Learning Concepts Definable in First-Order Logic with Counting

Steffen van Bergerem
RWTH Aachen University
vanbergerem@informatik.rwth-aachen.de

***Abstract*—We study classification problems over relational background structures for hypotheses that are defined using logics with counting. The aim of this paper is to find learning algorithms running in time sublinear in the size of the background structure. We show that hypotheses defined by FOCN(P)-formulas over structures of polylogarithmic degree can be learned in sublinear time. Furthermore, we prove that for structures of unbounded degree there is no sublinear learning algorithm for first-order formulas.**

I. Introduction

In this paper, we study Boolean classification problems over relational structures. We consider the relational structure, also called *background structure*, as fixed. For fixed $k \in \mathbb{N}$, the goal is to learn a classification function, called a *hypothesis*, that maps k-ary tuples from the relational structure to Booleans. Given a sequence of training examples, each of which consists of a k-ary tuple from the background structure and a Boolean, we aim to find a hypothesis that is consistent with the examples. In other words, our goal is to learn a description of a k-ary relation that is consistent with a given sequence of positive and negative examples, i.e., the relation contains all positive and no negative examples.

Example I.1. Consider a relational database containing data from an online encyclopedia. The universe of the structure consists of all pages of the encyclopedia. We have a binary relation representing hyperlinks between pages and a unary relation representing category pages. Pages that are not category pages are article pages. Our task is to learn a binary relation containing tuples of pages, where the first element of the tuple is a category page and the second element is a page that belongs to the category. For this we are given a training sequence of classified tuples, e.g. tuples that have been classified by experts beforehand. The goal is to learn the description of a relation that is consistent with the training sequence.

Consider the background structure and the training examples given in Figure 1. A description of a consistent relation would be the following. The relation contains all tuples, where the first element is a category page and the second page is linked from the category page or there is another page linked from the category page and both pages have at least two common linked pages. The corresponding relation can be seen in Figure 2. For example the tuple $(1, 8)$ is contained in the relation since Page 1 is a category, Page 2 is linked from the category, and there are at least two pages (Pages 3, 4, and 5) that both Page 2 and Page 8 link to. Note that the relation is consistent with the training examples.

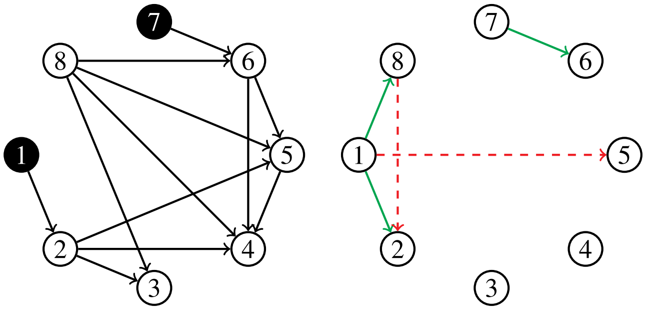

Figure 1. (Left) A background structure viewed as a directed graph. Vertices represent pages in the online encyclopedia, category pages are black and edges represent hyperlinks. (Right) Training examples. Green edges denote positive examples, i.e., the tuple is contained in the relation. Red edges denote negative examples.

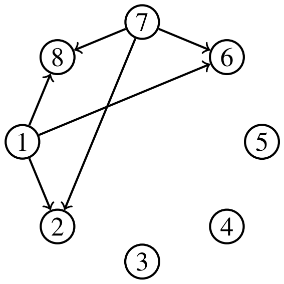

Figure 2. The learned relation from Example I.1.

Since the data are contained in a relational database, it would be convenient to learn an SQL query that defines the relation. Figure 3 shows an SQL query for the learned relation. ⌟

This paper studies the classification problems in the declarative framework that has been introduced in [1], [2], where logics are used to describe the hypotheses. Grohe and Ritzert showed in [2] that learning hypotheses in first-order logic is possible in time polynomial in the number of training examples and the degree of the background structure. For structures of polylogarithmic degree and training sequences of polylogarithmic length, measured in the size of the background structure, this yields a learning algorithm that runs in polylogarithmic time. We are interested in learning hypotheses that can be expressed in SQL. While first-order logic can be seen as the "logical core" of SQL, there are aggregating operators

S. van Bergerem, "Learning Concepts Definable in First-Order Logic with Counting," 2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), 2019, pp. 1-13. DOI: 10.1109/LICS.2019.8785811.

```
SELECT C.'page', CatLink.'to'
FROM Categories C, Links CatLink
WHERE CatLink.'from' = C.'page'
UNION
SELECT C.'page', L1.'from'
FROM Categories C, Links CatLink,
     Links L1, Links L2
WHERE CatLink.'from' = C.'page'
AND CatLink.'to' = L2.'from'
AND L1.'to' = L2.'to'
GROUP BY C.'page', L1.'from'
HAVING count(*) >= 2;
```

Figure 3. An SQL query that defines the relation learned in Example I.1.

in SQL, namely COUNT, AVG, SUM, MIN, and MAX, that do not have corresponding expressions in first-order logic. Motivated by this, we study the logic FOCN(P), which Kuske and Schweikardt introduced in [3] and which extends first-order logic by cardinality conditions similar to the COUNT operator in SQL. The logic depends on a collection of numerical predicates P, i.e., functions $P\colon \mathbb{Z}^m \to \{0,1\}$, that we use in FOCN(P)-formulas to express restrictions on the results of counting terms.

Let $\mathcal{B} = (U(\mathcal{B}), L, C)$ be the background structure from Example I.1, where $U(\mathcal{B})$ is the set of all pages, L is the binary relation of links between pages and C is the unary relation of category pages. The SQL query from Figure 3 can be expressed as the FOCN(P)-formula

$$\varphi(c,p) = Cc \wedge \Big(Lcp \vee \exists x \big(Lcx \wedge \#(y).(Lxy \wedge Lpy) \geqslant 2\big)\Big),$$

where $\geqslant \in$ P. The counting term $\#(y).(Lxy \wedge Lpy)$ counts the number of pages y such that both x and p link to y. The formula $\#(y).(Lxy \wedge Lpy) \geqslant 2$ checks whether this number is at least 2. In a more general approach, we may use the formula

$$\varphi'(c,p;\kappa) = Cc \wedge \Big(Lcp \vee \exists x \big(Lcx \wedge \#(y).(Lxy \wedge Lpy) \geqslant \kappa\big)\Big)$$

with the free number variable κ. When viewed as a parameter, for every assignment of κ we obtain a new hypothesis.

A hypothesis consists of the following components: a formula $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ as well as two parameter tuples $\bar{v}$ in $U(\mathcal{B})^\ell$ and $\bar{\lambda}$ in $\mathbb{Z}^m$. Together, they describe a function $[\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}}\colon U(\mathcal{B})^k \to \{0,1\}$ with

$$[\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}}(\bar{u}) := \begin{cases} 1 & \text{if } \mathcal{B} \models \varphi(\bar{u}\,;\bar{v},\bar{\lambda}) \\ 0 & \text{otherwise.} \end{cases}$$

Given a finite training sequence T of tuples $(\bar{u}, c)$ with $\bar{u} \in U(\mathcal{B})^k$ and $c \in \{0,1\}$, our goal is to learn a hypothesis $(\varphi(\bar{x}\,;\bar{y},\bar{\kappa}), \bar{v}, \bar{\lambda})$ that is consistent with the training examples, i.e., $[\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}}(\bar{u}) = c$ for all training examples $(\bar{u}, c)$. In the context of machine learning, the hypothesis is a *parametric model* and the described learning problem is called *model learning*.

A. Our Results

We study the model-learning problem for FOCN(P) over relational background structures. Instead of random access to the background structure, the algorithms we consider are granted only *local access* (see Section II for details). We measure the complexity of FOCN(P)-formulas in terms of their *binding width* and *binding rank* [3]. The binding width bounds the number of variables that occur in quantifier and counting terms. The binding rank bounds the nesting depth of these constructs. We bound the complexity of first-order formulas in Hanf normal form in terms of their *locality radius*. This allows us to consider only local neighborhoods in the model-checking problem for the given formula. We give a more detailed description of these parameters in Section II.

We give an algorithm for the model-learning problem for FOCN(P) that runs in time polylogarithmic in the size of the background structure, polynomial in the length of the training sequence, and quasipolynomial in the degree of the background structure. The behavior of the algorithm depends on the existence of a *target hypothesis* that is consistent with the training sequence and whose FOCN(P)-formula has binding rank at most r and binding width at most w. Both r and w are parameters of the algorithm. If there is no such hypothesis, then the algorithm may reject the input. Otherwise it always returns a hypothesis. Although the FOCN(P)-formula of the target hypothesis might contain counting terms, the algorithm will only return first-order formulas. This is due to the surprising fact that on a fixed background structure first-order formulas are as expressive as FOCN(P)-formulas. The following theorem is the main result of this paper.

Theorem I.1. *Let $k, \ell, r, w \in \mathbb{N}$. Then there is a learning algorithm $\mathfrak{L}_{con}$ for the k-ary learning problem over some finite relational structure $\mathcal{B}$, that receives a training sequence T and the degree of the structure $\Delta\mathcal{B}$ as input, with the following properties:*

1. *If there are an FOCN(P)-formula $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ of binding rank at most r and binding width at most w and parameter tuples $\bar{v} \in U(\mathcal{B})^\ell$ and $\bar{\lambda} \in \{0,\dots,|\mathcal{B}|\}^{|\bar{\kappa}|}$ such that $[\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}}$ is consistent with T, then $\mathfrak{L}_{con}$ always returns a hypothesis.*
2. *If the algorithm returns a hypothesis H, then H is of the form $(\varphi^*(\bar{x}\,;\bar{y}), \bar{v}^*)$ for some first-order formula $\varphi^*(\bar{x}\,;\bar{y})$ in Hanf normal form with locality radius smaller than $(2w{+}1)^r$ and $\bar{v}^* \in U(\mathcal{B})^\ell$, and $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with the input sequence T of training examples.*
3. *The algorithm runs in time $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d)}$ with only local access to $\mathcal{B}$, where $n := |\mathcal{B}|$, $d := \Delta B$ and $t := |T|$.*
4. *The hypothesis returned by the algorithm can be evaluated in time $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d)}$ with only local access to $\mathcal{B}$.*

When the degree of the background structure and the length of the training sequence are bounded by $\mathrm{polylog}(|\mathcal{B}|)$, then the running time of the algorithm is sublinear in the size of the background structure. Thus we obtain the following result.

Corollary I.2. *There is a consistent model-learning algorithm for FOCN(P)-formulas with only local access on background structures with polylogarithmic degree that runs in sublinear time on training sequences of polylogarithmic length, measured in the size of the background structure.*

Theorem I.1 is a direct generalization of the corresponding theorem for first-order logic due to Grohe and Ritzert [2], albeit with a slightly worse running time that is quasipolynomial in the degree. This generalization is well-motivated by the fact that typical machine-learning models have numerical parameters; our results may be seen as a first step towards including numerical aspects in the declarative framework. While the generalization may seem straightforward at first sight, at least for background structures of small (say, logarithmic) but unbounded degree, it is not obvious that an extension of the first-order result to FOCN(P) holds at all. The reason is that FOCN(P) loses its strong locality properties on structures of unbounded degree. For example, by comparing the degree sequences of the neighbors of nodes one can establish quite complex relations that may range over long distances. Indeed, as shown by Grohe and Schweikardt [4], various algorithmic meta theorems, whose proofs are also based on locality properties, fail when extended from first-order logic to first-order logic with counting. We show that it suffices to consider only FO-formulas to find a consistent hypothesis. However, the quantifier rank of the FO-formulas is polynomial in the degree of the background structure. Hence, a direct application of Grohe's and Ritzert's results [2] would not yield a sublinear-time learning algorithm, since the running time of their algorithm is non-elementary in the quantifier rank.

Thus it is not surprising that, even though our theorem looks similar to the corresponding result for first-order logic, there are significant differences in the proofs. The proof of the first-order result in [2] is based on Gaifman's theorem, but there is no analogue of Gaifman's theorem for the counting logic FOCN(P). Instead, our proof is based on Hanf's theorem. But this causes the technical difficulty that we have to deal with isomorphism types of local neighborhoods in our structures. To be able to do this within the desired time bounds, we apply a recent new graph isomorphism test running in time $n^{\mathrm{polylog}(d)}$ for n-vertex graphs graphs of maximum degree d [5].

In addition to the results for structures of polylogarithmic degree, in Section IV we obtain a *probably approximately correct* (PAC) learning algorithm for structures of bounded degree, i.e., the algorithm returns on most of the training sets (*probably*) a hypothesis that has a small expected error on new examples (*approximately*). One could also say that the hypotheses returned by the algorithm *generalize well* on unseen examples with high probability.

We also investigated learnability on background structures without a degree restriction and obtained the following negative result.

Theorem I.3. *There is no consistent model-learning algorithm for first-order formulas with only local access on background structures with no degree restriction whose running time is sublinear in the size of the background structure.*

B. Related Work

The descriptive framework has been considered in [1], [2], [6]. Grohe and Ritzert [2] showed that first-order formulas are PAC-learnable over background structures of polylogarithmic degree. We did not obtain an analogous result for FOCN(P)-formulas, but we prove PAC-learnability over structures of bounded degree and learnability of consistent hypotheses over structures of polylogarithmic degree. Grohe, Löding and Ritzert [6] obtained learning algorithms for monadic second-order logic over string data.

The framework of inductive logic programming (ILP) is closely related to the framework we consider (see, for example, [7], [8], [9], [10], [11]). One of the main differences is that we encode the background knowledge in a relational structure, whereas in ILP it is represented in a background theory. Furthermore, ILP focuses on first-order logic, whereas in our framework different logics have been considered. Other related learning frameworks in the context of databases can be found in [12], [13].

In [3] Kuske and Schweikardt introduced FOCN(P), which extends first-order logic by counting quantifiers and numerical predicates. The logic generalizes logics like FO(Cnt) from [14] and FO+C from [15]. Our results rely on the fact that Hanf normal forms for FOCN(P) always exist. We use the structure of the normal form to argue that considering FO-formulas for our hypotheses suffices for fixed background structures. Hella et al. studied other aggregating operators in [16].

II. Background from Logics

A. Structures

We only consider finite structures. A *signature* is a finite set σ of *relation symbols* and *constant symbols*. Every relation symbol $R \in \sigma$ has an arity $\mathrm{ar}(R) \in \mathbb{N}$. A signature is called *relational* if it does not contain any constant symbol. A *(σ-)structure* $\mathcal{A}$ consists of a finite set $U(\mathcal{A})$, called the *universe* of $\mathcal{A}$, a relation $R(\mathcal{A}) \subseteq (U(\mathcal{A}))^{\mathrm{ar}(R)}$ for every relation symbol $R \in \sigma$, and an element $c(\mathcal{A})$ for every constant symbol $c \in \sigma$. The *order* of $\mathcal{A}$ is $|\mathcal{A}| := |U(\mathcal{A})|$.

A structure $\mathcal{B}$ is a *substructure* of $\mathcal{A}$, denoted by $\mathcal{B} \subseteq \mathcal{A}$, if $U(\mathcal{B}) \subseteq U(\mathcal{A})$, $R(\mathcal{B}) \subseteq R(\mathcal{A})$ for every relation symbol $R \in \sigma$ and $c(\mathcal{B}) = c(\mathcal{A})$ for every constant symbol $c \in \sigma$. For a relational structure $\mathcal{A}$ and a set $V \subseteq U(\mathcal{A})$, the structure *induced* by $\mathcal{A}$ on V is the structure $\mathcal{A}[V]$ with universe V and $R(\mathcal{A}[V]) = R(\mathcal{A}) \cap V^{\mathrm{ar}(R)}$ for every relation symbol $R \in \sigma$. The *union* of two relational structures $\mathcal{A}$ and $\mathcal{B}$ is the structure $\mathcal{A} \cup \mathcal{B}$ with universe $U(\mathcal{A}) \cup U(\mathcal{B})$ and $R(\mathcal{A} \cup \mathcal{B}) = R(\mathcal{A}) \cup R(\mathcal{B})$ for all $R \in \sigma$. The *intersection* is defined analogously.

The *Gaifman graph* of a σ-structure $\mathcal{A}$ is the graph $G_{\mathcal{A}}$ with vertex set $U(\mathcal{A})$ and an edge between two vertices $u, v \in U(\mathcal{A})$ if there is a relation symbol $R \in \sigma$ and a tuple $(u_1, \ldots, u_{\mathrm{ar}(R)}) \in R(\mathcal{A})$ with $u, v \in \{u_1, \ldots, u_{\mathrm{ar}(R)}\}$. The *degree* of $\mathcal{A}$ is the maximum degree of its Gaifman graph, i.e., the maximum number of neighbors of a vertex in $G_{\mathcal{A}}$.

The *distance* $\operatorname{dist}^{\mathcal{A}}(u,v)$ between two vertices $u,v \in U(\mathcal{A})$ is the length of a shortest path between u and v in $G_{\mathcal{A}}$, and $\operatorname{dist}^{\mathcal{A}}(u,v) = \infty$ if there is no path between u and v. For $r \in \mathbb{N}$, the *r-neighborhood* of a vertex $u \in U(\mathcal{A})$ is the set $N_r^{\mathcal{A}}(u) = \{v \in U(\mathcal{A}) \mid \operatorname{dist}^{\mathcal{A}}(u,v) \leqslant r\}$ and the r-neighborhood of a tuple $\bar{u} = (u_1, \dots, u_k) \in U(\mathcal{A})^k$ is the set $N_r^{\mathcal{A}}(\bar{u}) = \bigcup_{i=1}^{k} N_r^{\mathcal{A}}(u_i)$.

We say that an algorithm has *local access* to a σ-structure $\mathcal{A}$ if it may use queries such as "Is $(u_1, \dots, u_{\operatorname{ar}(R)}) \in R(\mathcal{A})$?" and "Return a list of all neighbors of $u \in U(\mathcal{A})$".

B. First-Order Logic with Counting

We assume that the reader is familiar with first-order logic. Let σ be a relational signature. Let P a countable set of predicate names, $\operatorname{ar}\colon \mathrm{P} \to \mathbb{N}_{\geqslant 1}$ an arity function, and $\llbracket P \rrbracket \subseteq \mathbb{Z}^{\operatorname{ar}(P)}$ the semantics of the predicate name $P \in \mathrm{P}$. Then we call the tuple $(\mathrm{P}, \operatorname{ar}, \llbracket . \rrbracket)$ a numerical predicate collection.

Definition II.1 (FOCN(P) [3]). Let $(\mathrm{P}, \operatorname{ar}, \llbracket . \rrbracket)$ be a numerical predicate collection, and let vars and nvars be disjoint sets of structure variables and number variables, respectively. The set of *formulas* for FOCN(P) is built according to the following rules.

(F1) $x_1 = x_2$ and $R(x_1, \dots, x_{\operatorname{ar}(R)})$ are formulas for $R \in \sigma$ and structure variables $x_1, \dots, x_{\operatorname{ar}(R)} \in \mathsf{vars}$.
(F2) If φ and ψ are formulas, then $\neg\varphi$, $(\varphi \wedge \psi)$, and $(\varphi \vee \psi)$ are also formulas.
(F3) $\exists x\, \varphi$ and $\forall x\, \varphi$ are formulas for $x \in \mathsf{vars}$ and a formula φ.
(F4) If $t_1, \dots, t_{\operatorname{ar}(P)}$ are counting terms and $P \in \mathrm{P}$, then $P(t_1, \dots, t_{\operatorname{ar}(P)})$ is a formula.
(F5) $\exists \kappa\, \varphi$ is a formula for every number variable $\kappa \in \mathsf{nvars}$ and every formula φ.

The set of *counting terms* for FOCN(P) is built according to the following rules.

(C1) $\#\bar{x}.\varphi$ is a counting term for $s \in \mathbb{N}$, $\bar{x} = (x_1, \dots, x_s) \in \mathsf{vars}^s$ pairwise distinct structure variables and a formula φ.
(C2) Every $i \in \mathbb{Z}$ is a counting term.
(C3) If t_1 and t_2 are counting terms, then $(t_1 + t_2)$ and $(t_1 \cdot t_2)$ are also counting terms.
(C4) Every $\kappa \in \mathsf{nvars}$ is a counting term.

Let $\mathcal{I} = (\mathcal{A}, \beta)$ be an interpretation, where $\mathcal{A}$ is a relational structure with universe $U(\mathcal{A})$, and $\beta\colon \mathsf{vars} \cup \mathsf{nvars} \to U(\mathcal{A}) \cup \mathbb{Z}$ with $\beta(x) \in U(\mathcal{A})$ for $x \in \mathsf{vars}$ and $\beta(\kappa) \in \mathbb{Z}$ for $\kappa \in \mathsf{nvars}$.

For $k, \ell \in \mathbb{N}$, $x_1, \dots, x_k \in \mathsf{vars}$, $\kappa_1, \dots, \kappa_\ell \in \mathsf{nvars}$, $a_1, \dots, a_k \in U(\mathcal{A})$, and $\lambda_1, \dots, \lambda_\ell \in \mathbb{Z}$, let $\mathcal{I}\frac{a_1, \dots, a_k}{x_1, \dots, x_k}\frac{\lambda_1, \dots, \lambda_\ell}{\kappa_1, \dots, \kappa_\ell} := (\mathcal{A}, \beta')$ with $\beta'(x_i) := a_i$ for all $i \in [k]$, $\beta'(\kappa_i) := \lambda_i$ for all $i \in [\ell]$, and $\beta'(z) := \beta(z)$ for all $z \in (\mathsf{vars} \cup \mathsf{nvars}) \setminus \{x_1, \dots, x_k, \kappa_1, \dots, \kappa_\ell\}$. Then the semantics for a formula or a counting term is defined as follows.

(F1) $\llbracket x_1 = x_2 \rrbracket^{\mathcal{I}} := 1$ if $\beta(x_1) = \beta(x_2)$, and $\llbracket x_1 = x_2 \rrbracket^{\mathcal{I}} := 0$ otherwise.
$\llbracket R(x_1, \dots, x_{\operatorname{ar}(R)}) \rrbracket^{\mathcal{I}} := 1$ if $(\beta(x_1), \dots, \beta(x_{\operatorname{ar}(R)})) \in R^{\mathcal{A}}$, and $\llbracket R(x_1, \dots, x_{\operatorname{ar}(R)}) \rrbracket^{\mathcal{I}} := 0$ otherwise.
(F2) $\llbracket \neg\varphi \rrbracket^{\mathcal{I}} := 1 - \llbracket \varphi \rrbracket^{\mathcal{I}}$, $\llbracket \varphi \wedge \psi \rrbracket^{\mathcal{I}} := \min\left\{\llbracket \varphi \rrbracket^{\mathcal{I}}, \llbracket \psi \rrbracket^{\mathcal{I}}\right\}$ and $\llbracket \varphi \vee \psi \rrbracket^{\mathcal{I}} := \max\left\{\llbracket \varphi \rrbracket^{\mathcal{I}}, \llbracket \psi \rrbracket^{\mathcal{I}}\right\}$.
(F3) $\llbracket \exists x\, \varphi \rrbracket^{\mathcal{I}} := \max\{\llbracket \varphi \rrbracket^{\mathcal{I}\frac{a}{x}} \mid a \in U(\mathcal{A})\}$ and $\llbracket \forall x\, \varphi \rrbracket^{\mathcal{I}} := \min\{\llbracket \varphi \rrbracket^{\mathcal{I}\frac{a}{x}} \mid a \in U(\mathcal{A})\}$.
(F4) $\llbracket P(t_1, \dots, t_{\operatorname{ar}(P)}) \rrbracket^{\mathcal{I}} := 1$ if $(\llbracket t_1 \rrbracket^{\mathcal{I}}, \dots, \llbracket t_{\operatorname{ar}(P)} \rrbracket^{\mathcal{I}}) \in \llbracket P \rrbracket$ and otherwise $\llbracket P(t_1, \dots, t_{\operatorname{ar}(P)}) \rrbracket^{\mathcal{I}} := 0$.
(F5) $\llbracket \exists \kappa\, \varphi \rrbracket^{\mathcal{I}} := \max\{\llbracket \varphi \rrbracket^{\mathcal{I}\frac{k}{\kappa}} \mid k \in \{0, \dots, |U(\mathcal{A})|\}\}$.
(C1) $\llbracket \#\bar{x}.\varphi \rrbracket^{\mathcal{I}} := \left|\{\bar{a} \in (U(\mathcal{A}))^s \mid \llbracket \varphi \rrbracket^{\mathcal{I}\frac{a_1, \dots, a_s}{x_1, \dots, x_s}} = 1\}\right|$.
(C2) $\llbracket i \rrbracket^{\mathcal{I}} := i$.
(C3) $\llbracket (t_1 + t_2) \rrbracket^{\mathcal{I}} := \llbracket t_1 \rrbracket^{\mathcal{I}} + \llbracket t_2 \rrbracket^{\mathcal{I}}$ and $\llbracket (t_1 \cdot t_2) \rrbracket^{\mathcal{I}} := \llbracket t_1 \rrbracket^{\mathcal{I}} \cdot \llbracket t_2 \rrbracket^{\mathcal{I}}$.
(C4) $\llbracket \kappa \rrbracket^{\mathcal{I}} := \beta(\kappa)$. ⌟

Let φ be an FOCN(P)-formula. The *binding rank* $\operatorname{br}(\varphi)$ of φ is the maximal nesting depth of constructs of the form $\exists x$ and $\forall x$ with $x \in \mathsf{vars}$ and $\#\bar{x}$, where $\bar{x}$ is a tuple in vars. The *binding width* $\operatorname{bw}(\varphi)$ of φ is the maximal arity of an $\bar{x}$ of a term $\#\bar{x}.\psi$ in φ. If φ contains no such term, then $\operatorname{bw}(\varphi) = 1$ if φ contains a quantifier $\exists x$ or $\forall x$ with $x \in \mathsf{vars}$, and $\operatorname{bw}(\varphi) = 0$ otherwise.

C. Types, spheres and sphere formulas

Let $r \geqslant 0$ and $n \geqslant 1$, and let $c_1, c_2, \dots$ be a sequence of pairwise distinct constant symbols. An *r-type* (with *n centers*) is a structure $\tau = (\mathcal{A}, a_1, \dots, a_n)$ over the signature $\sigma \uplus \{c_1, \dots, c_n\}$, where $\mathcal{A}$ is a σ-structure with $a_1, \dots, a_n \in U(\mathcal{A})$ and $U(\mathcal{A}) = N_r^{\mathcal{A}}(a_1, \dots, a_n)$. The elements $a_1, \dots, a_n$ are the *centers* of τ. For every tuple $\bar{a} \in (U(\mathcal{A}))^n$, the *$r$-sphere of $\bar{a}$ in $\mathcal{A}$* is the r-type with n centers

$$\mathcal{N}_r^{\mathcal{A}}(\bar{a}) = (\mathcal{A}[N_r^{\mathcal{A}}(\bar{a})], \bar{a}).$$

Let τ be an r-type. A first-order formula $\operatorname{sph}_\tau(\bar{x})$ is a *sphere-formula* if for every σ-structure $\mathcal{B}$ and every tuple $\bar{b} \in U(\mathcal{B})^n$ we have

$$\mathcal{B} \models \operatorname{sph}_\tau(\bar{b}) \iff \mathcal{N}_r^{\mathcal{B}}(\bar{b}) \cong \tau.$$

The *locality radius* of $\operatorname{sph}_\tau(\bar{x})$ is r.

D. Numerical oc-type conditions and hnf-formulas

A *basic counting term* is a counting term of the form $\#(x).\operatorname{sph}_\tau(x)$, where x is a structure variable in vars and τ is an r-type with a single center. The radius r is called the *locality radius* of the basic counting term. In a σ-structure $\mathcal{A}$, a basic counting term evaluates to the number of elements $a \in U(\mathcal{A})$ with $\mathcal{N}_r^{\mathcal{A}}(a) \cong \tau$, i.e., the number of r-neighborhoods with one center in $\mathcal{A}$ that are isomorphic to τ.

A *numerical condition on occurrences of types with one center* (or *numerical oc-type condition*) is an FOCN(P)-formula that is built from basic counting terms and rules (F2), (F4), (F5), (C2), (C3), and (C4), i.e., using number variables and integers, and combining them by addition, multiplication, numerical predicates from $P \cup \{P_\exists\}$, Boolean combinations, and quantification of number variables. Its locality radius is the maximal locality radius of the involved basic counting terms.

Numerical oc-type conditions do not have any free structure variables.

A formula is in *Hanf normal form for* FOCN(P) or an *hnf-formula for* FOCN(P) if it is a Boolean combination of numerical oc-type conditions and sphere formulas. The locality radius of an hnf-formula is the maximal locality radius of the involved conditions and formulas.

E. Local hnf-formulas

Let $\mathcal{A}$ be a relational structure over σ, $\bar{u} \in U(\mathcal{A})^k$ for some $k \in \mathbb{N}$ and $r \in \mathbb{N}$. Then the *local hnf-formulas (for FOCN(P)) of $\bar{u}$ with locality radius smaller than r in $\mathcal{A}$* are

$$\begin{aligned}\mathrm{lhf}_r(\mathcal{A}, \bar{u}) = \{\varphi(\bar{x}) \text{ hnf-formula} \mid \mathcal{A} \models \varphi(\bar{u}),\\ \text{locality radius of } \varphi \text{ is smaller than } r\}.\end{aligned}$$

The following results help us to show that reduced formula and parameter spaces suffice to find consistent hypotheses. The first lemma states that two tuples satisfy the same local hnf-formulas if their neighborhoods are isomorphic.

Lemma II.2. *Let $\mathcal{A}$ be a structure over a relational signature σ, $k, r \in \mathbb{N}$ and $\bar{u}, \bar{u}' \in U(\mathcal{A})^k$. If $\mathcal{N}_r^{\mathcal{A}}(\bar{u}) \cong \mathcal{N}_r^{\mathcal{A}}(\bar{u}')$, then $\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}) = \mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}')$.*

Proof: Let $\varphi(\bar{x})$ be an hnf-formula with locality radius at most r. Then φ is a Boolean combination of numerical oc-type conditions and sphere formulas with locality radius at most r. The numerical oc-type conditions do not have any free structure variables and are thus independent from the assignment for $\bar{x}$. The free variables of the sphere formulas are a subset of $\mathrm{free}(\varphi)$. Let $\mathrm{sph}_\tau(x_{i_1}, \ldots, x_{i_\ell})$ be a sphere-formula used in φ with $x_{i_1}, \ldots, x_{i_\ell} \in \mathrm{free}(\varphi)$ and τ an r'-type with ℓ centers for $r' \leqslant r$. Then $\mathcal{N}_r^{\mathcal{A}}(u_{i_1}, \ldots, u_{i_\ell}) \cong \mathcal{N}_r^{\mathcal{A}}(u'_{i_1}, \ldots, u'_{i_\ell})$ and $\mathcal{N}_{r'}^{\mathcal{A}}(u_{i_1}, \ldots, u_{i_\ell}) \cong \mathcal{N}_{r'}^{\mathcal{A}}(u'_{i_1}, \ldots, u'_{i_\ell})$ and hence

$$\begin{aligned} & \mathcal{A} \models \mathrm{sph}_\tau(u_{i_1}, \ldots, u_{i_\ell}) \\ \iff & \mathcal{N}_{r'}^{\mathcal{A}}(\bar{u}) \models \mathrm{sph}_\tau(u_{i_1}, \ldots, u_{i_\ell}) \\ \iff & \mathcal{N}_{r'}^{\mathcal{A}}(\bar{u}') \models \mathrm{sph}_\tau(u'_{i_1}, \ldots, u'_{i_\ell}) \\ \iff & \mathcal{A} \models \mathrm{sph}_\tau(u'_{i_1}, \ldots, u'_{i_\ell}). \end{aligned}$$

This holds for all sphere formulas in φ. Thus $\mathcal{A} \models \varphi(\bar{u})$ if and only if $\mathcal{A} \models \varphi(\bar{u}')$. □

The following result is a local variant of the Feferman-Vaught Theorem [17] translated to our context. It allows us to analyze the parameters we choose by splitting them into two parts with disjoint neighborhoods.

Lemma II.3 (Local Composition Lemma)**.** *Let $\mathcal{A}, \mathcal{A}'$ be structures over a relational signature σ, $\bar{u} \in U(\mathcal{A})^k$, $\bar{v} \in U(\mathcal{A})^\ell$, $\bar{u}' \in U(\mathcal{A}')^k$, $\bar{v}' \in U(\mathcal{A}')^\ell$, and $r \in \mathbb{N}$, such that $N_r^{\mathcal{A}}(\bar{u}) \cap N_r^{\mathcal{A}}(\bar{v}) = N_r^{\mathcal{A}'}(\bar{u}') \cap N_r^{\mathcal{A}'}(\bar{v}') = \emptyset$, $\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}) = \mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}')$ and $\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{v}) = \mathrm{lhf}_{r+1}(\mathcal{A}', \bar{v}')$. Then*

$$\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}\bar{v}) = \mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}'\bar{v}').$$

Proof: All hnf-sentences φ with locality radius at most r that satisfy $\mathcal{A} \models \varphi$ are contained in $\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u})$ and thus also in $\mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}')$. Hence $\mathcal{A}$ and $\mathcal{A}'$ model the same hnf-sentences with locality radius at most r.

We know that the sphere-formula $\mathrm{sph}_{\mathcal{N}_r^{\mathcal{A}}(\bar{u})}(\bar{x})$ is contained in $\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}) = \mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}')$, so $\mathcal{N}_r^{\mathcal{A}}(\bar{u}) \cong \mathcal{N}_r^{\mathcal{A}'}(\bar{u}')$. Analogously it follows that $\mathcal{N}_r^{\mathcal{A}}(\bar{v}) \cong \mathcal{N}_r^{\mathcal{A}'}(\bar{v}')$. Since $N_r^{\mathcal{A}}(\bar{u}) \cap N_r^{\mathcal{A}}(\bar{v}) = \emptyset$ and $N_r^{\mathcal{A}'}(\bar{u}') \cap N_r^{\mathcal{A}'}(\bar{v}') = \emptyset$, we also obtain $\mathcal{N}_r^{\mathcal{A}}(\bar{u}\bar{v}) \cong \mathcal{N}_r^{\mathcal{A}'}(\bar{u}'\bar{v}')$. Thus for all sphere formulas $\psi(x_1, \ldots, x_m)$ with locality radius at most r, we know that $\mathcal{A} \models \psi(w_1, \ldots, w_m)$ for $w_1, \ldots, w_m \in \bar{u} \cup \bar{v}$ if and only if $\mathcal{A}' \models \psi(w'_1, \ldots, w'_m)$.

Let $\varphi \in \mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}\bar{v})$. Then φ is a Boolean combination of hnf-sentences and sphere formulas. These hnf-sentences and sphere formulas hold for $(\mathcal{A}, \bar{u}\bar{v})$ if and only if they hold for $(\mathcal{A}', \bar{u}'\bar{v}')$. Thus $\varphi \in \mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}'\bar{v}')$ and $\mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}\bar{v}) \subseteq \mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}'\bar{v}')$. Analogously we can show that $\mathrm{lhf}_{r+1}(\mathcal{A}', \bar{u}'\bar{v}') \subseteq \mathrm{lhf}_{r+1}(\mathcal{A}, \bar{u}\bar{v})$. □

For $d \in \mathbb{N}$, two formulas φ, φ' are called *d-equivalent* if $[\![\varphi]\!]^{\mathcal{I}} = [\![\varphi']\!]^{\mathcal{I}}$ for all interpretations $\mathcal{I} = (\mathcal{A}, \beta)$ for all structures $\mathcal{A}$ of degree at most d. The following result is due to Kuske and Schweikardt [3].

Theorem II.4. *Let $(P, \mathrm{ar}, [\![.]\!])$ be a numerical predicate collection. For any relational signature σ, any degree bound $d \in \mathbb{N}$, and any FOCN(P)[σ]-formula φ, there exists a d-equivalent hnf-formula ψ for FOCN(P)[σ] of locality radius smaller than $(2\ \mathrm{bw}(\varphi) + 1)^{\mathrm{br}(\varphi)}$ with $\mathrm{free}(\psi) = \mathrm{free}(\varphi)$.*

Using this result, we can show that a formula behaves the same for two different assignments of structure variables if the local hnf-formulas of the assigned tuples are the same.

Lemma II.5. *Let $\mathcal{A}$ be a structure over a relational signature σ, $\varphi(\bar{x}, \bar{\kappa})$ an FOCN(P)-formula, $\bar{u}, \bar{u}' \in U(\mathcal{A})^{|x|}$ and $\bar{\lambda} \in \{0, \ldots, |\mathcal{A}|\}^{|\kappa|}$. If*

$$\mathrm{lhf}_{(2\ \mathrm{bw}(\varphi)+1)^{\mathrm{br}(\varphi)}}(\mathcal{A}, \bar{u}) = \mathrm{lhf}_{(2\ \mathrm{bw}(\varphi)+1)^{\mathrm{br}(\varphi)}}(\mathcal{A}, \bar{u}'),$$

then

$$\mathcal{A} \models \varphi(\bar{u}, \bar{\lambda}) \iff \mathcal{A} \models \varphi(\bar{u}', \bar{\lambda}).$$

Proof: Let $\varphi'(\bar{x}) := \varphi(\bar{x}, \bar{\lambda})$, $r := \mathrm{br}(\varphi) = \mathrm{br}(\varphi')$ and $w := \mathrm{bw}(\varphi) = \mathrm{bw}(\varphi')$. Using Theorem II.4 we obtain an hnf-formula ψ with locality radius smaller than $(2w+1)^r$ that is $\Delta\mathcal{A}$-equivalent to φ' and, just like φ', doesn't have any free number variables. Then

$$\begin{aligned} \mathcal{A} \models \varphi(\bar{u}, \bar{\lambda}) \quad & \iff \quad \mathcal{A} \models \psi(\bar{u}) \\ & \iff \quad \psi \in \mathrm{lhf}_{(2w+1)^r}(\mathcal{A}, \bar{u}) \\ & \iff \quad \psi \in \mathrm{lhf}_{(2w+1)^r}(\mathcal{A}, \bar{u}') \\ & \iff \quad \mathcal{A} \models \psi(\bar{u}') \\ & \iff \quad \mathcal{A} \models \varphi(\bar{u}', \bar{\lambda}). \end{aligned}$$

□

III. Learning FOCN(P)-Definable Concepts over Structures of Polylogarithmic Degree

In this section, we prove Theorem I.1 and a variant of it, as well as negative results, including Theorem I.3. Let $k, \ell, t, r, w \in \mathbb{N}$ be fixed, $\bar{x} := (x_1, \ldots, x_k)$ instance variables, $\bar{y} := (y_1, \dots, y_\ell)$ parameter variables, $\mathcal{B}$ a background structure over a relational signature σ and $\mathcal{T} := (U(\mathcal{B})^k \times \{0,1\})^t$ training sequences. For $s \in \mathbb{N}$ and $T \in \mathcal{T}$, $T = ((\bar{u_1}, c_1), \dots, (\bar{u_t}, c_t))$, let $N_s^{\mathcal{B}}(T) := \bigcup_{i=1}^t N_s^{\mathcal{B}}(\bar{u}_i)$. Let Φ be the set of FOCN(P)-formulas $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ with binding width at most w, binding rank at most r and free number variables $\bar{\kappa}$, and

$$\mathcal{C} := \{[\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}} \mid \varphi(\bar{x}\,;\bar{y},\bar{\kappa}) \in \Phi,\ \bar{v} \in U(\mathcal{B})^\ell,\\ \bar{\lambda} \in \{0, \dots, |\mathcal{B}|\}^{|\bar{\kappa}|}\}.$$

To prove Theorem I.1, we give an algorithm that uses brute force. Given a training sequence that is consistent with some hypothesis in $\mathcal{C}$, the algorithm finds a consistent hypothesis consisting of a first-order formula and a parameter tuple $\bar{v} \in U(\mathcal{B})^\ell$. The following lemma states that it suffices to search in a reduced parameter space and to check a single formula per parameter. This enables us to find a consistent hypothesis with a sufficient running time.

Lemma III.1. *Let $T = ((\bar{u}_1, c_1), \dots, (\bar{u}_t, c_t)) \in \mathcal{T}$ be consistent with some $C \in \mathcal{C}$. Then there is a tuple $\bar{v}^* = (v_1, \dots, v_\ell) \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^\ell$ and some $m \leqslant \ell$ such that $[\![\varphi^*(\bar{x}, \bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T for*

$$\varphi^*(\bar{x}\,;\bar{y}) := \bigvee_{i \in [t],\, c_i = 1} \vartheta_i(\bar{x}\,;\bar{y}^\circ),$$

$\vartheta_i(\bar{x}\,;\bar{y}^\circ) := \mathrm{sph}_{\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}^\circ)}(\bar{x}\bar{y}^\circ)$, $v^\circ := (v_1, \dots, v_m)$, and $\bar{y}^\circ := (y_1, \dots, y_m)$.

Proof: Let $\varphi(\bar{x}\,;\bar{y},\bar{\kappa}) \in \Phi$ and $\bar{v} = (v_1, \dots, v_\ell) \in U(\mathcal{B})^\ell$, $\bar{\lambda} \in \{0, \dots, |\mathcal{B}|\}^{\bar{\kappa}}$ such that $C = [\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}}$ is consistent with T. Now define for some $m \leqslant \ell$ the elements $v^{(1)}, \dots, v^{(m)} \in \{v_1, \dots, v_\ell\}$ and the sets $N^{(0)}, \dots, N^{(m)} \subseteq U(\mathcal{B})$ by setting $N^{(0)} := N^{\mathcal{B}}_{(2w+1)^r-1}(T)$ and defining the rest as follows: Given $N^{(i)}$, if there is some $v \in \{v_1, \dots, v_\ell\} \setminus \{v^{(1)}, \dots, v^{(i)}\}$ such that $\mathrm{dist}^{\mathcal{B}}(v, N^{(i)}) \leqslant (2w+1)^r$, then let $v^{(i+1)} := v$. Choose arbitrarily if there is more than one. Let $N^{(i+1)} := N^{(i)} \cup N^{\mathcal{B}}_{(2w+1)^r-1}(v^{(i+1)})$. If there is no such v, then set $m := i$ and stop.

Let $N^\circ := N^{(m)}$ and w.l.o.g. let $v^{(i)} = v_i$ for $i \in [m]$. Let $\bar{v}^\circ := (v_1, \dots, v_m)$ and $\bar{v}^\bullet := (v_{m+1}, \dots, v_\ell)$. Then $\bar{v}^\circ \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^m$ and

$$N^\circ = \bigcup_{i=1}^t N^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i) \cup \bigcup_{i=1}^m N^{\mathcal{B}}_{(2w+1)^r-1}(v_i) \tag{1}$$

and

$$N^{\mathcal{B}}_{(2w+1)^r-1}(\bar{v}^\bullet) \cap N^\circ = \emptyset. \tag{2}$$

Claim 1. *Let $i, j \in [t]$ such that $\mathcal{B} \models \vartheta_i(\bar{u}_j\,;\bar{v}^\circ)$. Then $c_i = c_j$.*

Proof: From $\mathcal{B} \models \vartheta_i(\bar{u}_j\,;\bar{v}^\circ)$ it follows that $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}^\circ) \cong \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_j\bar{v}^\circ)$. Using Lemma II.2 we obtain $\mathrm{lhf}_{(2w+1)^r}(\mathcal{B}, \bar{u}_i) = \mathrm{lhf}_{(2w+1)^r}(\mathcal{B}, \bar{u}_j)$. Furthermore, from Equation (1) and Equation (2) it follows that $N^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}^\circ) \cap N^{\mathcal{B}}_{(2w+1)^r-1}(\bar{v}^\bullet) = \emptyset$ and $N^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_j\bar{v}^\circ) \cap N^{\mathcal{B}}_{(2w+1)^r-1}(\bar{v}^\bullet) = \emptyset$. Thus with Lemma II.3 we obtain

$$\mathrm{lhf}_{(2w+1)^r}(\mathcal{B}, \bar{u}_i\bar{v}) = \mathrm{lhf}_{(2w+1)^r}(\mathcal{B}, \bar{u}_j\bar{v}).$$

and hence with Lemma II.5 we get

$$\mathcal{B} \models \varphi(\bar{u}_i\,;\bar{v},\bar{\lambda}) \iff \mathcal{B} \models \varphi(\bar{u}_j\,;\bar{v},\bar{\lambda}).$$

This implies $c_i = c_j$. $\lrcorner$

Set

$$\bar{v}^* := (v_1, \dots, v_m, \underbrace{v, \dots, v}_{\ell - m \text{ times}})$$

for some arbitrary $v \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)$. Then $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^\ell$. It remains to show that $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T. If $\mathcal{B} \models \varphi^*(\bar{u}_i\,;\bar{v}^*)$, then there is some $p \in [t]$ with $c_p = 1$ and $\mathcal{B} \models \vartheta_p(\bar{u}_i\,;\bar{v}^*)$. Using the claim it follows that $c_i = c_p = 1$. On the other hand, if $c_i = 1$, then $\mathcal{B} \models \varphi^*(\bar{u}_i\,;\bar{v}^*)$. Thus $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T. □

```
Algorithm $\mathfrak{L}_{con}$
Input: Training sequence $T = ((\bar{u}_1, c_1)), \dots, (\bar{u}_t, c_t)) \in \mathcal{T}$,
    $d = \Delta\mathcal{B}$, local access to background structure $\mathcal{B}$
N ← $N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)$
for all $\bar{v}^* \in N^m,\ m \leqslant \ell$ do
    for $i = 1, \dots, t$ do
        $\mathcal{N}_i \leftarrow \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}^*)$
        $\vartheta_i(\bar{x}\,;\bar{y}) \leftarrow \mathrm{sph}_{\mathcal{N}_i}(\bar{x}\bar{y})$        ▷ a $d$-bounded
    $[(2w+1)^r - 1]$-type with $k + m$ centers
    $\varphi^*(\bar{x}\,;\bar{y}) \leftarrow \bigvee_{i \in [t],\, c_i = 1} \vartheta_i(\bar{x}\,;\bar{y})$
    consistent ← true
    for $i \in [t]$ with $c_i = 0$ do
        for $j \in [t]$ with $c_j = 1$ do
            if $\mathcal{N}_i \cong \mathcal{N}_j$ then
                consistent ← false
    if consistent then
        return $(\varphi^*(\bar{x}\,;\bar{y}),\ \bar{v}^*)$
reject
```

Figure 4. Learning algorithm $\mathfrak{L}_{con}$ of Theorem I.1

In our algorithm, we have to compare isomorphism types of local neighborhoods. To do this within the desired time bounds, we apply the following result due to Grohe, Neuen and Schweitzer [5].

Theorem III.2. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two σ-structures with $n := \max\{|\mathcal{A}_1|, |\mathcal{A}_2|\}$, $d := \max\{\Delta\mathcal{A}_1, \Delta\mathcal{A}_2\}$ and $m := \max_{R \in \sigma} \mathrm{ar}(R)$. One can check whether $\mathcal{A}_1 \cong \mathcal{A}_2$ in time $n^{\mathcal{O}(m \cdot (\log d)^c)}$ for some constant c.*

A. Consistent Hypotheses

We now prove the main theorem.

Proof of Theorem I.1: The pseudocode for our algorithm is shown in Figure 4. Because of Lemma III.1 the algorithm satisfies Condition (1). Let $(\varphi^*(\bar{x}\,;\bar{y}), \bar{v}^*)$ be the hypothesis returned by the algorithm. Note that $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*) \models \varphi^*(\bar{u}, \bar{v}^*)$ iff $\mathcal{B} \models \varphi^*(\bar{u}, \bar{v}^*)$ for all $(\bar{u}, c) \in T$, because φ^* is a disjunction of sphere formulas of locality radius smaller than $(2w+1)^r$. For $(\bar{u}_i, 1) \in T$ we know by construction that $\mathcal{N}_i \models \varphi^*(\bar{u}_i\,; \bar{v}^*)$. For all $(\bar{u}_i, 0) \in T$ the algorithm checks that there is no $(\bar{u}_j, 1) \in T$ with $\mathcal{N}_i \cong \mathcal{N}_j$ and thus $\mathcal{N}_i \not\models \varphi^*(\bar{u}_i\,; \bar{v}^*)$. Hence the hypothesis is consistent with the input sequence and the algorithm satisfies Condition (2) of the theorem.

Algorithm $\mathfrak{L}_{con}$ computes $\mathcal{N}_i = \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}^*)$ for all $i \in [t]$. $|\mathcal{N}_i| \leqslant (k+\ell) \cdot d^{(2w+1)^r}$ and the representation size is in $(\log n + d)^{\mathcal{O}(1)}$, because k, ℓ, r and w are constant. Each sphere-formula can be computed in time $(\log n \cdot (k+\ell) \cdot d^{(2w+1)^r})^{\mathcal{O}(\|\sigma\|)} = (\log n + d)^{\mathcal{O}(1)}$. ([3]) Thus φ^* can be computed in time $(\log n + d + t)^{\mathcal{O}(1)}$.

When checking whether φ^* is consistent with T, we check whether $\mathcal{N}_i$ and $\mathcal{N}_j$ are isomorphic. Using Theorem III.2 a single isomorphism test takes time

$$
\begin{aligned}
&(\max\{|\mathcal{N}_i|\,, |\mathcal{N}_j|\})^{\mathcal{O}(\max_{R\in\sigma} \operatorname{ar}(R)\cdot(\log\max\{\Delta\mathcal{N}_i,\Delta\mathcal{N}_j\})^c)}\\
\leqslant\ &\Big((k+\ell)\cdot d^{(2w+1)^r}\Big)^{\mathcal{O}((\log d)^c)} \leqslant d^{\mathcal{O}((\log d)^c)}
\end{aligned}
$$

Before checking whether two substructures are isomorphic, we rename the vertices in the substructure such that the representation size no longer depends on $\log n$, but on the size of the substructures. This can be done in time $(\log n)^{\mathcal{O}(1)}$. Hence the consistency of the formula can be checked in time $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathcal{O}((\log d)^c)}$.

The maximum number of iterations in the outer loop is $\sum_{m=1}^{\ell} |N|^m \leqslant \ell \cdot (2tkd^{2\ell(2w+1)^r})^\ell \in (t+d)^{\mathcal{O}(1)}$ and N can be computed with only local access to $\mathcal{B}$ in time $(t+d)^{\mathcal{O}(1)}$. All in all the running time of the algorithm is in $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathcal{O}((\log d)^c)}$. This shows that the algorithm satisfies Condition (3).

To evaluate the formula returned by the algorithm for $\bar{u}$, we only have to evaluate it using the structure $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u})$ with only local access to $\mathcal{B}$, so the running time is in $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathcal{O}((\log d)^c)}$ and Condition (4) is satisfied. □

If we consider k, ℓ, r and w as part of the input, then a more thorough analysis shows that the running time of the algorithm is exponential in $\operatorname{polylog}(d)\log(k)\ell^2(2w+1)^r + \ell\cdot\log(t)$ and polylogarithmic in n.

Theorem I.3 stated that there is no consistent model-learning algorithm for first-order formulas on background structures with no degree restriction that runs in sublinear time with only local access. In the proof we use the fact that an algorithm is unable to see all vertices of the background structure in sublinear time.

Proof of Theorem I.3: Let $k, \ell = 1$ and consider undirected graphs G_{ij} of size n for $i \in \{1,2\}$ and $1 \leqslant j \leqslant t$, where t is an even integer. Let G_{11} and G_{21} be graphs with no edges and G_{12} and G_{22} be graphs with a single edge. For $j \geqslant 3$ let G_{ij} be the empty graph iff $(j-i)$ is even and let G_{ij} be the graph with a single edge else.

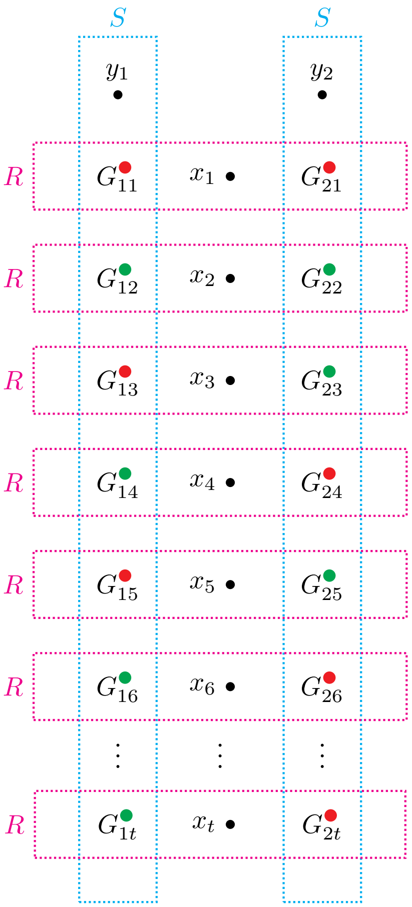

Figure 5. The background structure $\mathcal{B}$ in the proof of Theorem I.3. The red dot marks graphs with no edges. The green dot marks graphs with a single edge.

Define the background structure $\mathcal{B}$ for the relational signature $\sigma = \{E, R, S\}$ by

$$U(\mathcal{B}) := \{x_1, \dots, x_t, y_1, y_2\} \biguplus_{\substack{i\in\{1,2\},\\ 1\leqslant j\leqslant t}} V(G_{ij}),$$

$$E^{\mathcal{B}} := \biguplus_{\substack{i\in\{1,2\},\\ 1\leqslant j\leqslant t}} E(G_{ij}),$$

$$R^{\mathcal{B}} := \{(x_j, v) \mid 1 \leqslant j \leqslant t, v \in G_{ij} \text{ for some } i \in \{1,2\}\},$$

and

$$S^{\mathcal{B}} := \{(y_i, v) \mid i \in \{1,2\}, v \in G_{ij} \text{ for some } 1 \leqslant j \leqslant t\}.$$

The background structure can be seen in Figure 5. The size of the background structure is $|\mathcal{B}| = t \cdot (2\,|G_{11}| + 1) + 2$ and thus linear in the size of the graphs $|G_{ij}|$. Let

$$\varphi(x\,; y) := \exists v_1 \exists v_2\ Rxv_1 \wedge Rxv_2 \wedge Syv_1 \wedge Syv_2 \wedge Ev_1v_2$$

and consider training examples $(x_1, c_{i1}), \dots, (x_t, c_{it})$ with $c_{ij} = [\![\varphi(x_j\,; y_i)]\!]^{\mathcal{B}}$ for $i \in \{1,2\}$ and $1 \leqslant j \leqslant t$. The formula

$\varphi(x_j\,; y_i)$ is evaluated to 1 if and only if G_{ij} contains an edge. Define two training sequences

$$\begin{aligned} T_1 \coloneqq &((x_1, c_{11}), \dots, (x_t, c_{1t})) \\ =&((x_1, 0), (x_2, 1), (x_3, 0), (x_4, 1), \\ &(x_5, 0), (x_6, 1), \dots, (x_{t-1}, 0), (x_t, 1)) \text{ and} \\ T_2 \coloneqq &((x_1, c_{21}), (x_2, c_{22}), (x_4, c_{24}), (x_3, c_{23}), \\ &(x_6, c_{26}), (x_5, c_{25}), \dots, (x_t, c_{2t}), (x_{t-1}, c_{2(t-1)})) \\ =&((x_1, 0), (x_2, 1), (x_4, 0), (x_3, 1), \\ &(x_6, 0), (x_5, 1), \dots, (x_t, 0), (x_{t-1}, 1)). \end{aligned}$$

Let $\mathfrak{L}$ be a sublinear deterministic model-learning algorithm and run it on the training sequences T_1 and T_2. Then there is some vertex order such that $\mathfrak{L}$ is unable to find even a single edge from E. In both sequences the algorithm receives the same sequence of Booleans and all visited vertices have a single R-edge, a single S-edge and no E-edge. Thus both training sequences are indistinguishable for the learner and hence it has to return the same formula $\psi(x\,; y) \in \mathrm{FO}$ for both sequences T_1 and T_2.

The learner $\mathfrak{L}$ can distinguish two different vertex sets $\{x_j\} \cup V(G_{1j}) \cup V(G_{2j})$ and $\{x_{j'}\} \cup V(G_{1j'}) \cup V(G_{2j'})$ only by the values of c_{ij} and $c_{ij'}$. Hence, by choosing a suitable vertex ordering, we can assume that the algorithm chooses the parameters only from the set $\{y_1, y_2, x_1, x_2\} \cup V(G_{11}) \cup V(G_{12}) \cup V(G_{21}) \cup V(G_{22})$. Since T_1 and T_2 are indistinguishable, it has to choose the same parameter for T_1 and T_2, so $v_1 = v_2$.

But then $\mathfrak{L}$ can't be consistent with both T_1 and T_2, because $c_{13} = 0 \neq 1 = c_{23}$ and both values would have to be equal to $[\![\psi(x_3, v_1)]\!]^{\mathcal{B}} = [\![\psi(x_3, v_2)]\!]^{\mathcal{B}}$. □

Although we do not give a formal introduction into *probably approximately correct* (PAC) learning, we would like to mention that one can easily extend Theorem I.3 analogously to Grohe, Löding and Ritzert [6] to show that there is no sublinear model-learning algorithm for first-order formulas, that is a PAC-learning algorithm, on background structures with no degree restriction and only local access.

In addition to model learning, Grohe and Ritzert [2] also considered *parameter learning*, where we assume a fixed formula and we only want to find parameters such that the resulting hypothesis is consistent with the training examples.

Grohe and Ritzert showed that parameter learning is not possible in sublinear time with only local access. Here we prove a stronger result that shows that parameter learning is at least as hard as solving q-CLIQUE.

Theorem III.3. *If the exponential-time hypothesis (ETH) holds, then there is no consistent parameter-learning algorithm for first-order formulas φ of quantifier rank at most q on background structures $\mathcal{B}$ with no degree restriction running in time $f(q) \cdot |\mathcal{B}|^{o(q)}$ for some function f, i.e. that, given φ and a sequence of training examples T, returns a tuple $\bar{v}$ such that $[\![\varphi(\bar{x}\,; \bar{v})]\!]^{\mathcal{B}}$ is consistent with all training examples.*

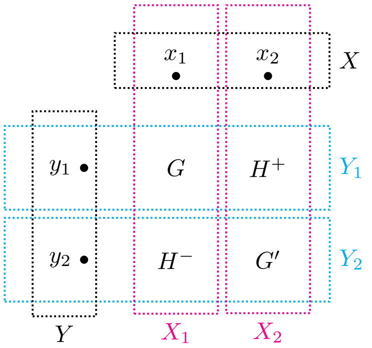

Figure 6. The background structure $\mathcal{B}$ in the proof of Theorem III.3.

Proof: Let $q \in \mathbb{N}$. For the background structure let G be a graph of size $|G| \gg q$, G' a copy of G and H_+ and H_- graphs of size $|G|$ such that H_+ has a q-clique and H_- does not, e.g. choose $H_+ = K_q \uplus \overline{K}_{|G|-q}$ and $H_- = \overline{K}_{|G|}$ where K_s and $\overline{K}_s$ are the complete graph and the empty graph on s vertices for $s \in \mathbb{N}$.

Define the background structure $\mathcal{B}$ for the relational signature $\sigma \coloneqq \{E, X, Y, X_1, X_2, Y_1, Y_2\}$ by

$$\begin{aligned} U(\mathcal{B}) &\coloneqq \{x_1, x_2, y_1, y_2\} \uplus V(G) \uplus V(G') \\ &\quad \uplus V(H_+) \uplus V(H_-), \\ E(\mathcal{B}) &\coloneqq E(G) \uplus E(G') \uplus E(H_+) \uplus E(H_-), \\ X(\mathcal{B}) &\coloneqq \{x_1, x_2\}, \\ Y(\mathcal{B}) &\coloneqq \{y_1, y_2\}, \\ X_1(\mathcal{B}) &\coloneqq \{x_1\} \uplus V(G) \uplus V(H_+), \\ X_2(\mathcal{B}) &\coloneqq \{x_2\} \uplus V(G') \uplus V(H_-), \\ Y_1(\mathcal{B}) &\coloneqq \{y_1\} \uplus V(G) \uplus V(H_-), \text{ and} \\ Y_2(\mathcal{B}) &\coloneqq \{y_2\} \uplus V(G') \uplus V(H_+). \end{aligned}$$

The background structure can be seen in Figure 6. Let

$$\psi_{ij}(x\,; y) \coloneqq X_i x \wedge Y_j y \wedge \exists v_1 \cdots \exists v_q \left(\bigwedge_{s=1}^{q} (X_i v_s \wedge Y_j v_s) \wedge \bigwedge_{1 \leqslant s_1 < s_2 \leqslant q} E v_{s_1} v_{s_2} \right)$$

and

$$\varphi(x\,; y) \coloneqq Xx \wedge Yy \wedge \bigvee_{i=1}^{2} \bigvee_{j=1}^{2} \psi_{ij}(x\,; y).$$

For a learned parameter v the hypothesis $[\![\varphi(x\,; v)]\!]^{\mathcal{B}}$ is only consistent with the training sequence $((x_1, \text{true}), (x_2, \text{false}))$, if $v \in \{y_1, y_2\}$. The parameter v is equal to y_1 if and only if G has a q-clique. Assuming ETH, there is no algorithm that checks whether a graph G has a q-clique in time $f(q) \cdot |G|^{o(q)}$ [18]. One can check whether G has a q-clique by computing $\mathcal{B}$ in time quadratic in $|G|$ (or linear in $|V(G)| + |E(G)|$), learning the parameter v and then checking whether v is equal to y_1. Thus learning the parameter is not possible in time $f(q) \cdot |G|^{o(q)} = f(q) \cdot |\mathcal{B}|^{o(q)}$. □

B. Minimizing the training error

We continue the analysis of model learning. In Theorem I.1 we allow the algorithm to reject a training sequence T if there is no consistent hypothesis within the given binding rank and binding width bounds. Instead of requiring a consistent hypothesis, we now try to find a hypothesis $H\colon U(\mathcal{B})^k \to \{0,1\}$ that minimizes the number of errors, i.e., that minimizes the *training error*

$$\mathrm{err}_T(H) = \frac{1}{|T|}\left|\{(\bar{u},c) \in T \mid H(\bar{u}) \neq c\}\right|.$$

In the next theorem we generalize the results of Theorem I.1.

Theorem III.4. *Let $k,\ell,r,w \in \mathbb{N}$. Then there is a learning algorithm $\mathfrak{L}_{min}$ for the k-ary learning problem over some finite relational structure $\mathcal{B}$, that receives a training sequence T and the degree of the structure $\Delta\mathcal{B}$ as input, with the following properties:*

(1) The algorithm always returns a hypothesis H of the form $(\varphi^*(\bar{x}\,;\bar{y}),\bar{v}^*)$ for some first-order formula $\varphi^*(\bar{x}\,;\bar{y})$ that is a Boolean combination of sphere formulas with locality radius smaller than $(2w+1)^r$ and $\bar{v}^* \in U(\mathcal{B})^\ell$.

(2) If there are an FOCN(P)-formula $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ of binding rank at most r and binding width at most w and parameter tuples $\bar{v} \in U(\mathcal{B})^\ell$ and $\bar{\lambda} \in \{0,\dots,|\mathcal{B}|\}^{|\bar{\kappa}|}$ such that $\mathrm{err}_T(\llbracket\varphi(\bar{x}\,;\bar{v},\bar{\lambda})\rrbracket^{\mathcal{B}}) \leqslant \varepsilon$ then $\mathfrak{L}_{min}$ returns a hypothesis $(\varphi^*(\bar{x}\,;\bar{y}),\bar{v}^*)$ with $\mathrm{err}_T(\llbracket\varphi^*(\bar{x}\,;\bar{v}^*)\rrbracket^{\mathcal{B}}) \leqslant \varepsilon$.

(3) The algorithm runs in time $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d)}$ with only local access to $\mathcal{B}$, where $n := |\mathcal{B}|$, $d := \Delta B$ and $t := |T|$.

(4) The hypothesis returned by the algorithm can be evaluated in time $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d)}$ with only local access to $\mathcal{B}$.

Proof: The pseudocode for our algorithm is shown in Figure 7. Let $(\varphi^*(\bar{x}\,;\bar{y}),\bar{v}^*)$ be the hypothesis returned by the algorithm. By construction we know that the hypothesis satisfies (1).

Let $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ be a FOCN(P)-formula of binding width at most w and binding rank at most r and some tuples $\bar{v} \in U(\mathcal{B})^\ell$ and $\bar{\lambda} \in \{0,\dots,|\mathcal{B}|\}^{|\bar{\kappa}|}$ of parameters such that $\mathrm{err}_T(\llbracket\varphi(\bar{x}\,;\bar{v},\bar{\lambda})\rrbracket^{\mathcal{B}})$ is minimal, especially $\mathrm{err}_T(\llbracket\varphi(\bar{x}\,;\bar{v},\bar{\lambda})\rrbracket^{\mathcal{B}}) \leqslant \varepsilon$ for the input sequence T of training examples.

Note that there is a $\Delta\mathcal{B}$-equivalent hnf-formula for φ of locality radius smaller than $(2w+1)^r - 1$ and thus $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}) \models \varphi(\bar{u},\bar{v},\bar{\lambda})$ iff $\mathcal{B} \models \varphi(\bar{u},\bar{v},\bar{\lambda})$ for all $(\bar{u},c) \in T$. Furthermore note that $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*) \models \varphi^*(\bar{u},\bar{v}^*)$ iff $\mathcal{B} \models \varphi^*(\bar{u},\bar{v}^*)$ for all $(\bar{u},c) \in T$, because φ^* is a Boolean combination of sphere formulas of locality radius smaller than $(2w+1)^r$. Let $T' \subseteq T$ be the subsequence of examples that are consistent with $\llbracket\varphi(\bar{x}\,;\bar{v},\bar{\lambda})\rrbracket^{\mathcal{B}}$. For every (u_i,c_i) in T let

$$\begin{aligned}\mathrm{pos}_i := \{j \in [t] \mid c_j = 1 \text{ and}&\\ &\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}) \cong \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_j\bar{v})\}\end{aligned}$$

and

$$\begin{aligned}\mathrm{neg}_i := \{j \in [t] \mid c_j = 0 \text{ and}&\\ &\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v}) \cong \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_j\bar{v})\}.\end{aligned}$$

```
Algorithm $\mathfrak{L}_{min}$
Input: Training sequence $T = ((\bar{u}_1,c_1)),\dots,(\bar{u}_t,c_t)) \in \mathcal{T}$,
    $d = \Delta\mathcal{B}$, local access to background structure $\mathcal{B}$
$N \leftarrow N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)$
$consistent_{max} \leftarrow -1$    ▷ maximal number of examples
    consistent with a chosen formula
for all $\bar{v} \in N^m,\ m \leqslant \ell$ do
    for $i = 1,\dots,t$ do
        $\mathcal{N}_i \leftarrow \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v})$
        $\vartheta_i(\bar{x}\,;\bar{y}) \leftarrow \mathrm{sph}_{\mathcal{N}_i}(\bar{x}\bar{y})$    ▷ a $d$-bounded
    $[(2w+1)^r-1]$-type with $k+m$ centers
    for $i = 1,\dots,t$ do
        $neg_i \leftarrow 0$
        $pos_i \leftarrow 0$
        for all $j = 1,\dots,t$ do
            if $\mathcal{N}_i \cong \mathcal{N}_j$ then
                if $c_j = 1$ then
                    $pos_i \leftarrow pos_i + 1$
                else
                    $neg_i \leftarrow neg_i + 1$
    $\varphi(\bar{x}\,;\bar{y}) \leftarrow \bigvee_{i\in[t],\, pos_i \geqslant neg_i} \vartheta_i(\bar{x}\,;\bar{y})$
    $consistent_{cur} \leftarrow |\{i \in [t] \mid (pos_i \geqslant neg_i$ and $c_i =$
    $1)$ or $(pos_i < neg_i$ and $c_i = 0)\}|$
    if $consistent_{cur} > consistent_{max}$ then
        $\varphi^* \leftarrow \varphi$
        $\bar{v}^* \leftarrow \bar{v}$
        $consistent_{max} \leftarrow consistent_{cur}$
return $(\varphi^*(\bar{x}\,;\bar{y}),\ \bar{v}^*)$
```

Figure 7. Learning algorithm $\mathfrak{L}_{min}$ of Theorem III.4

Claim 1. *If $(u_i,1)$ in T', then $|\mathrm{pos}_i| \geqslant |\mathrm{neg}_i|$.*

Proof: Consider the formula

$$\varphi'(\bar{x}\,;\bar{y},\bar{\kappa}) := \varphi(\bar{x}\,;\bar{y},\bar{\kappa}) \wedge \neg\mathrm{sph}_{\mathcal{N}_i}(\bar{x}\bar{y})$$

with $\mathcal{N}_i = \mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}_i\bar{v})$. We know that $c_i = 1$ and thus $\{(u_j,c_j) \mid j \in \mathrm{pos}_i\} = \{(u_j,1) \mid j \in \mathrm{pos}_i\} \subseteq T'$ and $\{(u_j,c_j) \mid j \in \mathrm{neg}_i\} \subseteq T \setminus T'$. The hypothesis $\llbracket\varphi'(\bar{x}\,;\bar{v},\bar{\lambda})\rrbracket^{\mathcal{B}}$ is consistent with the examples

$$T' \setminus \{(u_j,c_j) \mid j \in \mathrm{pos}_i\} \cup \{(u_j,c_j) \mid j \in \mathrm{neg}_i\}.$$

The cardinality of this set is $|T'| - |\mathrm{pos}_i| + |\mathrm{neg}_i|$. The claim follows from the optimality of $(\varphi,\bar{v},\bar{\lambda})$. ⌟

Claim 2. *If $(u_i,0)$ in T', then $|\mathrm{neg}_i| \geqslant |\mathrm{pos}_i|$.*

Proof: The proof is analogous to the proof of the first claim. Here we consider the formula

$$\varphi'(\bar{x}\,;\bar{y},\bar{\kappa}) := \varphi(\bar{x}\,;\bar{y},\bar{\kappa}) \vee \mathrm{sph}_{\mathcal{N}_i}(\bar{x}\bar{y})$$

and again use the optimality of $(\varphi,\bar{v},\bar{\lambda})$. ⌟

When using $\bar{v}^* = \bar{v}$, our algorithm constructs a formula that is consistent with all examples $(u_i,0)$ where $|\mathrm{pos}_i| < |\mathrm{neg}_i|$ and all examples $(u_i,1)$ where $|\mathrm{pos}_i| \geqslant |\mathrm{neg}_i|$. Using both claims we can follow that the hypothesis returned by the algorithm is consistent with at least as many examples as the optimal solution. This proves (2).

Analogous to Theorem I.1 there are at most $(t+d)^{\mathcal{O}(1)}$ tuples in $\bigcup_{m=1}^{\ell} N^m$ and N can be computed with only local access to $\mathcal{B}$ in time $(t+d)^{\mathcal{O}(1)}$. The algorithm in Figure 7 can compute all $\mathcal{N}_i$ and sphere formulas ϑ_i in time $(\log n + d + t)^{\mathcal{O}(1)}$. The isomorphism tests can be done in time $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d))}$. All in all the running time of the algorithm is in $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d)}$. This shows that the algorithm satisfies Condition (3).

To evaluate the formula returned by the algorithm for $\bar{u}$, we only have to evaluate it using the structure $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u})$ with only local access to $\mathcal{B}$, so the running time is in $(\log n + t)^{\mathcal{O}(1)} \cdot d^{\mathrm{polylog}(d)}$ and Condition (4) is satisfied. □

IV. Structures of Bounded Degree

In this section we consider structures of bounded degree. We prove results that improve the running times we obtain for structures of polylogarithmic degree and we also extend these results to prove that there is a sublinear-time PAC-learning algorithm for hypotheses using FOCN(P)-formulas on structures of bounded degree. Let $d \in \mathbb{N}$ be fixed and let $\mathcal{B}$ be a background structure with maximum degree at most d over a relational signature σ.

As in Section III, let Φ be the set of FOCN(P)-formulas $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ with binding width at most w, binding rank at most r and free number variables $\bar{\kappa}$, and

$$\mathcal{C} := \{[\![\varphi(\bar{x}\,;\bar{v},\bar{\lambda})]\!]^{\mathcal{B}} \mid \varphi(\bar{x}\,;\bar{y},\bar{\kappa}) \in \Phi,\ \bar{v} \in U(\mathcal{B})^{\ell},\\ \bar{\lambda} \in \{0,\dots,|\mathcal{B}|\}^{|\bar{\kappa}|}\}.$$

Let Φ^* be the set of normalized formulas $\varphi^*(\bar{x}\,;\bar{y}) = \bigwedge_i \bigvee_j \psi_{ij}(\bar{x},\bar{y})$ where ψ_{ij} are (possibly negated) sphere formulas (modulo equivalence) with locality radius smaller than $(2w+1)^r$ for structures of degree at most d and

$$\mathcal{C}^* := \{[\![\varphi^*(\bar{x}\,;\bar{v})]\!]^{\mathcal{B}} \mid \varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*,\ \bar{v} \in U(\mathcal{B})^{\ell}\}.$$

Lemma IV.1. *$|\Phi^*|$ is finite and does not depend on $|\mathcal{B}|$.*

Proof: There are at most $\nu_d(r) := 1 + d \cdot \sum_{i=0}^{r-1}(d-1)^i \leqslant d^{r+1} + 2r + 1$ elements in a $((2w+1)^r - 1)$-type with a single center [3]. Hence there are at most

$$E(d,k,\ell,r,w) := (k+\ell) \cdot \nu_d((2w+1)^r - 1)\\ \leqslant (k+\ell) \cdot (d^{(2w+1)^r} + 2(2w+1)^r)$$

elements in a $((2w+1)^r - 1)$-type with $k+\ell$ centers [3]. Thus there are at most

$$F(d,k,\ell,r,w,\sigma) := \prod_{R \in \sigma} 2^{\left(E(d,k,\ell,r,w)^{\mathrm{ar}(R)}\right)}$$

non-isomorphic $((2w+1)^r - 1)$-types with $k+\ell$ centers.

$$\varphi^* = \underbrace{\bigwedge_i \underbrace{\bigvee_j \underbrace{(\neg)\psi_{ij}}_{\leqslant 2F(\dots)\cdot 2^k k!}}_{\leqslant 2^{2F(\dots)\cdot 2^k k!}}}_{\leqslant 2^{2^{2F(\dots)\cdot 2^k k!}}}$$

Hence the number of normalized formulas in Φ^* is at most $2^{2^{2F(d,k,\ell,r,w,\sigma)\cdot 2^k k!}}$. □

Lemma IV.2. *$|\mathcal{C}|$ is finite and independent from $|\mathcal{B}|$.*

Proof: The set Φ^* is finite and thus also $\mathcal{C}^*$. For all formulas in Φ there is a d-equivalent formula in Φ^* and thus $\mathcal{C} \subseteq \mathcal{C}^*$. Hence $\mathcal{C}$ is finite as well. □

A. Consistent Hypotheses

As in Theorem I.1, we use brute force to learn a consistent hypothesis. Instead of building a formula from the training examples, we show that it suffices to use a formula from Φ^*. This makes the complexity of the formula independent from the number of training examples and thus we obtain a running time for the evaluation of the hypothesis that is independent from the number of training examples.

Corollary IV.3. *Let $T \in \mathcal{T}$ be consistent with some $C \in \mathcal{C}$. Then there is a formula $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$ and a tuple $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^{\ell}$ such that $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T.*

Proof: Let $\varphi^*(\bar{x}\,;\bar{y})$ and $\bar{v}^*$ be as in Lemma III.1. Then, after some normalization for φ^*, $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$, $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^{\ell}$ and $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T. □

To check the consistency of a hypothesis, we use the following model-checking result due to Grohe [19].

Definition IV.4. We say that a class C of structures has *low degree* if for every $\varepsilon > 0$ there is an integer N_{ε} such that for all $\mathcal{A} \in C$ with $|\mathcal{A}| \geqslant N_{\varepsilon}$ we have $\Delta\mathcal{A} \leqslant |A|^{\varepsilon}$. ⌟

Theorem IV.5. *There is an algorithm $\mathfrak{A}$ for* FO-Model-Checking *and a function f such that for every class C of structures that has low degree and for every $\varepsilon > 0$ the running time of $\mathfrak{A}$ on an input $(\mathcal{A},\varphi) \in C \times \mathrm{FO}$ is in $\mathcal{O}(f(\|\varphi\|) \cdot |\mathcal{A}|^{1+\varepsilon})$.*

Using this result we obtain a model-learning algorithm with an improved running time for hypothesis evaluation.

Theorem IV.6. *Let $d,k,\ell,r,w \in \mathbb{N}$. Then there is a learning algorithm $\mathfrak{L}^d_{con}$ for the k-ary learning problem over some finite relational structure $\mathcal{B}$ of degree at most d, that receives a training sequence T as input, with the following properties:*

(1) If the algorithm returns a hypothesis H, then H is of the form $(\varphi^*(\bar{x}\,;\bar{y}),\bar{v}^*)$ for some first-order formula $\varphi^*(\bar{x}\,;\bar{y})$ that is a Boolean combination of sphere formulas with locality radius smaller than $(2w+1)^r$ and $\bar{v}^* \in U(\mathcal{B})^{\ell}$, and $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T.

(2) If there are an FOCN(P)-formula $\varphi(\bar{x}\,;\bar{y},\bar{\kappa})$ of binding rank at most r and binding width at most w and parameter tuples $\bar{v} \in U(\mathcal{B})^{\ell}$ and $\bar{\lambda} \in \{0,\dots,|\mathcal{B}|\}^{|\bar{\kappa}|}$ such that $[\![\varphi(\bar{x}\,;\bar{y},\bar{\lambda})]\!]^{\mathcal{B}}$ is consistent with the input sequence T, then $\mathfrak{L}^d_{con}$ always returns a hypothesis.

(3) The algorithm runs in time $(\log n + t)^{\mathcal{O}(1)}$ with only local access to $\mathcal{B}$, where $n := |\mathcal{B}|$ and $t := |T|$.

```
Algorithm $\mathfrak{L}^d_{con}$
Input: Training sequence $T \in \mathcal{T}$, local access to background structure $\mathcal{B}$
$N \leftarrow N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)$
for all $\bar{v}^* \in N^\ell$ do
    for all $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$ do
        $consistent \leftarrow$ true
        for all $(\bar{u}, c) \in T$ do
            if ($\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*) \models \varphi^*(\bar{u}\,;\bar{v}^*)$ and $c = 0$) or ($\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*) \not\models \varphi^*(\bar{u}\,;\bar{v}^*)$ and $c = 1$) then
                $consistent \leftarrow$ false
        if $consistent$ then
            return $(\varphi^*(\bar{x}\,;\bar{y}),\, \bar{v}^*)$
reject
```

Figure 8. Learning algorithm $\mathfrak{L}^d_{con}$ of Theorem IV.6

(4) The hypothesis returned by the algorithm can be evaluated in time $(\log n)^{\mathcal{O}(1)}$ with only local access to $\mathcal{B}$.

Proof: The pseudocode for our algorithm is shown in Figure 8. The algorithm goes through all tuples $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^\ell$ and all formulas $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$ and checks, whether $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with T. The algorithm returns the first consistent $(\varphi^*, \bar{v}^*)$ and rejects if there is none. Note that $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1} \models \varphi^*(\bar{u}\,;\bar{v}^*)$ iff $\mathcal{B} \models \varphi^*(\bar{u}\,;\bar{v}^*)$ for all $(\bar{u}, c) \in T$, because φ^* is a Boolean combination of sphere formulas of locality radius smaller than $(2w+1)^r$. Thus the algorithm satisfies Condition (1) of Theorem IV.6 and because of Corollary IV.3 it satisfies (2) as well.

Let $n := |\mathcal{B}|$ and $t := |T|$. Then for all $\bar{u} \in U(\mathcal{B})^k$ and $\bar{v}^* \in U(\mathcal{B})^\ell$ we have $|\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*)| \leqslant (k+\ell) \cdot 2d^{(2w+1)^r}$ and the representation size is $(\log n)^{\mathcal{O}(1)}$, because d, k, ℓ, w and r are constant. Lemma IV.1 tells us that the number of formulas in Φ^* to check is constant. Thus for every real valued function f it follows that $\max_{\psi \in \Phi^*} f(\|\psi\|)$ is finite. Hence according to Theorem IV.5 it takes time polynomial in the size of $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*)$ to check whether the structure satisfies a formula $\varphi^*(\bar{u}\,;\bar{v}^*)$. The number of tuples in N^ℓ is $|N|^\ell \leqslant (2tkd^{2\ell(2w+1)^r})^\ell \in t^{\mathcal{O}(1)}$ and N can be computed with only local access to $\mathcal{B}$ in time $t^{\mathcal{O}(1)}$. All in all the running time of the algorithm is in $t^{\mathcal{O}(1)} \cdot t \cdot (\log n)^{\mathcal{O}(1)} \leqslant (\log n + t)^{\mathcal{O}(1)}$. This shows that the algorithm satisfies Condition (3).

To evaluate the formula returned by the algorithm for $(\bar{u}, \bar{v})$, we only have to evaluate it using the structure $\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v})$ with only local access to $\mathcal{B}$, so the running time is in $(\log n)^{\mathcal{O}(1)}$ and Condition (4) is satisfied. □

B. Minimizing the training error

Corollary IV.7. *Let $T \in \mathcal{T}$ be such that $\operatorname{err}_T(C) \leqslant \varepsilon$ for some $C \in \mathcal{C}$. Then there is a formula $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$ and a tuple $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^\ell$ such that $\operatorname{err}_T([\![\varphi^*(\bar{x}, \bar{v}^*)]\!]^{\mathcal{B}}) \leqslant \varepsilon$.*

Proof: If $\operatorname{err}_T(C) \leqslant \varepsilon$, then there is some $S \subseteq T$ with $|S| \geqslant (1-\varepsilon) \cdot |T|$ such that C is consistent with S. Using Corollary IV.3 on S we obtain a formula $\varphi^*(\bar{x}\,;\bar{y})$ and a tuple $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(S)^\ell \subseteq N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^\ell$ such that $[\![\varphi^*(\bar{x}, \bar{v}^*)]\!]^{\mathcal{B}}$ is consistent with S and thus $\operatorname{err}_T([\![\varphi^*(\bar{x}, \bar{v}^*)]\!]^{\mathcal{B}}) \leqslant \varepsilon$. □

```
Algorithm $\mathfrak{L}^d_{min}$
Input: Training sequence $T \in \mathcal{T}$, local access to background structure $\mathcal{B}$
$N \leftarrow N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)$
$err_{min} \leftarrow |T| + 1$
for all $\bar{v}^* \in N^\ell$ do
    for all $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$ do
        $err_{cur} \leftarrow 0$
        for all $(\bar{u}, c) \in T$ do
            if ($\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*) \models \varphi^*(\bar{u}\,;\bar{v}^*)$ and $c = 0$) or ($\mathcal{N}^{\mathcal{B}}_{(2w+1)^r-1}(\bar{u}\bar{v}^*) \not\models \varphi^*(\bar{u}\,;\bar{v}^*)$ and $c = 1$) then
                $err_{cur} \leftarrow err_{cur} + 1$
        if $err_{cur} < err_{min}$ then
            $err_{min} \leftarrow err_{cur}$
            $\varphi^*_{min} \leftarrow \varphi^*$
            $\bar{v}^*_{min} \leftarrow \bar{v}^*$
return $(\varphi^*_{min},\, \bar{v}^*_{min})$
```

Figure 9. Learning algorithm $\mathfrak{L}^d_{min}$ of Theorem IV.8 and Theorem IV.10

Theorem IV.8. *Let $d, k, \ell, r, w \in \mathbb{N}$. Then there is a learning algorithm $\mathfrak{L}^d_{min}$ for the k-ary learning problem over some finite relational structure $\mathcal{B}$ of degree at most d, that receives a training sequence T as input, with the following properties:*

(1) The algorithm always returns a hypothesis H of the form $(\varphi^*(\bar{x}\,;\bar{y}), \bar{v}^*)$ for some first-order formula $\varphi^*(\bar{x}\,;\bar{y})$ that is a Boolean combination of sphere formulas with locality radius smaller than $(2w+1)^r$ and $\bar{v}^* \in U(\mathcal{B})^\ell$.

(2) If there are an FOCN(P)-formula $\varphi(\bar{x}\,;\bar{y}, \bar{\kappa})$ of binding rank at most r and binding width at most w and parameter tuples $\bar{v} \in U(\mathcal{B})^\ell$ and $\bar{\lambda} \in \{0, \ldots, |\mathcal{B}|\}^{|\bar{\kappa}|}$ such that $\operatorname{err}_T([\![\varphi(\bar{x}\,;\bar{v}, \bar{\lambda})]\!]^{\mathcal{B}}) \leqslant \varepsilon$, then $\mathfrak{L}^d_{min}$ returns a hypothesis $(\varphi^*(\bar{x}\,;\bar{y}), \bar{v}^*)$ with $\operatorname{err}_T([\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}) \leqslant \varepsilon$.

(3) The algorithm runs in time $(\log n + t)^{\mathcal{O}(1)}$ with only local access to $\mathcal{B}$, where $n := |\mathcal{B}|$ and $t := |T|$.

(4) The hypothesis returned by the algorithm can be evaluated in time $(\log n)^{\mathcal{O}(1)}$ with only local access to $\mathcal{B}$.

Proof: The pseudocode for our algorithm is shown in Figure 9. The algorithm goes through all tuples $\bar{v}^* \in N^{\mathcal{B}}_{2\ell[(2w+1)^r-1]}(T)^\ell$ and all formulas $\varphi^*(\bar{x}\,;\bar{y}) \in \Phi^*$ and counts the number of errors that $[\![\varphi^*(\bar{x}\,;\bar{v}^*)]\!]^{\mathcal{B}}$ makes on T. The algorithm returns the hypothesis with minimum error. Using Corollary IV.7 one can show analogously to Theorem IV.6 that (1) and (2) hold. The running-time analysis for (3) and (4) is also analogous to the proof of Theorem IV.6. □

C. Agnostic PAC Learning

We give a short introduction to *probably approximately correct* (PAC) learning. For more background, we refer to [20]. Instead of focussing on the training error, we are now interested in hypotheses that generalize well. We assume that there is an (unknown) probability distribution $\mathcal{D}$ on $U(\mathcal{B})^k \times \{0,1\}$ and that training examples are drawn independently from this distribution. Our goal is to find an algorithm that with high probability on training examples drawn from $\mathcal{D}$, returns a hypothesis that has a small expected error on instances drawn from the same distribution $\mathcal{D}$.

The *generalization error* of a hypothesis $H\colon U(\mathcal{B})^k \to \{0,1\}$ for a probability distribution $\mathcal{D}$ on $U(\mathcal{B})^k \times \{0,1\}$ is

$$\mathrm{err}_{\mathcal{D}}(H) := \underset{(\bar{u},c)\sim\mathcal{D}}{\mathcal{P}}(H(\bar{u}) \neq c).$$

A hypothesis class $\mathcal{H}$ is *agnostic PAC-learnable* if there is some function $t_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ and a learning algorithm $\mathfrak{L}$ such that for every $\varepsilon, \delta \in (0,1)$ and for every distribution $\mathcal{D}$ over $U(\mathcal{B})^k \times \{0,1\}$, when running the algorithm on a sequence T of $t_{\mathcal{H}}(\varepsilon,\delta)$ examples drawn i.i.d. from $\mathcal{D}$, it satisfies

$$\underset{T\sim\mathcal{D}}{\mathcal{P}}\left[\mathrm{err}_{\mathcal{D}}(\mathfrak{L}(T)) \leqslant \inf_{H\in\mathcal{H}} \mathrm{err}_{\mathcal{D}}(H) + \varepsilon\right] \geqslant 1-\delta.$$

We use the following lemma from [20] to bound the generalization error.

Lemma IV.9 (Uniform Convergence)**.** *Let $\mathcal{H}$ be a finite hypothesis class of hypotheses $H\colon U(\mathcal{B})^k \to \{0,1\}$ and consider training sequences T of length*

$$t \geqslant t_{\mathcal{H}}^{UC}(\varepsilon,\delta) := \left\lceil \frac{\log(2|\mathcal{H}|/\delta)}{2\varepsilon^2} \right\rceil$$

where the examples are drawn i.i.d. from a probability distribution $\mathcal{D}$ over $U(\mathcal{B})^k \times \{0,1\}$. Then

$$\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\left[|\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_T(H)| \leqslant \varepsilon \text{ for all } H \in \mathcal{H}\right] \geqslant 1-\delta.$$

Now we show that there is an agnostic PAC-learning algorithm for hypotheses using FOCN(P)-formulas on background structures of bounded degree.

Theorem IV.10. *Let $d,k,\ell,r,w \in \mathbb{N}$. Then there is some $s \in \mathbb{N}$ such that the learning algorithm $\mathfrak{L}_{min}^d$ for the k-ary learning problem over some finite relational structure $\mathcal{B}$ of degree at most d has the following properties:*

(1) The algorithm always returns a hypothesis H of the form $(\varphi^*(\bar{x}\,;\bar{y}^*))$ for some first-order formula $\varphi^*(\bar{x}\,;\bar{y})$ that is a Boolean combination of sphere formulas with locality radius smaller than $(2w+1)^r$ and $\bar{v}^* \in U(\mathcal{B})^\ell$ such that for an input sequence T of at least $|T| =: t = s\left\lceil\frac{\log(|\mathcal{B}|/\delta)}{\varepsilon^2}\right\rceil$ training examples it holds that

$$\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\left[\mathrm{err}_{\mathcal{D}}(\llbracket\varphi^*(\bar{x},\bar{v}^*\rrbracket^{\mathcal{B}}) - \min_{C\in\mathcal{C}} \mathrm{err}_{\mathcal{D}}(C) \leqslant \varepsilon\right] \geqslant 1-\delta.$$

(2) If $|T| = s\left\lceil\frac{\log(|\mathcal{B}|/\delta)}{\varepsilon^2}\right\rceil$, then the algorithm runs in time $(\log|\mathcal{B}| + 1/\varepsilon + \log 1/\delta)^{\mathcal{O}(1)}$ with only local access to $\mathcal{B}$.

(3) The hypothesis returned by the algorithm can be evaluated in time $(\log n)^{\mathcal{O}(1)}$ with only local access to $\mathcal{B}$.

Proof: By Lemma IV.2, the hypothesis class $\mathcal{H}^* := \Phi^* \times U(\mathcal{B})^\ell$ is finite. Thus we can bound the generalization error of a hypothesis H returned by $\mathfrak{L}_{min}^d$ using the uniform convergence Lemma IV.9 by

$$\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\left[|\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_T(H)| \leqslant \varepsilon/2\right] \geqslant 1-\delta/2$$

for input sequences T of length at least $t = \left\lceil\frac{\log(4|\mathcal{H}^*|/\delta)}{2(\varepsilon/2)^2}\right\rceil = \left\lceil\frac{4\log|\Phi^*|\cdot\ell\cdot\log(|\mathcal{B}|/\delta)}{\varepsilon^2}\right\rceil$. Furthermore for all $C \in \mathcal{C}$,

$$\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\left[|\mathrm{err}_T(C) - \mathrm{err}_{\mathcal{D}}(C)| \leqslant \varepsilon/2\right] \geqslant 1-\delta/2$$

for input sequences T of length at least $t = \left\lceil\frac{\log(4|\mathcal{C}|/\delta)}{2(\varepsilon/2)^2}\right\rceil \leqslant \left\lceil\frac{4\log(|\Phi^*|/\delta)}{\varepsilon^2}\right\rceil$, because $|\mathcal{C}^*| \leqslant |\Phi^*|$ and $\mathcal{C} \subseteq \mathcal{C}^*$. Using Corollary IV.7 we know that $\mathrm{err}_T(H) \leqslant \mathrm{err}_T(C)$ for all $C \in \mathcal{C}$. Thus for all $C \in \mathcal{C}$

$$\begin{aligned}
&\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\left[\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_{\mathcal{D}}(C) \leqslant \varepsilon\right] \\
=\ &\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\big[\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_T(H) + \mathrm{err}_T(H) \\
&\qquad - \mathrm{err}_T(C) + \mathrm{err}_T(C) - \mathrm{err}_{\mathcal{D}}(C) \leqslant \varepsilon\big] \\
\geqslant\ &\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\big[\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_T(H) \leqslant \varepsilon/2 \text{ and} \\
&\qquad \mathrm{err}_T(H) \leqslant \mathrm{err}_T(C) \text{ and } \mathrm{err}_T(C) - \mathrm{err}_{\mathcal{D}}(C) \leqslant \varepsilon/2\big] \\
=\ &\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\big[\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_T(H) \leqslant \varepsilon/2 \text{ and} \\
&\qquad \mathrm{err}_T(C) - \mathrm{err}_{\mathcal{D}}(C) \leqslant \varepsilon/2\big] \\
\geqslant\ &\underset{T\sim\mathcal{D}^t}{\mathcal{P}}\big[|\mathrm{err}_{\mathcal{D}}(H) - \mathrm{err}_T(H)| \leqslant \varepsilon/2 \text{ and} \\
&\qquad |\mathrm{err}_T(C) - \mathrm{err}_{\mathcal{D}}(C)| \leqslant \varepsilon/2\big] \\
\geqslant\ &1-\delta.
\end{aligned}$$

This holds especially for $C = \mathrm{argmin}_{C'\in\mathcal{C}}\,\mathrm{err}_{\mathcal{D}}(C)$ and hence (1) follows with $s = 4\ell\cdot\log|\Phi^*|$. The statements (2) and (3) follow immediately from Theorem IV.8. □

V. Conclusions

We prove that FOCN(P)-definable concepts over structures of polylogarithmic degree can be learned in sublinear time. For structures with no degree bound we show that there is no consistent model-learning algorithm that runs in sublinear time with only local access to the structure. Furthermore, we show how to use a consistent parameter-learning algorithm to solve q-Clique.

It remains open whether one can obtain a similar result for model learning and improve the lower bound on the running time for consistent model-learning algorithms. Our results imply PAC-learnability on structures of bounded degree. It would be interesting to investigate structures of polylogarithmic degree in this context.

In addition to the COUNT operator that we analyze within the logic FOCN(P), another direction for future research is the analysis of learning algorithms for stronger logics that implement other aggregating operators from SQL.

REFERENCES

[1] M. Grohe and G. Turán, "Learnability and definability in trees and similar structures," *Theory Comput. Syst.*, vol. 37, no. 1, pp. 193–220, 2004.

[2] M. Grohe and M. Ritzert, "Learning first-order definable concepts over structures of small degree," in *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pp. 1–12, IEEE Computer Society, 2017.

[3] D. Kuske and N. Schweikardt, "First-order logic with counting: At least, weak hanf normal forms always exist and can be computed!," *CoRR*, vol. abs/1703.01122, 2017.

[4] M. Grohe and N. Schweikardt, "First-order query evaluation with cardinality conditions," in *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Houston, TX, USA, June 10-15, 2018* (J. V. den Bussche and M. Arenas, eds.), pp. 253–266, ACM, 2018.

[5] M. Grohe, D. Neuen, and P. Schweitzer, "A faster isomorphism test for graphs of small degree," in *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018* (M. Thorup, ed.), pp. 89–100, IEEE Computer Society, 2018.

[6] M. Grohe, C. Löding, and M. Ritzert, "Learning MSO-definable hypotheses on strings," in *International Conference on Algorithmic Learning Theory, ALT 2017, 15-17 October 2017, Kyoto University, Kyoto, Japan* (S. Hanneke and L. Reyzin, eds.), vol. 76 of *Proceedings of Machine Learning Research*, pp. 434–451, PMLR, 2017.

[7] W. W. Cohen and C. D. Page Jr., "Polynomial learnability and inductive logic programming: Methods and results," *New Generation Comput.*, vol. 13, no. 3&4, pp. 369–409, 1995.

[8] J. Kietz and S. Dzeroski, "Inductive logic programming and learnability," *SIGART Bulletin*, vol. 5, no. 1, pp. 22–32, 1994.

[9] S. Muggleton, "Inductive logic programming," *New Generation Comput.*, vol. 8, no. 4, pp. 295–318, 1991.

[10] S. Muggleton, "Inductive logic programming," in *Inductive Logic Programming* (S. Muggleton, ed.), vol. 38 of *The APIC Series*, pp. 1–27, Academic Press, 1992.

[11] S. Muggleton and L. D. Raedt, "Inductive logic programming: Theory and methods," *J. Log. Program.*, vol. 19/20, pp. 629–679, 1994.

[12] A. Abouzied, D. Angluin, C. H. Papadimitriou, J. M. Hellerstein, and A. Silberschatz, "Learning and verifying quantified boolean queries by example," in *Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2013, New York, NY, USA - June 22 - 27, 2013* (R. Hull and W. Fan, eds.), pp. 49–60, ACM, 2013.

[13] A. Bonifati, R. Ciucanu, and S. Staworko, "Learning join queries from user examples," *ACM Trans. Database Syst.*, vol. 40, no. 4, pp. 24:1–24:38, 2016.

[14] L. Libkin, *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series, Springer, 2004.

[15] M. Grohe, *Descriptive Complexity, Canonisation, and Definable Graph Structure Theory*. Lecture Notes in Logic, Cambridge University Press, 2017.

[16] L. Hella, L. Libkin, J. Nurmonen, and L. Wong, "Logics with aggregate operators," *J. ACM*, vol. 48, no. 4, pp. 880–907, 2001.

[17] S. Feferman and R. L. Vaught, "The first order properties of products of algebraic systems," *Fundamenta Mathematicae*, vol. 47, no. 1, pp. 57–103, 1959.

[18] D. Lokshtanov, D. Marx, and S. Saurabh, "Lower bounds based on the exponential time hypothesis," *Bulletin of the EATCS*, vol. 105, pp. 41–72, 2011.

[19] M. Grohe, "Generalized model-checking problems for first-order logic," in *STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science, Dresden, Germany, February 15-17, 2001, Proceedings* (A. Ferreira and H. Reichel, eds.), vol. 2010 of *Lecture Notes in Computer Science*, pp. 12–26, Springer, 2001.

[20] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. New York, NY, USA: Cambridge University Press, 2014.