A Learnability Analysis on Neuro-Symbolic Learning

Hao-Yuan He, Ming Li ✉
National Key Laboratory for Novel Software Technology, Nanjing University
School of Artificial Intelligence, Nanjing University
{hehy, lim}@lamda.nju.edu.cn

Abstract

This paper presents a comprehensive theoretical analysis of the learnability of neuro-symbolic (NeSy) tasks within hybrid systems. We characterize the learnability of NeSy tasks by their derived constraint satisfaction problems (DCSPs), demonstrating that a task is learnable if and only if its corresponding DCSP admits a unique solution. Under mild assumptions, we establish the sample complexity for learnable tasks and show that, for general tasks, the asymptotic expected concept error is controlled by the degree of disagreement among DCSP solutions. Our findings unify the characterization of learnability and the phenomenon of reasoning shortcuts, providing theoretical guarantees and actionable guidance for the principled design of NeSy systems.

1 Introduction

Neuro-symbolic learning (NeSy) seeks to integrate data-driven learning with knowledge-driven reasoning within a unified framework (Hitzler & Sarker, 2022; Marra et al., 2024). State-of-the-art NeSy approaches predominantly employ hybrid systems that map input queries $\boldsymbol{x}$ to concepts $\hat{\boldsymbol{z}}$ via a learning model, subsequently utilizing a symbolic system $\texttt{KB}$ to deduce the final answer $\hat{y}$. Feedback from $\texttt{KB}$—such as pseudo-labels in abductive learning (ABL) (Zhou, 2019; Dai et al., 2019) or weighted model counting in DeepProbLog (Manhaeve et al., 2018, 2021a)—guides further learning. This paradigm has found broad application in domains including puzzle solving, code generation, and autonomous path planning (Jiao et al., 2024; Li et al., 2024; Hu et al., 2025a).

A central challenge in NeSy is that systems are typically trained end-to-end in a weakly supervised manner, relying solely on $(\boldsymbol{x}, y)$ pairs, with the underlying concepts $\boldsymbol{z}$ remaining unobserved. The objective is to learn a model $f : \mathcal{X} \to \mathcal{Z}$ that generalizes effectively, minimizing the concept risk:

$$R_{0/1}(f) = \mathbb{E}_{(x,z)} \left[\mathbb{I} \left(f(x) \neq z \right) \right].$$

However, in the absence of supervision $\boldsymbol{z}$, only the surrogate NeSy risk is accessible:

$$R_{\text{NeSy}}(f) = \mathbb{E}_{(\boldsymbol{x},y)} \left[\mathbb{I} \left(f(\boldsymbol{x}) \wedge \texttt{KB} \not\models y \right) \right].$$

This motivates a fundamental question regarding the *learnability* of NeSy tasks:

Under what conditions can the concept risk be minimized through empirical risk minimization over the NeSy risk, given a finite sample set as its size approaches infinity?

Recently, Marconato et al. (2023b) identified the reasoning shortcut problem, wherein a reasoning shortcut (RS) refers to a concept distribution that achieves maximal likelihood on training data yet deviates from the true underlying concept distribution. This phenomenon is closely related to the statistical learnability of NeSy systems, as models may minimize empirical NeSy risk without necessarily minimizing the concept risk. While several approaches have been proposed to address reasoning shortcuts Marconato et al. (2023a, 2024), a rigorous theoretical framework connecting RSs to the statistical learnability of NeSy tasks remains underexplored.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

To address this gap, we analyze the learnability of the NeSy task within the probably-approximately-correct (PAC) framework (Valiant, 1984), focusing on restricted hypothesis spaces (cf. section 3.3). Our key insight is to formulate NeSy tasks as *derived constraint satisfaction problems* (DCSPs): a task is learnable if and only if its DCSP has a unique solution.

We further introduce *disagreement* d among DCSP solutions as a finer-grained measure of uncertainty. For learnable tasks, we establish the sample complexity $1/\kappa \cdot \log(|\mathcal{B}|/\epsilon)$, where $\kappa, |\mathcal{B}|$ are task-specific and ϵ is the desired concept error (cf. theorem 3.6). For general NeSy tasks, we derive an upper bound on the expected concept error, showing it is bounded by d/L, where L is the concept space size (cf. theorem 3.7). Moreover, with the DCSP perspective, we find that aggregating unlearnable tasks in a multi-task learning manner reduces the degree of ambiguity, thereby enhancing overall task learnability.

Our analysis aligns with the RS phenomenon (Marconato et al., 2023a,b), wherein models may achieve low empirical risk yet incur high concept risk. The existence of deterministic RSs corresponds to the presence of multiple DCSP solutions. We show that concept error correlates more strongly with solution disagreement than with the number of solutions (cf remark 1), underscoring disagreement d as a more informative indicator.

In summary, our contributions are as follows:

- We establish a rigorous theoretical foundation for NeSy learnability within the hybrid systems paradigm and derive both sample complexity and asymptotic bounds on concept error.
- The DCSP framework proposed facilitates intuitive analysis of learnability using CSP solvers, while solution disagreement provides a detailed perspective on concept error analysis.
- Empirical studies are conducted to validate our theoretical findings.

The remainder of the paper is organized as follows: Section 2 reviews the background and related work on NeSy. Section 3 presents our theoretical analysis of NeSy task learnability. Section 4 details empirical validations of our theoretical results. Section 5 discusses limitations, and section 6 concludes the paper.

2 Preliminaries

This section first introduces the problem setup of neuro-symbolic learning, followed by a presentation of the two principal neuro-symbolic methods: probabilistic neuro-symbolic learning and abductive learning. Subsequently, we discuss the most related works, including those on reasoning shortcuts and theoretical analyses of NeSy.

2.1 Problem Setup

A typical hybrid neuro-symbolic system consists of two components: a *machine learning* model (e.g., a neural network) and a *logical reasoning* model (e.g., a first-order logic solver). The learning model $f : \mathcal{X} \to \mathcal{Z}$ maps an instance x (e.g., image, text, or audio) from the input space $\mathcal{X}$ to an intermediate concept z (e.g., primitive facts or predicates) within the symbol space $\mathcal{Z}$, where $|\mathcal{Z}| = L$. The reasoning model $\texttt{KB}$ consists of rules over the concept space and can be implemented using any logic-based system, such as ProbLog (De Raedt et al., 2007) or answer set programming (Dimopoulos et al., 1997). Assume that a labeling function $g : \mathcal{X} \to \mathcal{Z}$ exists such that $z = g(x)$. The learning model, belonging to a hypotheses family $\mathcal{F}$, is parameterized by θ, and $p_\theta(\cdot)$ represents the likelihood estimated by the model, where $f(x) = \arg\max_{z \in \mathcal{Z}} p_\theta(z \mid x)$.

During inference, the learning model f accepts multiple instances as a sequence $\boldsymbol{x} = (x_1, \ldots, x_m)$ and outputs a sequence of concepts $\hat{\boldsymbol{z}} = (\hat{z}_1, \ldots, \hat{z}_m)$. The output $\hat{\boldsymbol{z}}$ is then passed to the reasoning model $\texttt{KB}$, which infers the final label $y \in \mathcal{Y}$ through logical entailment, i.e., $\hat{\boldsymbol{z}} \wedge \texttt{KB} \models y$. To simplify the inference process of $\texttt{KB}$, we represent it by a logical forward operator $\sigma(\cdot)$ such that $\sigma(\hat{\boldsymbol{z}}) = y$. In a standard neuro-symbolic learning setup (Dai et al., 2019; Manhaeve et al., 2021a; Li et al., 2023; Marra et al., 2024), the training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ are sampled from data distribution $\mathcal{D} = (\mathcal{X}^m, \mathcal{Y})$. Therefore, a neuro-symbolic task can be formally defined as a triple $\mathcal{T} = \langle \mathcal{X}, \mathcal{Y}, \texttt{KB} \rangle$.

Example 1 (Addition)**.** The input $\boldsymbol{x} \in \mathcal{X}^2$, where $\mathcal{X}$ denotes digit images. The concepts $\mathcal{Z}$ consist of digits from `0` to `9`. The data takes the form ( , ) $\mapsto$ `1`, with the logical forward function $\sigma(\cdot) := \texttt{SUM}(\cdot, \cdot)$. The label space $\mathcal{Y}$ is defined as the addition results, i.e., from `0` to `18`.

Note that when all final labels y are identical (e.g., $\boldsymbol{z} \wedge \texttt{KB} \models \top$), as in the case of code generation where all code must satisfy a syntax constraint (Jiao et al., 2024), the final label y can be omitted for simplicity. The analysis presented in this paper can be easily adapted to such cases.

The success of NeSy systems highly depends on the recognition of intermediate concepts. To evaluate the concept-level performance, we define the concept risk as follows:

$$R_{0/1}(f; g) = \mathbb{E}_x \left[\mathbb{I}\left(f(x) \neq g(x)\right)\right]. \tag{1}$$

For simplicity, we omit g and denote (1) as $R_{0/1}(f)$. However, optimizing (1) is challenging due to the lack of supervision regarding the intermediate concept.

2.2 Neuro-Symbolic Methods

To minimize the concept risk (1), the key idea of current NeSy methods (Manhaeve et al., 2018; Dai et al., 2019) is to optimize the neuro-symbolic risk as a surrogate, which aims to minimize the discrepancy between the learning and reasoning models.

$$R_{\text{NeSy}}(f) = \mathbb{E}_{(\boldsymbol{x},y)} \left[\mathbb{I}\left(f(\boldsymbol{x}) \wedge \texttt{KB} \not\models y\right)\right]. \tag{2}$$

The optimization process is to select the optimal $f^* \in \mathcal{F}$ that minimizes the NeSy risk; that is:

$$f^* = \arg\min_{f \in \mathcal{F}} R_{\text{NeSy}}(f). \tag{3}$$

Probabilistic Neuro-Symbolic Learning. Probabilistic neuro-symbolic learning (PNL, Manhaeve et al. 2021b) methods adopt reasoning models via probabilistic logic programming, such as DeepProblog (Manhaeve et al., 2018, 2021a), NeurASP (Yang et al., 2020), and Scallop (Li et al., 2023). Since the NeSy risk (2) is non-continuous, PNL aims to minimize the following objective:

$$-\mathbb{E}_{(\boldsymbol{x},y)} p\left[y \mid \boldsymbol{x}; f, \texttt{KB}\right]. \tag{4}$$

Reformulating (4), we can express the objective as follows:

$$-\mathbb{E}_{(\boldsymbol{x},y)} \log \sum_{\boldsymbol{z}} \mathbb{I}(\boldsymbol{z} \wedge \texttt{KB} \models y) \cdot p\left[\boldsymbol{z} \mid \boldsymbol{x}; f, \texttt{KB}\right]. \tag{5}$$

Equation (5) is referred to as the probabilistic neuro-symbolic learning risk, denoted as $R_{\text{PNL}}(f)$.

The key operation for calculating the PNL risk is $\sum_{\boldsymbol{z}} \mathbb{I}(\boldsymbol{z} \wedge \texttt{KB} \models y) \cdot p\left[\boldsymbol{z} \mid \boldsymbol{x}; f, \texttt{KB}\right]$, also well-known as *weighted model counting* (WMC), which requires enumerating all possible worlds that satisfy the constraints of the symbolic system. This operation can be performed using various approaches, such as ProbLog (De Raedt et al., 2007), answer set programming Dimopoulos et al. (1997), and so on. However, in general, the computational complexity of WMC is #P (Maene et al., 2024), which makes PNL methods challenging to scale.

Abductive Learning. Unlike PNL methods, abductive learning methods (ABL) (Dai et al., 2019; Huang et al., 2021; Hu et al., 2025a) infer the *most* plausible concepts through abductive reasoning and use them to update the model. The objective of ABL is to minimize the risk:

$$R_{\text{ABL}}(f) = -\mathbb{E}_{(\boldsymbol{x},y)} \log\left(p\left[y, \bar{\boldsymbol{z}} \mid \boldsymbol{x}; f, \texttt{KB}\right]\right), \tag{6}$$

where $\bar{\boldsymbol{z}} = \min_{\boldsymbol{z} \in A(y)} \text{Score}(\boldsymbol{z}, f(\boldsymbol{x}))$ represents the most likely candidate in the abduction set. The abduction set $A(y)$ includes all possible concepts $\boldsymbol{z}$ that satisfy the constraints of KB and have a non-zero measure, i.e., $p[\boldsymbol{z}] > 0$. The score function measures the alignment between a candidate $\boldsymbol{z}$ and the model's prediction $f(\boldsymbol{x})$. For instance, Dai et al. (2019) use the Hamming distance.

ABL enhances computational efficiency by concentrating on the most plausible candidates, thereby avoiding the enumeration of all possible worlds. However, the inherent ambiguity of abduction can lead to incorrect candidate selection, introducing bias into the learning process (He et al., 2024b).

We now present a unified perspective: both PNL and ABL approaches are capable of effectively optimizing (2), thereby allowing the analysis to be uniformly applicable to both methods. Accordingly, we formally state the following theorem, with its proof provided in section B.1.

Theorem 2.1. *A minimizer of R_{PNL} or R_{ABL} is also a minimizer of R_{NeSy}. For each surrogate $R_s \in \{R_{PNL}, R_{ABL}\}$, we have:*

$$\arg\min_{f \in \mathcal{F}} R_s(f) \subseteq \arg\min_{f \in \mathcal{F}} R_{NeSy}(f),$$

2.3 Related Works

Here we review related works below that focus on reasoning shortcuts and the theoretical analysis of neuro-symbolic learning. A more comprehensive review of related works is available in section A.

Reasoning Shortcuts. A reasoning shortcut (RS) is formally defined as a distribution that maximizes the training likelihood while deviating from the true concept distribution (Marconato et al., 2023b). To address RS, a variety of mitigation strategies have been proposed, including continual learning paradigms that structure tasks sequentially to promote knowledge retention (Marconato et al., 2023a), entropy regularization to encourage more faithful concept representations (Marconato et al., 2024), and the development of dedicated benchmarks for systematic evaluation (Bortolotti et al., 2024). Recent works have also introduced new metrics and theoretical analyses to better characterize and quantify shortcut risks (Yang et al., 2024). Despite these advances, a comprehensive theoretical understanding of RS remains limited, particularly regarding the specific concept error analysis under the statistical learning framework. Our work builds upon these foundations by providing a unified theoretical framework that elucidates the learnability of neuro-symbolic tasks, offering new insights into this phenomenon.

Theoretical Analyses. Prior theoretical frameworks, such as multi-instance partial label learning with the M-unambiguity condition (Wang et al., 2023), provide concept error bounds based on the VC-dimension. However, this analysis assume repetitive input patterns, such as $[z, z, \dots]$, thereby limiting the applicability to real-world scenarios that require heterogeneous predicates and facts as logical inputs. Tao et al. (2024) examine scenarios where randomly selecting abduction candidates results in a consistent optimization objective within the ABL framework. They formulate the learning process as a weakly supervised learning problem and analyze its consistency through a probabilistic matrix $\tilde{Q}$. However, construction of such a matrix $\tilde{Q}$ requires full knowledge of the underlying concept sequence distribution, which is not easily obtainable. Additionally, Yang et al. (2024) introduce a shortcut risk metric R_s to quantify the discrepancy between true risk and surrogate risk. While they establish error bounds for this metric, a low shortcut risk does not necessarily guarantee a low concept error. Thus, a comprehensive theoretical analysis of the concept error remains lacking.

In summary, our work advances the field by providing a unified and comprehensive theoretical framework for analyzing the learnability of neuro-symbolic learning tasks. We establish necessary and sufficient conditions for learnability under mild assumptions, introduce a constraint satisfaction perspective that enables systematic verification, and derive meaningful error bounds even in unlearnable cases.

3 Learnability Analysis

In this section, we examine the learnability of neuro-symbolic (NeSy) tasks, specifically whether the concept risk can be minimized via empirical risk minimization (ERM) over the NeSy risk as the sample size approaches infinity. While learnability is attainable in certain scenarios, it is not universally guaranteed. To elucidate the underlying reasons, we conduct a rigorous learnability analysis to address the question: *Which classes of NeSy tasks are learnable?*

Analogous to the standard probably approximately correct (PAC) learning framework (Valiant, 1984), we formalize the learnability of a NeSy task as follows:

Definition 3.1. Let N denote the size of samples drawn i.i.d. from $\mathcal{D}$, $\mathcal{T}$ represent a NeSy task, and $\mathcal{F}$ denote the hypothesis space. We say that $\mathcal{T}$ is *learnable* if: for any $0 < \epsilon$, $\delta < 1$ and distribution $\mathcal{D}$, there exists an algorithm $\mathcal{A}$ and an integer $N_{\epsilon,\delta}$ such that, whenever $N \geq N_{\epsilon,\delta}$, the selected hypothesis $\hat{f} = \mathcal{A}(D)$ satisfies $p[R_{0/1}(\hat{f}) \leq \epsilon] \geq 1 - \delta$. Otherwise, we say that it is *unlearnable*.

Our analysis focuses on the ERM algorithm as $\mathcal{A}$, given its proven efficacy in common learning settings such as supervised classification and regression, where a problem is learnable if and only if it is learnable by ERM (Blumer et al., 1989; Alon et al., 1997).

3.1 Restricted Hypothesis Space

Unlike conventional supervised learning, which learns a hypothesis $x \mapsto y$ from pairs (x, y), a NeSy task seeks a mapping $\boldsymbol{x} \mapsto \boldsymbol{z}$ from pairs $(\boldsymbol{x}, y)$, where $\boldsymbol{z}$ denotes latent symbolic variables constrained by the reasoning module. Ideally, each y determines a unique $\boldsymbol{z}$; that is, $\forall y \in \mathcal{Y},\ |A(y)| = 1$. In this case the task reduces to a standard learning problem (Vapnik, 1999). In practice this is rarely true: many y admit multiple feasible solutions $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_k$, making the task inherently ambiguous. We call the task *ambiguous* if there exists $y \in \mathcal{Y}$ with $|A(y)| \geq 2$. In statistical learning theory, the complexity of a hypothesis space is characterized by the notion of *shattering*.

Definition 3.2 (Shattering)**.** A hypothesis space $\mathcal{F}$ *shatters* a finite set $X = \{x_1, \ldots, x_m\}$ with respect to a label space $\mathcal{Z}$ if, for every labeling function $\ell : X \to \mathcal{Z}$, there exists a hypothesis $f \in \mathcal{F}$ such that $f(x_i) = \ell(x_i)$ for all $x_i \in X$.

Using this notion, we obtain the following proposition.

Proposition 3.3. *For an ambiguous NeSy task $\mathcal{T}$, if the hypothesis space $\mathcal{F}$ shatters the task, then there exists a hypothesis f^* that minimizes R_{NeSy} but does not minimize $R_{0/1}$.*

The proof is provided in section B.2. Proposition 3.3 suggests that ambiguous NeSy tasks may be unlearnable when the hypothesis space is very complex, such as nearest neighbor, whose Vapnik-Chervonenkis dimension is infinite (Karacali & Krim, 2003), or deep neural networks (Bartlett & Maass, 2003) without any regularization terms. This issue arises due to overfitting caused by the high memorization capacity of models (Zhang et al., 2021). Previous studies emphasize the importance of constraining the hypothesis space in NeSy tasks (Yang et al., 2024). For example, pre-training models or self-supervised learning methods (Sohn et al., 2020) have been shown to promote clustering properties in neural networks, further enhancing generalization performance.

Consider a scenario where a pre-trained model satisfies a clustering property (Huang et al., 2021), meaning that instances representing the same concept are grouped together in feature space. In the ambiguous task described in example 1, if the model correctly processes a key sample such as $\texttt{SUM}(\boxed{0}, \boxed{0}) = \texttt{0}$, it can reliably identify `0`. This, in turn, simplifies subsequent tasks. For example, once the model recognizes $\texttt{SUM}(\boxed{0}, \boxed{1}) = \texttt{1}$, it can correctly identify `1`. By iteratively applying this process, the model can learn to recognize all relevant concepts despite initial ambiguity.

The above process highlights the need to restrict the hypothesis space for the learning system. This hypothesis space ensures consistent mappings between concepts and labels. Let $\mathcal{F}^*$ be a restricted hypothesis space which ensures that instances with the same label correspond to the same concept, and vice versa. Given the labeling function g, formally, for any $f \in \mathcal{F}^*$:

$$\forall x_1, x_2 \in \mathcal{X}, \quad g(x_1) = g(x_2) \iff f(x_1) = f(x_2).$$

3.2 Derived Constraint Satisfaction Problem

The restricted hypothesis space implicitly partitions the raw input space $\mathcal{X}$ into L clusters. We use $\langle x \rangle_i$ to denote the cluster $\{x \mid x \in \mathcal{X}, f(x) = i\}$. The learning process is to establish a mapping between the clusters $\{\langle x \rangle_1, \ldots, \langle x \rangle_L\}$ and $\mathcal{Z}$ that minimizes the NeSy risk. This process inherently transforms the NeSy learning problem into a *constraint satisfaction problem* (CSP). In this paper, we refer to it as a derived CSP (DCSP).

The derived constraint satisfaction problem for a NeSy task $\mathcal{T}$ is defined as a triple $\langle \mathsf{V}, \mathsf{D}, \mathsf{C} \rangle$, where: $\mathsf{V} = \{V_1, \ldots, V_L\}$ are the variables, $\mathsf{D} = \{D_1 = \mathcal{Z}, \ldots, D_L = \mathcal{Z}\}$ are the domains, and $\mathsf{C} = \{C_1, \ldots, C_N\}$ are the constraints. Each V_i corresponds to a mapping from $\langle x \rangle_i$ to a concept label. For convenience, we slightly abuse notation by letting $\mathsf{V}(\boldsymbol{x})$ denote a mapping from an input sequence to the corresponding concept sequence determined by the mapping set V. Each C_j corresponds to a constraint $(\boldsymbol{x}_j, y_j)$, e.g., $\mathsf{V}(\boldsymbol{x}_j) \wedge \texttt{KB} \models y_j$. Solving the DCSP is to find a consistent assignment I that satisfies all constraints.

A DCSP solution I corresponds to an assignment of values to variables, expressed as $I = \{(V_1, v_1), \ldots, (V_L, v_L)\}$, where each v_i is the value assigned to the variable V_i. For simplicity, we denote the solution as $I = (v_1, \ldots, v_L)$ by omitting the variables. Here we only discuss the case when the DCSP has solution; Otherwise, the learning model will inevitably conflict with the background `KB`.

3.3 Conditions of Learnability

In general, the solution to a DCSP may not be unique, i.e., multiple distinct solutions may exist. We denote the solution space as $\mathcal{S} = \{I_1, \ldots, I_k\}$. To characterize the relationships among these solutions, we define an operation Union(), which captures the common assignments among the solutions. When the input set consists of a single element, this operation simply returns that element. The *DCSP solution disagreement* d quantifies the inconsistency among all solutions:

$$d = L - |\text{Union}(\mathcal{S})|.$$

The disagreement d measures the number of variables whose values differ across the solutions in $\mathcal{S}$. If $d = 0$, i.e., $|\mathcal{S}| = 1$, there is a unique solution. In this case, the optimal hypothesis can be identified by minimizing the NeSy risk. Formally, we have:

Lemma 3.4. *For a NeSy task $\mathcal{T}$, if the DCSP solution disagreement $d = 0$, then the NeSy risk is consistent with the concept risk. Formally, for any $f \in \mathcal{F}$:*

$$R_{NeSy}(f) \to 0 \iff R_{0/1}(f) \to 0.$$

Proof Sketch. The direction from the right-hand side to the left-hand side is straightforward; here, we focus on proving the reverse direction. We demonstrate this by showing that *if the concept risk is non-zero, then the NeSy risk cannot be zero* (contraposition). If the concept risk is non-zero, there must be at least one misclassified instance where f assigns an incorrect label. Given that the DCSP solution is unique and there is no disagreement (i.e., $d = 0$), any such misclassification directly results in a non-zero NeSy risk. Therefore, if the NeSy risk is zero, it follows that the concept risk must also be zero. □

The detailed proof is in section B.3. To proceed, we introduce the following mild assumption.

Assumption 3.5. *The set of possible concept sequences, $\mathcal{B} = \bigcup_{y \in \mathcal{Y}} A(y)$, where $A(y)$ is the set of valid concept combinations for label y, has finite cardinality; and the probability of sampling a concept sequence is at least $\kappa > 0$*

With this assumption, we formally present the main result of this paper as follows.

Theorem 3.6. *For a neuro-symbolic task $\mathcal{T}$ with a restricted hypothesis space $\mathcal{F}^*$, learnability is determined by the following conditions:*

- *If the derived constraint satisfaction problem has a unique solution, the task is* learnable. *Specifically, the concept error is bounded by ϵ, provided that the sample size N satisfies:*

$$N > \frac{1}{\kappa} \cdot \log\left(|\mathcal{B}|/\epsilon\right).$$

- *Otherwise, the task is* unlearnable.

The proof is in section B.4 . Theorem 3.6 establish that a NeSy task $\mathcal{T}$ is *learnable* if and only if the DCSP solution is unique, i.e., disagreement $d = 0$. Conversely, if the DCSP has multiple solutions (i.e., $d \geq 1$), the task is *unlearnable*, implying that concept error remains *unavoidable* regardless of additional training data.

Building upon the concept of DCSP solution disagreement, we derive a more general theorem offering deeper insights into learning errors in a restricted hypothesis space $\mathcal{F}^*$. As the sample size approaches infinity, the hypotheses learned via ERM asymptotically converge to: $\mathcal{F}^*_{\text{ERM}} = \arg\min_{f \in \mathcal{F}^*} R_{\text{NeSy}}(f)$. The average error of the ERM result, denoted by $\mathcal{E}^*$, is the expected concept risk of an arbitrarily selected hypothesis: $\mathcal{E}^* = \mathbb{E}_{f \in \mathcal{F}^*_{\text{ERM}}}\left[R_{0/1}(f)\right]$.

Theorem 3.7. *The average error $\mathcal{E}^*$ is bounded by:*

$$\mathcal{E}^* \leq \frac{d}{L}.$$

The proof is in section B.5. Theorem 3.7 provides an asymptotic error analysis for NeSy tasks, indicating that as the DCSP solution disagreement d increases, the upper bound of the concept error also increases. Revealing that the disagreement d is crucial to the learnability of NeSy tasks.

3.4 Examples

Here, we present examples to better understand the learnability conditions of a NeSy task. To illustrate the distinction between *learnable* and *unlearnable* tasks, we use digital images as input data. We model the data as $\boldsymbol{x} = (x_1, x_2) \in \mathcal{X}^2$, where $\mathcal{X}$ represents the space of digit images (e.g., {  , , ...}). The intermediate concept space $\mathcal{Z}$ and the label space $\mathcal{Y}$ depend on the specific knowledge base. Table 1 summarizes these examples,where in modular addition task $2 \leq k \leq 10$.

Table 1: Examples of (un)learnable tasks.

Learnable	Addition	$y = z_1 + z_2$
	Multiplication	$y = z_1 \times z_2$
Unlearnable	Exclusive OR	$y = z_1 \oplus z_2$
	Modular Addition	$y = (z_1 + z_2) \bmod k$

For the XOR task ($d/L = 1$), interchanging the concepts 0 and 1, i.e., $\mapsto$ 0, $\mapsto$ 1 and vice versa, minimizes the NeSy risk. For the modular addition task ($k = 9, d/L = 0.2$), swapping the mappings of 0 and 9, i.e., $\mapsto$ 0, $\mapsto$ 9 and vice versa, minimizes the NeSy risk.

Remark 1. A direct implication from theorem 3.7 is that the expected concept error does not increase monotonically with the number of DCSP solutions but is instead with the disagreement d. For example, in the modular addition task: (i) When $k = 8$, there are 4 solutions, with $d/L = 0.4$; However, (ii) when $k = 10$, there are only 2 solutions, yet $d/L = 1$. This occurs because the significant disagreement between the two solutions leads to an unbounded worst-case concept error.

3.5 Aggregation of Unlearnable Tasks

Certain NeSy tasks are inherently unlearnable because they admit multiple solutions to their DCSPs, resulting in ambiguity. This ambiguity cannot be resolved by increasing data or improving the learning algorithm, as it stems from intrinsic task properties. Interestingly, however, such unlearnable tasks may become learnable when combined under a multi-task learning paradigm. The key insight is that tasks can mutually constrain each other, reducing ambiguity.

Consider two unlearnable tasks, $\mathcal{T}_1$ and $\mathcal{T}_2$, with their solution spaces $\mathcal{S}_1$ and $\mathcal{S}_2$, where $|\mathcal{S}_1| \geq 2$ and $|\mathcal{S}_2| \geq 2$. In a multi-task learning setting, the combined task requires satisfying constraints from both tasks at the same time, creating the solution set $\mathcal{S}_{\text{agg}} = \mathcal{S}_1 \cap \mathcal{S}_2$. For the combined task to become learnable, two key conditions must hold: (1) *concept space overlap*: $\mathcal{Z}_1 \cap \mathcal{Z}_2 \neq \varnothing$; and (2) *reduced DCSP disagreement*: $d_{\text{agg}} = |\mathcal{Z}_1 \cup \mathcal{Z}_2| - |\text{Union}(\mathcal{S}_{\text{agg}})| < \min(d_1, d_2)$. This reduction of the solution space reduces ambiguity and may lead to a unique solution, making the combined task learnable. Therefore, by using the mutual constraints from overlapping solution spaces, combining unlearnable tasks in an aggregation framework can enable learnability. From the perspective of DCSP, we can formally state the corollary as follows:

Corollary 3.8. *NeSy tasks become learnable in an aggregation framework if combining their DCSPs results in a unique solution.*

4 Empirical Study

To empirically validate the theoretical results, we conducted a series of experiments, including arithmetic tasks shown in table 1 and BDD-OIA Xu et al. (2020), which is evaluated in Marconato et al. (2023b) as a realistic application. Due to space limitations, some experimental results are presented in the appendix.

Setup Manhaeve et al. (2018) proposed the digit addition task by incorporating the handwritten MNIST (LeCun et al., 1994) and predefined addition rules. We extend the setup by including KMNIST (Clanuwat et al., 2018), CIFAR10 Krizhevsky (2009), and SVHN (Netzer et al., 2011), mapping class indices to digits, e.g., CIFAR-10 classes (`airplane` $= 0, \ldots$) , and enriching the background knowledge as depicted in table 1. The learning model for MNIST and KMNIST is LeNet (LeCun & Bengio, 1998), while ResNet50 (He et al., 2016) is used for CIFAR10 and SVHN. Besides that, we also adopt BDD-OIA from Bortolotti et al. (2024), which is a multi-label autonomous driving task for studying RSs in real-world, high-stakes scenarios. All experiments were conducted five times with different random seeds. Details can be seen in section C.

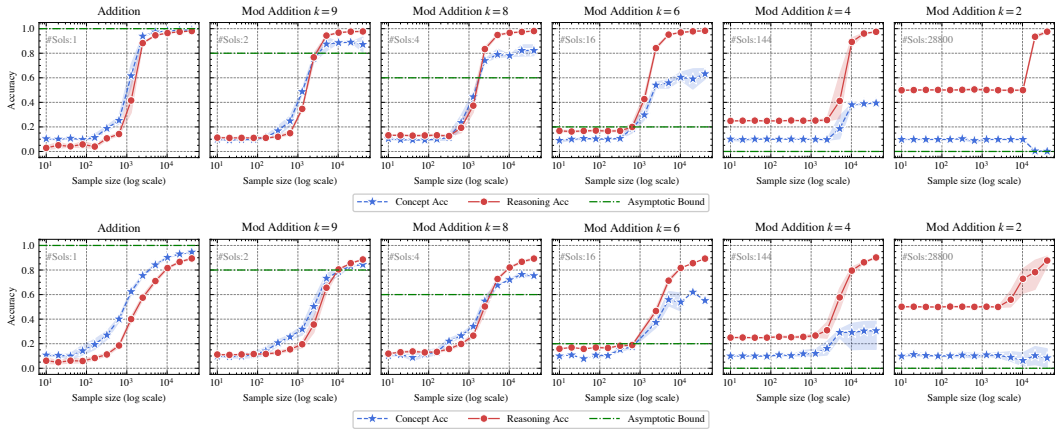


Figure 1: *Accuracies* versus *sample size* for different NeSy tasks (top MNIST and bottom KMNIST). The shadowed area denotes the standard error. The number of the DCSP solutions (#Sols) is shown at the top left of each plot. The asymptotic bound (green line) from theorem 3.7 indicates that concept accuracy should exceed this bound as the sample size grows.

Method To effectively optimize the NeSy risk (2), we adopt the following surrogate (cf. proof in section B.1):

$$-\mathbb{E}_{(\boldsymbol{x},y)} \log \left(\sum_{\bar{\boldsymbol{z}} \in \mathcal{N}(y)} p\left[y, \bar{\boldsymbol{z}} \mid \boldsymbol{x}; f, \mathtt{KB}\right] \right) , \tag{7}$$

which is flexible, where $\mathcal{N}(y) \subseteq A(y)$ represents several valid candidates for the final answer y. By restricting the size of $\mathcal{N}(y)$ from the entire set $A(y)$ to the most likely candidate $\bar{\boldsymbol{z}}$, we achieve a balance between PNL and ABL, and we set the size of $\mathcal{N}(y)$ is $\min(16, |A(y)|)$. The implementation is based on the code of He et al. (2024b). For brevity, detailed experiments on PNL and ABL are provided in section C.

4.1 Empirical Analysis on Learnability

We empirically evaluate the learnability of NeSy tasks based on theorem 3.6, focusing on two key aspects: (i) validating that minimizing the NeSy risk consistently minimizes the concept risk for learnable tasks, and (ii) examining how DCSP solution disagreement affects learnability.

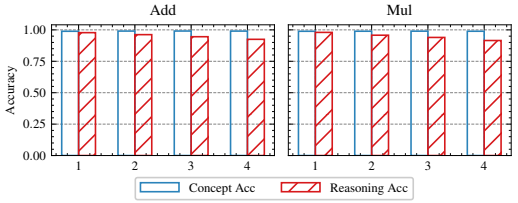


Figure 2: Accuracies on the learnable tasks.

(i) Validation of learnable tasks. We first validate the learnability conditions (cf. theorem 3.6) by examining addition and multiplication tasks (cf. table 1). Solving the DCSP shows that both tasks are learnable, and their learnability remains unaffected by increases in digit size (e.g., from $\mathtt{PROD}(\boxed{1}, \boxed{1}) = 2$ to $\mathtt{PROD}(\boxed{10}, \boxed{10}) = 200$). The raw dataset in figure 2 is MNIST, and additional results for other datasets are in the appendix. We further substantiate learnability by examining tasks with varying digit sizes, ranging from one to four digits. As depicted in figure 2, the results confirm that: (a) Optimization of the surrogate risk (7) effectively minimizes the NeSy risk. (b) For learnable tasks, a good minimizer of the NeSy risk also serves as a reliable minimizer of the concept risk.

(ii) Impact of DCSP solution disagreement. We further investigate how disagreement in DCSP solutions impacts learnability. According to theorem 3.7, the asymptotic error is bounded by the ratio of DCSP solution disagreement d to the size of the concept space L. Experiments involving addition and modular addition tasks with varying modular bases k reveal that altering the knowledge base changes the DCSP solution space, directly influencing learnability. For clarity, we plot the asymptotic accuracy bound for each task, i.e., $1 - d/L$, showing that higher disagreement results in a lower bound line (green). As shown in figure 1: (a) Tasks with a unique DCSP solution are learnable; (b) Tasks with high DCSP disagreement struggle to achieve low concept risk, even as the sample size increases.

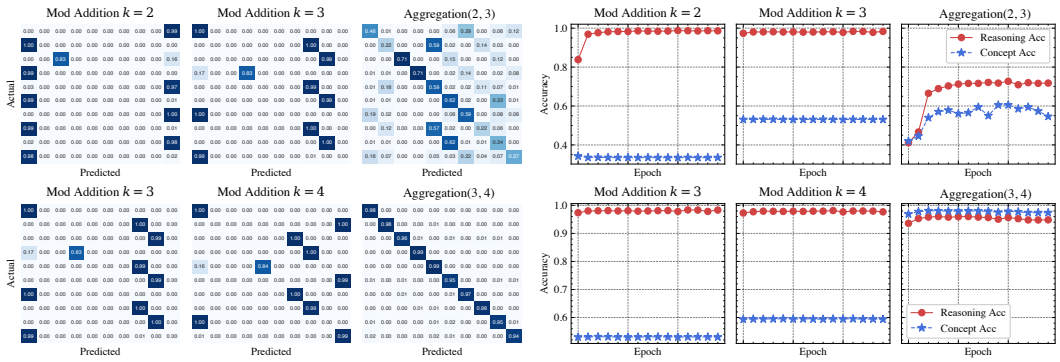


Figure 3: *Aggregation* of unlearnable NeSy tasks. The left shows confusion matrices and the right displays accuracy curves. (a) The top row illustrates an *unlearnable* case, where combining the tasks still results in multiple DCSP solutions. (b) The bottom row illustrates a *learnable* case, where combining the tasks reduces the DCSP solutions to a single one.

In summary, our empirical analysis confirms the theoretical learnability conditions by demonstrating that minimizing the NeSy risk reliably minimizes the concept risk for learnable tasks. Furthermore, tasks with lower disagreement exhibit better learnability, while those with high disagreement suffer from ambiguity due to multiple conflicting solutions.

4.2 Further Evaluation on a Realistic Task: BDD-OIA

We further extend our experiments to the realistic BDD-OIA task, a multi-label autonomous driving benchmark in real-world, high-stakes scenarios. The knowledge base KB encodes, for example, that it is unsafe to move `forward` when pedestrians are present, using a set of 21 binary concepts indicating various obstacles on the road. The constraints specify conditions for proceeding ($\texttt{green\_light} \vee \texttt{follow} \vee \texttt{clear} \Rightarrow \texttt{forward}$), stopping ($\texttt{red\_light} \vee \texttt{stop\_sign} \vee \texttt{obstacle} \Rightarrow \texttt{stop}$), turning left and right, and relationships between actions (e.g., $\texttt{stop} \Rightarrow \neg\texttt{forward}$). In this task, there are 74,240 DCSP solutions, and the disagreement among these solutions is 15, indicating a typically unlearnable problem under our criterion. We evaluate several methods on this task: LTN and DeepProbLog are implemented using the `rsbench` codebase (Bortolotti et al., 2024), whereas ABL and A^{3}BL use their official implementations.

Table 2: Accuracies and standard deviations on the BDD-OIA task across five seeds.

Method	Reasoning Acc	Concept Acc
LTN (Badreddine et al., 2022)	$27.22_{\pm 5.52}$	$20.41_{\pm 8.05}$
DeepProbLog (Manhaeve et al., 2021a)	$44.69_{\pm 0.19}$	$0.00_{\pm 0.00}$
ABL (Dai et al., 2019)	$75.16_{\pm 0.12}$	$66.92_{\pm 5.86}$
A^{3}BL (He et al., 2024b)	$60.91_{\pm 0.01}$	$88.37_{\pm 0.00}$

4.3 Aggregation of Unlearnable NeSy Tasks

With the DCSP framework, we find that certain NeSy tasks are inherently unlearnable because their DCSPs admit multiple solutions, resulting in inherent ambiguity. However, when combined in an aggregation framework within a multi-task learning setting, such tasks may become learnable by enforcing mutual consistency, as shown in corollary 3.8. We evaluate corollary 3.8 using mod addition tasks with mod bases k_1 and k_2 under two specific configurations: an unlearnable aggregation ($k_1 = 2, k_2 = 3$) and a learnable aggregation ($k_1 = 3, k_2 = 4$). For $k = 2, 3, 4$, the degree of DCSP solution disagreement d is 10. The experiments in figure 3 are based on the raw MNIST dataset. Additional details and experiments are provided in section C.2.3.

In the top of figure 3, the unlearnable case ($k_1 = 2, k_2 = 3$) shows that while the aggregation narrows the solution space, reducing the disagreement d to 8, it does not converge to a unique solution, and the task remains unlearnable. In the bottom of figure 3, the learnable case ($k_1 = 3, k_2 = 4$) illustrates that both tasks initially admit multiple DCSP solutions, causing reasoning accuracy to exceed concept accuracy, as shown in figure 3. Through the aggregation, the intersection of solution spaces shrinks, with the disagreement d reduced to 0, making the aggregation task learnable.

This experimental result supports corollary 3.8, demonstrating that forming aggregations of different NeSy tasks can enhance learnability by mutually constraining DCSP solution spaces. This finding suggests that collecting tremendous NeSy tasks and jointly learning them in an aggregation manner could improve the learnability and potentially introduce a "scaling law" (Kaplan et al., 2020) in the NeSy domain.

5 Limitations and Future Directions

This paper focuses exclusively on hybrid neuro-symbolic systems, e.g., probabilistic neuro-symbolic and abductive learning methods. Thus the findings may not directly extend to other types of neuro-symbolic methods. The analysis of this study relies on a restricted hypothesis space, which is inherently satisfied by models such as neural networks equipped with manifold regularization (Belkin et al., 2006) or self-supervised pretraining (Liu et al., 2021). However, extending the framework to encompass more general hypothesis spaces without requiring this specific property remains an open challenge.

Future work may involve a deeper investigation into extending the learnability framework to encompass a broader range of NeSy systems. Additionally, exploring the learnability of the semi-supervised case of NeSy tasks, where some training examples are supervised for intermediate concepts, could be an interesting direction. Developing practical strategies for constructing effective task aggregations aslo represents a promising avenue for improving learnability in many scenarios.

Moreover, because solving CSPs is NP-hard in general, the DCSP framework may face computational scalability challenges for large-scale knowledge bases. Future directions include approximation methods such as sampling-based estimation (e.g., uniform sampling Heradio et al. 2020), incorporating practical heuristics (e.g., exploiting symmetries in the knowledge base), and decomposing complex knowledge bases into simpler sub-tasks with tractable CSPs (Hu et al., 2025b).

6 Conclusion

We establish that a neuro-symbolic task is learnable if and only if the derived constraint satisfaction problem (DCSP) has a unique solution. This conclusion is consistent with previously found reasoning shortcuts problem. With the DCSP framework, we can conduct a comprehensive analysis on *sample complexity* and *concept error* based on the disagreement d among these solutions. This framework also implies that forming aggregations of unlearnable tasks reduces the disagreement d, thereby enhancing overall task learnability.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (62076121) and the Major Program (JD) of Hubei Province (2023BAA024). Prof. Ming Li is the corresponding author. The authors thank Lue Tao for helpful feedback on earlier drafts of the paper and the anonymous reviewers for their valuable comments.

References

Agiollo, A., Rafanelli, A., Magnini, M., Ciatto, G., and Omicini, A. Symbolic Knowledge Injection Meets Intelligent Agents: QoS Metrics and Experiments. *Autonomous Agents and Multi-Agent Systems*, 37(2), June 2023.

Alon, N., Ben-David, S., Cesa-Bianchi, N., and Haussler, D. Scale-sensitive Dimensions, Uniform Convergence, and Learnability. *Journal of the ACM*, 44(4):615–631, July 1997.

Badreddine, S., d'Avila Garcez, A., Serafini, L., and Spranger, M. Logic Tensor Networks. *Artificial Intelligence*, 303(C), 2022.

Bartlett, P. L. and Maass, W. Vapnik-Chervonenkis Dimension of Neural Nets. *The Handbook of Brain Theory and Neural Networks*, pp. 1188–1192, 2003.

Belkin, M., Niyogi, P., and Sindhwani, V. Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. *Journal of Machine Learning Research*, 7(11), 2006.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. Learnability and The Vapnik-Chervonenkis Dimension. *Journal of the ACM*, 36(4):929–965, October 1989.

Bortolotti, S., Marconato, E., Carraro, T., Morettin, P., van Krieken, E., Vergari, A., Teso, S., and Passerini, A. A Neuro-Symbolic Benchmark Suite for Concept Quality and Reasoning Shortcuts. In *Advances in Neural Information Processing Systems 37*, pp. 115861–115905, 2024. Datasets and Benchmarks Track.

Cai, L.-W., Dai, W.-Z., Huang, Y.-X., Li, Y.-F., Muggleton, S., and Jiang, Y. Abductive Learning with Ground Knowledge Base. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pp. 1815–1821, Virtual, 2021.

Ciatto, G., Sabbatini, F., Agiollo, A., Magnini, M., and Omicini, A. Symbolic Knowledge Extraction and Injection with Sub-symbolic Predictors: A Systematic Literature Review. *ACM Computing Surveys*, 56(6), March 2024.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. Deep Learning for Classical Japanese Literature. *CoRR*, abs/1812.01718, 2018.

Dai, W.-Z., Xu, Q., Yu, Y., and Zhou, Z.-H. Bridging Machine Learning and Logical Reasoning by Abductive Learning. In *Advances in Neural Information Processing Systems 32*, pp. 2815–2826, Vancouver, BC, Canada, 2019.

De Raedt, L., Kimmig, A., and Toivonen, H. ProbLog: A Probabilistic Prolog and Its Application in Link Discovery. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 2468–2473, Hyderabad, India, 2007.

Dimopoulos, Y., Nebel, B., and Koehler, J. Encoding Planning Problems in Nonmonotonic Logic Programs. In *Proceedings of the 4th European Conference on Planning*, pp. 169–181, Toulouse, France, 1997.

Gao, E.-H., Huang, Y.-X., Hu, W.-C., Zhu, X.-H., and and, W.-Z. D. Knowledge-enhanced Historical Document Segmentation and Recognition. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pp. 8409 – 8416, Vancouver, BC, Canada, 2024.

Garcez, A. S. d., Gabbay, D. M., and Broda, K. B. *Neural-Symbolic Learning System: Foundations and Applications*. Springer-Verlag, Berlin, Heidelberg, 2002.

He, H.-Y., Dai, W.-Z., and Li, M. Reduced Implication-bias Logic Loss for Neuro-symbolic Learning. *Machine Learning*, 113:3357–3377, 2024a.

He, H.-Y., Sun, H., Xie, Z., and Li, M. Ambiguity-aware Abductive Learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 18019–18042, Vienna, Austria, 2024b.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, 2016.

Heradio, R., Fernandez-Amoros, D., Galindo, J. A., and Benavides, D. Uniform and Scalable SAT-sampling for Configurable Systems. In *Proceedings of the 24th ACM Conference on Systems and Software Product Line*, SPLC '20, New York, NY, USA, 2020. Association for Computing Machinery.

Hitzler, P. and Sarker, M. K. (eds.). *Neuro-Symbolic Artificial Intelligence: The State of the Art*. IOS Press, Amsterdam, 2022.

Hu, W.-C., Dai, W.-Z., Jiang, Y., and Zhou, Z.-H. Efficient Rectification of Neuro-Symbolic Reasoning Inconsistencies by Abductive Reflection. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, Philadelphia, USA, 2025a. 17333–17341.

Hu, W.-C., Li, Q.-J., Jia, L.-H., Ge, C., Li, Y.-F., Jiang, Y., and Zhou, Z.-H. Curriculum Abductive Learning. In *Advances in Neural Information Processing Systems, 38*, pp. To appear, San Diego, CA, USA, 2025b.

Huang, Y.-X., Dai, W.-Z., Cai, L.-W., Muggleton, S. H., and Jiang, Y. Fast Abductive Learning by Similarity-based Consistency Optimization. In *Advances in Neural Information Processing Systems 34*, pp. 26574–26584, Virtual, 2021.

Huang, Y.-X., Hu, W.-C., Gao, E.-H., and Jiang, Y. ABLkit: A Python Toolkit for Abductive Learning. *Frontiers of Computer Science*, 18(6):186354, 2024.

Jiao, Y., De Raedt, L., and Marra, G. Valid Text-to-SQL Generation with Unification-Based DeepStochLog. In *Neural-Symbolic Learning and Reasoning: 18th International Conference, NeSy*, pp. 312–330, Berlin, Heidelberg, 2024. Springer-Verlag.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling Laws for Neural Language Models, 2020. URL https://arxiv.org/abs/2001.08361.

Karacali, B. and Krim, H. Fast Minimization of Structural Risk by Nearest Neighbor Rule. *IEEE Transactions on Neural Networks*, 14(1):127–137, 2003.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. In *The 2nd International Conference on Learning Representations*, San Diego, CA, USA, 2015.

Krizhevsky, A. Learning Multiple Layers of Features from Tiny Images. *Technical Report, Department of Computer Science, University of Toronto*, 2009.

LeCun, Y. and Bengio, Y. Convolutional Networks for Images, Speech, and Time Series. In *The Handbook of Brain Theory and Neural Networks*, pp. 255–258. MIT Press, Cambridge, MA, USA, 1998.

LeCun, Y., Cortes, C., and Burges, C. J. C. The MNIST Database of Handwritten Digits, 1994. URL http://yann.lecun.com/exdb/mnist/.

Li, Z., Huang, J., and Naik, M. Scallop: A Language for Neurosymbolic Programming. *Proceedings of the ACM on Programming Languages*, 7(PLDI), June 2023.

Li, Z., Huang, Y., Li, Z., Yao, Y., Xu, J., Chen, T., Ma, X., and Lü, J. Neuro-Symbolic Learning Yielding Logical Constraints. In *Advances in Neural Information Processing Systems, 38*, pp. 21635 – 21657, New Orleans, LA, USA, 2024.

Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., and Tang, J. Self-Supervised Learning: Generative or Contrastive. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):857–876, 2021.

Maene, J., Derkinderen, V., and De Raedt, L. On the Hardness of Probabilistic Neurosymbolic Learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34203–34218, Vienna, Austria, 2024.

Magnini, M., Ciatto, G., and Omicini, A. On the Design of PSyKI: A Platform for Symbolic Knowledge Injection into Sub-symbolic Predictors. In *Explainable and Transparent AI and Multi-Agent Systems: 4th International Workshop, EXTRAAMAS 2022*, pp. 90–108, Berlin, Heidelberg, 2022.

Manginas, N., Paliouras, G., and Raedt, L. D. NeSyA: Neurosymbolic Automata. In *Proceedings of the 34th International Joint Conference on Artificial Intelligence*, pp. 5950–5958, Montreal, Canada, 2025.

Manhaeve, R., Dumančić, S., Kimmig, A., Demeester, T., and De Raedt, L. DeepProbLog: Neural Probabilistic Logic Programming. In *Advances in Neural Information Processing Systems 31*, pp. 3753–3763, Montréal, Canada, 2018.

Manhaeve, R., Dumančić, S., Kimmig, A., Demeester, T., and De Raedt, L. Neural Probabilistic Logic Programming in DeepProbLog. *Artificial Intelligence*, 298(C):103504, 2021a.

Manhaeve, R., Marra, G., Demeester, T., Dumancic, S., Kimmig, A., and De Raedt, L. Neuro-Symbolic AI = Neural + Logical + Probabilistic AI. In *Neuro-Symbolic Artificial Intelligence: The State of the Art*, volume 342 of *Frontiers in Artificial Intelligence and Applications*, pp. 173–191. IOS Press, 2021b.

Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B., and Wu, J. The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision. In *The 7th International Conference on Learning Representations*, New Orleans, LA, USA, 2019.

Marconato, E., Bontempo, G., Ficarra, E., Calderara, S., Passerini, A., and Teso, S. Neuro-Symbolic Continual Learning: Knowledge, Reasoning Shortcuts and Concept Rehearsal. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 23915–23936. JMLR.org, 2023a.

Marconato, E., Teso, S., Vergari, A., and Passerini, A. Not All Neuro-Symbolic Concepts Are Created Equal: Analysis and Mitigation of Reasoning Shortcuts. In *Advances in Neural Information Processing Systems*, volume 36, pp. 72507–72539, 2023b.

Marconato, E., Bortolotti, S., van Krieken, E., Vergari, A., Passerini, A., and Teso, S. BEARS Make Neuro-Symbolic Models Aware of their Reasoning Shortcuts. In *Proceedings of the 40th Conference on Uncertainty in Artificial Intelligence*, volume 244, pp. 2399–2433. PMLR, 2024.

Marra, G., Dumančić, S., Manhaeve, R., and De Raedt, L. From Statistical Relational to Neuro-symbolic Artificial Intelligence: A survey. *Artificial Intelligence*, 328:104062, 2024.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading Digits in Natural Images with Unsupervised Feature Learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, Granada, Spain, 2011.

Prud'homme, C. and Fages, J.-G. Choco-solver: A Java Library for Constraint Programming. *Journal of Open Source Software*, 7(78):4708, 2022.

Roychowdhury, S., Diligenti, M., and Gori, M. Regularizing Deep Networks with Prior Knowledge: A Constraint-based Approach. *Knowledge-Based Systems*, 222:106989, 2021.

Smet, L. D., Venturato, G., Raedt, L. D., and Marra, G. Relational Neurosymbolic Markov Models. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, Philadelphia, USA, 2025. 16181–16189.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C., Cubuk, E. D., Kurakin, A., and Li, C. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In *Advances in Neural Information Processing Systems 33*, pp. 596 – 608, Virtual Event, 2020.

Sun, R. *Integrating Rules and Connectionism for Robust Commonsense Reasoning*, pp. 273. John Wiley & Sons, Inc., 1994.

Tao, L., Huang, Y.-X., Dai, W.-Z., and Jiang, Y. Deciphering Raw Data in Neuro-Symbolic Learning with Provable Guarantees. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pp. 15310 – 15318, Vancouver, BC, Canada, 2024.

Towell, G. G. and Shavlik, J. W. Knowledge-based Artificial Neural Networks. *Artificial Intelligence*, 70(1):119–165, 1994.

Valiant, L. G. A Theory of The Learnable. *Communications of the ACM*, 27(11):1134–1142, November 1984.

van Krieken, E., Acar, E., and van Harmelen, F. Analyzing Differentiable Fuzzy Logic Operators. *Artificial Intelligence*, 302:103602, 2022.

Vapnik, V. N. An Overview of Statistical Learning Theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

Verreet, V., De Raedt, L., and Bekker, J. Modeling PU Learning Using Probabilistic Logic Programming. *Machine Learning*, 113(3):1351–1372, 2023.

von Rueden, L., Mayer, S., Beckh, K., Georgiev, B., Giesselbach, S., Heese, R., Kirsch, B., Pfrommer, J., Pick, A., Ramamurthy, R., Walczak, M., Garcke, J., Bauckhage, C., and Schuecker, J. Informed Machine Learning – A Taxonomy and Survey of Integrating Prior Knowledge into Learning Systems. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):614–633, 2023.

Wang, J., Deng, D., Xie, X., Shu, X., Huang, Y.-X., Cai, L.-W., Zhang, H., Zhang, M.-L., Zhou, Z.-H., and Wu, Y. Tac-Valuer: Knowledge-based Stroke Evaluation in Table Tennis. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 3688–3696, Virtual Event, Singapore, 2021.

Wang, K., Tsamoura, E., and Roth, D. On Learning Latent Models with Multi-instance Weak Supervision. In *Advances in Neural Information Processing Systems 36*, pp. 9661 – 9694, New Orleans, LA, USA, 2023.

Xu, J., Zhang, Z., Friedman, T., Liang, Y., and Van den Broeck, G. A Semantic Loss Function for Deep Learning with Symbolic Knowledge. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 5502–5511, Stockholm, Sweden, 2018.

Xu, Y., Yang, X., Gong, L., Lin, H.-C., Wu, T.-Y., Li, Y., and Vasconcelos, N. Explainable Object-Induced Action Decision for Autonomous Vehicles. In *Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9520–9529, Vritual, 2020.

Yang, X.-W., Wei, W.-D., Shao, J.-J., Li, Y.-F., and Zhou, Z.-H. Analysis for Abductive Learning and Neural-Symbolic Reasoning Shortcuts. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 56524–56541, Vienna, Austria, 2024.

Yang, Z., Ishay, A., and Lee, J. NeurASP: Embracing Neural Networks into Answer Set Programming. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, pp. 1755–1762, Yokohama, Japan, 2020.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding Deep Learning (Still) Requires Rethinking Generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Zhou, Z.-H. Abductive Learning: Towards Bridging Machine Learning and Logical Reasoning. *Science China Information Sciences*, 62(7):76101, 2019.

Appendix

The appendix is structured as follows.

- Section A contains discussion about reasoning shortcuts and previous theoretical works.
- Section B contains proofs omitted in the main paper, because of the space limit.
- Section C contains more details and additional experiments.

A Extended Related Works

To contextualize and highlight the distinct contributions of this work, we undertake a comprehensive and critical analysis of several closely related studies in neuro-symbolic learning.

A.1 Neuro-Symbolic Methods

The combination of learning and reasoning remains the holy grail problem of AI for decades now (Towell & Shavlik, 1994; Sun, 1994; Garcez et al., 2002). One promising approach is to directly incorporate logical constraints into the loss function as the optimization objective (Xu et al., 2018; Roychowdhury et al., 2021; Badreddine et al., 2022). However, since the logical constraint is discrete, the optimization must be projected into the continuous space. This requires an approximation of logical reasoning. Such an approach can lead to issues when approximating discrete logical computations (van Krieken et al., 2022; He et al., 2024a). It is also worth noting that there is a line of research on knowledge extraction and injection for perception models (Ciatto et al., 2024; von Rueden et al., 2023; Agiollo et al., 2023; Magnini et al., 2022).

A more effective approach is the hybrid system, where both the learning and reasoning models function at their full capacity. For instance, DeepProbLog (Manhaeve et al., 2018, 2021a), NeurASP (Yang et al., 2020), and Scallop (Li et al., 2023) employ probabilistic logic programming as their reasoning model. ABL (Zhou, 2019; Dai et al., 2019) employs abductive reasoning for logical inference. Recently, there have been some studies on NeSy with auto-regressive or temporal models (Manginas et al., 2025; Smet et al., 2025). Building on the hybrid approach, there have been many successful applications (Mao et al., 2019; Wang et al., 2021; Cai et al., 2021; Verreet et al., 2023; Gao et al., 2024; Jiao et al., 2024). Since the hybrid approach has shown its superiority, it is worthwhile to establish a theoretical framework for analyzing its learnability.

A.2 Reasoning Shortcuts Works

Marconato et al. (2023b) gave a formal definition of reasoning shortcut (RS):

> *A reasoning shortcut is a distribution* $p_\theta(\mathbf{C} \mid \mathbf{X})$ *that achieves maximal log-likelihood on the training set but does not match the ground truth concept distribution,*
>
> $$\mathcal{L}(p_\theta, \mathcal{D}, \mathsf{K}) = \max_{\theta' \in \Theta} \mathcal{L}(p_{\theta'}, \mathcal{D}, \mathsf{K}) \quad \wedge \quad p_\theta(\mathbf{C} \mid \mathbf{X}) \not\equiv p^*(\mathbf{G} \mid \mathbf{X})\,.$$

They further proposed several approaches to mitigate this issue. Furthermore, Marconato et al. (2023a) propose a new paradigm that combines continual learning with neuro-symbolic learning by structuring multiple NeSy tasks as a sequential process, which may help mitigate the RS problem. Subsequently, Marconato et al. (2024) use entropy regularization to address the RS problem. Bortolotti et al. (2024) construct a benchmark to evaluate this problem. Although the RS problem has been studied, an in-depth theoretical explanation and analysis is still lacking.

Our analysis provides new insights into the RS problem. From the perspective of DCSP, we can gain a deeper understanding of the RS problem. The presence of multiple DCSP solutions directly corresponds to the existence of deterministic RSs, as defined in Marconato et al. (2023b). Instead of simply counting solutions, we argue that disagreement d offers a more nuanced characterization of concept error. Moreover, a key implication of theorem 3.7 is that concept error does not increase monotonically with the number of DCSP solutions; instead, it depends on their level of disagreement (cf. remark 1).

A.3 Theoretical Works

Several works have explored theoretical insights into neuro-symbolic learning. Below, we provide a discussion of these efforts and contrast them with our contributions.

Wang et al. (2023) introduces multi-instance partial label learning (MI-PLL) and proposes an "M-unambiguity" condition for learnability. They define this condition as follows:

> *A transition σ is M-unambiguous if, for any two diagonal label vectors $\boldsymbol{z}$ and $\boldsymbol{z}' \in \mathcal{Z}^M$ such that $\boldsymbol{z} \neq \boldsymbol{z}'$, we have $\sigma(\boldsymbol{z}) \neq \sigma(\boldsymbol{z}')$.*

Here, we slightly modify the notation to ensure consistency with this paper. A diagonal label vector consists of identical elements in every position. When $M = 1$, the concept label can be directly inferred from the transition function σ, reducing the problem to a standard supervised learning setting with well-established learnability guarantees. Building upon this condition, they derive a concept error bound based on the VC-dimension.

Compare with Wang et al. (2023), the advantages of our work are:

a) Their learnability condition assumes repetitive input patterns (e.g., $[z, z, \dots]$) to satisfy symbolic constraints, which is unrealistic in practical applications. For example, in autonomous driving scenarios, neuro-symbolic systems usually process diverse predicates and facts, contradicting this assumption. In contrast, our framework provides a more general learnability analysis that is both easier to verify using modern constraint satisfaction problem (CSP) solvers and applicable to both probabilistic neuro-symbolic learning and abductive learning.

b) Even in unlearnable cases, our approach establishes a lower bound for concept error by analyzing the disagreement among DCSP solutions, offering insights into these unlearnable scenarios.

Tao et al. (2024) examine scenarios where randomly selecting abduction candidates results in a consistent optimization objective within the abductive learning (ABL) framework. They formulate the learning process as a weakly supervised learning problem and analyze its consistency through a probabilistic matrix $\tilde{Q} \in \mathbb{R}^{c \times (m \cdot |\mathcal{Z}|)}$ (cf. Section 3 in Tao et al. (2024)).

> *Let $t \in \{0, \dots, |\mathcal{Y}| - 1\}$ and $o = tm + k$, $\tilde{Q}_{jo}$ represents the probability of the class j occuring at the k-th position in a sequence of final label t.*

If $\tilde{Q}$ is full-rank, the concept-level supervision can be fully recovered, ensuring that the task is learnable under their ABL formulation. However, construction of such a matrix $\tilde{Q}$ requires full knowledge of the backhind concept sequence distribution, which is not easy to derive.

Yang et al. (2024) introduce a shortcut risk metric R_s to quantify the discrepancy between true risk and surrogate risk:

$$R_s := -\mathbb{E}_{\boldsymbol{x},\boldsymbol{z}} \log p_\theta(\boldsymbol{z} \mid \boldsymbol{x}) + \frac{1}{N} \sum_{(\boldsymbol{x},y) \in D} \log \left(\sum_{\boldsymbol{z} \in \mathcal{Z}} \mathbb{I}[\boldsymbol{z} \wedge \mathtt{KB} \models y] \cdot p_\theta(\boldsymbol{z} \mid \boldsymbol{x}) \right).$$

They further establish an upper bound on R_s based on the complexity of the knowledge base:

$$\mathbb{E}_{\boldsymbol{x},y}[R_s] \leq \frac{1}{2} \log (C - D_{\mathtt{KB}}) + \gamma,$$

where $D_{\mathtt{KB}} := \mathbb{E}_{\boldsymbol{x},y} \left[\sum_{\boldsymbol{z} \in \mathcal{Z}} \mathbb{I}(\boldsymbol{z} \wedge \mathtt{KB} \not\models y) \right]$ measures knowledge base complexity, C is the size of concept sequence space, and γ is a constant dependent on the hypothesis space. However, a low R_s does not necessarily imply a low concept error, meaning their results do not directly establish the learnability of NeSy tasks. Furthermore, their complexity measure does not account for dependencies between concept sequences. Consequently, the derived bound is loose and they classify the addition task as "unbounded" (cf. Definition 4.1 in Yang et al. (2024)). In contrast, our analysis and experiments demonstrate that it is learnable under a restricted hypothesis space.

B Proofs

In this section, we give the proofs that are omitted in the main text. For the convenience of the reader, we re-state the assumptions, lemmas, propositions, and theorems in the appendix again.

B.1 Proof of Theorem 2.1

Theorem 2.1. *A minimizer of R_{PNL} or R_{ABL} is also a minimizer of R_{NeSy}. For each surrogate $R_s \in \{R_{PNL}, R_{ABL}\}$, we have:*

$$\arg\min_{f\in\mathcal{F}} R_s(f) \subseteq \arg\min_{f\in\mathcal{F}} R_{NeSy}(f),$$

Proof. First, we recall the risks as follows:

$$R_{\text{PNL}}(f) = -\mathbb{E}_{(\boldsymbol{x},y)} \log \sum_{\boldsymbol{z}} \mathbb{I}(\boldsymbol{z} \wedge \texttt{KB} \models y) \cdot p\left[\boldsymbol{z} \mid \boldsymbol{x}; f, \texttt{KB}\right] ,$$
$$R_{\text{ABL}}(f) = -\mathbb{E}_{(\boldsymbol{x},y)} \log \left(p\left[y, \bar{\boldsymbol{z}} \mid \boldsymbol{x}; f, \texttt{KB}\right]\right) .$$

To unify the proof, we introduce (7) as a surrogate form:

$$R_{\text{A}^3}(f) = -\mathbb{E}_{(\boldsymbol{x},y)} \log \left(\sum_{\bar{\boldsymbol{z}}\in\mathcal{N}(y)} p\left[y, \bar{\boldsymbol{z}} \mid \boldsymbol{x}; f, \texttt{KB}\right] \right) ,$$

where the set $\mathcal{N}(y)$ denotes the candidate set satisfying:

$$\forall \boldsymbol{z} \in \mathcal{N},\ \boldsymbol{z} \wedge \texttt{KB} \models y\,.$$

Furthermore, we reformulate the PNL risk using a union-based representation:

$$\begin{aligned} R_{\text{PNL}}(f) &= -\mathbb{E}_{(\boldsymbol{x},y)} \log \sum_{\boldsymbol{z}\in A(y)} \mathbb{I}(\boldsymbol{z} \wedge \texttt{KB} \models y) \cdot p\left[\boldsymbol{z} \mid \boldsymbol{x}; f, \texttt{KB}\right] \\ &= -\mathbb{E}_{(\boldsymbol{x},y)} \log \left(\sum_{\boldsymbol{z}\in A(y)} p\left[y, \boldsymbol{z} \mid \boldsymbol{x}; f, \texttt{KB}\right] \right) . \end{aligned}$$

Consequently, R_{A^3} emerges as a flexible surrogate. By adjusting the size of the candidate set, we can interpolate between the ABL risk and the PNL risk. Thus, it suffices to prove that for any candidate set $\mathcal{N}(y)$, R_{A^3} achieves the desired objective.

Since the following properties hold:

- For any $f \in \mathcal{F}$, the risk $R_{\text{A}^3}(f) \geq 0$.
- For the labeling function g, the risk $R_{\text{A}^3}(g) = 0$.

Therefore, the minimum achievable value of this risk is strictly 0. For any $f^* \in \arg\min_{f\in\mathcal{F}} R_{\text{A}^3}(f)$, we have $R_{\text{A}^3}(f^*) = 0$, which implies that, for a fixed candidate set $\mathcal{N}(y)$ and any $(\boldsymbol{x}, y)$:

$$\begin{aligned} &\sum_{\bar{\boldsymbol{z}}\in\mathcal{N}(y)} p\left[y, \bar{\boldsymbol{z}} \mid \boldsymbol{x}; f^*, \texttt{KB}\right] \\ =& \sum_{\bar{\boldsymbol{z}}\in\mathcal{N}(y)} \mathbb{I}(\bar{\boldsymbol{z}} \wedge \texttt{KB} \models y) \cdot p_{\theta^*}(\bar{\boldsymbol{z}} \mid \boldsymbol{x}) = 1\,. \end{aligned}$$

Consequently, any $\bar{\boldsymbol{z}}$ predicted by the learning model f^* with a probability greater than zero will satisfy the knowledge base, i.e., $\bar{\boldsymbol{z}} \wedge \texttt{KB} \models y$. This ensures that the NeSy risk $R_{\text{NeSy}}(f) = -\mathbb{E}_{(\boldsymbol{x},y)}\left[\mathbb{I}\left(f^*(\boldsymbol{x}) \wedge \texttt{KB} \not\models y\right)\right]$ is also zero. Since the NeSy risk should also be greater or equal to zero, which means the hypothesis f^* is a minimizer of NeSy risk. Hence, the proof is complete. □

B.2 Proof of Proposition 3.3

Proposition 3.3. *For an ambiguous NeSy task $\mathcal{T}$, if the hypothesis space $\mathcal{F}$ shatters the task, then there exists a hypothesis f^* that minimizes R_{NeSy} but does not minimize $R_{0/1}$.*

Proof. By the definitions of R_{NeSy} and $R_{0/1}$, we have:

$$R_{\text{NeSy}}(f) = \mathbb{E}_{(\boldsymbol{x},y)}\left[\mathbb{I}\left(f(\boldsymbol{x}) \wedge \texttt{KB} \not\models y\right)\right],$$

and, thus,

$$R_{0/1}(f) = \mathbb{E}_{x,z}\left[\mathbb{I}(f(x) \neq z)\right].$$

Since $\mathcal{T}$ is ambiguous, i.e., there exists a $y \in \mathcal{Y}$ such that $|A(y)| \geq 2$, we assume, without loss of generality, a sample pair $(\boldsymbol{x}_0, y_0) \in (\mathcal{X}^m, \mathcal{Y})$ such that $\{\boldsymbol{z}_1, \boldsymbol{z}_2\} \subseteq A(y_0)$.

Given that the hypothesis space $\mathcal{F}$ is sufficiently complex to shatter the task, we assume the existence of two hypotheses f_1 and f_2 that yield identical correct predictions for all inputs except $\boldsymbol{x}_0$:

$$\begin{cases} f_1(\boldsymbol{x}_0) = \boldsymbol{z}_1, \quad f_2(\boldsymbol{x}_0) = \boldsymbol{z}_2 & \text{if } \boldsymbol{x} = \boldsymbol{x}_0, \\ f_1(\boldsymbol{x}) = f_2(\boldsymbol{x}) & \text{otherwise.} \end{cases}$$

By definition, as f_1 and f_2 yield identical predictions except at $\boldsymbol{x}_0$, we have $R_{\text{NeSy}}(f_1) = R_{\text{NeSy}}(f_2)$. However, since $\boldsymbol{z}_1 \neq \boldsymbol{z}_2$, there exists at least one index $k \in [m]$ such that $(\boldsymbol{z}_1)_k \neq (\boldsymbol{z}_2)_k$. Thus, by the definition of $R_{0/1}$, we observe that $R_{0/1}(f_1) \neq R_{0/1}(f_2)$, as at the sample $(\boldsymbol{x}_0)_k$, they produce two distinct recognition results.

In this scenario, even if f_1 represents the underlying ground truth mapping function, it is indistinguishable from f_2, as both achieve zero risk under the optimized objective R_{NeSy}. This concludes that $(R_{\text{NeSy}} \to 0) \not\Rightarrow (R_{0/1} \to 0)$. □

B.3 Proof of Lemma 3.4

Lemma 3.4. *For a NeSy task $\mathcal{T}$, if the DCSP solution disagreement $d = 0$, then the NeSy risk is consistent with the concept risk. Formally, for any $f \in \mathcal{F}$:*

$$R_{NeSy}(f) \to 0 \iff R_{0/1}(f) \to 0.$$

Proof. We first prove the direction:

$$(R_{\text{NeSy}} \to 0) \Leftarrow (R_{0/1} \to 0)\ .$$

This is evident because, as $R_{0/1}$ approaches zero, f must correctly classify all input-output pairs, which consequently drives R_{NeSy} to zero as well.

Next, we prove the direction: $(R_{\text{NeSy}} \to 0) \Rightarrow (R_{0/1} \to 0)$. Equivalently, we prove the contrapositive:

$$(R_{0/1} \not\to 0) \Rightarrow (R_{\text{NeSy}} \not\to 0)\ .$$

Suppose $R_{0/1} \not\to 0$. Then, there exist integers $i, j \in [L]$ with $i \neq j$ such that f misclassifies elements of the set

$$\langle x \rangle_i = \{x \mid x \in \mathcal{X}, g(x) = i\}$$

as belonging to class j.

Recall that the DCSP of $\mathcal{T}$ has a unique solution, which ensures that the correct labels are unambiguous. As the training set size grows, there must exist a sample $(\boldsymbol{x}, y) \in (\mathcal{X}^m, \mathcal{Y})$ such that $\text{set}(\boldsymbol{x}) \cap \langle x \rangle_i \neq \varnothing$ and $f(\boldsymbol{x}) \notin A(y)$. This implies that $R_{\text{NeSy}} \not\to 0$.

Thus, by proving both directions, we complete the proof. □

B.4 Proof of Theorem 3.6

First, we recall that the learnability analysis depends on two assumptions.

Assumption 3.5. *The set of possible concept sequences, $\mathcal{B} = \bigcup_{y \in \mathcal{Y}} A(y)$, where $A(y)$ is the set of valid concept combinations for label y, has finite cardinality; and the probability of sampling a concept sequence is at least $\kappa > 0$*

Based on the assumptions, we first prove the below lemma, which states the sample complexity under the learnable case, when the hypothesis space is restricted hypotheses space.

Lemma B.1. *Consider a NeSy task $\mathcal{T}$ with above assumptions and $d = 0$. By applying empirical risk minimization, the hypothesis $\hat{f} = \arg\min_{f \in \mathcal{F}^*} \hat{R}_{NeSy}(f)$ ensures that $R_{0/1}(\hat{f}) \leq \epsilon$ for any $\epsilon > 0$, provided that the training size N satisfies the inequality:*

$$N > \frac{1}{\kappa} \cdot \log\left(\frac{|\mathcal{B}|}{\epsilon}\right).$$

Proof. Recall that empirical risk minimization, based on the neuro-symbolic risk $\hat{R}_{\text{NeSy}}$, corresponds to solving a derived constraint satisfaction problem over the restricted hypothesis space $\mathcal{F}^*$. By lemma 3.4, if the training set includes all possible concept sequences, the minimum value of R_{NeSy} becomes zero. This ensures that $R_{0/1}$ also attains a value of zero. Therefore, it is crucial to analyze the sampling process of the training data.

Let Q denote the event that "not all concept sequences are sampled in the training data". The concept risk is the probability of event P that "for a random sample (x, z), the learned hypotheses wrongly classified, i.e., $p[\hat{f}(x) \neq z]$". It is obvious that event P is included by event Q in the probabilistic space. Therefore, we conclude that the true risk is bounded by the probability of event Q:

$$R_{0/1}(\hat{f}) = p[\hat{f}(x) \neq z] \leq p[Q].$$

To bound $R_{0/1}$, it suffices to bound $p[Q]$. For any individual concept sequence $\boldsymbol{z}_i$, the probability that it is not sampled after N draws is given by:

$$(1 - p_i)^N \leq (1 - \kappa)^N .$$

Applying the union-bound inequality, we derive:

$$p[Q] \leq |\mathcal{B}| (1 - \kappa)^N .$$

Since $(1 - x) \leq \exp(-x)$ holds for $x \geq 0$, we can further bound $R_{0/1}(\hat{f})$ as follows:

$$R_{0/1}(\hat{f}) \leq p[Q] \leq |\mathcal{B}| \exp(-N \cdot \kappa).$$

Given that $R_{0/1}(\hat{f}) \leq \epsilon$, it follows that:

$$N \geq \frac{1}{\kappa} \cdot \log\left(\frac{|\mathcal{B}|}{\epsilon}\right).$$

This completes the proof of the proposition. □

Theorem 3.6. *For a neuro-symbolic task $\mathcal{T}$ with a restricted hypothesis space $\mathcal{F}^*$, learnability is determined by the following conditions:*

- *If the derived constraint satisfaction problem has a unique solution, the task is* learnable. *Specifically, the concept error is bounded by ϵ, provided that the sample size N satisfies:*

$$N > \frac{1}{\kappa} \cdot \log\left(|\mathcal{B}|/\epsilon\right).$$

- *Otherwise, the task is* unlearnable.

Proof. The proof is divided into two parts:

1. If the disagreement d equals zero, the task is *learnable*, and the sample complexity is $\mathcal{O}(\frac{1}{\kappa} \cdot \log(|\mathcal{B}|/\epsilon))$.
2. If the disagreement d is greater than zero, the DCSP solution space contains at least two distinct solutions, making the task *unlearnable*.

The first part follows directly from lemma B.1. Hence, we focus on proving the second part by contradiction.

If the DCSP has multiple solutions, there exists $(\boldsymbol{x}, y) \in (\mathcal{X}^m, \mathcal{Y})$ such that two distinct concept sequences $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ are valid, i.e., $\boldsymbol{z}_1 \wedge \mathtt{KB} \models y$ and $\boldsymbol{z}_2 \wedge \mathtt{KB} \models y$.

Since both $\hat{f}_1(\boldsymbol{x}) = \boldsymbol{z}_1$ and $\hat{f}_2(\boldsymbol{x}) = \boldsymbol{z}_2$ are valid solutions for $(\boldsymbol{x}, y)$, and $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ are distinct, it follows that their true risks cannot be simultaneously zero. Thus, at least one of them must have a $R_{0/1}$ greater than zero. Without loss of generality, assume that $R_{0/1}(\hat{f}_1) = \epsilon_0 > 0$.

Since both $\hat{f}_1$ and $\hat{f}_2$ achieve the minimal NeSy risk (which is zero), it is impossible to distinguish between them using learning techniques or by adding more data. Consequently, there is no integer N_ϵ such that for any $0 < \epsilon < \epsilon_0$, $R_{0/1}(\hat{f}) < \epsilon$ holds when $N \geq N_\epsilon$. This implies that the task is unlearnable.

Combining both parts completes the proof. □

B.5 Proof of Theorem 3.7

Theorem 3.7. *The average error $\mathcal{E}^*$ is bounded by:*

$$\mathcal{E}^* \leq \frac{d}{L}.$$

Proof. Recall that the DCSP solution disagreement d is given by $d = L - \text{Union}(\mathcal{S})$, where $\text{Union}(\mathcal{S})$ represents the common assignments among the solutions. Since the restricted hypothesis space ensures that instances with the same assigned label correspond to the same concept and vice versa. In the worst case, errors occur in at d classes, so the maximum true risk is $\max_{f \in \mathcal{F}^*_{\text{ERM}}} R_{0/1}(f) = d/L$. Thus, the average error is bounded by: $\mathcal{E}^* \leq \max_{f \in \mathcal{F}^*_{\text{ERM}}} R_{0/1}(f) = d/L$. □

C Experiments

We first introduce the experimental details, including data preparation, model setup, optimizer configurations, hyperparameters, and implementation details. After that, we present experiments omitted from the main context due to space constraints.

C.1 Experiment Details

Arithmetic tasks The raw datasets are based on MNIST (LeCun et al., 1994), KMNIST (Clanuwat et al., 2018), CIFAR-10 (Krizhevsky, 2009), and SVHN (Netzer et al., 2011). For MNIST-style datasets, the learning model is based on LeNet (LeCun & Bengio, 1998); other datasets use ResNet (He et al., 2016).

BDD-OIA The dataset are based on Xu et al. (2020), all evaluated methods are using the same backbone, i.e., `Conceptizer` defined in `rsbench` (Bortolotti et al., 2024). For the abduction-based methods, we adopt their official implementations.

The results were obtained using an Intel Xeon Platinum 8538 CPU and an NVIDIA A100-PCIE-40GB GPU on an Ubuntu 20.04 platform.

C.1.1 Preparing data and model

The construction of datasets is heavily based on algorithmic operations; thus, we rely on digit indices mapping from class indices to digit indices. After that, different knowledge bases require different rules. Here, we base our approach on ABLKit (Huang et al., 2024)[1] and the code of He et al. (2024b)[2]. During the dataset construction process, we control the sample size by resampling data until the sequence size exceeds a threshold, denoted as `sample_size`. For figure 2, the `sample_size` is set to $30,000$, while for aggregation experiments it is set to $120,000$; other values are specified in the respective plots. The reasoning model employs abductive reasoning, implemented using a cache-based search program (Huang et al., 2024).

For an example of addition, the knowledge base is programmed as follows:

```
class add_KB(KBBase):
  ...
  def logic_forward(self, nums):
    nums1, nums2 = split_list(nums)
    return digits_to_number(nums1) + digits_to_number(nums2)
```

Figure 4: Example of addition knowledge base with Python program form.

For the modular addition task, the knowledge base is more complex:

```
class Mod_KB(KBBase):
  ...
  def logic_forward(self, lsts):
      nums1, nums2, mod = parse_nums_and_mods(lsts)
      nums1, nums2 = digits_to_number(nums1), digits_to_number(
nums2)
      return (nums1 + nums2) % mod
```

Figure 5: Example of modular addition knowledge base with Python program form.

[1] https://github.com/AbductiveLearning/ABLkit

[2] https://github.com/Hao-Yuan-He/A3BL

Algorithm 1 DCSP Solution

Require: NeSy task $\mathcal{T}$ and training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$
$\mathsf{V}, \mathsf{D}, \mathsf{C} \leftarrow \{V_i\}_{i=1}^{L}, \{D_i = \mathcal{Z}\}_{i=1}^{L}, \{\}$ ▷ Initialize the CSP triple
for $i = 1 \ldots N$ **do**
  $\mathsf{C} \leftarrow \mathsf{C} \cup \{\mathsf{V}(\boldsymbol{x}_i) \wedge \mathtt{KB} \models y_i\}$ ▷ Initialize the constraints
end for
$\mathcal{S} \leftarrow \textsc{SolveCSP}(\mathsf{V}, \mathsf{D}, \mathsf{C})$ ▷ Call the CSP solver
$d \leftarrow L - \text{Union}(\mathcal{S})$
return $\mathcal{S}, d$

C.1.2 Implementation details

For all experiments, the random seeds are set to $\{2023, 2024, 2025, 2026, 2027\}$ for repeating five times. To ensure the clustering property depicted in the definition of the restricted hypothesis space, the learning models are pre-trained. For LeNet5, we use self-supervised methods, with the weights available in the supplementary materials. For ResNet50, we load the pre-trained weights from the official PyTorch library, named `ResNet50_Weights.IMAGENET1K_V2`, and replace the last linear layer with `Linear(2048, 10)`.

Optimizer Configurations. All experiments use AdamW, a weight-decay variant of Adam (Kingma & Ba, 2015), as the optimizer, with a learning rate of `0.0015` and betas set to $(0.9, 0.99)$. The batch size is set to 256, and unless otherwise noted, the number of epochs is set to 10. The loss function used for optimization is cross-entropy, with further details available in the support code.

C.2 Additional Experiments

The additional experiments include setups using raw datasets not covered in the main text, specifically the CIFAR-10 and SVHN cases. Additionally, beyond the empirical analysis on the surrogate (7), here we refer to this as A^{3}BL (He et al., 2024b), we also include an analysis of ABL and PNL to ensure a comprehensive evaluation.

C.2.1 Observations of the structure of DCSP solution space

The configurations of the modular addition task and their aggregations are illustrated in figure 6. These configurations were computed using algorithm 1 with the open-source library Choco (Prud'homme & Fages, 2022). Through the investigation of the modular addition task, we observe that: *the number of DCSP solutions is highly related to DCSP solution disagreement; however, this relationship is not monotonic.* Specifically, even a small number of DCSP solutions can result in high disagreement, as observed in the modular addition task with base $k = 10$.

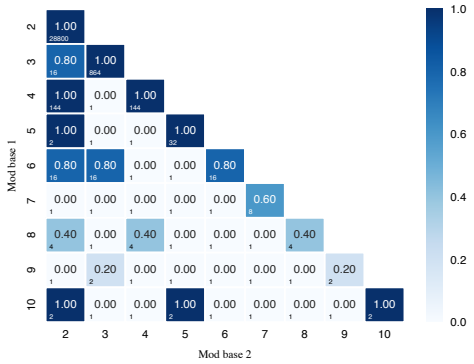
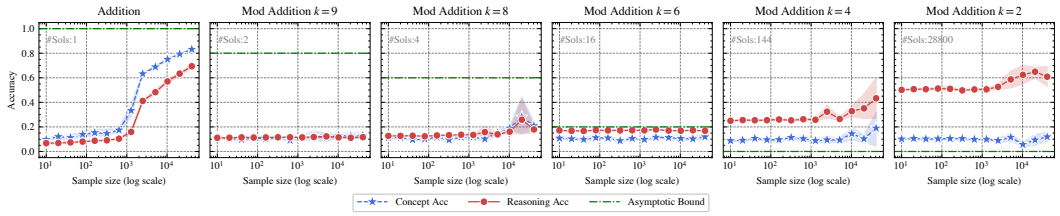
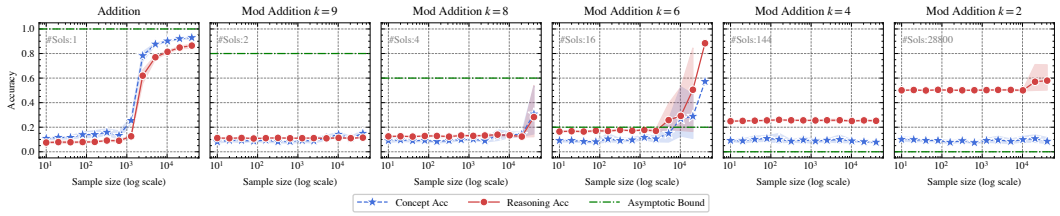


Figure 6: Configurations of modular additions and their aggregations. The center value represents the ratio of disagreement d to concept size L, while the number of DCSP solutions is shown at the bottom-left.



(a) CIFAR-10

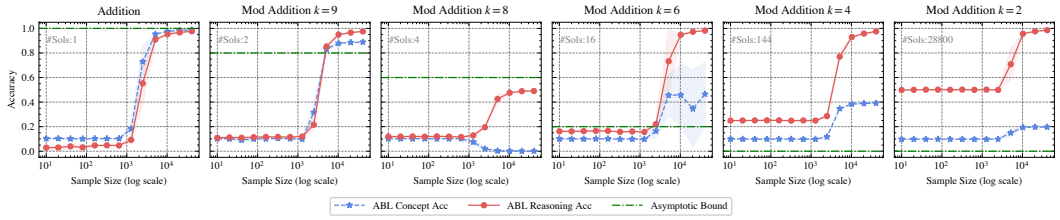


(b) SVHN

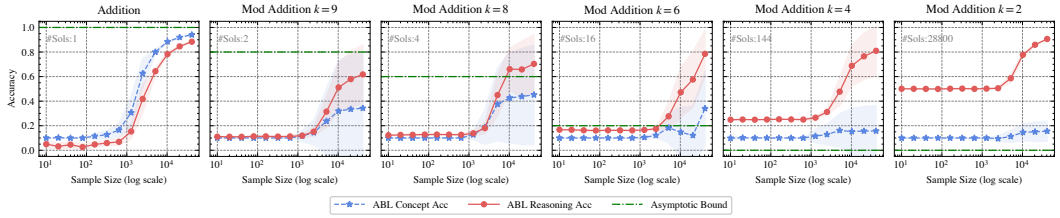
Figure 7: *Accuracies* versus *sample size* for different NeSy tasks of A^{3}BL. The shadowed area denotes the standard error. The number of the DCSP solutions (#Sols) is shown at the top left of each plot. The asymptotic bound (green line) from theorem 3.7 indicates that concept accuracy should exceed this bound as the sample size grows.

C.2.2 Impact of DCSP solution disagreement

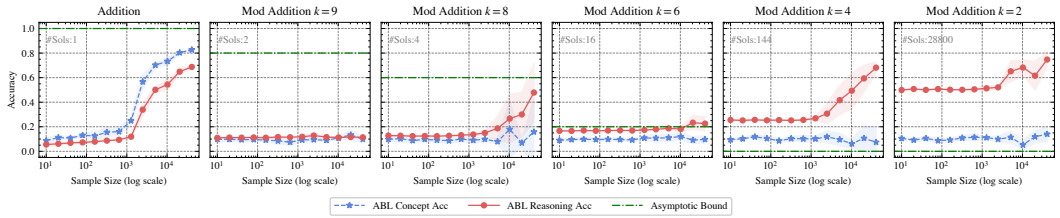
In figure 7, we present results under the same settings but using different raw datasets, specifically CIFAR-10 and SVHN, as shown in figure 7. As illustrated in the figure, the learnable case follows a trend similar to that in figure 1. However, in the unlearnable case, optimization becomes significantly more challenging due to high conflicts among valid DCSP solutions. While one might suspect that this issue stems from the specific surrogate used, applying the same settings to ABL and PNL produces similar results (cf. figure 8 and figure 9 respectively), confirming the generality of this observation.



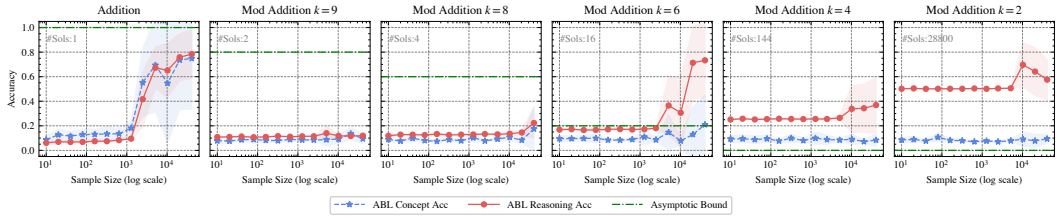
(a) MNIST



(b) KMNIST

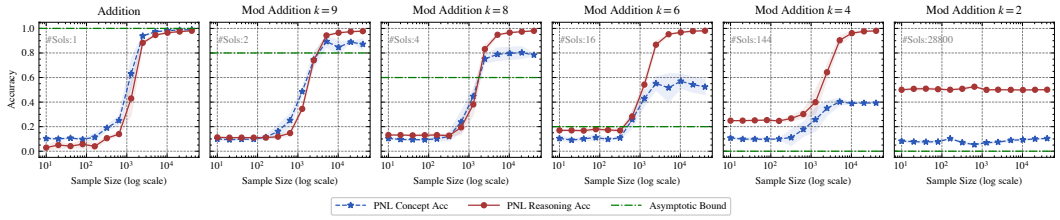


(c) CIFAR-10

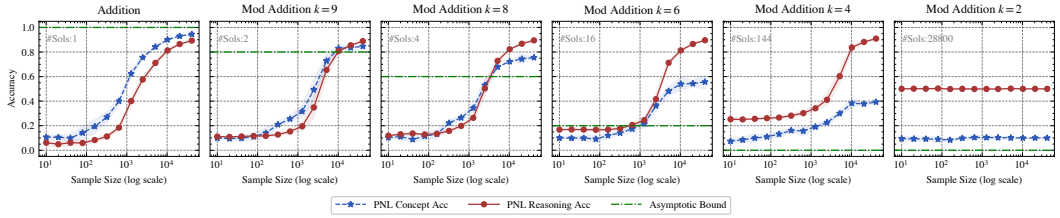


(d) SVHN

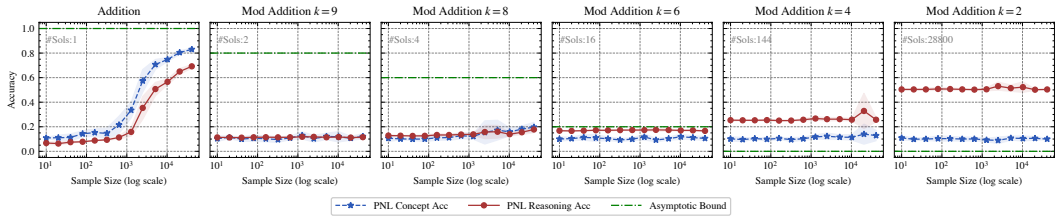
Figure 8: *Accuracies* versus *sample size* for different NeSy tasks of ABL. The shadowed area denotes the standard error. The number of the DCSP solutions (#Sols) is shown at the top left of each plot. The asymptotic bound (green line) from theorem 3.7 indicates that concept accuracy should exceed this bound as the sample size grows.



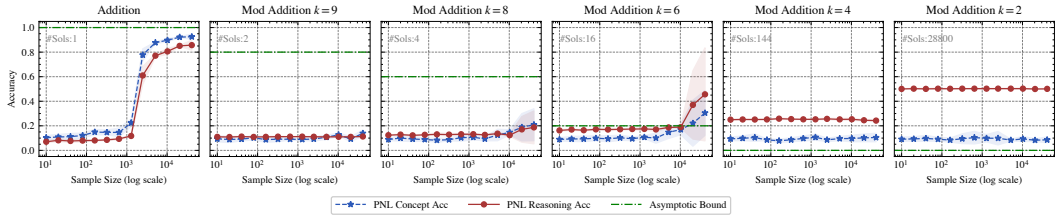
(a) MNIST



(b) KMNIST



(c) CIFAR-10



(d) SVHN

Figure 9: *Accuracies* versus *sample size* for different NeSy tasks of PNL. The shadowed area denotes the standard error. The number of the DCSP solutions (#Sols) is shown at the top left of each plot. The asymptotic bound (green line) from theorem 3.7 indicates that concept accuracy should exceed this bound as the sample size grows.

C.2.3 Aggregation of unlearnable NeSy tasks

Here we present additional combinations of unlearnable NeSy tasks. In figure 10, we provide an overview of all aggregation combinations using a heatmap. After that, we present a more fine-grained analysis. In figure 11 and figure 12, we show cases where the aggregation approach fails and succeeds respectively.

Figure 10: *Heatmaps* of aggregation mod addition tasks. Left: reasoning accuracy for different aggregations of mod bases; Right: concept accuracy for different aggregations of mod bases. The bottom-left corner of each cell shows the number of DCSP solutions.

By observing the experiments, we find that adopting the aggregation perspective can enrich the benchmark diversity in the NeSy field (e.g., `rsbench`, Bortolotti et al. 2024), providing a clear and controllable methodology to achieve this.

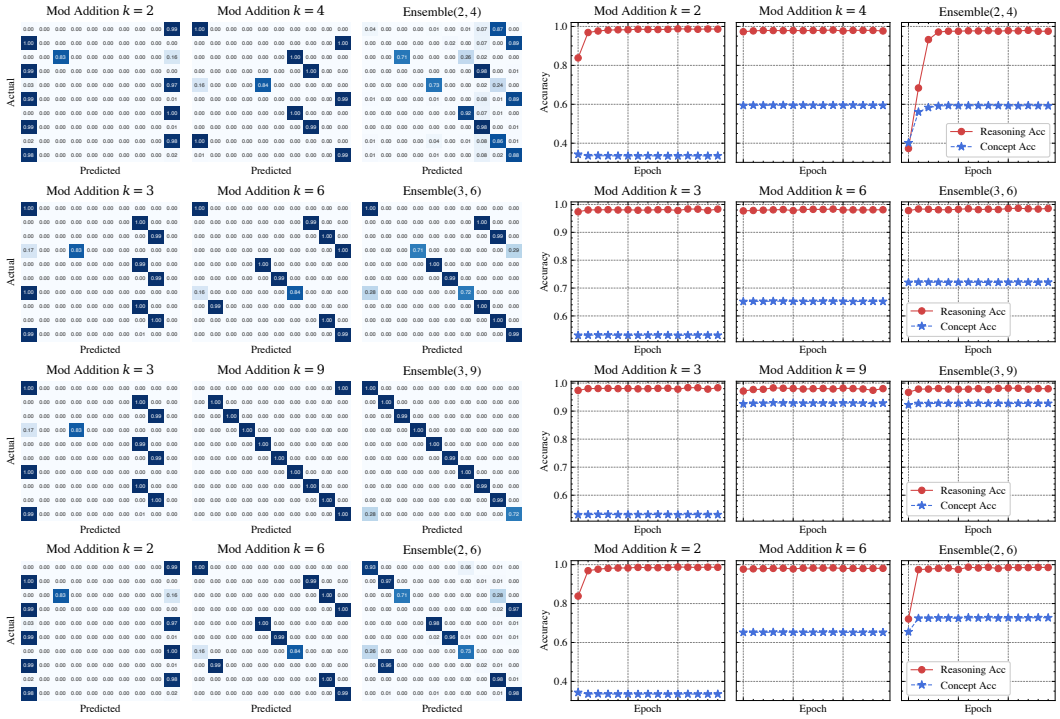


Figure 11: *aggregation* of unlearnable NeSy tasks, *failed* case. The left shows confusion matrices and the right displays accuracy curves. After the aggregation, the tasks are still unlearnable.

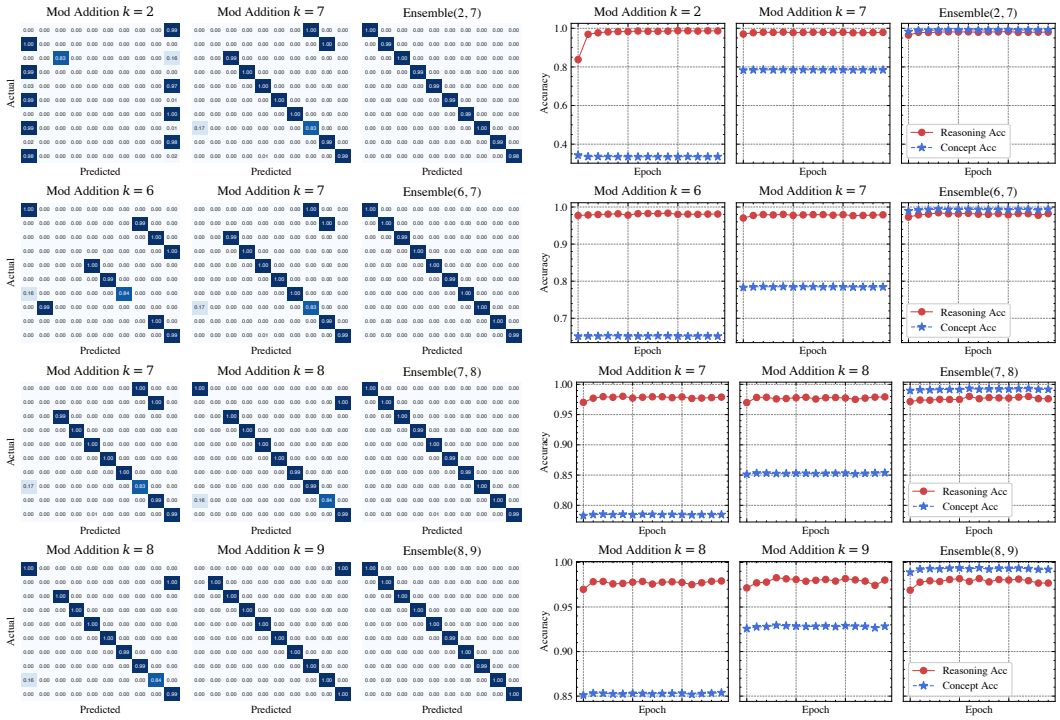


Figure 12: *aggregation* of unlearnable NeSy tasks, *successed* case. The left shows confusion matrices and the right displays accuracy curves. After the aggregation, the tasks become learnable.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: As stated in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See the section of limitation.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All the proofs are included.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: See the supplemental materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We include standard deviation color block in all figures.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computational requirements are included in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Of course.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See impact statement.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This is a theoretical work.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All code/data/model are included with their url.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: We do not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This is a theoretical work.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: N/A

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: This research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.