Implicit Learning for Reasoning in First-Order Probabilistic Logic

Luise Ge, Brendan Juba, Kris Nilsson, and Alison Shao[1]

[1]Washington University in St. Louis

Abstract

Reasoning under uncertainty is a fundamental challenge, particularly when inference involves complex computations over large relational domains. Existing probabilistic inference methods typically answer probabilistic queries by leveraging symmetries to perform efficient, lifted inference assuming access to a complete model. However, the upstream task of learning such a model from partial and noisy observations is generally intractable. We propose the first polynomial-time framework for "implicit learning to reason" in a first-order relational probability logic, even over infinite domains. Our approach combines statistical signals from data with semidefinite relaxations of the moment problem, jointly informed by new observations and prior knowledge from the knowledge base. This enables verifying or refuting first-order queries without ever constructing an explicit model. We establish soundness and completeness guarantees for our algorithm, as well as a tractability result.

1 INTRODUCTION

Many domains, including for example robotics, scientific reasoning, and analysis of social networks, involve reasoning about uncertain relationships among entities, including which entities exist, how they relate, and what properties they satisfy. Probabilistic relational inference is the problem of drawing sound inferences in such settings, where both the relational structure (who relates to whom and/or satisfies which properties) and the probabilistic dependencies (how likely various configurations are) must be represented and reasoned about jointly. This requires combining the expressiveness of a first-order logic, which naturally captures relational structure through quantified variables and predicates, with probabilistic semantics that quantify uncertainty. However, this combination faces a fundamental challenge: expressive logical languages often lead to computational intractability in large domains. In the first place, such applications demand *lifted* reasoning, as reasoning explicitly about all combinations of objects in large domains quickly becomes intractable. Probabilistic relational models have been developed to address precisely these settings. Richardson and Domingos [2006] and Getoor et al. [2001] for example define distributions over relational structures using weighted logical rules, and probabilistic logic programming [De Raedt et al., 2007, Bruynooghe et al., 2010, Manhaeve et al., 2018] defines distributions on relational structures using probabilistic logic programs.

Such approaches typically maintain a strict separation between learning and inference. A complete probabilistic model is first learned from data – often an NP-hard task in expressive settings (see, e.g., Chickering [1996]) – after which inference is performed over that fixed model, using lifted inference techniques [e.g. Poole, 2003, Van den Broeck et al., 2011, Kersting, 2012, Malhotra et al., 2026, Van Bremen and Kuželka, 2023] to exploit symmetry. This modularity is brittle: partial observations obscure dependencies, approximate learning can destroy the structural regularities needed for efficient inference, and the resulting systems are often both computationally expensive and misaligned with the reasoning tasks they must ultimately support. Even more recent tractable probabilistic models such as tractable Markov logic [Domingos and Webb, 2012], probabilistic theorem proving [Gogate and Domingos, 2016], sum-product networks [Poon and Domingos, 2011], cutset networks [Rahman et al., 2014], probabilistic sentential decision diagrams [Kisa et al., 2014], and more generally probabilistic circuits [Choi et al., 2020] rely on the Expectation-Maximization (E-M) algorithm or gradient ascent on the model log-likelihood function to learn the model. Tractability in this case means that the individual iterations are computationally tractable, but the learning methods still lack overall guarantees on the number of iterations until convergence or the ultimate solution quality. This matters

most in domains where a certifiable bound is required: a high-likelihood learned model alone need not rule out rare but unacceptable outcomes.

An alternative approach to probabilistic inference is based on the sum-of-squares (SOS) logic [Juba, 2019, Ge et al., 2025]. Following Halpern and Pucella [2007], one can generalize reasoning about probabilities to reasoning about expected values: each propositional variable φ corresponds a random variable $x := \mathbf{1}[\varphi \text{ holds}]$, with expected value $\Pr[\varphi]$. The language allows furthermore reasoning about numerical variables, such as a subject's blood sugar level, as a random variable $x := Bloodsugar(alex)$, for example bounding its moments as in "the expected value of x^2 is at most 10." The SOS hierarchy [Lasserre, 2001] provides a tractable fragment of such a logic: if the constraints are inconsistent, a bounded-degree polynomial refutation exists and can be found in polynomial time (for fixed degree d), yielding formal certificates rather than approximate solutions. Following the approach of Lasserre, consistency of a knowledge base and query is thus checked via a semidefinite program: one asks whether any probability distribution over the variables can simultaneously satisfy all of the encoded constraints. While the approach proposed by Lasserre was essentially propositional, Ge et al. [2025] recently proposed a lifted sum-of-squares logic. Based on an adaptation of the grounding trick [Belle, 2017], this logic allows tractable reasoning about open universes and large domains. But, Ge et al. only considered inference, not learning.

1.1 OUR APPROACH TO INTEGRATED LEARNING AND INFERENCE

In this work we consider how learning can be achieved in such a relational logic of expected value, while retaining guarantees on the quality of the output and polynomial running time. Learning here means that we infer constraints on the distribution using example sets of values for the random variables that are sampled from the probability distribution of interest. Specifically, we demonstrate the tractability of acquiring new knowledge from incomplete examples (meaning, not all variables' values are observed in all examples) and integrating this new knowledge with an explicit knowledge base for relational probabilistic reasoning.

We follow the "implicit learning to reason" paradigm [Juba, 2013, Belle and Juba, 2019]. Learning is "implicit" in the sense that, in the course of answering a query (deciding consistency), we draw on rules explicitly provided by the knowledge base together with rules inferable from the data to reach this new conclusion *without* producing explicit representations of the rules encoding this additional inferred knowledge. This contrasts with the "explicit" paradigm, in which a knowledge base is first constructed from data before reasoning begins. The advantage, following Juba [2013] and Belle and Juba [2019], is that when the explicit learning problem is intractable, implicit learning provides an approach to tractable learning with guarantees of correctness.

In this work we extend the approach of implicit learning at the lifted level to probabilistic knowledge. We encode our reasoning problem, using the SOS framework, as deciding the feasibility of a semidefinite program. For a given query, the system combines the explicitly given rules with rules that are implicitly learned from the partial data. These constraints are then used as the foundation for a proof by contradiction: we combine these constraints with a constraint asserting the query fails to hold, and check for inconsistency. Such inconsistency is witnessed by a bounded degree polynomial refutation corresponding to a (dual feasible) solution to the SOS program. Thus, our framework pivots from constructing a probabilistic model to constructing a certificate of inconsistency. By answering queries through this algebraic feasibility problem, we are able to bypass the bottleneck of constructing a complete probabilistic model. This yields a more efficient algorithm that is capable of polynomial time lifted probabilistic reasoning even over infinite domains.

We still face substantial technical obstacles. In prior SOS-based inference for probabilistic logic, empirical estimates were directly embedded into the semidefinite relaxation. By contrast, earlier work on implicit learning in first-order settings focused on Boolean logic [Belle and Juba, 2019] and relied on the grounding strategy of Belle [2017]: for each partially observed example, one enumerates candidate grounding sets and searches for a suitable instantiation. In the probabilistic setting, however, empirical quantities such as marginal probabilities must be estimated consistently across examples. Naively allowing each example to choose its own grounding would correspond to an exponential number of constraint combinations in the semidefinite program.

The key observation underlying our solution is that empirical estimates require a fixed grounding across all examples. To estimate the probability of a relational event, the same set of ground variables must be used throughout the sample; otherwise, one could artificially inflate estimates by selecting favorable groundings on a per-example basis. (An analogy is a lottery in which one is allowed to retrospectively choose the ticket considered in each draw.) Consequently, we solve a semidefinite program separately for each candidate grounding set, ensuring that empirical constraints are defined over a consistent domain.

This leads to a structural distinction between two types of knowledge in our framework. Expectation bounds are learned from empirical estimates with confidence intervals and therefore depend on consistent groundings across the dataset. Support constraints, in contrast, can be refuted by exhibiting a single violating grounding. The interaction between probabilistic expressions and relational structure thus forces a refined treatment of what it means for a rule to be "witnessed" by data.

1.2 OUR CONTRIBUTIONS

We propose the first implicit learning to reason framework that enables lifted probabilistic reasoning with *simultaneous* soundness, completeness and polynomial running time guarantee, even over *infinite* domains.

- We characterize precise conditions under which implicit learning to reason is complete in first-order probabilistic logic in terms of "testability".
- We design an algorithm that integrates empirical moment bounds with lifted sum-of-squares reasoning, and we prove matching soundness and completeness guarantees for bounded-degree refutations.
- We demonstrate that for a fixed SOS degree and quantifier rank, the resulting inference procedure runs in time polynomial in the bit-complexity of the input and the number of partial examples.

Our work establishes the first polynomial-time guarantee for combining relational probabilistic reasoning, implicit learning, and partial observations within a unified framework.

2 RELATED WORK

Inference Approaches Tractable Markov Logic (TML) [Domingos and Webb, 2012] and Weighted First-Order Model Counting (WFOMC) [Poole, 2003, Braz et al., 2005, Van den Broeck et al., 2011, Gogate and Domingos, 2016] are important frameworks for reasoning under uncertainty. In contrast to our approach, they operate in settings where an explicitly specified model is assumed. Recent lifted-inference work extends the classes of first-order theories and statistical statements for which inference can be performed efficiently Azzolini and Riguzzi [2023], Malhotra et al. [2026], and Ge et al. [2025] develop lifted SOS inference for open universes. These works focus on tractable inference within a given model or knowledge base; they do not address the implicit acquisition of expectation bounds from partial observations.

Model-Free Approaches Model-free methods answer queries without constructing a full probabilistic model. The closest antecedents are the implicit learning-to-reason frameworks of Juba [2013, 2019], Belle and Juba [2019], Mocanu et al. [2020], and Rader et al. [2021]. However, none of these extend to first-order probabilistic relational domains. Our approach generalizes implicit reasoning to first-order Probability Relational Logic (FOPRL) Ge et al. [2025] via a lifted Sum-of-Squares relaxation, enabling tractable inference with *double lifting* (grounding-lift and world-lift) directly on axioms and data.

Classical Model-Based Approaches Probabilistic relational models [Richardson and Domingos, 2006, Getoor et al., 2001] define joint distributions over relational structures via weighted logical rules. There is a substantial literature on learning Markov logic networks, including discriminative structure and parameter learning, bottom-up structure learning, motif-based structure learning, lifted generative learning, and boosting-based learning with missing data [Lowd and Domingos, 2007, Mihalkova and Mooney, 2007, Huynh and Mooney, 2008, Kok and Domingos, 2010, Khot et al., 2015, Van Haaren et al., 2016]. Other work exploits symmetries to scale relational training or construct lifted models [Ahmadi et al., 2013, Luttermann et al., 2024]. These methods learn or transform explicit probabilistic models. Our objective is different: we certify the consequence of the explicit knowledge base together with empirical expectation bounds without first selecting a full parametric joint distribution. ILP methods [Cropper and Dumančić, 2022] learn symbolic rules from data but typically ignore uncertainty or require extensive hand-crafting.

Neural-Symbolic Approaches DeepProbLog [Manhaeve et al., 2018] extends ProbLog with trainable neural predicates, but relies on grounding and Sentential Decision Diagram (SDD) compilation, limiting scalability in evolving domains. Neural Markov Logic Networks (NMLNs) [Marra and Kuželka, 2021] replace symbolic rules with neural potentials over fragments, learning a smooth approximation of the joint distribution. Both remain fundamentally model-based—requiring sampling or approximate inference over an explicit generative model—and so do not eliminate the bottlenecks of structure learning or grounding. Approaches like Neural Theorem Provers [Rocktäschel and Riedel, 2017], Logic Tensor Networks [Donadello et al., 2017] similarly attempt to reconcile symbolic reasoning with neural networks, but ultimately face similar model-based limitations requiring approximate inference or sampling over an explicit representation.

3 LOGICAL PRELIMINARIES

We study probabilistic inference over relational structures. Each ground relation is viewed as a random variable, and our goal is to reason about the unknown joint distribution $\mathscr{D}$ over all random variables. We assume a knowledge base Δ that provides a partial characterization of the distribution through two types of constraints: Support constraints define the support of $\mathscr{D}$ by specifying relationships among the values of the random variables that must hold almost surely. Expectation constraints restrict the possible expected values of polynomials in these random variables. Both types of constraints are defined by polynomial inequalities. We begin by defining the language used to formalize these expressions.

Language: We work in a first-order relational language $\mathscr{L}$ that includes a countably infinite set of *names* $\mathscr{N} \cong \mathbb{N}$ serving as the domain of quantification. Formally, the set

$\mathscr{N}$ will correspond to the set of integers $\mathbb{N}$, but informally we will use proper names such as $\{aaron, adam, alex, ...\}$ for readability. The language also includes a countable set of *variables* $\mathscr{V} = \{u, v, w, \dots\}$ that can represent arbitrary elements in the domain, and *relational symbols* of various finite arities (e.g., $\{P(v), Q(u, v), ...\}$). We also consider a finite set of *constants* $\mathscr{C} \subseteq \mathscr{N}$, which are fixed elements of the domain. We refer to $\mathscr{G} = \mathscr{N} \setminus \mathscr{C}$ as *generic names* that can be freely permuted without affecting the constants. A *renaming substitution* is any permutation on $\mathscr{G}$ that fixes $\mathscr{C}$. A *term* is $R(t_1, \dots, t_k)$ with R a k-ary predicate and each t_i is either a variable or a name. If all $t_i \in \mathscr{N}$, we call the term an *atom*. We write $\text{ATOMS} = \{\, R(a_1, \dots, a_k) \mid R \text{ is } k\text{-ary},\ a_i \in \mathscr{N}\}$. Importantly, in our interpretation of this language, the random variables corresponding to atoms will take **real values**, and we will use the atoms in arithmetic expressions that define our constraints.

Support Constraints: Each term $\tau \in \mathscr{T}$ corresponds to a random variable x_τ in our framework that represents the value of a relational variable. In particular, the distribution we aim to reason about is a joint distribution over all atoms (i.e., fully instantiated relational terms). To express more complicated interactions among relations, we allow monomials: products of these variables raised to nonnegative integer powers, limited in degree to keep expressions manageable. Specifically, fixing a maximum degree d, a *monomial* of degree at most d is $\mu := \prod_{\tau \in \mathscr{T}} x_\tau^{\alpha_\tau}$, $\alpha_\tau \in \mathbb{Z}_{\geq 0}$, $\sum_i \alpha_\tau \leq d$. Given a finite set M of such monomials and real coefficients $\{c_\mu\}$, a polynomial inequality over terms (equivalently, a linear inequality over their monomials) is $p(\vec{x}) = \sum_{\mu \in M} c_\mu\, \mu \geq 0$, $\max_{\mu \in M} \deg(\mu) \leq d$. We bind these inequalities under universal quantifiers analogous to the language of Lakemeyer and Levesque [2002], whose domain is carved out by Boolean combinations of equalities (e.g., $u = v$, $x = adam$) over $\mathscr{V} \cup \mathscr{C}$.

By pairing an equality expression Ξ with a polynomial inequality Λ we can obtain a *support constraint* formula $\Phi = \forall \Xi \supset \Lambda$. This can be shortened to $\forall \Lambda$ when Ξ is a tautology. The *quantifier rank* of Φ can be given by the number of distinct variables occurring in Ξ and Λ. The semantics of Φ is that given a substitution of names θ satisfies Ξ then that same substitution will satisfy Λ.

Example 1. *The support constraint* $\forall x[height(x) \geq 0]$ *asserts that every individual must have nonnegative height.*

Expectation Constraints: To express probabilistic knowledge, we introduce expectation constraints in the spirit of Halpern and Pucella [2007]. For any monomial μ, let $e(\mu)$ denote its expected value under the unknown distribution $\mathscr{D}$. A (first-order) expectation constraint is a linear inequality over moment variables of the form: $\forall \Xi \supset [\sum c_{e(\mu)}\, e(\mu)\, \{\geq, \leq\} c_0]$. Its quantifier rank and semantics are defined analogously to the support constraints.

Example 2. *The expectation constraint* $\forall x \neq alex \supset [e(\text{height}(x)) - 60 \geq 0]$ *asserts that the average height of all individuals, except Alex, is over 60 inches.*

Knowledge Base and Query: A *knowledge base* Δ is a finite set of support and expectation constraints. A *query* φ is another constraint whose validity we test by checking whether Δ together with the negation of φ is inconsistent.

Semantics: A $\mathscr{L}$-model $\vec{x}$ assigns real-valued functions to each relational symbol over the domain $\mathscr{N}$, along with a probability measure on this space, such that all support constraints hold almost surely and all expectation constraints hold exactly.

Grounding: A ground knowledge base $GND(\Delta)$ is one that only contains *ground* terms. Starting with a general Δ we can obtain a $GND(\Delta)$ by substituting variables with names. We will also define the rank of Δ to be the maximum quantifier rank of any formula in Δ. This allows us to define $GND(\Delta) = \{\phi\theta | [\forall \Xi \supset \phi] \in \Delta, \models \Xi\theta\}$. In an open universe where we have a countably infinite number of names the grounding can also be infinite. This makes the following form more useful: $GND(\Delta, k) = \{\phi\theta | [\forall \Xi \supset \phi] \in \Delta, \models \Xi\theta\ \theta \in \mathscr{C}\}$ where $\mathscr{C}$ consists of all constants present in Δ as well as k additional generic names. This definition limits our substitutions to a finite subset of our possible names. Specifically, one that contains all constants, and k additional generic names.

Example 3. *Given* $\Delta = \{\forall x[P(x, alex) \geq 0]\}$ *then* $GND(\Delta, 2) = \{P(alex, alex) \geq 0, P(adam, alex) \geq 0, P(aaron, alex) \geq 0\}$.

3.1 LIFTED RELATIONAL SUM-OF-SQUARES REFUTATIONS

We now recall the construction of lifted relational SOS refutations introduced by Ge et al. [2025]. These refutations enable reasoning over knowledge bases consisting of both support and expectation constraints by demonstrating that no joint distribution—that is, no model—can satisfy all the constraints simultaneously. This approach extends Putinar's Positivstellensatz [Putinar, 1993] to the relational setting.

Let Δ consist of a collection of logical inequalities $\{g_i \geq 0\}_{i \in I}$, and expectation constraints $\{b_j \geq 0\}_{j \in J}$. A *sum-of-squares polynomial* is given by a sum of squares of polynomials $\sum_{\ell \in L} (p_\ell)^2$. A *sum-of-squares refutation* is given by the sum-of-squares polynomials σ_0 and σ_i for $i \in I$, and positive constants $\{r_j\}_J$ satisfying the formal equation

$$\sigma_0 + \sum_{i \in I} \sigma_i g_i + \sum_{j \in J} r_j b_j = -1. \tag{1}$$

Equation 1 is a syntactic polynomial identity. Its soundness comes from the constraints in Δ: in any distribution satisfying the support constraints, each product $\sigma_i g_i$ has

nonnegative expectation, and each expectation constraint $b_j \geq 0$ contributes a nonnegative term because $r_j > 0$. The sum therefore cannot evaluate to -1 in any model of Δ. Thus, Equation 1 certifies that no such distribution exists: Δ is inconsistent.

This converts probabilistic inference into an algebraic feasibility problem: finding σ_0 and σ_i for $i \in I$, along with positive constants r_j that satisfy Equation 1. Assuming the system is "explicitly compact" (see Definition 2), refutations may be found in polynomial time so long as the degree of the refutation is bounded by an absolute constant [Shor, 1987, Nesterov, 2000, Parrilo, 2000, Lasserre, 2001].

This is achieved by solving a semidefinite program constructed as follows: first, supposing $\vec{v}$ is a vector of monomials of degree up to $d/2$, abusing notation, let $e(\vec{v}\vec{v}^\top)$ denote the matrix obtained by taking the $e(\mu)$ variable for each monomial μ in the matrix of variables $\vec{v}\vec{v}^\top$, known as the *moment matrix*. For a polynomial h_j of degree d_j, for each monomial μ of degree up to $d - d_j$, we consider the *shifted* polynomial μh_j in which each monomial ν of h_j is substituted by the variable $e(\mu \cdot \nu)$ (obtained by summing the degree vectors). Next, for a given polynomial g_i of degree d_i, let v' be a vector of monomials of degree up to $(d - d_i)/2$, and let $e(g_i\vec{v'}\vec{v'}^\top)$ denote the matrix of shifts of g_i with the monomial ν is the monomial that would appear in the corresponding entry of the moment matrix $e(\vec{v'}\vec{v'}^\top)$. We refer to this as the *localizing matrix* for g_i. Now, the *sum-of-squares program* is the semidefinite program that asserts that the moment matrix is positive semidefinite, the localizing matrices for the inequality constraints are positive semidefinite, the shifted polynomials for the equality constraints are all equal to 0, and the expectation bounds are simply included with each monomial μ replaced by the corresponding $e(\mu)$.

The lifted variant is obtained by grounding Δ, and showing that the ground refutation is sound and complete for the lifted model.

Definition 1 (Ge et al. [2025])**.** *A* lifted sum-of-squares system *is given by the union of $GND(\Delta, k)$ and the set of equality constraints $e(\mu) - e(\mu') = 0$ for each pair of ground monomials used in $GND(\Delta, k)$ such that for a renaming substitution θ, $\mu = \mu'\theta$.*

Theorem 1 (Ge et al. [2025])**.** (Transferability from open to closed universe) *For any fixed degree d, the grounded system $GND(\Delta)$ admits a degree-d sum-of-squares refutation if and only if the lifted sum-of-squares system over $GND(\Delta, k)$ admits a degree-d refutation, where k is the quantifier rank of Δ.*

This theorem guarantees that every $GND(\Delta)$ that uses an infinite set of names also has a refutation when limited to $GND(\Delta, k)$. For inference, now, we need the analogue of explicit compactness for our first-order systems:

Definition 2 (Explicit Compactness of a Knowledge Base)**.** *The system Δ is* explicitly compact *if for each monomial μ, Δ includes constraints of the form $\mu - L_\mu \geq 0$ and $U_\mu - \mu \geq 0$ for some L_μ, U_μ that are finite.*

Explicit compactness guarantees every μ falls within the bounded range $[L_\mu, U_\mu]$.

Theorem 2. (Soundness and Completeness, Ge et al. [2025]) *Given an explicitly compact knowledge base Δ and a grounding $GND(\Delta, k)$, a lifted sum-of-squares program using $GND(\Delta, k)$ is sound and complete for degree-d refutations of Δ.*

The second theorem ensures that lifted sum-of-squares is sound and complete on a $GND(\Delta, k)$. These two theorems together are sufficient to show that lifted sum-of-squares is sound and complete for an infinite set of names.

We remark that under certain conditions, some finite-degree sum-of-squares program is equivalent to the infinite version (without a bound on the degree) [Ghasemi et al., 2016]. In this case, the program is only feasible if a consistent probability distribution exists, and the system is complete in a strong sense. The degree may be large, however, so this is generally not computationally useful.

4 FORMULATION OF LEARNING WITH PARTIAL OBSERVATIONS

In many real-world domains, especially those involving complex relational structures (e.g., healthcare, scientific experiments, or social networks), observations are often partial or incomplete. Crucially, these missing values are not merely noise—they arise from systematic processes, such as sensor limitations, privacy constraints, or study design. Following prior works by Rubin [1976] and Michael [2010], we explicitly model this partial observability by introducing a probabilistic masking process Θ over full relational models. This allows us to reason about uncertainty both from the underlying probabilistic structure and from the incompleteness of the data, in a unified framework. It also enables us to answer queries that are robust to missing information, rather than relying on imputation or assuming fully observed inputs.

Definition 3. *A **partial model** $\vec{\rho}$ maps* ATOMS *to $\mathbb{R} \cup \{*\}$. We say $\vec{\rho}$ is consistent with a full model $\vec{x} \in \mathscr{M}$ if $\vec{\rho}_\tau = \vec{x}_\tau$ for all coordinates such that $\vec{\rho}_\tau \neq *$. We call the set of partial models $\mathscr{P}$.*

Definition 4. *A **masking function** is a function $\theta : \mathscr{M} \to \mathscr{P}$ such that $\theta(\vec{x})$ is consistent with $\vec{x}$. A* masking process Θ *is a random variable over such functions. The induced distribution over partial models is denoted $\Theta(D)$.*

Our goal is to answer relational-probabilistic queries using two sources of information: (i) a partial knowledge base Δ specified in the first-order relational probabilistic logic from Section 3 describing $\mathscr{D}$, and (ii) a collection of *partial models* drawn i.i.d. from $\Theta(D)$.

Example 4. *Consider a laboratory studying several candidate drugs that might shrink tumors. Their initial knowledge base Δ encodes support constraints indicating that the relations take Boolean values, only one drug is given to a mouse at a time, and a criterion for progressing to a second trial stage, requiring* $\Pr[\texttt{shrunk}(x)|\texttt{treated}(x,d)] \geq 0.8$*:*

$$\forall x, d: \ \texttt{treated}(x,d)^2 = \texttt{treated}(x,d),$$
$$\forall x: \ \texttt{shrunk}(x)^2 = \texttt{shrunk}(x),$$
$$\forall d: \ \texttt{second_stage}(d)^2 = \texttt{second_stage}(d),$$
$$\forall (x,d) \neq (x',d'): \texttt{treated}(x,d)\,\texttt{treated}(x',d') = 0,$$
$$\forall x, d: \ \big(0.8\,\mathbb{E}[\texttt{treated}(x,d)\,\texttt{second_stage}(d)] -$$
$$\mathbb{E}[\texttt{treated}(x,d)\,\texttt{shrunk}(x)\,\texttt{second_stage}(d)]\big) \leq 0$$

Suppose our samples come from a mixture of six "worlds". Because of limitations in the imaging equipment, each observation always shows which drug was given but independently hides the value of $\texttt{shrink}(x)$ *with 20% probability. The hypothesis under test is "every drug advances to Phase II", or equivalently—* $\forall d, \texttt{second_stage}(d) = 1$.

World	Prob.	Drug	shrink(mouse)
1	0.15	A	1
2	0.25	A	0
3	0.1	A	*
4	0.3	B	1
5	0.1	B	0
6	0.1	B	*

Table 1: Trial outcomes in each world (* denotes a hidden value).

Under partial observability, we can only partially evaluate the expressions in polynomial constraints.

Definition 5 (Partial Evaluation)**.** *For a polynomial $p(\vec{x})$, we will let $p|_{\vec{\rho}}(\vec{x})$ denote the polynomial with ρ_τ plugged in for x_τ whenever $\rho_\tau \neq *$, and collecting terms with the same monomial. We refer to this as the partial evaluation of p under $\vec{\rho}$.*

Example 5. *Continuing Example 4, notice that because we have the observation that* $\texttt{treated}(x^*, d^*) = 1$ *in each partial example $\rho^{(i)}$, partial evaluation of the mutual exclusion constraint under $\rho^{(i)}$ for each $(x',d') \neq (x^*,d^*)$ gives that* $\texttt{treated}(x',d') = 0$*, even though this was not directly recorded in the data.*

In general, the partial evaluation may not be adequate to determine whether a polynomial constraint was satisfied by the full joint assignment drawn from the distribution. We can only infer frequently *witnessed* support constraints: those that remain verifiable despite missing entries. Indeed, at least in isolation, if an expression could be falsified by some setting of the unobserved values, then in general it may be that the full assignment does not satisfy the corresponding constraint. Since monomial bounds are not directly observed, we must infer these via sum-of-squares reasoning. On each example, witnessing is defined as:

Definition 6 (Witnessing)**.** *Let $\vec{\rho}$ be a partial model, and suppose that for each monomial μ the program constraints include some explicit upper and lower bound U_μ and L_μ. Let $L_\mu(\vec{\rho}) \leq \mu \leq U_\mu(\vec{\rho})$ be the tightest lower and upper bounds on the monomial μ consistent with $\vec{\rho}$ and the constraints of the given program. We say a polynomial inequality $p(\vec{x}) = \sum_{\mu \in M} c_\mu \mu \geq 0$ with $c_\mu \in \mathbb{R}$ is* witnessed *by $\vec{\rho}$ if, after substituting each μ by its worst-case value*

$$\hat{\mu} \ = \ \begin{cases} L_\mu(\vec{\rho}), & c_\mu > 0, \\ U_\mu(\vec{\rho}), & c_\mu < 0, \end{cases}$$

the resulting constant remains nonnegative: $\sum_\mu c_\mu \, \mu \ \geq \ 0$.

Definition 7 (Testability of Support Constraints)**.** *A system of polynomial inequalities $\{p_\ell(\vec{x}) \geq 0\}_{\ell \in L}$ is* $(1-\gamma)$-testable *under $\mathscr{D}$ and Θ if, with probability at least $1-\gamma$ over $\vec{\rho} \sim \Theta(D)$, every p_ℓ is witnessed under $\vec{\rho}$.*

Example 6. *A scientist may assert that DrugA does not guarantee shrinkage, based on a constraint:* $g(\vec{x}) := 1 - \texttt{treated}(mouse, \text{DrugA}) - \texttt{shrink}(mouse) \geq 0$. *In the example distribution, $mouse$ is treated with DrugA 50% of the time. The constraint is only witnessed in the worlds where the mouse is treated with DrugA,* $\texttt{shrink}(mouse_1) = 0$*, and the shrinkage label is observed. These events have a joint probability of* 0.15*, respectively, so the witnessing probability under $\Theta(D)$ is* 0.15*, i.e., g is* 0.15*-testable .*

For expectation constraints, we use confidence intervals derived from empirical averages. Because our system is explicitly compact, we can obtain these confidence intervals using Hoeffding's inequality:

Theorem 3 (Hoeffding's Inequality)**.** *Let $X_1, \ldots, X_m$ be independent random variables such that $a \leq X_i \leq b$ for all i. Then for the empirical mean $\bar{X} = \frac{1}{m}\sum_{i=1}^m X_i$ and any $\varepsilon > 0$,* $\Pr[\bar{X} - \mathbb{E}[X] \geq \varepsilon] \leq \exp\left(-\frac{2m\varepsilon^2}{(b-a)^2}\right)$.

Definition 8 (Naive Norm)**.** *Let U_μ and L_μ be the upper and lower bounds on the monomial μ as in an explicitly compact system. Given a polynomial expression $p(\vec{x})$, we define its* naive upper bound *to be $U_p = \sum_{\mu: c_\mu \geq 0} c_\mu U_\mu + \sum_{\mu: c_\mu < 0} c_\mu L_\mu$ and its* naive lower bound *to be $L_p =$*

$\sum_{\mu:c_\mu \geq 0} c_\mu L_\mu + \sum_{\mu:c_\mu<0} c_\mu U_\mu$. *Its naive norm is then* $|p| := U_p - L_p$.

Definition 9 (Testability of Expectation Constraints)**.** *An expectation constraint* $e(p) = \sum c_{e(\mu)}\, e(\mu) \geq \ell$ *or* $e(p) = \sum c_{e(\mu)}\, e(\mu) \leq u$ *is* $(1-\gamma)$-testable *from* m *partial examples if it is statistically certified by the empirical bounds as follows. For the naive norm* $|p|$ *of* p*, and* m *examples with parameter* γ*, consider the accuracy parameter*

$$\epsilon_m(p,\gamma) = |p|\sqrt{\frac{\ln(2/\gamma)}{2m}}.$$

The empirical estimates of a polynomial p *with respect to our monomial bounds are* $\bar{U}(p) = \frac{1}{m}\sum_t U_p(\vec{\rho}^{(t)})$ *and* $\bar{L}(p) = \frac{1}{m}\sum_t L_p(\vec{\rho}^{(t)})$*. If the constraint is an upper bound* $e(p) \leq u$*, then we require* $u \geq \bar{U}(p) + \epsilon_m(p,\gamma)$*; if it is a lower bound* $e(p) \geq \ell$*, then we require* $\ell \leq \bar{L}(p) - \epsilon_m(p,\gamma)$*.*

These inequalities are obtained by substituting $\epsilon_m(p,\gamma)$ into the Hoeffding bounds for the random variables $U_p(\vec{\rho})$ and $L_p(\vec{\rho})$, whose ranges have width at most $|p|$.

Example 7. *A lab assistant gave a rough estimate instead:*

$$e(\texttt{treated}(mouse, \mathrm{DrugA}) \cdot \texttt{shrink}(mouse)) \leq 0.12$$

$$e(\texttt{treated}(mouse, \mathrm{DrugA})) \geq 0.5$$

Let us denote $t = \texttt{treated}(mouse, \mathrm{DrugA})$; $s = \texttt{shrink}(mouse)$*. Suppose the empirical upper/lower bounds from* 1200 *masked samples yield* $\bar{U}(ts) = 0.22$*,* $\bar{L}(ts) = 0.17$*,* $\bar{U}(t) = .52$*, and* $\bar{L}(t) = .44$*. We can then check if these estimates are* .99*-testable using the definitions above. Both are Boolean so* $|ts| = |t| = 1$*, and*

$$\epsilon_m(t,\gamma) = \epsilon_m(ts,\gamma) = \sqrt{\frac{\ln(2/.01)}{2400}} \approx 0.047$$

So, for each bound, we respectively require that:

$$0.12 \leq 0.17 - 0.047 = 0.123$$

$$0.5 \geq 0.52 - 0.047 = 0.473$$

This means that the first estimate on ts *is* .99*-testable, while the second estimate on* t *is not* .99*-testable.*

5 ALGORITHM FOR INFERENCE WITH IMPLICIT LEARNING AND ITS ANALYSIS

In the usual way, we prove a query by introducing its negation to our knowledge base Δ, and refuting the resulting, extended knowledge base. Thus, henceforth, we simply assume that our task is to refute a given collection Δ.

In the algorithm, we will compute the tightest lower bound for each variable v that is consistent with a given partial model $\vec{\rho}^{(i)}$ by determining the minimum value v can take while satisfying all witnessed constraints in Δ restricted to $\vec{\rho}^{(i)}$, which we denote by $\mathrm{LBound}(v, \vec{\rho}^{(i)})$. Similarly, $\mathrm{UBound}(v, \vec{\rho}^{(i)})$ computes the maximum value. For directly observed variables in $\vec{\rho}^{(i)}$, these functions simply return the observed value.

Algorithm 1 Implicit Learning to Reason

Require: degree bound d, naive norm bound S, confidence parameter δ, partial models $\vec{\rho}^{(1)}, \ldots, \vec{\rho}^{(m)}$, KB Δ, the quantifier rank k of Δ

Initialize V to the set of monomials of degree at most d formed over terms in $\mathrm{GND}(\Delta, k)$.
Initialize arrays $L[v] \leftarrow 0$ and $U[v] \leftarrow 0$ for all $v \in V$
for $i = 1, \ldots, m$ **do**
  for all $v \in V$ **do**
    $L[v] \leftarrow L[v] + \mathrm{LBound}(v, \vec{\rho}^{(i)})$ ▷ via SOS
    $U[v] \leftarrow U[v] + \mathrm{UBound}(v, \vec{\rho}^{(i)})$ ▷ via SOS
  end for
end for
for all $v \in V$ **do**
  $L[v] \leftarrow L[v]/m$ ▷ average bounds
  $U[v] \leftarrow U[v]/m$
end for
Let $N = |V|$ and $\epsilon = S\sqrt{\ln(2N/\delta)/(2m)}$
Run SOS solver on $\{\Delta|_{\vec{\rho}^{(t)}}\}_{t\in[m]}$ with moment bounds $B = \{e(v) \geq L[v] - \epsilon\}_{v\in V} \cup \{e(v) \leq U[v] + \epsilon\}_{v\in V}$
return True if the SOS program is feasible, else **False**

Theorem 4 (Soundness)**.** *Let* $\delta \in (0,1)$ *and* $d, k \in \mathbb{N}$ *be given and let* S *be the input naive norm parameter. Let* V *be the set of monomials used by Algorithm 1 and let* $N = |V|$*. Suppose we have* $m = \Omega(S^2 \log(N/\delta))$ *partial models* $\vec{\rho}^{(1)}, \vec{\rho}^{(2)}, \ldots, \vec{\rho}^{(m)}$ *drawn i.i.d. from* $\Theta(D)$ *and every monomial* $v \in V$ *has range* $U_v - L_v \leq S$*. If* $\mathscr{D}$ *satisfies the input system* Δ *then Algorithm 1 will return True with probability* $1 - \delta$*.*

Proof. Starting with the support constraints of Δ, we see that every $\vec{\rho}^{(t)}$ is consistent with Δ. For the expectation bounds, we observe that every moment $e(v)$ is a bounded random variable in $[L_v, U_v]$. Hoeffding's inequality gives

$$\Pr[|e(v) - \widehat{e}(v)| > \epsilon] \leq \delta/N$$

for $\epsilon = S\sqrt{\ln(2N/\delta)/(2m)}$, where $\widehat{e}(v)$ denotes the empirical upper or lower estimator used by the algorithm. A union bound over all $v \in V$ implies that all true moments lie inside the algorithm's intervals simultaneously with probability at least $1 - \delta$. Since $\mathscr{D}$ satisfies Δ, these intervals admit the true moment sequence as a feasible point of the SOS relaxation. Therefore the algorithm returns True. □

Theorem 5 (Completeness)**.** *Let $\delta \in (0,1)$, $d, k \in \mathbb{N}$ be given, and S be the naive norm bound. Let V be the set of monomials used by Algorithm 1 and let $N = |V|$. Suppose $m \in \Omega(S^2 \log(N/\delta))$, there is a set of support constraints I_s that is $(1 - \frac{1}{2S})$-testable under $\Theta(D)$ and a set of expectation constraints I_m that is $1 - \delta/2$-testable under $\Theta(D)$, that, together with the input system Δ have a degree-d SOS refutation with naive norm bounded by S.*

Then, with probability $1 - \delta$, Algorithm 1 also returns a refutation.

Proof. By assumption, there is an SOS refutation over the union of the implicit knowledge base and explicit knowledge base $\Delta \cup I_s \cup I_m$, where I_s is the set of support constraints and I_m is the set of expectation constraints:

$$\sigma_0 + \sum_k \sigma_k g_k + \sum_j c_j^+ q_j = -1, \tag{2}$$

where each σ is a sum of squares, $\{g_k\}$ are all the support constraints from the combined knowledge base and $\{q_j\}$ are all the expectation constraints, and the naive norm of the overall expression is $\leq S$.

Note this is a formal polynomial identity, hence under any evaluations of the variables, the identity will hold. Thus, if we take the average over the expression evaluated over m examples, formally we will still have an identity with right hand side equal to -1:

$$\frac{1}{m}\sum_{t=1}^{m}\Big[\sigma_0|_{\vec{\rho}^{(t)}} + \sum_k \sigma_k|_{\vec{\rho}^{(t)}}\, g_k|_{\vec{\rho}^{(t)}} + \sum_j c_j^+\, q_j|_{\vec{\rho}^{(t)}}\Big] = -1.$$

Because I_m is $1 - \delta/2$ testable and each q_j is linear in the expectations $e(v)$, the empirical bounds in B imply the corresponding constraints with enough margin to absorb the Hoeffding radius. Since the full refutation has naive norm at most S, this replacement introduces at most $\frac{1}{2}$ total slack after the sample-size choice above.

Therefore, we can replace the moment bound with the empirical bound up to the slack constant. Hence we still have the expression:

$$\frac{1}{m}\sum_{t=1}^{m}\Big[\sigma_0|_{\vec{\rho}^{(t)}} + \sum_k \sigma_k|_{\vec{\rho}^{(t)}}\, g_k|_{\vec{\rho}^{(t)}} + \sum_{j'} c_{j'}^+\, B_{j'}\Big] = -1 + \frac{1}{2} < 0. \tag{3}$$

By rescaling, we still have a refutation.

Next, by the testability of I_s, with probability at least $1-\delta/2$, all but at most $m/(2S)$ examples witness every support constraint used in the refutation. On the witnessed examples, each restricted constraint satisfies $g_k|_{\vec{\rho}^{(t)}} \geq 0$. On the remaining examples, explicit compactness bounds the contribution of the restricted proof by its naive norm, at most S. Hence we can remove the unwitnessed support-constraint contributions and get

$$\frac{1}{m}\sum_{t=1}^{m}\Big[\sigma_0|_{\vec{\rho}^{(t)}} + \sum_k \sigma_k|_{\vec{\rho}^{(t)}}\, g_k|_{\vec{\rho}^{(t)}} + \sum_{j'} c_{j'}^+\, B_{j'}\Big] = -1 + \frac{m \cdot S}{m \cdot 2S} = -1 + \frac{1}{2} < 0. \tag{4}$$

Rescaling again, we get another refutation. Hence with probability $1 - \delta$ overall, there also exists a SOS refutation over just $\Delta_{\vec{\rho}}$ and the upper lower bounds for the moments, which is the system our algorithm is checking.

Our guarantee for this lifted knowledge base now follows from Theorem 2. □

Example 8. *Continuing our example, the implicit, testable knowledge base from Example 7 first proves the premise of DrugA to end before next trial, i.e., there is a sum-of-square proof* $[0.8e(t) - e(ts)] = 0.8[e(t) - 0.9] + [0.5 - e(ts)] + 0.22$. *Then using the original knowledge base, we get* $\texttt{second_stage}(\text{DrugA}) = 0$*, hence we will refute the hypothesis that every drug will enter Phase II. Moreover, using the more accurate bounds calculated by our algorithm will also lead to the same conclusion as* $[0.8e(t) - e(ts)] = 0.8[e(t) - 1] + [0.44 - e(ts)] + 0.36$.

We now show that, for any fixed SOS degree d and quantifier-rank k, our lifted semidefinite-programming (SDP) based inference algorithm runs in time polynomial in the *bit-complexity* of the input and the number of partial examples. The number of examples affects preprocessing and the number of linear constraints; it does not enter the lifting exponent that determines the PSD matrix dimension.

Theorem 6 (Polynomial Time in Bit Complexity)**.** *Let Δ be an explicit knowledge base whose encoding has bit length B, let φ be a ground query of bit length b, let m be the number of partial examples, and let d and k be the chosen SOS degree and quantifier-rank bounds. Then Algorithm 1 can be compiled into an SDP whose PSD matrix dimension and number of PSD constraints are bounded by*

$$(B + b)^{O(d+k)},$$

and whose number of linear constraints and preprocessing time are bounded by

$$m\,(B + b)^{O(d+k)},$$

up to polynomial factors in the coefficient bit precision. Hence, for fixed d and k, the algorithm runs in time polynomial in $B + b + m$ and the input bit precision. In particular, m contributes through the ordinary input size and empirical linear constraints, but it does not enter the exponent controlling the lifted PSD matrix dimension.

Proof. First fix d and k. Grounding only requires the constants appearing in $\Delta \cup \{\varphi\}$ and k generic names. The number of resulting ground atoms is therefore polynomial in the explicit input size for fixed predicate arities and fixed k. Degree-d monomials over these atoms form the moment vector used by the SOS relaxation, so after quotienting by renaming equivalence the lifted moment vector has size $(B + b)^{O(d+k)}$. The moment and localizing matrices are indexed by this vector, which gives the stated bound on PSD matrix dimension and on the number of PSD constraints.

The m partial examples are used to compute, for every lifted monomial, empirical upper and lower bounds by the calls to LBound and UBound. Each example contributes a polynomial number of such bound computations and linear inequalities, but it does not create a new family of PSD variables. Thus the preprocessing time and number of linear constraints scale by a multiplicative factor m, while the lifted PSD dimension remains independent of m.

Finally, all coefficients from Δ, φ, and the Hoeffding radii have polynomial bit length in the input encoding and the requested confidence precision. Standard bit-complexity bounds for semidefinite and convex optimization [Grötschel et al., 2012, Ghadiri et al., 2023] then give a running time polynomial in the SDP description length. Combining the matrix-size and linear-constraint bounds yields polynomial time in $B + b + m$ for fixed d and k. □

A direct propositional SOS encoding would require distinct moment variables for each ground monomial, yielding an SDP of size $\Omega(|\mathscr{U}|^d)$, where $\mathscr{U}$ is the (possibly infinite) universe. Our double-lifting avoids this blow-up entirely.

6 CONCLUSION AND LIMITATIONS

In this work, we presented a polynomial-time framework for implicitly learning to reason in first-order relational probabilistic logic, advancing beyond previous work limited in pure inference. Our approach unifies incomplete first-order axioms with partial observations through a bounded-degree SOS hierarchy, enabling tractable inference over infinite domains with quantified variables.

Our approach simultaneously handles first-order symmetries and probabilistic semantics, without imposing restrictions on predicate arity or clause length. Because the method avoids reliance on explicit models or independence assumptions, it naturally accommodates complex noise and interdependent variables, with flexibility as a core strength.

The boundedness assumption is required for explicit compactness and for the Hoeffding concentration bounds used to learn expectation constraints from finite samples. In most cases, when we record data in a table, we use an encoding that has a fundamentally bounded range (e.g., 64-bit integer or floating point), and our guarantees hold for the bounded values we obtain in this representation. Replacing an unbounded quantity by such a large finite truncation yields a formal guarantee for the bounded problem, but it can make the guarantee loose: the confidence radius and sample complexity scale with the range parameter S. This reflects a genuine statistical limitation of distribution-free learning, since a variable that is usually small but occasionally extremely large has an expectation that, in general, cannot be bounded from finitely many partial observations.

We do not claim to have presented an empirically scalable system in this paper. However, the practical viability of our approach is relevant for future work, and we emphasize that existing tools, such as SparsePOP [Waki et al., 2006], are capable of solving the kinds of SOS instances required by our approach. SOS solvers more generally remain an area of active research [Huang et al., 2022, Jiang et al., 2023]. The presence of constants and reliance on moderately low-degree SOS proofs may pose challenges for scaling to large, richly relational domains, requiring further algorithmic and engineering advances.

Our polynomial-time framework for combining partial observations with incomplete first-order background knowledge offers reliable decision support in complex domains, where we desire sound inference in the absence of strong independence assumptions, from early-stage biomedical trials to environmental monitoring. By making uncertainty explicit via witnessed constraints and confidence bounds, it also lays the groundwork for risk-aware systems. Although the approach is still theoretical at this stage, our SOS proof certificates could be released to support transparency, reproducibility, and community-driven validation on real-world challenges.

Acknowledgements

We thank the reviewers, area chairs, and program chairs for their constructive feedback and suggestions. Luise Ge is supported by National Science Foundation (IIS-2214141) and Army Research Office (W911NF-25-1-0059). Brendan Juba and Kris Nilsson were supported by NSF award IIS-1942336.

References

Babak Ahmadi, Kristian Kersting, Martin Mladenov, and Sriraam Natarajan. Exploiting symmetries for scaling loopy belief propagation and relational training. *Machine Learning*, 92(1):91–132, 2013.

Damiano Azzolini and Fabrizio Riguzzi. Lifted inference for statistical statements in probabilistic answer set programming. *International Journal of Approximate Reasoning*, 163:109040, 2023.

Vaishak Belle. Open-universe weighted model counting. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pages 3701–3708, 2017.

Vaishak Belle and Brendan Juba. Implicitly learning to reason in first-order logic. In *Advances in Neural Information Processing Systems 32*, pages 3376–3386, 2019.

Rodrigo de Salvo Braz, Eyal Amir, and Dan Roth. Lifted first-order probabilistic inference. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 1319–1325, 2005.

Maurice Bruynooghe, Theofrastos Mantadelis, Angelika Kimmig, Bernd Gutmann, Joost Vennekens, Gerda Janssens, and Luc De Raedt. Problog technology for inference in a probabilistic first order logic. In *ECAI 2010*, pages 719–724. IOS Press, 2010.

David Maxwell Chickering. Learning bayesian networks is np-complete. In *Learning from data: Artificial intelligence and statistics V*, pages 121–130. Springer, 1996.

YooJung Choi, Antonio Vergari, and Guy Van den Broeck. Probabilistic circuits: A unifying framework for tractable probabilistic models. UCLA. URL: http://starai.cs.ucla.edu/papers/ProbCirc20.pdf, 2020.

Andrew Cropper and Sebastijan Dumančić. Inductive logic programming at 30: a new introduction. *Journal of Artificial Intelligence Research*, 74:765–850, 2022.

Luc De Raedt, Angelika Kimmig, and Hannu Toivonen. ProbLog: A probabilistic Prolog and its application in link discovery. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 2462–2467, 2007.

Pedro Domingos and William Webb. A tractable first-order probabilistic logic. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 1902–1909, 2012.

Ivan Donadello, Luciano Serafini, and Artur d'Avila Garcez. Logic tensor networks for semantic image interpretation. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 1596–1602, 2017.

Luise Ge, Brendan Juba, and Kris Nilsson. Polynomial-time relational probabilistic inference in open universes. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)*, pages 9058–9067, 2025.

Lise Getoor, Nir Friedman, Daphne Koller, and Avi Pfeffer. Learning probabilistic relational models. In *Relational data mining*, pages 307–335. Springer, 2001.

Mehrdad Ghadiri, Richard Peng, and Santosh S. Vempala. The bit complexity of efficient continuous optimization. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 2059–2070. IEEE, 2023.

Mehdi Ghasemi, Salma Kuhlmann, and Murray Marshall. Moment problem in infinitely many variables. *Israel Journal of Mathematics*, 212:1012–1012, 2016.

Vibhav Gogate and Pedro Domingos. Probabilistic theorem proving. *Communications of the ACM*, 59(7):107–115, 2016.

Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*. Springer Science & Business Media, 2nd edition, 2012.

Joseph Y. Halpern and Riccardo Pucella. Characterizing and reasoning about probabilistic and non-probabilistic expectation. *Journal of the ACM (JACM)*, 54(3):15–es, 2007.

Baihe Huang, Shunhua Jiang, Zhao Song, Runzhou Tao, and Ruizhe Zhang. Solving sdp faster: A robust ipm framework and efficient implementation. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 233–244. IEEE, 2022.

Tuyen N. Huynh and Raymond J. Mooney. Discriminative structure and parameter learning for Markov logic networks. In *Proceedings of the 25th International Conference on Machine Learning*, pages 416–423. ACM, 2008.

Shunhua Jiang, Bento Natura, and Omri Weinstein. A faster interior-point method for sum-of-squares optimization. *Algorithmica*, 85(9):2843–2884, 2023.

Brendan Juba. Implicit learning of common sense for reasoning. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 939–946, 2013.

Brendan Juba. Polynomial-time probabilistic reasoning with partial observations via implicit learning in probability logics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7866–7875, 2019.

Kristian Kersting. Lifted probabilistic inference. In *ECAI 2012*, pages 33–38. IOS Press, 2012.

Tushar Khot, Sriraam Natarajan, Kristian Kersting, and Jude Shavlik. Gradient-based boosting for statistical relational learning: The Markov logic network and missing data cases. *Machine Learning*, 100(1):75–100, 2015.

Doga Kisa, Guy Van den Broeck, Arthur Choi, and Adnan Darwiche. Probabilistic sentential decision diagrams. In *Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning*, 2014.

Stanley Kok and Pedro Domingos. Learning Markov logic networks using structural motifs. In *Proceedings of the 27th International Conference on Machine Learning*, pages 551–558, 2010.

Gerhard Lakemeyer and Hector J. Levesque. Evaluation-based reasoning with disjunctive information in first-order knowledge bases. In *Eighth International Conference on the Principles of Knowledge Representation and Reasoning*, pages 73–81, 2002.

Jean Bernard Lasserre. Global optimzation with polynomials and the problem of moments. *SIAM J. Optimization*, 11(3):796–817, 2001.

Daniel Lowd and Pedro Domingos. Efficient weight learning for Markov logic networks. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 200–211. Springer, 2007.

Malte Luttermann, Marcel Gehrke, and Tanya Braun. Colour passing revisited: Lifted model construction with commutative factors. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pages 20491–20499, 2024.

Siddharth Malhotra, Davide Bizzaro, and Luciano Serafini. Lifted inference beyond first-order logic. *Artificial Intelligence*, 342:104310, 2026.

Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. DeepProbLog: Neural probabilistic logic programming. 2018.

Giuseppe Marra and Ondřej Kuželka. Neural Markov logic networks. In *Uncertainty in Artificial Intelligence*, pages 908–917. PMLR, 2021.

Loizos Michael. Partial observability and learnability. *Artificial Intelligence*, 174(11):639–669, 2010.

Lilyana Mihalkova and Raymond J. Mooney. Bottom-up learning of Markov logic network structure. In *Proceedings of the 24th International Conference on Machine Learning*, pages 625–632. ACM, 2007.

Ionela G. Mocanu, Vaishak Belle, and Brendan Juba. Polynomial-time implicit learnability in SMT. In *ECAI 2020*, pages 1152–1158. IOS Press, 2020.

Yurii Nesterov. Squared functional systems and optimization problems. *High performance optimization*, 13:405–440, 2000.

Pablo A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, 2000.

David Poole. First-order probabilistic inference. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 985–991, 2003.

Hoifung Poon and Pedro Domingos. Sum-product networks: A new deep architecture. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 689–690. IEEE, 2011.

Mihai Putinar. Positive polynomials on compact semi-algebraic sets. *Indiana U. Math. J.*, 42:969–984, 1993.

Alexander P. Rader, Ionela G. Mocanu, Vaishak Belle, and Brendan Juba. Learning implicitly with noisy data in linear arithmetic. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pages 1410–1417, 2021.

Tahrima Rahman, Prasanna Kothalkar, and Vibhav Gogate. Cutset networks: A simple, tractable, and scalable approach for improving the accuracy of chow-liu trees. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 630–645. Springer, 2014.

Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine learning*, 62:107–136, 2006.

Tim Rocktäschel and Sebastian Riedel. End-to-end differentiable proving. In *Advances in Neural Information Processing Systems*, pages 3788–3800, 2017.

Donald B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

Naum Z. Shor. An approach to obtaining global extremums in polynomial mathematical programming problems. *Cybernetics and Systems Analysis*, 23(5):695–700, 1987.

Timothy Van Bremen and Ondřej Kuželka. Lifted inference with tree axioms. *Artificial Intelligence*, 324:103997, 2023.

Guy Van den Broeck, Nima Taghipour, Wannes Meert, Jesse Davis, and Luc De Raedt. Lifted probabilistic inference by first-order knowledge compilation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pages 2178–2185, 2011.

Jan Van Haaren, Guy Van den Broeck, Wannes Meert, and Jesse Davis. Lifted generative learning of Markov logic networks. *Machine Learning*, 103(1):27–55, 2016.

Hayato Waki, Sunyoung Kim, Masakazu Kojima, and Masakazu Muramatsu. Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity. *SIAM Journal on Optimization*, 17(1):218–242, 2006.