealing
with conservative learning, we can show that there are no indexed families with a higher oracle-complexity. Secondly, we prove that strong-monotonic learning is weaker than conservative and monotonic learning in the sense that every indexed family which is identifiable by an oracle machine \( M[A] \) is already identifiable by an OIM having access to \( \mathcal{K} \). Hence, the indexed family \( \mathcal{L}^\mathcal{K} \) defined in Theorem 3.1.11 has maximal oracle-complexity within strong-monotonic learning.

We start with the result concerning conservative learning. Stephan [96] proved that \( LIM[A] \subseteq LIM[K] \), and \( ECOV[A'] = LIM[A] \) for every oracle \( A \). Thus, for every oracle \( A \), \( ECOV[A] \) is contained in \( LIM[A] \) which is contained in \( LIM[K] = ECOV[\text{TOT}] \). Thus, the next corollary follows.

**Corollary 3.2.22.** Let \( A \) be an oracle, and let \( \mathcal{L} \) be an indexed family with \( \mathcal{L} \in ECOV[A] \). Then \( \mathcal{O}_{ECOV}(\mathcal{L}) \) contains an oracle \( B \) with \( B \leq_T \text{TOT} \).

Consequently, the indexed family \( \mathcal{L} \) defined in the proof of Theorem 3.2.20, and the indexed family constructed in the proof of Theorem 3.2.17 have maximal oracle-complexity with respect to proper conservative oracle learning.

Let \( A \) be an oracle. The following result shows that every indexed family \( \mathcal{L} \in ESMON[A] \) is already strong-monotonically identifiable by an OIM having access to \( \mathcal{K} \).

**Theorem 3.2.23.** Let \( \mathcal{L} \) be an indexed family which is strong-monotonically identifiable by an OIM \( M[B] \) with respect to a class comprising hypothesis space \( \mathcal{G} \). Then \( \mathcal{L} \) is properly strong-monotonically identifiable by an oracle machine having access to \( \mathcal{K} \).

**PROOF.** Let \( \mathcal{L} = (L_i)_{i \in \mathbb{N}} \) be an indexed family which is strong-monotonically identifiable by an OIM \( M[B] \) with respect to a class comprising hypothesis space \( \mathcal{G} \). Then \( \mathcal{L} \) has the following property. For every \( k \in \mathbb{N} \), and every \( \tau \in \text{text}(L_k) \), there exists an \( y \in \mathbb{N} \) such that for all \( j \in \mathbb{N} \), the following holds. Either \( \text{range}(\tau_y) \not\subseteq L_j \) or \( L_k \subseteq L_j \). Such a sequence \( \tau_y \) exists, since otherwise \( M[B] \) could not identify \( \mathcal{L} \) strong-monotonically. For a related proof see [109].

Now we are able to define an OIM \( M[\mathcal{K}] \) identifying \( \mathcal{L} \) properly strong-monotonically. Let \( k \) be a natural number, and let \( \tau \in \text{text}(L_k) \). Let \( x \) be a
natural number.

**OIM** $M[\mathcal{K}]$: On input $\tau_x$, $M[\mathcal{K}]$ works as follows.

$M[\mathcal{K}]$ searches whether there is an $i \in \mathbb{N}$, $i \leq x$, such that the following conditions hold.

(A) $\text{range}(\tau_x) \subseteq L_i$, and

(B) for all $j \in \mathbb{N}$: either $\text{range}(\tau_x) \not\subseteq L_j$, or $w \not\in L_i \setminus L_j$ for all $w \in \Sigma^*$.

If there is such an $i$, then $M[\mathcal{K}]$ guesses $i$. If not, then it requests the next input string.

Notice that, for every $i \in \mathbb{N}$, $M[\mathcal{K}]$ can compute whether or not $\text{range}(\tau_x) \not\subseteq L_j$, or $L_i \subseteq L_j$ for all $j \in \mathbb{N}$ by using $\mathcal{K}$-oracle. It is easy to see that $M[\mathcal{K}]$ properly strong-monotonically identifies $L$. ⋄

Our last corollary can be drawn from Theorem 3.1.15 and Theorem 3.2.23 by using any indexed family containing an infinite language $L$ and all finite subsets of $L$.

**Corollary 3.2.24.** $\text{ESMON}_{\text{prob}}(1/2) \setminus \bigcup_{A \subseteq \mathbb{N}} \text{CSMON}[A] \neq \emptyset$. 

140
Summary and Conclusions

Our research was motivated by the fact that, in the field of probabilistic inductive inference, most of the learning models investigated do not reflect the human ability to learn with high probabilities (cf. Chapter 1.1.4). Moreover, only learning problems which are identifiable with a probability close to 1 are interesting with respect to potential applications.

We showed that a variant of the probabilistic learning model introduced by Freivalds [37], namely probabilistic inductive inference under monotonicity constraints, fulfills our intuition of probabilistic learning. In Section 2.1 and Section 2.2, we proved that the probabilistic hierarchies in the case of proper and class preserving probabilistic learning of indexed families under monotonicity constraints are highly structured. In particular, all these learning models yield dense or strictly decreasing probabilistic hierarchies in an interval $(r, 1]$ where $r < 1$. This is a completely new phenomenon in the field of probabilistic inductive inference.

Furthermore, we investigated how powerful the investigated probabilistic learning models are. For measuring the power of the probabilistic learning machines, we asked how much additional information the deterministic learners need for achieving at least the learning power of their probabilistic counterparts. We introduced a complexity measure for probabilistic learning classes and indexed families, namely oracle-complexity. In the following, we discuss whether oracle-complexity is able to give us an insight into the nature of the probabilistic learning models considered.

Consider the following examples by Schäfer-Richter [89]. Let $\mathcal{P}$ be the set of all partial recursive functions, and let $\mathcal{R}$ be the set of all recursive functions. Then the following results were proven.

1. $\mathcal{P}$ is identifiable from text by an oracle machine which has access to $\mathcal{K}$ (cf. [22]). Moreover, if an oracle machine $M[B]$ identifies $\mathcal{P}$ from text, then $\mathcal{K} \leq_T B$. $\mathcal{R}$ is identifiable from text by an oracle machine which has access to an oracle $A <_T K$ (cf. [1]).

2. Let $M[A]$ be an oracle machine which identifies $\mathcal{R}$ from text. Then $M[A]$ machine can be transformed into an oracle machine which identifies $\mathcal{P}$ from effective text. Moreover, the oracle machine $M[A]$ can be transformed into a solution for the problem to infer a minimal program for a $\varphi \in \mathcal{R}$, or into an oracle machine which identifies every $\varphi \in \mathcal{R}$.
consistently. (cf. [89]).

Hence, all these problems are equivalent with respect to oracle-complexity. Thus, oracle-complexity is not able to reflect the special quality of the complexity of these learning problems. However, in the setting of probabilistic learning under monotonicity constraints, the results are more meaningful.

First, our results show that the oracle-complexity of the investigated probabilistic learning models coincides with the threshold-results proven in Section 2.3. Let $\lambda \in \{E, \varepsilon\}$. For conservative learning, every indexed family which is $\lambda COV$-identifiable with a probability $p > 1/2$ is

\(\lambda COV\)-identifiable by an oracle machine which has access to $K$.

The same holds for strong-monotonic probabilistic learning with probability $p > 1/2$, and for monotonic probabilistic learning with probability $p > 2/3$. In the case of strong-monotonic learning with probability $p > 2/3$, we showed that every indexed family which is $SMON$-identifiable with probability $p > 2/3$ is $SMON$-identifiable. Furthermore, every indexed family which is $ESMON$-identifiable with probability $p > 2/3$ is $SMON$-identifiable.

\( ESMON\)-identifiable by an oracle machine which has access to an arbitrary Peano-complete oracle.

We may interpret these results in the following way. The probability $p = 1/2$ is a breakpoint for the power of class comprising conservative learning as well as for the power of conservative oracle identification. In the case of monotonic learning, the probability 2/3 plays the analogous role. For strong-monotonic learning, both parameters are relevant. Thus, the parameters 1/2 and 2/3 are characteristic for conservative, monotonic and strong-monotonic probabilistic learning, respectively, within two different learning models, namely class comprising probabilistic learning and oracle identification.

Next, we discuss whether the results proved in Section 3.2 do reflect the results concerning the set inclusions between the probabilistic learning classes (cf. Conclusion 2.4). Obviously, strong-monotonic probabilistic learning is weaker than conservative and monotonic probabilistic learning with respect to set inclusion. This coincides with the following result.

\(\lambda COV\)-identifiable by an oracle machine $M[A]$ with respect to a class preserving hypothesis space
is already properly strong-monotonically identifiable by an oracle machine having access to \( \mathcal{K} \). Thus, we cannot characterize \( TOT \) in the setting of strong-monotonic learning.

\(+\) However, there exists an indexed family \( \mathcal{L}^{1/2} \in ECOV_{\text{prop}}(1/2) \) which is properly conservatively identifiable by an oracle machine \( M[B] \) if and only if \( TOT \leq_T B \). The same holds for monotonic learning with probability \( p = 2/3 \).

Thus, by using oracle-complexity, we are able to express the weakness of strong-monotonic probabilistic learning in contrast to conservative and monotonic probabilistic learning, respectively.

In the case of conservative probabilistic learning and monotonic probabilistic learning, the situation is different. Proper conservative and proper monotonic probabilistic learning are not comparable with respect to set inclusion. Our results show that the characteristics of the two probabilistic learning models can be described in terms of oracle-complexity.

\(+\) Proper monotonic probabilistic learning is stronger than proper conservative probabilistic learning in the sense that it is not possible to compensate the power of proper monotonic probabilistic learning with probability \( p = 2/3 \). In particular, we can find indexed families with oracle-complexity \( \{ A \mid TOT \leq_T A \} \) in \( EMON_{\text{prob}}(2/3) \).

\(+\) Proper conservative probabilistic learning is stronger than proper monotonic probabilistic learning in the sense that it is possible to strongly characterize every oracle \( A \leq_T \mathcal{K} \) with respect to \( ECOV \).

This leads to the conclusion that oracle-complexity is a useful complexity measure in the setting of probabilistic learning indexed families under monotonicity constraints.

The results proven in Section 3.2 are also interesting from a recursion theoretic point of view. Since every oracle \( A \leq_T \mathcal{K} \) can be encoded in an indexed family \( \mathcal{L}^A \in ECOV_{\text{prop}}(1/2) \), there exists an injective operator which embeds the Turing degrees below \( \text{deg}_T(\mathcal{K}) \) into the probabilistic learning class \( ECOV_{\text{prob}}(1/2) \). The same result holds for arbitrary \( p > 1/2 \). Our results show that Turing complexity is not rich enough to measure the complexity of the learning problems in \( ECOV_{\text{prob}}(1/2) \). Thus, oracle-complexity is not a “trivial” measure. In particular, there are interesting classes of sets, for
example the set of Peano-complete oracles, which can be encoded in appropriate learning problems.

There are several interesting questions left open. The problem whether \( \text{MON}_{\text{prob}}(p) = \text{MON} \) or not is not yet solved. We suggest that an analogous result to Theorem 2.1.8 holds. Furthermore, it remains to show how the probabilistic hierarchies in the interval \([0, r]\) behave. We suggest that in the case of class preserving conservative and proper and class preserving monotonic probabilistic learning, it is possible to show that the corresponding probabilistic hierarchies are dense in the interval \([0, 1]\). Moreover, we conjecture that in the case of class preserving strong-monotonic learning, the probabilistic hierarchy is discrete in the interval \([1/2, 1]\) with breakpoints at \( p_n = n/2n - 1 \) for \( n \in \mathbb{N} \). Another open question concerns the relation between conservative and monotonic probabilistic learning for \( p > 2/3 \).

Furthermore, it would be of interest whether it is possible to measure the additional power of probabilistic learning in terms of amalgamation complexity, i.e., in the construction effort which is necessary for a deterministic machine in order to learn the indexed families in the probabilistic learning classes \( \lambda \mu_{\text{prob}}(p), \mu \in \{\text{COV}, \text{SMON}, \text{MON}\}, \lambda \in \{E, \varepsilon\} \). Some work in this field was done by Fritzsche [42]. Finally, further investigation is necessary in order to characterize the sets and classes of sets which can be characterized by an indexed family with respect to a given monotonicity constraint, and which oracle learning classes contain maximal complicated problems when dealing with class preserving conservative, proper and class preserving strong-monotonic and monotonic learning, respectively.
Bibliography


[33] T. Erlebach, P. Rossmanith, H. Stadtherr, A. Steger, and T. Zeugmann, Learning One-Variable Pattern Languages Very Efficiently on Average,


[88] P. Rossmanith, T. Zeugmann, Learning k-Variable Pattern Languages Efficiently Stochastically Finite on Average from Positive Data, in: V. Honavar and G. Slutzki, eds., *Proc. 4th International Colloquium on*
Grammatical Inference - ICGI’98 Lecture Notes in Artificial Intelligence


