WHAT GRADIENT DESCENT CAN SEE

Anonymous authors
Paper under double-blind review

ABSTRACT

The Boolean queries computed by DLOGTIME-uniform, polynomial-size, bounded-depth threshold networks are definable in first-order logic with majority quantifiers, the logic of uniform constant-depth threshold circuits. Expressibility is not learnability: polynomial-tolerance correlational queries cannot efficiently learn some targets that these networks represent, with sparse parity the standard witness. We show that this gap is itself logical. We introduce the correlationally visible fragment of that logic, in which a step may introduce new coordinates only when they carry a correlation with the residual of the target given what is already present. In the correlational statistical query model, this fragment is exactly the class learnable to any inverse-polynomial error in polynomial time, for definable targets of logarithmic arity, inverse-polynomial Fourier coefficients, and definable relevant sets. The lower half of the characterization is a leap lower bound proved in the model itself, with absolute constants, for arbitrary Boolean shapes of logarithmic arity. The arity bound cannot be dropped: we exhibit a sparse query with no small coefficient that introduces one coordinate per step, lies in the fragment, and is not learnable. The separation from the full logic is unconditional, witnessed by parity.

1 INTRODUCTION

At bounded depth and precision and polynomial size, DLOGTIME-uniform families of threshold networks compute Boolean functions in uniform TC^0, the class of constant-depth polynomial-size threshold circuits. That class has a second description, as the queries definable in first-order logic extended by the majority quantifier, a logic written FOM (Barrington et al., 1990). The graph neural networks studied by Grohe (2024) fall in the non-uniform version of the same class, and log-precision transformers translate into FOM formulas (Merrill & Sabharwal, 2023). Representational power does not by itself describe what gradient descent can learn. Sparse parity illustrates the gap in the correlational-query model. A small network represents it, and parity lies inside $\mathsf{TC}^0 = \mathsf{FOM}$. Yet inverse-polynomial-tolerance queries need super-polynomially many steps to learn its orbit when its size $k(n)$ tends to infinity and is at most $\log n$. This reflects the statistical-query obstruction (Kearns, 1998; Blum et al., 1994), and it is why gradient learnability of a sparse function is governed by the order in which its features can be reached (Abbe et al., 2022; 2023a).

If expressivity is captured by a logic, then learnability is captured by a fragment of that logic. We define such a fragment by restricting definable feature derivations: a step may introduce new coordinates only when those coordinates carry a correlation with the target's residual after the earlier features.

1.1 OUR RESULTS

We work over the Boolean hypercube $\{-1, 1\}^n$ with the uniform distribution and model gradient training of a two-layer network by correlational statistical queries, the abstraction under which one gradient coordinate is one correlation (Section 3). We write Learn for the queries whose orbit under coordinate permutations is learnable in that model to any accuracy ε in time polynomial in n and $1/\varepsilon$. A query is *tame* when it depends on $O(\log n)$ coordinates and its nonzero Fourier coefficients have magnitude at least $1/\mathrm{poly}(n)$, and *presented* when its relevant coordinates form a definable family, which every query written down here is. We define sFOM, the correlationally visible fragment of FOM (Definition 5.1): a query lies in sFOM of width w when it admits a *staircase derivation*, a

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

sequence of FOM-definable features, each introducing at most w new coordinates, each such that the coordinates it introduces carry an inverse-polynomial correlation with the residual of the target given the features already present, and a single threshold readout at the end.

Theorem 1.1 (restated as Theorem 6.2)**.** *On presented tame* FOM*-definable queries,* sFOM $=$ Learn*. If such a query lies in* sFOM *of width w, it is learnable in time $n^{O(w+c)}$, where c is a constant for which $c \log n$ bounds the number of coordinates the query depends on. If such a query is efficiently learnable, it lies in* sFOM *of some constant width.*

The two halves rest on a bridge between the syntax and the dynamics (Lemma B.8): a tame query has a width-w visible derivation if and only if its leap, the least over orderings of its Fourier support of the largest number of coordinates any step introduces, is at most w. The upper bound is Algorithm 1, the adaptive search that Joshi et al. (2024) analyze for a constant number of relevant coordinates, carried to arity $c \log n$ and shown to return the target exactly. The lower bound is where the characterization needs something the literature does not supply: the correlational leap lower bounds of Abbe et al. (2023a); Joshi et al. (2024) fix a target and let n grow, so their constants depend on the shape, while the equality above quantifies over shapes that vary with n.

Proposition 1.2 (restated as Proposition B.6)**.** *Let f be a query of arity at most $c \log n$, for instance a tame one, and let f_n be nonconstant with leap ℓ, arity a and P nonzero Fourier coefficients. For all large n, a deterministic learner that makes correlational queries of tolerance τ against $\mathcal{O}_n(f_n)$ and, against every legal oracle, outputs a hypothesis of error below $2^{-2a}/2$ makes at least $\frac{1}{2}\binom{n-a}{\ell}\tau^2/P$ queries. At tolerance $n^{-O(1)}$ this is at least $n^{(1-o(1))\ell - O(1)}$, super-polynomial once $\ell = \omega(1)$.*

The proof is Parseval, a count and Markov's inequality, applied not to parities but to the placements of the coordinates that a search at width $\ell - 1$ cannot reach; the constant is absolute. The accuracy it demands is not a convenience: Proposition B.10 exhibits a presented tame FOM-definable query that n queries of constant tolerance learn to error $1/(2n)$ and that lies outside sFOM, so at any fixed accuracy the equality of Theorem 1.1 is false. Nor is the arity bound inside tameness a convenience of the proof.

Theorem 1.3 (restated as Theorem B.4)**.** *Sparsity and a lower bound on the coefficients do not suffice in place of the arity bound. For every* FOM*-definable $k = k(n)$ with $k = \omega(\log n)$ that is a power of two with $2 \le k \le n/8$ for all $n \ge 16$, there is a presented* FOM*-definable query in* sFOM_1 *and not in* Learn *which, for every $n \ge 16$, has leap one, $12k^2 + 8$ Fourier coefficients, none of magnitude below $1/(8k)$, and arity $2k + \log_2 k + 2$.*

The witness is a multiplexer that reads each of k pairs of coordinates through the disjunction of the two, together with a secret parity, hidden behind a selector, that picks one coordinate from each pair. Transposing a pair leaves the multiplexer unchanged and moves the secret, so all 2^k secrets sit in one orbit and finding the secret is learning an unknown parity; queries of degree at most two find every relevant coordinate, and none of that identifies the secret. This is where the leap literature's picture stops: the bounds of Joshi et al. (2024) are for recovering the relevant coordinates, on the ground that what remains is a problem of fixed size, and that fails once the arity grows, a question Abbe et al. (2023a) raise when discussing what lies beyond sparsity.

Theorem 1.4 (restated as Theorem 6.3)**.** sFOM $\subsetneq$ FOM*, unconditionally, witnessed by the parity on $k(n) = \omega(1)$ coordinates.*

The non-learnability half is the classical correlational hardness of parity (Blum et al., 1994; Bshouty & Feldman, 2002); what is new is the pair of logical classes it separates, and the witness is a parity of logarithmic rather than full size because the orbit of the full parity is a single function that no query-complexity argument reaches.

1.2 Our Techniques

The technical core is a translation between two vocabularies that developed independently: descriptive complexity, which measures a query by quantifier rank and locality (Immerman, 1999; Libkin, 2004), and the analysis of gradient descent, which measures a target by leap complexity and statistical-query dimension (Abbe et al., 2022; 2023a; Blum et al., 1994). The visibility clause is exact rather than decorative: if the correlation certifying a step is allowed to come from coordinates the step already had, the step obtains its new coordinates for nothing and width stops bounding the leap.

1.3 Other Related Work

Logical Characterizations of Neural Networks. Uniform TC^0 coincides with FOM (Barrington et al., 1990; Immerman, 1999). Polynomial-weight, bounded-depth graph neural networks, under Grohe's activation assumptions, are characterized by a guarded counting logic with built-in relations and sit in non-uniform TC^0 (Grohe, 2024). An earlier correspondence between aggregate-combine graph neural networks and a guarded counting logic, for logical classifiers, is due to Barceló et al. (2020). Log-precision transformers translate into FOM formulas, in one direction (Merrill & Sabharwal, 2023). Parity already escapes AC^0 (Furst et al., 1984; Håstad, 1986). All of this concerns what is expressible; our subject is what, among the expressible, is learnable. Where a logic and a learnability notion have met before, the logic was the hypothesis class and the learner unconstrained: PAC and consistent learning of concepts definable in first-order logic and its extensions, over trees (Grohe & Turán, 2004), over structures of small degree (Grohe & Ritzert, 2017) and with counting (van Bergerem, 2025); here the logic is the target's and the algorithm is a correlational-query learner, and what is characterized is which definable targets it reaches.

Hardness and Possibility of Gradient Learning. The statistical-query model and its correlational variant abstract gradient-based training and supply the relevant lower bounds (Kearns, 1998; Blum et al., 1994; Bshouty & Feldman, 2002); the correlational restriction is a real one, since there are problems that general statistical queries solve and correlational ones do not (Goel et al., 2020). The accounting of one gradient coordinate as one correlational query has a known boundary: precision fine enough relative to the batch size takes stochastic gradient descent beyond statistical queries (Abbe et al., 2021b), suitable architectures remain polynomial-time universal under controlled gradient noise (Abbe & Sandon, 2023), and reusing batches can overcome the leap exponent of single-pass training (Dandi et al., 2024); our results concern correlational queries at inverse-polynomial tolerance. Within that frame, the merged-staircase property and leap complexity govern which sparse targets stochastic gradient descent reaches on a two-layer network (Abbe et al., 2022; 2023a), and the matching adaptive bounds of Joshi et al. (2024) are, like them, stated for a constant number of relevant coordinates; two positive results reach past that number on other hypotheses, the staircase class of Abbe et al. (2021a) and the constructible class of Ren et al. (2026), both from samples rather than from correlations. The polynomial method is optimal for distribution-free correlational learning (Gollakota et al., 2020); here we fix the uniform distribution to study coordinate discovery. Parity becomes learnable under distributions that supply low-order correlation (Daniely & Malach, 2020; Abbe et al., 2023b).

1.4 Open Problems

Theorem 1.3 shows that the characterization fails once the arity bound is dropped, which leaves the question of what takes its place.

Open Problem 1.5 (Which shapes replace the arity bound?)**.** Which queries of leap one, $\mathrm{poly}(n)$-size support and nonzero coefficient magnitudes at least $1/\mathrm{poly}(n)$ are efficiently learnable when their arity is unbounded? The staircase class of Abbe et al. (2021a), in which every support set of size at least two extends a smaller support set by one coordinate, is learned from samples by layer-wise coordinate descent without an arity bound. At each step, the candidates are the n extensions of support sets already found. The witness of Theorem B.4 fails this condition at one set: its 2^k possible placements are precisely what the learner cannot enumerate. A characterization needs a measure of how many placements of the target's shape are consistent with the sets a learner has found, and a proof that the learner pays for that number and nothing else.

2 Preliminaries

2.1 Queries, Distributions, and the Fourier Expansion

We study Boolean *queries*, by which we mean families $f = (f_n)_{n\geq 1}$ of functions $f_n \colon \{-1,1\}^n \to \{-1,1\}$. Inputs are drawn from the uniform distribution on $\{-1,1\}^n$ throughout. We write $\mathrm{poly}(n)$ for the class of functions bounded by a polynomial in n, and we call a quantity efficient when it is $\mathrm{poly}(n)$. Logarithms are to base 2, so that $2^{c\log n} = n^c$.

Write $[n] = \{1, \ldots, n\}$ for the coordinate positions of an input of length n. For a set $S \subseteq [n]$ of positions, the *character* χ_S is the function that multiplies the coordinates in S together, $\chi_S(x) = \prod_{i \in S} x_i$, the empty product $\chi_\emptyset$ being the constant 1. Each character is itself $\{-1, 1\}$-valued, and it records the parity of the coordinates it reads. There are 2^n characters, one for each subset of $[n]$, and under the uniform distribution they are orthonormal: the average of $\chi_S \chi_T$ is 1 when $S = T$ and 0 otherwise. Every $f \colon \{-1, 1\}^n \to \mathbb{R}$ therefore has a unique expansion in the characters,

$$f(x) = \sum_{S \subseteq [n]} \widehat{f}(S) \chi_S(x). \tag{1}$$

in which the coefficient at S is the correlation of f with the character there, $\widehat{f}(S) = \mathbb{E}_x \left[f(x) \chi_S(x) \right]$, the expectation being over the uniform distribution. These 2^n numbers are the *Fourier coefficients* of f, and each says how much of f the parity of its set accounts for. The *support* $\operatorname{supp}(f) = \{S \subseteq [n] : \widehat{f}(S) \neq 0\}$ is the collection of sets whose coefficient is nonzero, and it is the object almost every statement below is about. Parseval's identity states that $\sum_S \widehat{f}(S)^2 = \mathbb{E}_x \left[f(x)^2 \right]$. A $\{-1, 1\}$-valued f has $f(x)^2 = 1$ at every input, so its coefficients satisfy $\sum_S \widehat{f}(S)^2 = 1$. The parity on a set $T \subseteq [n]$ is the single character χ_T, a function whose support has the one member T, and the full parity is $\chi_{[n]}$, the parity on all n coordinates. We refer to O'Donnell (2014) for background on the expansion and for the facts just quoted.

The same expansion is available under any product distribution, that is, under any distribution that makes the coordinates independent, and Definition 5.1 is stated in those terms; every result below is proved under the uniform distribution. Let D be a product distribution under which no coordinate is constant. Recentering each coordinate to mean zero and rescaling it to variance one turns coordinate i into a function φ_i, and the products $\varphi_S = \prod_{i \in S} \varphi_i$ take the place of the characters: they are an orthonormal basis of $L^2(D)$, the space of real functions on $\{-1, 1\}^n$ under the inner product $\mathbb{E}_D [gh]$, and we write $\widehat{f}^D(S) = \mathbb{E}_D [f \varphi_S]$ for the coefficient of f at S in that basis. Under the uniform distribution $\varphi_S = \chi_S$ and $\widehat{f}^D = \widehat{f}$, and we drop the superscript. Section A sets the basis up in full.

A coordinate i is *relevant* to f when $i \in S$ for some $S \in \operatorname{supp}(f)$, that is, when some set carrying a nonzero coefficient contains i. The *arity* of f is the number of its relevant coordinates, so a function of arity a ignores all but a of its n inputs. One more convention is needed, a way of relabeling coordinates, and Definition 3.2 is what it is for. For a permutation σ of $[n]$ we write f^σ for the function $x \mapsto f(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$, which is f reading its coordinates in the order σ prescribes. Relabeling moves the support to $\{\sigma(S) : S \in \operatorname{supp}(f)\}$ and carries each coefficient along with its set, and Section A reads that off the expansion.

2.2 The Canonical Regime

Two quantities control everything below, and both need a bound: a target of unbounded arity leaves a learner too many subsets of the coordinates it has already found to enumerate, and one whose coefficients are exponentially small carries a correlation no learner of polynomial tolerance can see. We therefore fix the regime once.

Definition 2.1 (Tame query)**.** A query f is *tame* with parameters (c, q), for a positive integer c and a positive polynomial q with integer coefficients, when the following two conditions hold for every n. The function f_n has arity at most $c \log n$. Every nonzero Fourier coefficient of f_n satisfies $|\widehat{f_n}(S)| \geq 1/q(n)$. A query is *tame* when it is tame with some such parameters.

In words, a tame query reads only logarithmically many of its coordinates, and whatever correlations it does have are large enough for a learner of polynomial tolerance to measure. One further condition is needed, and unlike tameness it is about the logic rather than about the spectrum. A query is *presented* when the family of its relevant coordinate sets is FOM-definable, that is, when the relation that holds of n and i exactly when i is relevant to f_n is FOM-definable. What definability of a relation means is fixed in Section 2.3 below and in Section A; for the moment it is enough to read it as saying that one formula of the logic describes the relation at every input length. Presentedness therefore asks that the position of the relevant coordinates be describable, and not merely that the query itself be. Every family written down in this paper is presented, and Lemma B.1 is the one place the condition is spent.

A tame query has $|\mathrm{supp}(f_n)| \le 2^{c \log n} = n^c$, so tameness implies sparsity in the usual sense and we do not assume the two separately. The arity bound is what makes the coordinates a learner has already found searchable by enumeration, so that the only remaining obstruction is the search for coordinates it has not found; that division is the subject of the paper. Remark 4.4 says what goes wrong without it.

2.3 First-Order Logic with Majority

A logical formula does not read a string; it is interpreted in a structure, which is a set of elements together with relations on them. We therefore encode an input $x \in \{-1, 1\}^n$ as a finite structure whose elements are the coordinate positions $[n]$, equipped with the linear order on the positions, the BIT predicate that supplies arithmetic on them, and a unary predicate X with $X(i)$ true exactly when $x_i = 1$. Only X depends on the input, and it is what makes the input readable one coordinate at a time. Section A gives the three ingredients in full.

The logic FOM is first-order logic over such structures extended by one further quantifier, the majority quantifier, which is what lets a formula count. The formula $\mathsf{Maj}y\,\varphi(y)$ holds in a structure of size n when $\varphi(a)$ holds for more than $n/2$ of the elements a. A query f is FOM-*definable* when one sentence of the logic decides it: there is a sentence φ, meaning a formula with no free variables, such that $f_n(x) = 1$ if and only if the structure of x satisfies φ, for every n and every $x \in \{-1, 1\}^n$. The same sentence serves every input length, which is what makes definability a property of the family f and not of one f_n.

Several definitions below speak instead of a *relation*, a *function*, a family of *real numbers* or a family of *Boolean functions* being FOM-definable, and Section A fixes what each of the four means. Two features of those conventions are worth having in hand here. First, a number below n^c, for a constant c fixed in advance, is written in base n, so it is named by a c-tuple of elements of the universe rather than by a single variable. Second, a number the logic computes is described one digit at a time, because a formula returns a truth value and not a number; a definable family of reals is accordingly one with a definable sign relation and a definable digit relation, and Definition 5.1 is where that shape is asked for.

The next proposition fixes the outer limit of the logic, and it is the reason the paper can speak of networks and of formulas interchangeably. Its other side is a circuit class: TC^0, the queries decided by constant-depth polynomial-size families of Boolean circuits whose gates are unbounded fan-in AND, OR, NOT and majority gates, taken with the DLOGTIME-uniformity that makes such a family one object rather than a separate circuit chosen at each input length. Section A defines both.

Proposition 2.2 (Expressivity ceiling, (Barrington et al., 1990; Immerman, 1999))**.** *A query is* FOM*-definable if and only if it lies in* DLOGTIME*-uniform* TC^0*. In particular every query computed by a* DLOGTIME*-uniform, polynomial-size, bounded-depth, bounded-precision family of threshold networks is* FOM*-definable, and the parity query* χ_T *is* FOM*-definable for every definable family of sets* T*.*

One fact about FOM is used throughout the proofs below, and Section A states it: on numbers below n^c written as c-tuples in base n, the logic can quantify over them, compare them, extract their binary digits and count the tuples satisfying a definable condition (Immerman, 1999), so it handles such a number as though it were a single element of the universe. That is what a step below means when it calls something arithmetic on indices.

Two lemmas carry the paper's definability hypotheses over to its Fourier data, and Section B supplies both with their proofs: Lemma B.1 names the support sets and coefficients of a presented tame FOM-definable query as definable families, and Lemma B.2 names a leap-minimizing order of them definably, which is what lets Theorem 6.2 choose a derivation rather than read one off a trained network.

3 The Learning Model

3.1 Correlational Queries

We model gradient training of a two-layer network by correlational statistical queries, the standard abstraction for the correlations that a gradient extracts from data (Kearns, 1998; Blum et al., 1994; Bshouty & Feldman, 2002). Under the correlation loss $L(\theta) = -\mathbb{E}_x [f(x)N_\theta(x)]$, each coordinate of the gradient is $-\mathbb{E}_x [f(x)\,\partial N_\theta(x)/\partial\theta_j]$, a correlation with a test function fixed once the current parameter θ is fixed. When these derivative functions take values in $[-1, 1]$, a fresh independent sample of size $O(\tau^{-2}\log(1/\delta))$ estimates each correlation to within τ with probability $1 - \delta$, by Hoeffding's inequality, conditional on the preceding updates.

Definition 3.1 (Correlational query)**.** A *correlational statistical query* for a target g, with respect to the uniform distribution, is a pair (ψ, τ). Here ψ maps $\{-1, 1\}^n$ into $[-1, 1]$ and the tolerance τ is positive. The oracle returns a value v with $|v - \mathbb{E}_x [g(x)\psi(x)]| \le \tau$, chosen adversarially subject to that constraint. The learner accesses v through dyadic approximations: reading it to additive error 2^{-r} costs $O(r)$ time.

The learner chooses ψ, while the oracle holds the unknown target g. The adversarial reply models estimation error without assuming a particular estimator.

3.2 Learnability Is a Property of a Class

Learnability cannot be defined for one fixed target: an algorithm that must succeed only on a single known f_n may print f_n without querying anything, which is why correlational lower bounds are quantified over a class of targets that the learner cannot tell apart. We therefore attach to each query a class of targets: the smallest class that contains it and is closed under renaming coordinates.

Definition 3.2 (Orbit class)**.** The *orbit* of a query f is the family of classes $\mathcal{O}(f) = (\mathcal{O}_n(f))_{n\ge1}$ with $\mathcal{O}_n(f) = \{f_n^\sigma : \sigma \text{ a permutation of } [n]\}$.

The relabeling f_n^σ is fixed in Section 2.1: the learner knows the shape of f but not the placement of its relevant coordinates. Arity, the multiset of Fourier coefficients, and leap are invariant under coordinate permutations.

Definition 3.3 (Efficient learnability)**.** A class family $\mathcal{C} = (\mathcal{C}_n)$ is *learnable in time* $T(n, 1/\varepsilon)$ when there is a deterministic algorithm with the following property for every n, every accuracy $\varepsilon \in (0, 1)$ and every target $g \in \mathcal{C}_n$. Given n and ε and facing the target g, the algorithm makes correlational queries, each of tolerance at least $1/\mathrm{poly}(n, 1/\varepsilon)$. It runs in time at most $T(n, 1/\varepsilon)$. It outputs a hypothesis $h\colon \{-1, 1\}^n \to \{-1, 1\}$ with $\mathbf{Pr}_x [h(x) \ne g(x)] \le \varepsilon$. All three hold whatever answers the oracle returns subject to Definition 3.1. A query f is *efficiently learnable*, written $f \in \mathsf{Learn}$, when $\mathcal{O}(f)$ is learnable in time $\mathrm{poly}(n, 1/\varepsilon)$.

The inverse-polynomial lower bound on tolerance excludes the super-polynomial sample cost of arbitrarily precise estimates. The definition asks for a deterministic learner; the lower-bound obstructions also apply to randomized learners with constant success probability. A learner's *transcript* is the sequence of queries it makes together with the answers it receives. For the lower-bound families, the fraction of targets on which a fixed-coin learner succeeds is bounded uniformly over its coins; when that bound is $o(1)$, averaging preserves it.

The accuracy is a parameter rather than a constant, as in the strong form of the PAC model, and Proposition B.10 is why: at any fixed accuracy the equality of Theorem 6.2 fails.

The parity obstruction in this model is one lemma. The bound is the statistical-query dimension bound of Blum et al. (1994) in its correlational form, its consequence for parities is due to Bshouty & Feldman (2002), and the proof is the standard one. We prove the form needed by Theorem B.4: a correlation bound for a subfamily of the degree-k characters. Lemma B.3 in Section B states it with its proof: correlating at γ with every character of a family $\mathcal{P}$ costs at least $\left(|\mathcal{P}| - \gamma^{-2}\right)\tau^2$ queries of tolerance τ.

Remark 3.4 (From error to correlation)**.** Let h and g both be $\{-1, 1\}$-valued, so that the product $h(x)g(x)$ is 1 where the two agree and -1 where they differ. If $\mathbf{Pr}_x [h(x) \ne g(x)] \le \eta$ then $\mathbb{E}_x [h(x)g(x)] \ge 1 - 2\eta$. Run a learner of Definition 3.3 at accuracy $\varepsilon = 1/3$, at which its time

is polynomial in n alone. Its hypothesis then correlates with the target at $1 - 2/3 = 1/3$ or better, which is the case $\gamma = 1/3$ of Lemma B.3. A lower bound at that accuracy is a lower bound at every smaller value of ε.

4 Leap Complexity and Correlational Learning

Stochastic gradient descent discovers the relevant coordinates of a target a few at a time, and leap complexity measures the largest step in that climb (Abbe et al., 2023a); the case of a single step at a time is the merged-staircase property of Abbe et al. (2022).

Definition 4.1 (Leap)**.** Let f be nonzero with support $\mathrm{supp}(f) = \{S_1, \ldots, S_P\}$. An *ordering* of the support is a permutation π of $\{1, \ldots, P\}$, which lists the support sets as $S_{\pi(1)}, \ldots, S_{\pi(P)}$. For such an ordering, the *jump* at step i is the number of coordinates that $S_{\pi(i)}$ introduces beyond all earlier sets,

$$J_\pi(i) = \left| S_{\pi(i)} \setminus \bigcup_{j<i} S_{\pi(j)} \right|, \tag{2}$$

with the convention $\bigcup_{j<1} S_{\pi(j)} = \emptyset$. The *leap* of f is the smallest achievable largest jump,

$$\mathrm{Leap}(f) = \min_{\pi} \max_{1 \le i \le P} J_\pi(i). \tag{3}$$

The leap is invariant under permutations of the coordinates, so it is a property of an orbit rather than of a representative. In words, every listing of the support has a cost: the largest number of coordinates that any one of its steps introduces. The leap is the least cost over all listings. Throughout, *leap* names this invariant, and *width* names a bound on how many coordinates one step of an ordering, of a derivation or of an algorithm may introduce. Widths are nonnegative integers. An ordering whose jumps are all at most w is an ordering of width w, so that $\mathrm{Leap}(f) \le w$ says exactly that f admits an ordering of width w.

Example 4.2. The parity χ_T has a one-element support, so its only ordering has a single jump of size $|T|$ and $\mathrm{Leap}(\chi_T) = |T|$. A staircase such as $\chi_{\{1\}} + \chi_{\{1,2\}} + \cdots + \chi_{\{1,\ldots,k\}}$ admits an ordering whose jumps are each 1, so its leap is 1. A function that depends on its input only through one weighted sum of the coordinates behaves the same way, provided that all of its degree-one Fourier coefficients are nonzero. Each coordinate then has its own singleton in the support, so listing the singletons first introduces the coordinates one at a time, and every later set introduces nothing.

4.1 The Upper Bound: Search Where the Leap Allows

The adaptive search recovers the support and enough coefficient precision to recover the target exactly.

Theorem 4.3 (Learning at the leap, proved in Section B)**.** *Let f be tame with parameters (c, q), and let w be a width that bounds the leap at every length, so that $w \ge \mathrm{Leap}(f_n)$ for every n. Run Algorithm 1 with acceptance threshold $\tau = 1/q(n)$ and query tolerance $\theta = \min(\tau/3, n^{-c}/2)$. It makes $n^{O(w+c)}$ queries and runs in time $n^{O(w+c)}$. For every target $g \in \mathcal{O}_n(f)$ and every legal oracle it outputs the hypothesis $h = g$, the target itself. Hence $\mathcal{O}(f)$ is learnable in time $n^{O(w+c)}$, at every accuracy and at a tolerance that does not depend on the accuracy. In particular $f \in \mathsf{Learn}$ whenever $\mathrm{Leap}(f_n) = O(1)$.*

The search must also inspect subsets of the coordinates it has found.

Remark 4.4 (Where the arity bound is spent)**.** The proof spends the arity bound in one place that sparsity alone would not cover: the enumeration of the subsets $T \subseteq U$ of the coordinates already found, which costs a factor of $2^{|U|}$. Without a bound on $|U|$ that factor is not polynomial, and no argument here replaces it: a correlational query returns one number, and Algorithm 1 isolates the coefficient $\widehat{g}(T \cup F)$ only by naming T. The leap governs the search for coordinates the learner has not yet found; the arity governs the search among those it has.

4.2 The Arity Bound Is Necessary

Remark 4.4 says where the arity bound is spent; the expense is unavoidable in the worst case. The witness is the query of Theorem 1.3 of Section 1.1, which meets every requirement of Definition 2.1 except the arity bound, so Theorem 6.1 fails without that bound already at width one; Theorem B.4 in Section B states it with its proof.

The multiplexer makes the leap one while the arity is large: its support contains the singleton of every data coordinate, so degree-one queries find all $2k$ data coordinates at polynomial cost. What the learner cannot do is decide which coordinate of each data pair the secret uses. Nothing else in the target depends on that choice, because the multiplexer reads a pair only through a symmetric function of its two coordinates, as Eq. (12) in Section B sets it up. So the enumeration in Remark 4.4 is not an artifact of Algorithm 1: already at width one it is the problem itself.

The theorem addresses the growing-arity regime raised by Abbe et al. (2023a): the adaptive bounds of Joshi et al. (2024) are $\Theta(n^{\mathrm{Leap}})$ at constant tolerance and concern recovery of the relevant coordinates. Once those are known, the remaining problem has fixed size when the arity is constant; efficient learning can fail when the arity is $\omega(\log n)$. Recovering the relevant coordinates is cheap in this construction, since every one of them lies in a support set of size at most two with coefficient at least $1/(8k)$; what remains afterwards is the family of the 2^k transversals, and that costs $2^{\Omega(k)}$ queries, which is $2^{\Omega(\mathrm{arity})}$.

The theorem fixes the arity at $2k + \log_2 k + 2$ and leaves k free. Choosing k against a given bound gives the statement a reader wants: read together with Theorem 4.3, the logarithmic arity bound of Definition 2.1 is exactly the threshold. Corollary B.5 in Section B states that and proves it, against any FOM-definable arity bound $a(n) = \omega(\log n)$ that stays below $n/4$.

4.3 The Lower Bound

The converse of Theorem 4.3 is known for a fixed shape of constant arity, with constants that depend on the shape (Abbe et al., 2023a; Joshi et al., 2024). The characterization needs a bound with absolute constants, uniform over shapes that vary with n, for arity up to $c \log n$, and the paper proves one in its own model, from the three ingredients of Lemma B.3, namely Parseval, a count and Markov's inequality; of the two conditions of Definition 2.1 it uses only the arity bound, so its bound does not depend on q. Its accuracy requirement is the one Definition 3.3 lets us impose, and Proposition B.10 shows that a constant accuracy would not do. The bound is Proposition 1.2 of Section 1.1, and Proposition B.6 in Section B restates it with its hypotheses in full, adds the explicit form at tolerance $n^{-\beta}$, and proves it. The two bounds meet in Theorem B.7 of Section B: a tame query is efficiently learnable exactly when its leap is bounded.

5 Correlationally Visible FOM

We isolate the queries in FOM that admit Boolean features certified by residual correlations involving every newly introduced coordinate.

Definition 5.1 (Staircase derivation and sFOM)**.** Let $g\colon \{-1,1\}^n \to \{-1,1\}$ and let D be a product distribution on $\{-1,1\}^n$ under which no coordinate is constant. A *width-w correlationally visible staircase derivation* of g under D is a finite sequence of Boolean features $\psi_0, \psi_1, \ldots, \psi_m$, one for each position i from 0 to m. Each feature ψ_i carries with it a set of coordinates $C_i \subseteq [n]$, and ψ_i is a function of those coordinates alone. Write $U_i = \bigcup_{j \le i} C_j$ for the coordinates that the features up to position i use between them. Write $F_{i+1} = C_{i+1} \setminus U_i$ for the coordinates that ψ_{i+1} *introduces*, that is, the coordinates it uses that no earlier feature used. Write Π_i for the orthogonal projection in $L^2(D)$ onto the span of $\psi_0, \ldots, \psi_i$, and call $g - \Pi_i g$ the *residual* of g after the first $i+1$ features, the part of g that those features do not capture. The sequence is a derivation when the following four conditions hold.

- *Base.* The sequence starts from no coordinates at all: ψ_0 is the constant 1, so $C_0 = \emptyset$.
- *Bounded introduction.* No step introduces more than w coordinates: $|F_{i+1}| \le w$ for every $i < m$.

- *Correlational visibility.* At every $i < m$ the introduced coordinates carry the correlation: there is a set S with $F_{i+1} \subseteq S \subseteq C_{i+1}$ such that

$$\left|\widehat{(g - \Pi_i g)}^{D}(S)\right| \geq \frac{1}{\mathrm{poly}(n)} \qquad \text{and} \qquad \widehat{\psi_{i+1}}^{D}(S) \neq 0. \tag{4}$$

- *Readout.* The features, weighted and thresholded, give back g exactly: $g = \mathrm{sign}\left(\sum_i a_i \psi_i\right)$ for real weights a_i of $\mathrm{poly}(n)$-bit precision.

A derivation's *visibility margin* is the largest $\gamma = \gamma(n)$ for which Eq. (4) holds at every step with $1/\mathrm{poly}(n)$ replaced by γ. For a derivation with no steps, set the margin to $+\infty$. The margin is a function of n rather than a constant. For the derivations that Lemma B.8(2) builds, the margin is at least the inverse polynomial that bounds the nonzero Fourier coefficients of the target from below. A query f lies in $\mathsf{sFOM}_w(D)$ when two things hold. First, f is FOM-definable. Second, for every n the function f_n has a width-w derivation whose features and weights form FOM-definable families, indexed by n and by the position i in the sequence, with one polynomial visibility bound and one polynomial precision bound valid at every length. Spelled out, the second condition asks for two definable families. The relation that holds of n, i and x exactly when the feature ψ_i takes the value 1 at the input x must be FOM-definable. The weights (a_i) must form an FOM-definable family of reals in the sense of Section A.4. We write $\mathsf{sFOM}(D) = \bigcup_{w=O(1)} \mathsf{sFOM}_w(D)$ for the constant-width queries and drop D when it is uniform.

Bounded introduction caps how many coordinates a step may bring in, and visibility demands that those coordinates be the ones the correlation is about, since the witness S contains every coordinate the step introduces; this is the merged-staircase mechanism of Abbe et al. (2022) transcribed into the logic.

6 Soundness, Completeness, and the Separation

The bridge in Lemma B.8 of Section B identifies the least derivation width of a nonconstant query with its leap, provided its nonzero coefficients are inverse-polynomial; Corollary B.9 records the equality.

Theorem 6.1 (Soundness, proved in Section B)**.** *Let f be a tame query with parameters (c, q) and let $f \in \mathsf{sFOM}_w$. Then $\mathcal{O}(f)$ is learnable in time $n^{O(w+c)}$. Consequently every tame query in* sFOM *is efficiently learnable, that is* $\mathsf{sFOM} \cap \{\textit{tame}\} \subseteq \mathsf{Learn}$.

The converse is the half that needs the logic, since the derivation it builds has to be one a formula can name.

Theorem 6.2 (Completeness, restatement of Theorem 1.1, proved in Section B)**.** *Let f be a presented tame* FOM*-definable query with $f \in \mathsf{Learn}$. Then $f \in \mathsf{sFOM}_\ell$ for some constant ℓ. Together with Theorem 6.1, this gives* $\mathsf{sFOM} = \mathsf{Learn}$ *on presented tame* FOM*-definable queries.*

Parity separates the visible fragment from the full logic.

Theorem 6.3 (Unconditional separation, restatement of Theorem 1.4, proved in Section B)**.** *Let $k = k(n)$ be an* FOM*-definable function with $k(n) = \omega(1)$ and $k(n) \leq \log n$. Let $f_n = \chi_{[k(n)]}$ be the parity of the first $k(n)$ coordinates. Then f is presented, tame and* FOM*-definable, $f \notin \mathsf{sFOM}$, and $f \notin \mathsf{Learn}$. Hence* $\mathsf{sFOM} \subsetneq \mathsf{FOM}$*, and the sparse parity witnesses the separation.*

AI use statement

The authors posed the problem. Claude (Anthropic) helped develop the proofs, review the literature, and draft the manuscript. Codex (OpenAI) assisted with reviewing the mathematical arguments, citations, and prose of the ICLR submission. The authors reviewed the AI-assisted work and take responsibility for the paper and any errors. Every proof was rederived independently of the process that produced it, and every reference was checked against its source.

Under the ICLR 2027 AI Policy for Authors, model development, claim formulation, proof contributions, and proof writing require disclosure; drafting, literature identification, and reference format-

ting are recommended for disclosure. Other required tasks involving empirical work or translation do not arise here.

Reproducibility statement

This is a theoretical paper with no experiments. Every statement in the main text carries a complete proof in Section B, and the assumptions each result needs are stated with it. Definition 2.1 fixes the regime the results hold in, and Definition 3.1 and Definition 3.3 fix the learning model. The imported expressivity ceiling Proposition 2.2 is attributed where it is used and places the witnesses inside FOM. The counting and arithmetic facts used in the definability proofs are cited in Section A.5. The witnesses of Theorem B.4 and Proposition B.10 were additionally checked in exact rational arithmetic at small parameters. The counting argument of Proposition B.6 is proved in full in the appendix.

References

Emmanuel Abbe and Colin Sandon. Polynomial-time universality and limitations of deep learning. *Communications on Pure and Applied Mathematics*, 76(11):3493–3549, 2023. doi: 10.1002/cpa.22121. Conference version: On the universality of deep learning, NeurIPS 2020.

Emmanuel Abbe, Enric Boix-Adserà, Matthew Brennan, Guy Bresler, and Dheeraj Nagaraj. The staircase property: How hierarchical structure can guide deep learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 26989–27002. Curran Associates, Inc., 2021a.

Emmanuel Abbe, Pritish Kamath, Eran Malach, Colin Sandon, and Nathan Srebro. On the power of differentiable learning versus PAC and SQ learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34. Curran Associates, Inc., 2021b.

Emmanuel Abbe, Enric Boix-Adserà, and Theodor Misiakiewicz. The merged-staircase property: A necessary and nearly sufficient condition for SGD learning of sparse functions on two-layer neural networks. In *Proceedings of the 35th Conference on Learning Theory (COLT)*, volume 178 of *Proceedings of Machine Learning Research*, pp. 4782–4887. PMLR, 2022.

Emmanuel Abbe, Enric Boix-Adserà, and Theodor Misiakiewicz. SGD learning on neural networks: Leap complexity and saddle-to-saddle dynamics. In *Proceedings of the 36th Conference on Learning Theory (COLT)*, volume 195 of *Proceedings of Machine Learning Research*, pp. 2552–2623. PMLR, 2023a.

Emmanuel Abbe, Elisabetta Cornacchia, and Aryo Lotfi. Provable advantage of curriculum learning on parity targets with mixed inputs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 24291–24321. Curran Associates, Inc., 2023b.

Pablo Barceló, Egor V. Kostylev, Mikaël Monet, Jorge Pérez, Juan L. Reutter, and Juan Pablo Silva. The logical expressiveness of graph neural networks. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.

David A. Mix Barrington, Neil Immerman, and Howard Straubing. On uniformity within NC^1. *Journal of Computer and System Sciences*, 41(3):274–306, 1990.

Avrim Blum, Merrick L. Furst, Jeffrey C. Jackson, Michael J. Kearns, Yishay Mansour, and Steven Rudich. Weakly learning DNF and characterizing statistical query learning using Fourier analysis. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 253–262, 1994.

Nader H. Bshouty and Vitaly Feldman. On using extended statistical queries to avoid membership queries. *Journal of Machine Learning Research*, 2:359–395, 2002.

Yatin Dandi, Emanuele Troiani, Luca Arnaboldi, Luca Pesce, Lenka Zdeborová, and Florent Krzakala. The benefits of reusing batches for gradient descent in two-layer networks: Breaking the curse of information and leap exponents. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9991–10016. PMLR, 2024.

Amit Daniely and Eran Malach. Learning parities with neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 20356–20365. Curran Associates, Inc., 2020.

Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984.

Surbhi Goel, Aravind Gollakota, and Adam R. Klivans. Statistical-query lower bounds via functional gradients. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 2147–2158. Curran Associates, Inc., 2020.

Aravind Gollakota, Sushrut Karmalkar, and Adam R. Klivans. The polynomial method is universal for distribution-free correlational SQ learning. *arXiv preprint arXiv:2010.11925*, 2020. Revised 2023, version 3.

Martin Grohe. The descriptive complexity of graph neural networks. *TheoretiCS*, 3(25), 2024. doi: 10.46298/theoretics.24.25. Conference version in LICS 2023, pp. 1–14; arXiv:2303.04613.

Martin Grohe and Martin Ritzert. Learning first-order definable concepts over structures of small degree. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pp. 1–12. IEEE, 2017. doi: 10.1109/LICS.2017.8005080.

Martin Grohe and György Turán. Learnability and definability in trees and similar structures. *Theory of Computing Systems*, 37(1):193–220, 2004. doi: 10.1007/s00224-003-1112-8. Conference version in STACS 2002, pp. 645–657.

Johan Håstad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 6–20, 1986.

Neil Immerman. *Descriptive Complexity*. Graduate Texts in Computer Science. Springer, New York, 1999.

Nirmit Joshi, Theodor Misiakiewicz, and Nathan Srebro. On the complexity of learning sparse functions with statistical and gradient queries. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 103198–103241. Curran Associates, Inc., 2024.

Michael J. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45 (6):983–1006, 1998.

Leonid Libkin. *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, Berlin, 2004.

William Merrill and Ashish Sabharwal. A logic for expressing log-precision transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 52453–52463. Curran Associates, Inc., 2023.

Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

Yunwei Ren, Zihao Wang, and Jason D. Lee. Algorithmic separation between constant-depth and logarithmic-depth neural networks. *arXiv preprint arXiv:2607.25200*, 2026.

Steffen van Bergerem. Learning concepts definable in first-order logic with counting. *Logical Methods in Computer Science*, 21(3), 2025. doi: 10.46298/lmcs-21(3:9)2025. Article 9. Conference version in LICS 2019, pp. 1–13.

A Background on the Expansion and on the Logic

This appendix holds what the main text points to here. It carries the Fourier expansion under a product distribution and the identity that relabeling coordinates satisfies. It also carries the encoding of an input as a structure, and it fixes the conventions by which a relation, a function, a family of reals or a family of Boolean functions is called FOM-definable. Last, it records the closure fact the proofs use and the circuit class that Proposition 2.2 names. None of it is new and none of it is changed. It is the material of Section 2 in full, drawn on by the main text and by the proofs of Section B alike.

A.1 The Expansion under a Product Distribution

The same expansion is available under a distribution that is not uniform, provided the coordinates stay independent under it. Definition 5.1 is stated in terms of such a distribution, and every result in this paper is proved under the uniform one. Let D be a product distribution on $\{-1,1\}^n$, which is to say that the coordinates are independent under D. Write μ_i for the mean $\mathbb{E}_D[x_i]$ of coordinate i and v_i for its variance $\mathrm{Var}_D(x_i)$, which we take to be positive, so that no coordinate is constant. The function $\varphi_i(x) = (x_i - \mu_i)/\sqrt{v_i}$ is coordinate i recentered to mean zero and rescaled to variance one. Their products $\varphi_S = \prod_{i\in S}\varphi_i$ take the place of the characters: they form an orthonormal basis of $L^2(D)$, the space of real functions on $\{-1,1\}^n$ under the inner product $\mathbb{E}_D[gh]$. We write $\widehat{f}^D(S) = \mathbb{E}_D[f\varphi_S]$ for the coefficient of f at S in that basis. Under the uniform distribution every μ_i is zero and every v_i is one, so $\varphi_S = \chi_S$ and $\widehat{f}^D = \widehat{f}$, and we drop the superscript.

A.2 Relabeling the Coordinates

For a permutation σ of $[n]$, we write f^σ for the function $x \mapsto f(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$. This is f read with its coordinates taken in the order that σ prescribes. For a set $S \subseteq [n]$ write $\sigma(S) = \{\sigma(i) : i \in S\}$ for its image. Substituting $y_i = x_{\sigma(i)}$ into Eq. (1) gives $f^\sigma = \sum_S \widehat{f}(S)\chi_{\sigma(S)}$. Reading that expansion off, $\mathrm{supp}(f^\sigma) = \{\sigma(S) : S \in \mathrm{supp}(f)\}$, and the coefficient of f^σ at $\sigma(S)$ is $\widehat{f}(S)$. Relabeling therefore moves the support and carries each coefficient along with its set.

A.3 An Input as a Structure

A logical formula does not read a string; it is interpreted in a structure, which is a set of elements together with relations on them. We therefore encode an input $x \in \{-1,1\}^n$ as a finite structure whose elements are the coordinate positions $[n]$, and we equip that structure with three ingredients. The first is the linear order on the positions. The second is the BIT predicate on zero-based order ranks: $\mathrm{BIT}(u,v)$ holds exactly when digit $v-1$ of the binary expansion of $u-1$ is one. The third is a unary predicate X with $X(i)$ true exactly when $x_i = 1$. The order and BIT say nothing about x and are present in every structure of size n; what they provide is arithmetic on positions, which is what the two lemmas below use whenever they compute with indices. Only X depends on the input, and it is what makes the input readable one coordinate at a time.

A.4 What It Means for a Relation, a Function or a Family To Be Definable

Several definitions below speak instead of a *relation* being FOM-definable, and one convention covers all of them. The relations the paper needs take numbers as arguments, and those numbers can be larger than n, so the first thing to fix is how a number too large to be an element of the universe is handled. A number i below n^c is written in base n, where it has at most c digits. The digit $d \in \{0, \ldots, n-1\}$ is represented by the element $d+1$ of $[n]$; numerical arguments use these zero-based ranks, while coordinate labels remain $1, \ldots, n$. Such an i is therefore named not by one variable but by a c-tuple of variables, one per digit, with the constant c fixed in advance. At $n = 1$ the sole tuple denotes zero, and exceptional values needed only at that length are handled directly by finite formulas. Taking c to be a positive integer loses nothing: any positive real arity constant can be rounded up. Likewise, any positive polynomial denominator bound can be enlarged to a positive polynomial with integer coefficients, so the convention on q in Definition 2.1 loses no query and makes the learner's thresholds finitely specified.

Now consider a relation whose arguments are the size n together with numbers $i_1, \ldots, i_k$, each below n^c. The relation is FOM-definable when there is a formula $\varphi(\bar{y}_1, \ldots, \bar{y}_k)$, with each $\bar{y}_j$ a c-tuple of variables, satisfying the following condition. The relation holds of n and $i_1, \ldots, i_k$ exactly when φ is satisfied in the structure of size n by the base-n digit tuples of $i_1, \ldots, i_k$. The size n is not an argument of the formula at all; it is the size of the structure the formula is evaluated in. When the relation also takes an input $x \in \{-1,1\}^n$, that input is likewise not an argument: it is the structure, read through the predicate X as above.

Three further kinds of object are asked to be definable below, and each is reduced to a relation in the sense just fixed. A function $k\colon \mathbb{N} \to \mathbb{N}$ is FOM-*definable* when its values are readable digit by digit.

This means that the relation holding of n and m exactly when $k(n)$ has a one in binary digit m is itself FOM-definable. Functions enter this way because a formula returns a truth value rather than a number, so a number the logic computes has to be described one digit at a time.

A family of real numbers is described the same way, by its sign and its digits. Let $(a_{n,i})$ be a family of reals indexed by the size n and by a number i, each of absolute value at most one and each with a terminating binary expansion. The family is FOM-*definable* when the following two relations are FOM-definable. The *sign relation* holds of n and i exactly when $a_{n,i} < 0$. The *digit relation* holds of n, i and $m \geq 0$ exactly when digit m of $|a_{n,i}|$ is one in that expansion. Here digit 0 is the units digit and digit $m \geq 1$ is the m-th digit after the binary point. The units digit is what distinguishes $|a_{n,i}| = 1$ from 0, and it is the only reason the index starts at zero. Every weight family below is of this shape, and Definition 5.1 is where the shape is asked for.

Last, a family of Boolean functions is FOM-definable when one relation describes the whole family at once. That relation holds of n, i and x exactly when the i-th function of the family takes the value 1 at x, and it must itself be FOM-definable. Here x is the structure and i is a number below n^c, as the convention above prescribes. So the formula has c free variables and reads x through the predicate X.

A.5 Arithmetic on Indices

One fact about FOM is used throughout the proofs below, and it is stated here so that later steps can appeal to it rather than repeat it. Recall that a number below n^c, for a fixed constant c, is a c-tuple of elements of the universe written in base n. For such tuples, FOM is closed under quantifying over them, comparing them, extracting their binary digits, and counting the tuples that satisfy a definable condition (Barrington et al., 1990, Propositions 11.3–11.4); see also Immerman (1999). In effect the logic handles a number below n^c as though it were a single element of the universe, and that is what a step below means when it calls something arithmetic on indices.

A.6 Threshold Circuits and DLOGTIME-Uniformity

The other side of Proposition 2.2 is a circuit class. Here TC^0 is the class of queries decided by families of Boolean circuits of constant depth and polynomial size whose gates are unbounded fan-in AND, OR, NOT and majority gates, majority gates being interchangeable with threshold gates. DLOGTIME-uniformity is the further condition that makes such a family one object rather than a separate circuit chosen at each input length. It requires that the structure of the circuit for length n be decidable in time logarithmic in n, the notion fixed by Barrington et al. (1990) and developed by Immerman (1999).

B Proofs

This appendix carries the complete proof of every statement in the paper. A statement the main text only points to is stated here in full, in front of its proof. The first two lemmas below carry the paper's definability hypotheses over to its Fourier data, and Theorem 6.2 is where they are spent. Nothing is sketched and nothing is deferred further.

Lemma B.1 (The spectrum of a presented query is definable)**.** *Let f be a presented tame* FOM*-definable query with parameters (c, q). Write R_n for the set of coordinates relevant to f_n. Index the subsets of R_n by the numbers $j < 2^{|R_n|} \leq n^c$, taking the binary digits of j to be the characteristic vector of the subset. That is, number the elements of R_n from 0 upward in increasing order, and let binary digit m of j be one exactly when the element numbered m lies in the subset. Write S_j for the subset coded by j. Then three things are* FOM*-definable: the relation that holds of n, a code j and a coordinate i exactly when $i \in S_j$; the family of characters $\left(\chi_{S_j}\right)$; and the family of coefficients $\left(\widehat{f_n}(S_j)\right)$, the last two indexed by n and j.*

Proof of Lemma B.1. At $n = 1$, tameness forces $R_1 = \emptyset$. The only subset code is zero, membership is always false, its character is the constant 1, and its coefficient is the constant value $f_1 \in \{-1, 1\}$. The coefficient's absolute value has units digit one and every fractional digit zero, so all three families are definable directly at this length. Henceforth assume $n \geq 2$. *The relation $i \in S_j$ is definable.*

Tameness gives $|R_n| \leq c \log n$, so $2^{|R_n|} \leq 2^{c \log n} = n^c$. So every code j is a number below n^c, and a c-tuple of elements of the universe names it, as Section A.4 prescribes. Number the elements of R_n from 0 upward along the linear order on $[n]$, as in the statement. So a coordinate i has index m exactly when two conditions hold: $i \in R_n$, and exactly m relevant coordinates precede i. Here $m < |R_n| \leq n$, so m is a number below n and a single variable names it. Each of the two conditions on i is definable. That i is relevant to f_n is definable because f is presented, which is what presentedness says. That exactly m relevant coordinates precede i says that the number of coordinates $i' < i$ satisfying the definable relation "i' is relevant to f_n" equals m. The logic computes the number of elements satisfying a definable condition and compares it with m (Section A.5, applied to tuples of length one). So the relation "i has index m in R_n" is FOM-definable. The relation "$i \in S_j$" then says two things: that i has some index m in R_n, and that binary digit m of j is one. The relation just shown definable supplies the first of these. The second, that binary digit m of j is one, asks for a binary digit of j, a number below n^c named by a c-tuple rather than by a single element. The logic extracts the binary digits of such a number (Section A.5), whereas the BIT predicate alone reads only the digits of a single element. Hence the relation "$i \in S_j$", of n, j and i, is FOM-definable, which is the first claim of the lemma.

The family of characters is definable. Each character χ_{S_j} is then the parity of a definable set of coordinates, and the parity half of Proposition 2.2 carries over with the code j as a free variable, as follows. The value $\chi_{S_j}(x)$ is 1 exactly when an even number of the coordinates $i \in S_j$ have $x_i = -1$, that is, satisfy $\neg X(i)$. The number of those coordinates is the number of elements i satisfying the definable condition "$i \in S_j$ and $\neg X(i)$", a count the logic computes (Section A.5). That number is even exactly when its binary digit 0 is zero. So the relation that holds of n, j and x exactly when $\chi_{S_j}(x) = 1$ is FOM-definable, which is what definability of the family of characters asks (Section A.4).

The coefficient is a finite sum. Two things enter here: the function f_n depends only on the coordinates in R_n, and so does each character χ_{S_j}, because its set S_j is a subset of R_n. The first holds because Eq. (1) writes f_n as a combination of the characters χ_S with $S \in \mathrm{supp}(f_n)$, and each such S lies inside R_n by the definition of relevance. So the product $f_n \chi_{S_j}$ takes the same value at any two inputs that agree on R_n. Group the 2^n inputs by their restriction to R_n. There are $2^{|R_n|}$ groups, each of $2^{n-|R_n|}$ inputs, and each group contains exactly one input y that takes the value 1 outside R_n. The average $\widehat{f_n}(S_j) = 2^{-n} \sum_x f_n(x)\chi_{S_j}(x)$ is therefore $2^{-n} \cdot 2^{n-|R_n|} \sum_y f_n(y)\chi_{S_j}(y)$, so

$$\widehat{f_n}(S_j) = 2^{-|R_n|} \sum_y f_n(y)\chi_{S_j}(y), \tag{5}$$

the sum running over the at most n^c assignments y that take the value 1 outside R_n. Each summand is $+1$ or -1, since f_n and χ_{S_j} are both $\{-1, 1\}$-valued. Index the assignments in the sum by the codes $k < 2^{|R_n|}$: the assignment y_k takes the value -1 exactly at the coordinates of S_k and the value 1 at every other coordinate. An assignment in the sum is fixed by the set of coordinates of R_n at which it takes the value -1, and that set is S_k for exactly one code k. So $k \mapsto y_k$ is a bijection from the codes onto the assignments in the sum.

Each summand's sign is definable. Whether the summand at y_k is $+1$ is FOM-definable from n, j and k, for the following reasons. Let φ be a sentence that defines f, so that $f_n(x) = 1$ exactly when the structure of x satisfies φ. The structure of y_k has the same universe, order and BIT predicate as every structure of size n, and its predicate X holds at i exactly when $i \notin S_k$. Replace every atomic formula $X(t)$ in φ by the formula "$t \notin S_k$", which the first part of this proof makes FOM-definable. In that replacement formula, choose the variables that name k to be distinct from the variables of φ. Substituting a definable relation for a relation symbol an FOM formula uses leaves an FOM formula. So the result is an FOM formula with the variables naming k free, and it holds exactly when $f_n(y_k) = 1$. The summand at y_k is the product $f_n(y_k)\chi_{S_j}(y_k)$, and its other factor is $\chi_{S_j}(y_k) = (-1)^{|S_j \cap S_k|}$, since y_k takes the value -1 exactly on S_k. The parity of $|S_j \cap S_k|$ is binary digit 0 of the number of coordinates i with $i \in S_j$ and $i \in S_k$, a count the logic computes. The summand is $+1$ exactly when its two factors are equal, so the relation "the summand at y_k is $+1$", of n, j and k, is FOM-definable.

The sign relation is definable. What remains is to add the summands up inside the logic, and since each is $+1$ or -1 this is a matter of counting. Write N_+ for the number of codes $k < 2^{|R_n|}$ whose summand is $+1$, and N_- for the number whose summand is -1; both depend on n and j. The

bound $k < 2^{|R_n|}$ is itself definable. It says that no binary digit of k at a position at least $|R_n|$ is one, and $|R_n|$ is the number of relevant coordinates, a count the logic computes. So each of N_+ and N_- is the number of c-tuples satisfying a definable condition, and the logic computes both, compares them and extracts their binary digits (Section A.5). Both are at most $2^{|R_n|} \leq n^c$, so they are below n^{c+1} and $(c+1)$-tuples name them. Eq. (5) now reads $\widehat{f_n}(S_j) = 2^{-|R_n|}(N_+ - N_-)$, where $N_+ + N_- = 2^{|R_n|}$. Since $2^{-|R_n|} > 0$, the coefficient is negative exactly when $N_+ < N_-$, a comparison of two definable numbers. So the sign relation is FOM-definable.

The digit relation is definable. For the digits, let N be the larger of N_+ and N_-. Its binary digits are definable, since the comparison of N_+ with N_- decides which of the two it is. Then $|N_+ - N_-| = 2N - 2^{|R_n|}$, so $\left|\widehat{f_n}(S_j)\right| = 2^{1-|R_n|}N - 1$. Also $2^{|R_n|-1} \leq N \leq 2^{|R_n|}$, since N is at least half of $N_+ + N_- = 2^{|R_n|}$. Suppose first that $N = 2^{|R_n|}$, which happens exactly when binary digit $|R_n|$ of N is one. Then $\left|\widehat{f_n}(S_j)\right| = 1$, so its units digit is one and every other digit is zero. Suppose instead that $N < 2^{|R_n|}$. Then $|R_n| \geq 1$, because for $|R_n| = 0$ there is a single summand and $N = 1 = 2^{|R_n|}$. So $2^{|R_n|-1} \leq N < 2^{|R_n|}$, binary digit $|R_n|-1$ of N is one, and $N = 2^{|R_n|-1} + N'$, where N' is N with that digit cleared. Then $\left|\widehat{f_n}(S_j)\right| = 2^{1-|R_n|}N'$. A binary digit $b \leq |R_n| - 2$ of N' is also digit b of N. It contributes $2^{b-(|R_n|-1)}$ to the coefficient's absolute value, which is to say that it becomes digit $m = |R_n| - 1 - b$ after the binary point. So in this case the units digit is zero, digit m is binary digit $|R_n| - 1 - m$ of N for $1 \leq m \leq |R_n| - 1$, and every digit $m \geq |R_n|$ is zero. The position $|R_n| - 1 - m$ is itself a count, the number of t with $m < t < |R_n|$. So the digit relation is FOM-definable, in the convention of Section A.4 that digit 0 is the units digit.

The family of coefficients is definable. The family also meets the two conditions that convention puts on a family of reals. Its values satisfy $\left|\widehat{f_n}(S_j)\right| \leq 1$, because $|N_+ - N_-| \leq N_+ + N_- = 2^{|R_n|}$. Their binary expansions terminate, because each is an integer divided by $2^{|R_n|}$. The sign relation and the digit relation are therefore both FOM-definable, which is what the definition of a definable family of reals in Section A.4 asks. □

The second lemma puts those sets in an order, which is what Theorem 6.2 needs in order to choose a derivation rather than read one off a trained network.

Lemma B.2 (A leap-minimizing order can be chosen definably)**.** *Let f be a presented tame* FOM*-definable query with parameters (c, q), and let w be a constant. Suppose that* $\mathrm{Leap}(f_n) \leq w$ *for every n. Write R_n for the set of coordinates relevant to f_n. Code the subsets of R_n by the numbers $j < 2^{|R_n|}$, and write S_j for the subset coded by j, both as in Lemma B.1. Then there is a family of linear orders, one on* $\mathrm{supp}(f_n)$ *for each n, with the following two properties. Every jump in every one of these orders is at most w. The rank function is* FOM*-definable. By that we mean that one relation is* FOM*-definable: the relation that holds of n, a code j and a number i exactly when S_j occupies position i.*

Proof of Lemma B.2. Fix n and write $\mathcal{S} = \mathrm{supp}(f_n)$ for the support and $R = R_n$ for the relevant coordinates. Every relevant coordinate lies in some support set, and every support set consists of relevant coordinates, so $\bigcup \mathcal{S} = R$. Tameness gives $|R| \leq c \log n$. Throughout, Y and Z range over subsets of R, each coded by a number below $2^{|R|} \leq n^c$ as in Lemma B.1.

A closure operator. The closure under bounded jumps defined next, and the reachable sets it determines, are the objects the leap literature already works with, at width one in Abbe et al. (2022) and in general in Joshi et al. (2024). What is new here is only that the closure is used to name an order definably. Two notions are needed, and both are relative to a pair of sets $Z \subseteq Y \subseteq R$. Call a support set $S \in \mathcal{S}$ *available at Z inside Y* when $S \subseteq Y$ and $|S \setminus Z| \leq w$. In words: S stays within Y, and S adds at most w coordinates to Z. Say that Z is *w-closed in Y* when every set available at Z inside Y is contained in Z. That is to say that a step of width w taken inside Y gets nothing new out of Z. The whole of Y is w-closed in itself, since availability inside Y already asks that $S \subseteq Y$.

The w-closed subsets of Y are also closed under intersection. Let Z and Z' both be w-closed in Y, and let S be available at $Z \cap Z'$ inside Y. Both $S \setminus Z$ and $S \setminus Z'$ are contained in $S \setminus (Z \cap Z')$, so $|S \setminus (Z \cap Z')| \leq w$ gives $|S \setminus Z| \leq w$ and $|S \setminus Z'| \leq w$. Availability at $Z \cap Z'$ inside Y also

gives $S \subseteq Y$. So S is available at Z inside Y, and it is available at Z' inside Y as well. Now Z is w-closed in Y and so contains every set available at Z inside Y, and the same holds of Z', so $S \subseteq Z$ and $S \subseteq Z'$. Therefore $S \subseteq Z \cap Z'$. There are finitely many subsets of Y, and Y itself is one of the w-closed ones, so the collection of w-closed subsets of Y is finite and nonempty. Applying the fact just proved to two sets at a time, an induction on the number of sets intersected shows that the intersection of all of them is again w-closed in Y. That intersection lies inside every w-closed subset of Y, so it is the smallest subset of Y that is w-closed in Y. Write it $\langle Y \rangle$, suppressing the fixed w from the notation.

Equivalently $\langle Y \rangle$ is the limit of the greedy process inside Y,

$$Z_0 = \emptyset, \qquad Z_{r+1} = Z_r \cup \bigcup \{S \in \mathcal{S} : S \subseteq Y, |S \setminus Z_r| \leq w\}, \tag{6}$$

each step adding to Z_r exactly the sets available at Z_r inside Y. The sets Z_r only grow and all of them lie inside the finite set Y, so after at most $|Y|$ strict increases there is an r with $Z_{r+1} = Z_r$. From then on the process stops changing, because Z_{r+2} is computed from Z_{r+1} by the rule that computed Z_{r+1} from Z_r, and $Z_{r+1} = Z_r$. Its limit is the set it stops at.

That limit is w-closed in Y. The reason is that the step taken at the limit added nothing, that is, every set available at the limit inside Y was already contained in it. The limit also lies inside every set that is w-closed in Y, and an induction on r shows it. Let W be w-closed in Y. The claim holds at $r = 0$, since $Z_0 = \emptyset$ lies inside W. Suppose it holds at r, so that Z_r lies inside W. Let S be available at Z_r inside Y. Then $S \subseteq Y$, and $Z_r \subseteq W$ gives $S \setminus W \subseteq S \setminus Z_r$, so $|S \setminus W| \leq |S \setminus Z_r| \leq w$. So S is available at W inside Y, and it lies inside W, because W is w-closed in Y. The step of Eq. (6) adds exactly those sets to Z_r, which puts Z_{r+1} inside W as well. The limit is therefore the smallest w-closed subset of Y, which is exactly $\langle Y \rangle$.

Call Y *reachable* when $\langle Y \rangle = Y$. Since $\langle Y \rangle$ is the smallest subset of Y that is w-closed in Y, reachability says that no proper subset of Y is w-closed in Y. The empty set is reachable, since it has no proper subset. So when $\emptyset \in \mathcal{S}$, the level function defined below gives $\emptyset$ the value 0. Every nonempty support set S has level at least $|S| \geq 1$, since a set containing S has at least $|S|$ elements. Hence $\emptyset$, when it is a support set, comes first in the order.

R is reachable. Because $\mathrm{Leap}(f_n) \leq w$, the minimum in Eq. (3) is at most w, so some ordering $S^{(1)}, \ldots, S^{(P)}$ of the P sets of $\mathcal{S}$ has every jump at most w. The superscripts count positions in that ordering; they are not the codes of Lemma B.1. Fix such an ordering, and run the greedy process Eq. (6) with $Y = R$. Write $Q_r = \bigcup_{i \leq r} S^{(i)}$ for the union of the first r sets of that ordering. Then $Q_P = \bigcup \mathcal{S} = R$, and the jump bound reads $\left|S^{(r+1)} \setminus Q_r\right| \leq w$ for $0 \leq r < P$. An induction on r gives $Q_r \subseteq Z_r$. It holds at $r = 0$, where both sides are empty. Suppose it holds at r. Then $S^{(r+1)} \setminus Z_r \subseteq S^{(r+1)} \setminus Q_r$, so $\left|S^{(r+1)} \setminus Z_r\right| \leq \left|S^{(r+1)} \setminus Q_r\right| \leq w$, and $S^{(r+1)} \subseteq R = Y$. So $S^{(r+1)}$ is available at Z_r inside Y, and the step puts it inside Z_{r+1}. The union Q_{r+1} is $Q_r \cup S^{(r+1)}$, and $Q_r \subseteq Z_r \subseteq Z_{r+1}$, so $Q_{r+1} \subseteq Z_{r+1}$. The chain only grows, so Z_P lies inside its limit $\langle R \rangle$, and hence $\langle R \rangle \supseteq Z_P \supseteq Q_P = R$. Also $\langle R \rangle \subseteq R$, since $\langle R \rangle$ is by definition a subset of R. So $\langle R \rangle = R$, which says that R is reachable.

The order. For $S \in \mathcal{S}$ let the *level* of S be the size of the smallest reachable set containing it, that is, $\mathrm{lev}(S) = \min\{|Y| : Y \text{ reachable and } S \subseteq Y\}$. The minimum is over a nonempty collection, since R is reachable and contains S, so the level is well defined. Now order $\mathcal{S}$ by level, breaking ties by the code of Lemma B.1. That is, S precedes S' when $\mathrm{lev}(S) < \mathrm{lev}(S')$, or when $\mathrm{lev}(S) = \mathrm{lev}(S')$ and the code of S is smaller than the code of S'. Distinct subsets of R have distinct codes, so distinct support sets have distinct pairs $(\mathrm{lev}(S), \text{code of } S)$. The order is the lexicographic order on those pairs, and so it is a linear order on $\mathcal{S}$. A set of low level is one contained in a small reachable set, and the order puts such sets early.

The order in which the greedy process inside R first adds the support sets would also have every jump at most w, but it is not the order used here. The round at which a set is added is reached by iterating the step as many as $|R|$ times, a number that grows with n. A single formula of fixed quantifier depth does not directly express an iteration of growing length. The level is instead defined by a single quantification over codes, as the last part of this proof shows.

Every jump is at most w. Fix a support set $S \in \mathcal{S}$. The definition of the level supplies a reachable set Y with $S \subseteq Y$ and $|Y| = \mathrm{lev}(S)$, and we fix one. If $Y = \emptyset$ then $S = \emptyset$, since $S \subseteq Y$, and the

jump of $\emptyset$ is 0. So assume $Y \neq \emptyset$, and let $Z_0 \subseteq Z_1 \subseteq \cdots$ be the greedy chain Eq. (6) inside Y. That chain reaches Y: its limit is $\langle Y \rangle$, and $\langle Y \rangle = Y$ because Y is reachable. Let m be the least index with $Z_m = Y$. Since $Z_0 = \emptyset$ and $Y \neq \emptyset$, we have $m \geq 1$. And Z_{m-1} is a proper subset of Y, since m is the least index with $Z_m = Y$.

First, Z_{m-1} is reachable. To see this, write $Z'_0 \subseteq Z'_1 \subseteq \cdots$ for the greedy chain Eq. (6) run inside Z_{m-1} in place of Y. We show by induction on r that $Z'_r = Z_r$ for every $r \leq m-1$. At $r = 0$ both are empty. Suppose $Z'_r = Z_r$ for some $r < m-1$. A set available at Z_r inside Z_{m-1} is available at Z_r inside Y, because $Z_{m-1} \subseteq Y$. Conversely, a set available at Z_r inside Y is added by the step from Z_r, so it lies in Z_{r+1}, and $Z_{r+1} \subseteq Z_{m-1}$ because $r+1 \leq m-1$. So that set is available at Z_r inside Z_{m-1} as well. The two steps therefore add the same sets to the same set, and $Z'_{r+1} = Z_{r+1}$. At $r = m-1$ this gives $Z'_{m-1} = Z_{m-1}$, so the chain inside Z_{m-1} reaches all of Z_{m-1}, and its limit $\langle Z_{m-1} \rangle$ contains Z_{m-1}. Since $\langle Z_{m-1} \rangle$ is also a subset of Z_{m-1}, it equals Z_{m-1}.

Second, Z_{m-1} is a union of members of $\mathcal{S}$ that are contained in it, namely the sets the process added at steps $1, \ldots, m-1$. When $m = 1$ that union is empty. Each of those members is contained in the reachable set Z_{m-1}, so each has level at most $|Z_{m-1}|$. And $|Z_{m-1}| < |Y| = \mathrm{lev}(S)$, because Z_{m-1} is a proper subset of Y. So each of those members precedes S, the order being by level. Write U for the union of the sets preceding S in the order, so that $Z_{m-1} \subseteq U$.

Third, let T be one of the sets the last step adds, and consider the set $Z_{m-1} \cup T$. Since the last step adds exactly the sets available at Z_{m-1} inside Y, the set T is a member of $\mathcal{S}$ with $T \subseteq Y$ and $|T \setminus Z_{m-1}| \leq w$. That set $Z_{m-1} \cup T$ is reachable, by the argument that made Z_{m-1} reachable, run with $Z_{m-1} \cup T$ in place of Z_{m-1} and carried one step further. Write $Z''_0 \subseteq Z''_1 \subseteq \cdots$ for the greedy chain run inside $Z_{m-1} \cup T$. That set is contained in Y, because $Z_{m-1} \subseteq Y$ and $T \subseteq Y$, so every set available at Z_r inside it is available at Z_r inside Y. Every set the process inside Y adds at the step from Z_r with $r < m-1$ lies in $Z_{r+1} \subseteq Z_{m-1} \subseteq Z_{m-1} \cup T$, and so is available at Z_r inside $Z_{m-1} \cup T$ as well. Induction on r now gives $Z''_r = Z_r$ for every $r \leq m-1$. Both are empty at $r = 0$. If $Z''_r = Z_r$ with $r < m-1$, the two facts just stated say that the step from Z_r inside $Z_{m-1} \cup T$ and the step from Z_r inside Y add the same sets, so $Z''_{r+1} = Z_{r+1}$. The next step starts from $Z''_{m-1} = Z_{m-1}$ and absorbs T, which is available at Z_{m-1} inside $Z_{m-1} \cup T$ because $T \subseteq Z_{m-1} \cup T$ and $|T \setminus Z_{m-1}| \leq w$. So $Z''_m \supseteq Z_{m-1} \cup T$, and the chain inside $Z_{m-1} \cup T$ reaches the whole set. Its limit $\langle Z_{m-1} \cup T \rangle$ therefore contains $Z_{m-1} \cup T$ and is contained in it, so the two are equal. So $Z_{m-1} \cup T$ is a reachable set containing T, and its size is $|Z_{m-1}| + |T \setminus Z_{m-1}|$. Hence $\mathrm{lev}(T) \leq |Z_{m-1}| + |T \setminus Z_{m-1}|$, which is at most $|Z_{m-1}| + w$ because $|T \setminus Z_{m-1}| \leq w$.

Now split on whether S is itself one of the sets the last step adds. Suppose first that it is. Then $|S \setminus Z_{m-1}| \leq w$, which is what availability at Z_{m-1} asks. The jump of S is $|S \setminus U|$, and $Z_{m-1} \subseteq U$ makes it at most $|S \setminus Z_{m-1}|$, hence at most w. Suppose instead that it is not. The last step adds every set available at Z_{m-1} inside Y. Since S is not one of the sets it adds, S is not available at Z_{m-1} inside Y, and $S \subseteq Y$, so $|S \setminus Z_{m-1}| > w$. Since $S \subseteq Y$, the set Y contains Z_{m-1} together with the elements of $S \setminus Z_{m-1}$, of which there are more than w. Hence $|Y| \geq |Z_{m-1}| + w + 1$, which is more than $|Z_{m-1}| + w$. Every T the last step adds therefore has $\mathrm{lev}(T) \leq |Z_{m-1}| + w < |Y| = \mathrm{lev}(S)$. So every such T precedes S in the order. Then U contains Z_{m-1} and every such T. The last step adds to Z_{m-1} exactly those sets T, so U contains $Z_m = Y$, and Y contains S. The jump $|S \setminus U|$ of S is therefore 0. The two cases are exhaustive, so every jump is at most w.

The order is definable. By Lemma B.1 the relations "$i \in S_j$" and "$S_j \in \mathcal{S}$" are FOM-definable. The second of these holds because that lemma makes the coefficient family definable, and $S_j \in \mathcal{S}$ says exactly that the coefficient at S_j is nonzero. A coefficient is nonzero exactly when some digit of its absolute value is one, so this relation is an existential quantification over m applied to the digit relation. Here and below every quantification over codes ranges over the codes $j < 2^{|R|}$, which name each subset of R exactly once. The bound $j < 2^{|R|}$ is definable, since it says that no binary digit of j at a position at least $|R|$ is one, and $|R|$ is the number of relevant coordinates, a count the logic computes (Section A.5). An inclusion between two coded sets, such as $S_j \subseteq Y$, is a universal quantification over coordinates i of an implication between two instances of the relation "$i \in S_j$". The relation "Z is w-closed in Y" says two things. The first is that $Z \subseteq Y$. The second is that, for every code j, if $S_j \in \mathcal{S}$, $S_j \subseteq Y$ and $|S_j \setminus Z| \leq w$, then $S_j \subseteq Z$. It is thus a universal quantification over the codes j of a condition built from the relations "$i \in S_j$" and "$S_j \in \mathcal{S}$" and from the comparison $|S_j \setminus Z| \leq w$. That comparison is first-order, because w is a constant. It says that no $w+1$ distinct elements lie in $S_j \setminus Z$, and $w+1$ quantifiers express that.

That Y is reachable is a further universal quantification, over the codes of the proper subsets of Y. By the equivalence proved above, Y is reachable exactly when no code Z with $Z \subseteq Y$ and $Z \neq Y$ is w-closed in Y. Then $\mathrm{lev}(S) = \lambda$ says two things. First, some reachable Y with $S \subseteq Y$ has $|Y| = \lambda$. Second, every reachable Y with $S \subseteq Y$ has $|Y| \geq \lambda$. Here $|Y|$ is the number of coordinates i in the set coded by Y, a count the logic computes (Section A.5). The value $\mathrm{lev}(S)$ is at most $|R| \leq n$, since $R \subseteq [n]$, so the logic names and compares it like the counts above. Write $\mathrm{code}(S')$ for the canonical code $j < 2^{|R|}$ of a support set S'. The position of S is then the number of sets $S' \in \mathcal{S}$ whose pair $(\mathrm{lev}(S'), \mathrm{code}(S'))$ stands lexicographically below the pair $(\mathrm{lev}(S), \mathrm{code}(S))$. So positions are numbered from 0: the first set of the order has no set below it and occupies position 0. That number is a count over the codes $j < 2^{|R|}$ with $S_j \in \mathcal{S}$, and it counts each support set once, because each support set has exactly one such code. Every code is a number below n^c, hence a c-tuple of elements of the universe. For a fixed c, the logic FOM is closed under quantifying, comparing and counting over such tuples (Section A.5). This count and the count $|Y|$ above are the two places where the majority quantifier is needed. So the relation that holds of n, a code j and a number i exactly when $j < 2^{|R|}$, $S_j \in \mathcal{S}$, and the count for $S = S_j$ equals i, is FOM-definable. It holds exactly when S_j occupies position i. None of the formulas above depends on n, because w and c are constants, so one formula serves every n, which is what definability of the rank function asks. □

The next lemma is the parity obstruction in this model, in the form Theorem B.4 applies it: a bound on a correlation rather than on an error, and a family of characters that need not all be of one size.

Lemma B.3 (Correlational hardness of parities)**.** *Let $\mathcal{P}$ be a family of distinct characters on $\{-1,1\}^n$. Let $\tau > 0$ and $\gamma > 0$. Suppose a* deterministic *learner makes at most t correlational queries of tolerance τ against an unknown target from $\mathcal{P}$. Suppose also that the learner succeeds in the following sense. For every target χ_S in $\mathcal{P}$, and whatever answers the oracle returns subject to Definition 3.1, the learner outputs a hypothesis $h\colon \{-1,1\}^n \to [-1,1]$ whose correlation with that target satisfies $|\mathbb{E}_x[h(x)\chi_S(x)]| \geq \gamma$. Then*

$$t \geq \left(|\mathcal{P}| - \gamma^{-2}\right)\tau^2. \tag{7}$$

Take $\mathcal{P} = \{\chi_S : |S| = k\}$, the $\binom{n}{k}$ characters on sets of size k, with $k = \omega(1)$ and $k \leq n^{1/2}$. Also take $\tau, \gamma \geq 1/\mathrm{poly}(n)$. Then the right-hand side of Eq. (7) *is $n^{\Omega(k)}$, which is super-polynomial. The learner of Definition 3.3 is deterministic, so assuming determinism costs nothing where the lemma is applied.*

Proof of Lemma B.3. Against every query (ψ, τ) of Definition 3.1 the adversary answers 0. Since ψ takes its values in $[-1,1]$ we have $\psi(x)^2 \leq 1$ at every input, hence $\mathbb{E}_x\left[\psi(x)^2\right] \leq 1$. The characters are orthonormal, so Parseval's identity gives $\sum_{S\subseteq[n]} \widehat{\psi}(S)^2 = \mathbb{E}_x\left[\psi(x)^2\right]$, where $\widehat{\psi}(S) = \mathbb{E}_x[\chi_S(x)\psi(x)]$. The members of $\mathcal{P}$ are distinct characters, so the sum over $\mathcal{P}$ below takes each set S at most once, and it leaves out only terms of the full sum, each of which is nonnegative. Hence

$$\sum_{\chi_S \in \mathcal{P}} \mathbb{E}_x[\chi_S(x)\psi(x)]^2 = \sum_{\chi_S \in \mathcal{P}} \widehat{\psi}(S)^2 \leq \sum_{S\subseteq[n]} \widehat{\psi}(S)^2 = \mathbb{E}_x\left[\psi(x)^2\right] \leq 1. \tag{8}$$

Let m_ψ be the number of members χ_S of $\mathcal{P}$ with $|\mathbb{E}_x[\chi_S\psi]| > \tau$, and say that ψ *excludes* them. Each of them contributes more than τ^2 to the left-hand side of Eq. (8), and every term there is nonnegative, so $m_\psi\tau^2 \leq 1$, that is, $m_\psi \leq \tau^{-2}$. For every member that ψ does not exclude, the answer 0 lies within τ of the true correlation $\mathbb{E}_x[\chi_S\psi]$, so it is a legal answer to (ψ, τ) under Definition 3.1 when χ_S is the target. Suppose, for a contradiction, that $t\tau^{-2} + \gamma^{-2} < |\mathcal{P}|$. Now run the learner against the oracle that answers 0 to every query. Every prefix of at most t queries leaves at least $|\mathcal{P}| - t\tau^{-2} > 0$ possible targets for which all those answers are legal. The learner must therefore halt with an output after at most t queries: otherwise it violates its query bound or its output guarantee on a surviving target. Its first query is fixed before any answer arrives, and, the learner being deterministic, each later query is fixed by the earlier queries and their answers, all of which are 0. So this run is one sequence of queries $(\psi_1, \tau), \ldots, (\psi_{t'}, \tau)$ with $t' \leq t$, followed by one output h, and none of it depends on which member of $\mathcal{P}$ is the target. Call a member of $\mathcal{P}$ a *candidate* when none of $\psi_1, \ldots, \psi_{t'}$ excludes it. Each ψ_i excludes at most τ^{-2} members, so at most $t'\tau^{-2} \leq t\tau^{-2}$ members are not candidates, and at least $|\mathcal{P}| - t\tau^{-2}$ members are. For a

candidate χ_S every answer 0 in the run is legal. Extend that transcript to a legal oracle by answering 0 whenever the true correlation is within τ of zero, and the true correlation otherwise. Against this oracle the learner makes exactly the queries $\psi_1, \ldots, \psi_{t'}$ and outputs h. The output h takes its values in $[-1, 1]$, so the two steps above apply to it with h in place of ψ: $\sum_{\chi_S \in \mathcal{P}} \mathbb{E}_x [h\chi_S]^2 \leq \sum_{S \subseteq [n]} \widehat{h}(S)^2 = \mathbb{E}_x [h^2] \leq 1$. Let m_h be the number of members χ_S of $\mathcal{P}$ with $|\mathbb{E}_x [h\chi_S]| \geq \gamma$. Each of them contributes at least γ^2 to that sum, so $m_h \gamma^2 \leq 1$, and at most γ^{-2} members do. Under the contradiction assumption, the number of candidates, at least $|\mathcal{P}| - t\tau^{-2}$, exceeds γ^{-2}, so some candidate χ_S has $|\mathbb{E}_x [h\chi_S]| < \gamma$. Against the target χ_S and this legal oracle the learner outputs h, which violates the correlation requirement, so the learner fails. So the learner succeeds only if $t\tau^{-2} + \gamma^{-2} \geq |\mathcal{P}|$, and rearranging that inequality gives Eq. (7).

Now the asymptotic form. In that case $\mathcal{P}$ is the set of characters on the k-element subsets of $[n]$, so $|\mathcal{P}| = \binom{n}{k}$. Since $k = \omega(1)$ and $k \leq n^{1/2} \leq n$, for all sufficiently large n we have $1 \leq k \leq n$. Then $\binom{n}{k} = \prod_{i=0}^{k-1} \frac{n-i}{k-i}$, and each factor satisfies $\frac{n-i}{k-i} \geq \frac{n}{k}$, because $(n-i)k - n(k-i) = i(n-k) \geq 0$. So $\binom{n}{k} \geq (n/k)^k$. The hypothesis $k \leq n^{1/2}$ gives $n/k \geq n^{1/2}$, so $\binom{n}{k} \geq n^{k/2}$. Since τ and γ are at least $1/\mathrm{poly}(n)$, there is a constant $\beta \geq 1$ with $\tau \geq n^{-\beta}$ and $\gamma \geq n^{-\beta}$ for all sufficiently large n. Then $\tau^2 \geq n^{-2\beta}$ and $\gamma^{-2} \leq n^{2\beta}$. Since $k = \omega(1)$, also $k \geq 8\beta + 2$ for all sufficiently large n. Then $k/2 - 2\beta \geq 2\beta + 1 \geq 1$, so $n^{k/2} \geq 2n^{2\beta}$ for $n \geq 2$, and hence $|\mathcal{P}| - \gamma^{-2} \geq n^{k/2} - n^{2\beta} \geq n^{k/2}/2$. Multiplying by $\tau^2 \geq n^{-2\beta}$ gives $\left(|\mathcal{P}| - \gamma^{-2}\right)\tau^2 \geq n^{k/2-2\beta}/2$. Also $2\beta \leq k/4$, so $k/2 - 2\beta \geq k/4$, and $n^{k/4}/2 \geq n^{k/8}$ because $n^{k/8} \geq 2$ when $n \geq 2$ and $k \geq 8$. So the right-hand side of Eq. (7) is at least $n^{k/8}$ for all sufficiently large n, which is $n^{\Omega(k)}$. It is super-polynomial because $k = \omega(1)$: for every constant d, eventually $k/8 > d$, and then $n^{k/8} > n^d$. □

Algorithm 1 Visible search, at width w, acceptance threshold τ and query tolerance $\theta \leq \tau/3$. The set A holds the support sets accepted so far, and U holds the coordinates they use. Each round queries every set made of coordinates of U together with at most w coordinates from outside it. It reads each answer to error at most $\theta/4$ and accepts when the resulting absolute value is at least $\tau/2$; the loop stops when a round accepts nothing new.

$A \leftarrow \emptyset$; $\quad U \leftarrow \emptyset$
loop
  $N \leftarrow \emptyset$
  for all $T \subseteq U$ and $F \subseteq [n] \setminus U$ with $|F| \leq w$ **do**
    $S \leftarrow T \cup F$; query (χ_S, θ); read a dyadic v_S within $\theta/4$ of its answer
    if $|v_S| \geq \tau/2$ and $S \notin A$ **then** $N \leftarrow N \cup \{S\}$
  if $N = \emptyset$ **then exit loop**
  $A \leftarrow A \cup N$; $\quad U \leftarrow \bigcup_{S \in A} S$
return $h = \mathrm{sign}\left(\sum_{S \in A} v_S \chi_S\right)$, taking each v_S from the last iteration and $\mathrm{sign}(0) = 1$

Proof of Theorem 4.3. Fix n and a target $g \in \mathcal{O}_n(f)$. Then $g = f_n^\sigma$ for a permutation σ of $[n]$. As recorded in Section 2.1, relabeling moves each set S of $\mathrm{supp}(f_n)$ to $\sigma(S)$ and carries its coefficient along, so that $\mathrm{supp}(g) = \{\sigma(S) : S \in \mathrm{supp}(f_n)\}$ and $\widehat{g}(\sigma(S)) = \widehat{f_n}(S)$. So every nonzero coefficient of g is at least $1/q(n)$ in magnitude, and the relevant coordinates of g are the images under σ of those of f_n, at most $c \log n$ of them. Hence g meets both conditions of Definition 2.1 at length n with the parameters (c, q).

An ordering $S_{\pi(1)}, \ldots, S_{\pi(P)}$ of $\mathrm{supp}(f_n)$ gives the ordering $\sigma(S_{\pi(1)}), \ldots, \sigma(S_{\pi(P)})$ of $\mathrm{supp}(g)$ with the same jumps, because σ is a bijection, so that

$$\left|\sigma(S_{\pi(i)}) \setminus \bigcup_{j<i} \sigma(S_{\pi(j)})\right| = \left|S_{\pi(i)} \setminus \bigcup_{j<i} S_{\pi(j)}\right|. \tag{9}$$

The same holds with σ^{-1} in the other direction, so $\mathrm{Leap}(g) = \mathrm{Leap}(f_n) \leq w$.

The algorithm queries a set S as the pair (χ_S, θ) of Definition 3.1, which is a legal query because χ_S takes the values ± 1 and $\theta > 0$. It reads the answer to additive error at most $\theta/4$ and calls this dyadic

approximation v_S. The oracle answer lies within θ of the correlation $\mathbb{E}_x[g(x)\chi_S(x)] = \widehat{g}(S)$. The triangle inequality therefore gives $|v_S - \widehat{g}(S)| \le 5\theta/4 \le 5\tau/12$. The oracle may answer the same set differently in different iterations, but every answer it gives obeys this bound, and every step below uses only the bound, applied to the answer received in the iteration in question.

Every set collected is in the support. If S is added to A then $|v_S| \ge \tau/2$, so $|\widehat{g}(S)| \ge \tau/2 - 5\tau/12 = \tau/12 > 0$ and $S \in \mathrm{supp}(g)$.

Every set of the support is collected. The loop does exit, for the following reason. Every iteration but the last adds to A at least one set not already in it, and every such set lies in the finite family $\mathrm{supp}(g)$ by the previous step. The cost paragraph below counts these iterations. Suppose the loop exits with $A \ne \mathrm{supp}(g)$. Let $S_1, \dots, S_P$ be an ordering of $\mathrm{supp}(g)$ that attains the leap, so that every jump is at most $\mathrm{Leap}(g)$ and hence at most w. Every set of A lies in $\mathrm{supp}(g)$ by the previous step, and $A \ne \mathrm{supp}(g)$ by supposition, so some listed set is missing from A. Let i be the least index with $S_i \notin A$. The set A only grows, and the final iteration leaves it unchanged, since that iteration found $N = \emptyset$. So A already contained $S_1, \dots, S_{i-1}$ at the start of the final iteration, by the minimality of i.

Write $V = \bigcup_{j<i} S_j$ for the coordinates those earlier sets use. At the start of every iteration $U = \bigcup_{S\in A} S$, since A and U both start empty and the last step of each iteration updates them together. So $U \supseteq V$ at the start of the final iteration. Put $T = S_i \cap U$, the part of S_i the search has already reached, and $F = S_i \setminus U$, the part it has not. Since $U \supseteq V$, we get $F = S_i \setminus U \subseteq S_i \setminus V$. The set $S_i \setminus V$ holds the coordinates that S_i introduces in the chosen ordering, and every jump of that ordering is at most w, so $|F| \le |S_i \setminus V| \le w$. The pair (T, F) is therefore one that the final iteration enumerates, and $T \cup F = S_i$. Since $S_i \in \mathrm{supp}(g)$, tameness gives $|\widehat{g}(S_i)| \ge 1/q(n) = \tau$. So $|v_{S_i}| \ge |\widehat{g}(S_i)| - |v_{S_i} - \widehat{g}(S_i)| \ge \tau - 5\tau/12 = 7\tau/12 > \tau/2$. Since $S_i \notin A$ as well, the set S_i enters N. This contradicts $N = \emptyset$, so $A = \mathrm{supp}(g)$ at termination.

The hypothesis is the target. Write $u = \sum_{S\in A} v_S \chi_S$ for the weighted sum the algorithm forms, so that the output is $h = \mathrm{sign}(u)$. Let R be the relevant set of g, which has at most $c \log n$ elements by tameness. Every $S \in A = \mathrm{supp}(g)$ lies inside R, so $|A| \le 2^{|R|} \le n^c$. Each v_S in u is the answer that the final iteration received for S, and that answer exists. It exists because, at the start of the final iteration, $S \subseteq U$, so the pair $T = S$ and $F = \emptyset$ is enumerated. Like every rounded answer, it obeys $|v_S - \widehat{g}(S)| \le 5\theta/4$. Since $A = \mathrm{supp}(g)$, subtracting u from the expansion Eq. (1) of g gives $g - u = \sum_{S\in\mathrm{supp}(g)} (\widehat{g}(S) - v_S)\,\chi_S$. So the Fourier coefficients of $g - u$ are $\widehat{g}(S) - v_S$ on $\mathrm{supp}(g)$ and 0 elsewhere. Parseval's identity for $g - u$, then the tolerance bound on each of the at most n^c terms, then the choice $\theta \le n^{-c}/2$, give

$$\mathbb{E}_x\left[(g(x) - u(x))^2\right] = \sum_{S\in\mathrm{supp}(g)} (\widehat{g}(S) - v_S)^2 \le \frac{25}{16} n^c \theta^2 \le \frac{25}{64} n^{-c}. \tag{10}$$

Consider a point x at which h and g disagree, that is at which $\mathrm{sign}(u(x)) \ne g(x)$. The target is $\{-1, 1\}$-valued, so there are two cases. If $g(x) = 1$, then $\mathrm{sign}(u(x)) = -1$, which with the convention $\mathrm{sign}(0) = 1$ means $u(x) < 0$, so $g(x) - u(x) = 1 - u(x) > 1$. If $g(x) = -1$, then $\mathrm{sign}(u(x)) = 1$, which means $u(x) \ge 0$, so $|g(x) - u(x)| = 1 + u(x) \ge 1$. In both cases $(g(x) - u(x))^2 \ge 1$. So the function equal to 1 where h and g disagree and to 0 elsewhere is at most $(g(x) - u(x))^2$ at every x, since the right-hand side is nonnegative. Taking averages, $\mathbf{Pr}_x[h(x) \ne g(x)] \le \mathbb{E}_x\left[(g(x) - u(x))^2\right] \le 25n^{-c}/64$ by Eq. (10). Both h and g are functions of the coordinates in R alone, for the following reason. The expansion Eq. (1) of g uses only characters χ_S with $S \in \mathrm{supp}(g)$, each of which reads only coordinates of R, and u, hence $h = \mathrm{sign}(u)$, is built from the same characters. The event that they disagree is therefore a union of events, one for each assignment to R at which they disagree. Each of those events has probability $2^{-|R|}$, which is at least n^{-c} because $|R| \le c \log n$. The probability of the union was just bounded by $25n^{-c}/64$, which is smaller than n^{-c}, so no assignment can contribute to it. Hence h and g agree everywhere, and $h = g$.

The cost. Every set collected lies in $\mathrm{supp}(g)$, so the coordinates in U are relevant coordinates of g, of which there are at most $c \log n$. Hence $2^{|U|} \le n^c$, and that is the number of sets T an iteration enumerates. An iteration pairs each such T with a set F of at most w coordinates outside U, and

there are $\sum_{j\le w}\binom{n-|U|}{j} \le \sum_{j\le w}\binom{n}{j}$ such sets F. For $j \le w$ we have $\binom{n}{j} \le n^j \le \binom{w}{j}n^j$, so the binomial theorem gives $\sum_{j\le w}\binom{n}{j} \le \sum_{j=0}^{w}\binom{w}{j}n^j = (n+1)^w$. So an iteration issues at most $n^c(n+1)^w$ queries. Every iteration but the last adds a set to A, and $|A| \le |\mathrm{supp}(g)| \le n^c$, so there are at most n^c+1 iterations. The total is at most $(n^c+1)\,n^c(n+1)^w \le 2n^c \cdot n^c \cdot (2n)^w = 2^{w+1}n^{2c+w}$ queries, and since w does not depend on n this is $n^{O(w+c)}$. Each query names the set $S = T \cup F$, written down in time $O(n)$, and its query function χ_S is evaluable in time $O(n)$. Reading each response to error at most $\theta/4$ uses $O(\log(1/\theta)) = O(\log n)$ bits and time for fixed (c, q). Between queries the algorithm compares $|v_S|$ with $\tau/2$ and tests $S \notin A$ by comparing S with each of the at most n^c sets of A, in time $O(n^{c+1})$. Each update of A and U, and the final output of the at most n^c pairs (S, v_S) describing h, also takes time $O(n^{2c+1})$. So the running time is at most $2^{w+1}n^{2c+w} \cdot O(n^{c+1}) + (n^c+1)\cdot O(n^{2c+1})$, which is $n^{O(w+c)}$.

Finally, the tolerance $\theta = \min(\tau/3, n^{-c}/2)$ is at least $1/\mathrm{poly}(n)$, since $\tau = 1/q(n)$ for a polynomial q. A tolerance of that size is what Definition 3.3 allows at every accuracy, and the algorithm never reads the accuracy it is run at. Its error is $0 \le \varepsilon$, so $\mathcal{O}(f)$ is learnable in time $n^{O(w+c)}$.

If $\mathrm{Leap}(f_n) = O(1)$, then $\mathrm{Leap}(f_n)$ is at most a constant for all large n and takes finitely many values below that, so $w = \max_n \mathrm{Leap}(f_n)$ is a constant width that bounds the leap at every length. With that w the time $n^{O(w+c)}$ is a polynomial in n, so $\mathcal{O}(f)$ is learnable in time $\mathrm{poly}(n, 1/\varepsilon)$ and $f \in \mathsf{Learn}$. □

The next theorem is the witness that the arity bound of Definition 2.1 cannot be dropped, restated from Section 1.1 with its parameters in full.

Theorem B.4 (The arity bound is necessary, restatement of Theorem 1.3)**.** *Let $k = k(n)$ be an* FOM*-definable function with $k(n) = \omega(\log n)$. Suppose also that $k(n) \le n/8$ for every $n \ge 16$, and that $k(n) = 2^t$ for an integer $t = t(n) \ge 1$ at every such n. Then there is a presented* FOM*-definable query f with $f \in \mathsf{sFOM}_1$ and $f \notin \mathsf{Learn}$. For every $n \ge 16$ the function f_n has leap 1 and arity $2k+t+2$. Its Fourier support has $12k^2+8$ members, and every nonzero Fourier coefficient has magnitude at least $1/(8k)$. So f meets every requirement of Definition 2.1 except the arity bound. Hence Theorem 6.1 fails without that bound, already at width one.*

Proof of Theorem B.4. The plan is to make every coordinate cheap for a learner to find and one bit about each pair of coordinates expensive. The coordinates are grouped into k pairs, and a few address coordinates name one pair. A multiplexer returns the disjunction of the two coordinates of the named pair, which is symmetric in them, so it says nothing about which of the two is which. Two further coordinates switch the target between that multiplexer and the parity of a hidden set, which takes one coordinate from each pair. There are 2^k such sets to tell apart. The query is built first. The steps that follow check that it is presented and definable, count its spectrum, compute its leap, define its derivation, show that the transversals form one orbit, and rule out an efficient learner.

The construction. For $n < 16$ let f_n be the constant 1. Its support is $\{\emptyset\}$, with coefficient 1, so no coordinate is relevant to it. Its width-one derivation is the single feature $\psi_0 = 1$ with weight 1. That sequence has $m = 0$, so bounded introduction and visibility, which speak only of positions $i < m$, hold vacuously, and the readout $\mathrm{sign}(1 \cdot 1) = 1$ gives back f_n. The lower bound below is proved at lengths $n \ge 16$ alone, but the definability claims below cover every length. Each formula written below for $n \ge 16$ is conjoined with the first-order statement that the structure has at least 16 elements. It is then put in disjunction with the statement that the structure has fewer than 16 elements, conjoined with the answer at those lengths: $f_n(x) = 1$, no coordinate relevant, and the one feature $\psi_0 = 1$ with weight 1.

Fix $n \ge 16$, so that $k = 2^t \ge 2$ and $k \le n/8$ by hypothesis. Reserve the first t coordinates $1, \dots, t$ and call them the *address coordinates*. Reserve the next $2k$ coordinates as k *data pairs* $B_1, \dots, B_k$. The j-th pair is $B_j = \{u_j, v_j\}$ with $u_j = t + 2j - 1$ and $v_j = t + 2j$, for $j \le k$. Reserve two more as the *selector* coordinates $r_1 = t + 2k + 1$ and $r_2 = t + 2k + 2$. The remaining coordinates are unused. There are enough coordinates for all of this. Since $k = 2^t$ we have $t = \log_2 k \le k$, so the last reserved position is $t + 2k + 2 \le 3k + 2$. Moreover $3k + 2 \le 3n/8 + 2 \le n$, using $k \le n/8$ and $n \ge 16$.

For $I \subseteq \{1, \dots, t\}$ the character χ_I reads address coordinates only. Let $o_j(x)$ be 1 when at least one of the two data coordinates x_{u_j} and x_{v_j} is 1, and -1 otherwise. This is the disjunction of the

two coordinates of the j-th pair, written in $\{-1, 1\}$. Checking the four assignments of the pair gives

$$o_j = \tfrac{1}{2}\left(1 + \chi_{\{u_j\}} + \chi_{\{v_j\}} - \chi_{B_j}\right), \tag{11}$$

which is symmetric in u_j and v_j: exchanging the two coordinates of the pair leaves o_j unchanged.

For $1 \le i \le t$ let $b_i(x) = (1 + x_i)/2 \in \{0, 1\}$ be the bit read off the i-th address coordinate, which is one when that coordinate is 1 and zero when it is -1. Let $\iota(x) = 1 + \sum_{i=1}^{t} b_i(x)2^{i-1}$, which is one plus the integer whose binary digit $i - 1$ is $b_i(x)$, counting the units digit as digit 0. Since $\sum_{i=1}^{t} b_i(x)2^{i-1}$ ranges over $0, \ldots, 2^t - 1$, the value $\iota(x)$ lies in $\{1, \ldots, k\}$, so the address coordinates name one of the k data pairs. Put

$$M(x) = o_{\iota(x)}(x), \tag{12}$$

the multiplexer that reads the selected data pair through its disjunction.

Write $\varepsilon_{j,i} \in \{-1, 1\}$ for the sign that records binary digit $i - 1$ of $j - 1$, the digit of weight 2^{i-1}: it is 1 when that digit is one and -1 when it is zero. For a set I of address coordinates put $\varepsilon_{j,I} = \prod_{i \in I} \varepsilon_{j,i}$, the product of those signs over I. Then $b_i(x)$ agrees with digit $i - 1$ of $j - 1$ exactly when $x_i = \varepsilon_{j,i}$, since $b_i(x) = 1$ exactly when $x_i = 1$. So the indicator of the event $\iota(x) = j$ is the product $\prod_{i \le t} \frac{1 + \varepsilon_{j,i} x_i}{2}$. Each factor of that product is 1 when $x_i = \varepsilon_{j,i}$ and 0 when not, so the product is 1 exactly when the address names the j-th pair. Expanding that product over the subsets I of the address coordinates gives $2^{-t} \sum_{I \subseteq \{1,\ldots,t\}} \varepsilon_{j,I} \chi_I$. Exactly one j is named at each x, so M is the sum over j of that indicator times o_j. Substituting Eq. (11) and separating the term of o_j that carries no data coordinate from the three that do,

$$M = \frac{1}{2^{t+1}} \sum_{I} \left(\sum_{j=1}^{k} \varepsilon_{j,I} \right) \chi_I + \frac{1}{2k} \sum_{j=1}^{k} \sum_{I} \varepsilon_{j,I} \left(\chi_{I \cup \{u_j\}} + \chi_{I \cup \{v_j\}} - \chi_{I \cup B_j} \right). \tag{13}$$

As $j - 1$ runs over the 2^t strings of t bits, each pattern of the digits in I occurs $2^{t-|I|}$ times. Hence $\sum_j \varepsilon_{j,I} = 2^{t-|I|} \prod_{i \in I} \sum_{b \in \{-1,1\}} b$. The inner sum over b is zero, so the product vanishes for every nonempty I. For $I = \emptyset$ the empty product is 1 and the sum is $2^t = k$, so the first sum in Eq. (13) is the constant $k/2^{t+1} = 1/2$. In the second sum, each term carries a set $I \cup H$ with $H \in \{\{u_j\}, \{v_j\}, B_j\}$. Such a set determines I, j and H, since I is its address part and H its data part. So no two of the terms share a character. There are $3k \cdot 2^t = 3k^2$ of them: three for each pair index j and each of the $2^t = k$ address sets I. Hence $\mathrm{supp}(M)$ consists of $\emptyset$, with coefficient $1/2$, and $3k^2$ sets each meeting exactly one data pair, with coefficients of magnitude $1/(2k)$. Finally, M is $\{-1, 1\}$-valued, because at every x it equals one of the $\{-1, 1\}$-valued functions o_j.

Let ζ be the indicator of the event $x_{r_1} = x_{r_2} = 1$, which is 1 when both selector coordinates are 1 and 0 otherwise. Two coordinates are constrained, so $\mathbb{E}[\zeta] = 1/4$. Multiplying out the two factors $(1 + x_{r_1})/2$ and $(1 + x_{r_2})/2$ gives $\zeta = \frac{1}{4} \sum_{V \subseteq \{r_1, r_2\}} \chi_V$. Call $W \subseteq [n]$ a *transversal* when it contains exactly one coordinate of each data pair and nothing else, and set

$$f^W = \zeta M + (1 - \zeta)\, \chi_W. \tag{14}$$

Take $f_n = f^{W_0}$ for the transversal $W_0 = \{u_1, \ldots, u_k\}$. Figure 1 shows the layout.

Presented and FOM*-definable.* The function t is FOM-definable. The reason is that k is FOM-definable and that t is the position of the unique one in the binary expansion of k. So all the index arithmetic below is available. At a length $n \ge 16$ the relevant set is the initial segment $[t+2k+2]$, as the count of the relevant coordinates below shows, and at a length $n < 16$ it is empty. Its endpoint is definable from k and t, so f is presented. The query f is also FOM-definable, and directly so rather than through a circuit.

The condition $f_n(x) = 1$ says that either $x_{r_1} = x_{r_2} = 1$ and $M(x) = 1$, or one of the two selector coordinates is -1 and $\chi_{W_0}(x) = 1$. In the first disjunct, the condition $M(x) = 1$ says that some $j \le k$ satisfies two conditions. First, at every $i \le t$ the address coordinate x_i is 1 exactly when binary digit $i - 1$ of $j - 1$ is one, that is, exactly when $\mathrm{BIT}(j, i)$ holds under the rank convention of Section A.3. That is to say that the address names the j-th pair. Second, at least one of the two data coordinates x_{t+2j-1} and x_{t+2j} of that pair is 1. Both conditions are arithmetic on indices, so the disjunct is first-order over the order and BIT. In the second disjunct, the condition $\chi_{W_0}(x) = 1$

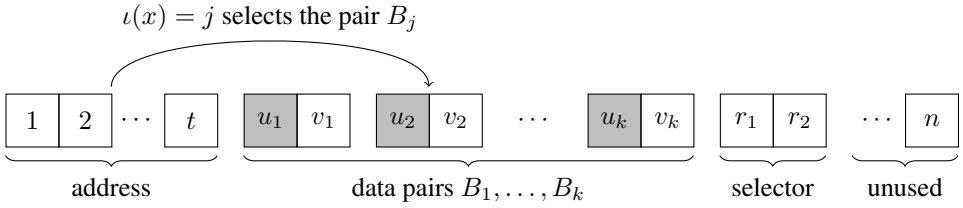


Figure 1: The witness of Theorem B.4 on $[n]$, drawn with the pair $j = 2$ selected. The $t = \log_2 k$ address coordinates name a data pair. The multiplexer M reads the named pair only through the disjunction o_j of its two coordinates. The two selector coordinates decide, through ζ, whether the target is M or the parity of the transversal W, shaded here, which takes one coordinate of each pair. Transposing the two coordinates of a pair fixes M and moves W, so the 2^k transversals lie in one orbit. The only support sets that depend on W are the four sets $W \cup V$ with $V \subseteq \{r_1, r_2\}$, and the smallest of them is W itself, of size k.

is the parity of the set $W_0 = \{u : t < u \leq t + 2k, u - t \text{ odd}\}$, which is definable from k and t, and that parity is FOM-definable by Proposition 2.2.

Sparsity, coefficients, arity. At every x exactly one of ζ and $1 - \zeta$ takes the value 1 and the other takes the value 0. So at each point f^W equals M or equals χ_W. Both of those functions are $\{-1, 1\}$-valued, so f^W is $\{-1, 1\}$-valued too. Expanding ζ and $1 - \zeta$,

$$f^W = \frac{1}{4} \sum_{V \subseteq \{r_1, r_2\}} \chi_V M + \frac{1}{4} \left(3 - \chi_{\{r_1\}} - \chi_{\{r_2\}} - \chi_{\{r_1, r_2\}}\right) \chi_W. \tag{15}$$

In the first group, each product $\chi_V M$ contributes the sets $S \cup V$ with $S \in \operatorname{supp}(M)$. Every $S \in \operatorname{supp}(M)$ consists of address and data coordinates by Eq. (13), so S and V are disjoint and $\chi_V \chi_S = \chi_{S \cup V}$. The coefficient of such a set is the coefficient of S in M divided by four. Taking $S = \emptyset$, whose coefficient in M is $1/2$, gives the four sets V alone with coefficient $1/8$. Taking the other $3k^2$ members of $\operatorname{supp}(M)$ gives $4 \cdot 3k^2$ sets with coefficients of magnitude $1/(8k)$. They are pairwise distinct, since intersecting with $\{r_1, r_2\}$ recovers V and the rest recovers S. The second group contributes the four sets $W \cup V$, since W holds data coordinates only and so $\chi_V \chi_W = \chi_{W \cup V}$. The coefficient is $3/4$ at $V = \emptyset$ and $-1/4$ at each of the three nonempty V, read off Eq. (15). No set occurs in both groups: a set of the first group meets at most one data pair, and W meets every one of the $k \geq 2$ data pairs. So there is no cancellation. The support has exactly $4 + 12k^2 + 4 = 12k^2 + 8$ members. The magnitudes that occur are $1/8$, $1/(8k)$, $3/4$ and $1/4$, and $1/(8k)$ is the smallest of them because $k \geq 2$.

Every reserved coordinate is relevant, and a support set containing it witnesses this. For an address coordinate i, take the set $\{i, u_1\}$. It is the set $I \cup H$ of the first group with address part $I = \{i\}$, pair index $j = 1$, data part $H = \{u_1\}$ and selector part $V = \emptyset$. For a data coordinate, take its singleton $\{u_j\}$ or $\{v_j\}$, which is the set with $I = V = \emptyset$. For a selector coordinate, take $\{r_1\}$ or $\{r_2\}$, which is a set of the first group whose part from $\operatorname{supp}(M)$ is empty. No other coordinate is relevant. Every set of the first group is $S \cup V$ with $S \in \operatorname{supp}(M)$, which consists of address and data coordinates, and every set of the second group is $W \cup V$. So every support set lies inside the reserved segment $[t + 2k + 2]$, and no unused coordinate lies in a support set. The arity is therefore exactly $t + 2k + 2$.

The leap is one. List the support in the following order. First $\emptyset$, then $\{r_1\}$ and $\{r_2\}$, then the $2k$ singletons $\{u_j\}$ and $\{v_j\}$, then $\{i, u_1\}$ for $i = 1, \ldots, t$. Each of these is a support set, by the witnesses of relevance given above, and each introduces at most one coordinate. The set $\emptyset$ introduces none. The sets $\{r_1\}$ and $\{r_2\}$ and the $2k$ singletons each introduce their one coordinate. The set $\{i, u_1\}$ comes after $\{u_1\}$, so it introduces only i. Together they introduce all $t + 2k + 2$ relevant coordinates. Every coordinate of a support set is relevant, by the definition of relevance, so every remaining set of the support lies inside the coordinates already introduced and introduces nothing, whatever order those sets are listed in. So every jump is at most one. No ordering does better. The support contains a nonempty set, for instance $\{r_1\}$. In any ordering, let S be the first nonempty set listed; every set before it is $\emptyset$, so the union of the earlier sets is empty and the jump of S is $|S| \geq 1$. Hence $\operatorname{Leap}(f^W) = 1$.

The derivation is definable. We apply Lemma B.8(2) with $g = f_n$, width $w = 1$ and $q(n) = 8k(n)$. Its hypotheses hold: f_n is $\{-1, 1\}$-valued, as shown above, and every nonzero Fourier coefficient of f_n has magnitude at least $1/(8k) = 1/q(n)$. So the ordering just given, whose jumps are all at most one, supplies a width-one derivation of f_n whose every visibility margin is at least $1/(8k)$. Since $k \leq n/8$, that margin is at least $1/n$. This is the $1/\mathrm{poly}(n)$ margin that Eq. (4) asks for. The derivation's features are the constant, with weight $\widehat{f_n}(\emptyset) = 1/8$, followed by the characters of the nonempty support sets in that order, with the corresponding Fourier coefficients as weights. Definition 5.1 asks in addition that the family of features and the family of weights be FOM-definable. Lemma B.1 is unavailable here, because its proof spends the arity bound, so we check the two families directly.

Index the sets of the first group that meet a data pair by quadruples (I, j, H, V). Here I is the address part, coded by an integer below k, and $j \leq k$ is the index of the data pair. The data part H is coded by one of three values, and V is the selector part. The four sets V alone complete the first group, and each of them is indexed by V itself. The second group's four sets are indexed by V as well. The relation "the coordinate u lies in the set with a given code" is then a disjunction of definable conditions, one for each part of the code.

For a quadruple there are three conditions. The first is that $u \leq t$ and digit $u - 1$ of the code of I is one, where the code of I is $\sum_{i \in I} 2^{i-1}$. This says that u is an address coordinate of I. The second is that u is $t + 2j - 1$ or $t + 2j$ and that H names it. The third is that u is a selector coordinate named by V. For a set of the second group, which is $W_0 \cup V$, there are two conditions. The first is that $t < u \leq t + 2k$ with $u - t$ odd, which says that u is the first coordinate of a data pair and so lies in W_0. The second is that u is a selector coordinate named by V. Each of these is arithmetic on indices, which the order and the BIT predicate supply.

So the support sets, indexed by their codes, form a definable family. Their characters form a definable family too. The character of the set with code c takes the value 1 at x exactly when the number of coordinates u of that set with $x_u = -1$, that is with $X(u)$ false, is even. That number is at most n, and the logic computes it, since it counts the elements satisfying a definable condition (Section A.5). The number is even exactly when its binary digit 0 is zero, which BIT reads. So there is an FOM-definable relation on n, a code c and an input x that holds exactly when the character of the set with code c is 1 at x.

The weights are $1/8$ on the four sets V alone, and $3/4$ and $-1/4$ on the second group. On the quadruple-indexed sets the weight is $\pm 1/(8k)$. Its sign is $\varepsilon_{j,I}$, multiplied by -1 exactly when $H = B_j$. That sign is FOM-definable. By its definition, $\varepsilon_{j,I}$ is -1 raised to the number of $i \in I$ at which digit $i - 1$ of $j - 1$ is zero. The parity of that number is definable. The magnitude is definable too. Since k is a power of two, so is $1/(8k) = 2^{-(t+3)}$, and a power of two is written exactly in $O(\log n)$ bits. Spelled out, the digit relation holds at digit $t + 3$ alone for the quadruple-indexed sets, and at digit 3 alone for the four sets V. For W_0, whose weight is $3/4$, it holds at digits 1 and 2. For the other three sets $W_0 \cup V$, it holds at digit 2 alone. Each of these compares m with a definable number, so the digit relation is FOM-definable. Among the quadruple-indexed sets, the sign relation holds exactly at those with $\varepsilon_{j,I} = 1$ and $H = B_j$, or with $\varepsilon_{j,I} = -1$ and $H \neq B_j$. Among the other sets, it holds exactly at the three sets $W_0 \cup V$ with $V \neq \emptyset$.

The ordering itself is definable. The $2k + t + 3$ sets listed above come first, in the order given, which is read off the definable indices j and i. Then come the remaining sets of the first group: first $\{r_1, r_2\}$, the one set V alone not listed above, and then the quadruple-indexed sets not listed above, in the lexicographic order of their quadruples (I, j, H, V). Last come the four sets $W_0 \cup V$, in a fixed order of the four values of V. Whether a quadruple indexes one of the first $2k + t + 3$ sets is itself arithmetic on indices. It does exactly when $I = V = \emptyset$ and H is a singleton, or when I is a singleton $\{i\}$, $j = 1$, $H = \{u_1\}$ and $V = \emptyset$.

The position of a nonempty set is the number of sets preceding it in this ranking, which a majority quantifier counts. The empty set heads the ranking. So the constant feature, which is $\chi_\emptyset$, occupies position 0, and the nonempty sets occupy positions $1, \ldots, 12k^2 + 7$, as Lemma B.8(2) indexes them. Every position is below $12k^2 + 8 \leq n^2$, so it is a pair of elements of the universe.

The feature at position i is the character of the unique support set whose position is i. So $\psi_i(x) = 1$ holds exactly when some code has position i and the character of its set is 1 at x, which quantifies over codes and is therefore FOM-definable. The weights at position i are read off that same code. The sign relation holds of n and i exactly when some code has position i and a negative weight. The

digit relation holds of n, i and m exactly when some code has position i and a weight with a one in digit m.

Every set after the first $2k + t + 3$ introduces nothing, since those already exhaust the relevant coordinates, so the width is unaffected by the tail order. Since f is itself FOM-definable, $f \in \mathsf{sFOM}_1$.

All transversals lie in one orbit. Let κ transpose the two coordinates of one data pair and fix everything else. Each o_i is invariant under κ. The permutation κ either fixes B_i pointwise or exchanges the two coordinates of B_i, and o_i is symmetric in those two coordinates by Eq. (11). The address coordinates are fixed as well, so M is invariant, and so is ζ, whose coordinates are the selectors. Applying κ to Eq. (14) therefore leaves the first term alone and sends χ_W to $\chi_{\kappa(W)}$, so $(f^W)^\kappa = f^{\kappa(W)}$.

Now let W be any transversal, and let σ be the product of the transpositions of those data pairs B_j on which W and W_0 pick different coordinates. Then $\sigma(W_0) = W$, because σ exchanges u_j with v_j exactly on those pairs and fixes u_j on the others. Relabeling composes: for permutations α and β, we have $(g^\alpha)^\beta = g^{\beta\circ\alpha}$, as substituting $y_i = x_{\beta(i)}$ into $g^\alpha(y) = g(y_{\alpha(1)}, \ldots, y_{\alpha(n)})$ shows. Applying the identity $(f^{W'})^\kappa = f^{\kappa(W')}$ once for each transposition in σ, starting from $W' = W_0$, gives $f_n^\sigma = f^{\sigma(W_0)} = f^W$. Hence every f^W lies in $\mathcal{O}_n(f_n)$, and there are 2^k transversals, one for each choice of a coordinate in each pair. This is the point of reading a pair through a symmetric function: the permutation that moves the secret leaves the rest of the target exactly where it was.

No efficient learner succeeds. Suppose $f \in \mathsf{Learn}$. Run the learner of Definition 3.3 at accuracy $1/3$. It is deterministic and makes $t_0 = \mathrm{poly}(n)$ queries, each of tolerance at least some $\tau = 1/\mathrm{poly}(n)$. For every member of $\mathcal{O}_n(f_n)$ and every legal oracle it outputs a $\{-1, 1\}$-valued hypothesis h whose error is at most $1/3$. Every f^W is such a member by the previous step, so the learner succeeds against each of them. Both h and f^W take values in $\{-1, 1\}$, so their product is 1 where they agree and -1 where they differ. Hence $\mathbb{E}_x\left[hf^W\right] = 1 - 2\,\mathbf{Pr}_x\left[h \neq f^W\right] \geq 1 - 2/3 = 1/3$.

Multiplying Eq. (14) by h and averaging gives $\mathbb{E}_x\left[hf^W\right] = \mathbb{E}_x\left[h\zeta M\right] + \mathbb{E}_x\left[h\left(1-\zeta\right)\chi_W\right]$. Since ζ takes values in $\{0, 1\}$ and $|hM| \leq 1$ pointwise, $\mathbb{E}_x\left[h\zeta M\right] \leq \mathbb{E}_x\left[\zeta\right] = 1/4$, so

$$\mathbb{E}_x\left[h\left(1-\zeta\right)\chi_W\right] \geq \frac{1}{3} - \frac{1}{4} = \frac{1}{12}, \tag{16}$$

and $h(1-\zeta)$ maps into $[-1, 1]$, so it correlates at least $1/12$ with χ_W.

Each query is simulated by one query to that character. For a query function ψ,

$$\mathbb{E}_x\left[f^W\psi\right] = \mathbb{E}_x\left[\zeta M\psi\right] + \mathbb{E}_x\left[\chi_W\left(1-\zeta\right)\psi\right]. \tag{17}$$

The first term of Eq. (17) is computed from ψ, ζ and M, none of which involves W. The second term is the correlation that the query $((1-\zeta)\psi, \tau)$ for the target χ_W asks for. That query is legal, because $(1-\zeta)\psi$ maps into $[-1, 1]$. Adding the first term to a legal answer to that query therefore produces a legal answer to (ψ, τ) for the target f^W. So the learner becomes a deterministic t_0-query learner for the family $\{\chi_W : W \text{ a transversal}\}$, and it outputs $h(1-\zeta)$.

It meets the hypothesis of Lemma B.3 for every target χ_W of that family and every legal oracle for it. The answers it passes on form a legal oracle for f^W, so the hypothesis h it receives is one the original learner outputs against f^W, and Eq. (16) applies to that h. The simulated learner computes the first term of Eq. (17) without a query, and Lemma B.3 counts queries rather than time, so that computation costs nothing in the count. The 2^k transversals give 2^k distinct characters, and an answer legal at the least tolerance τ is legal at every larger one. So Lemma B.3 applies with the family $\mathcal{P} = \{\chi_W : W \text{ a transversal}\}$ of 2^k characters, tolerance τ, $t = t_0$ and $\gamma = 1/12$. Since $\gamma^{-2} = 144$, this gives $t_0 \geq \left(2^k - 144\right)\tau^2$.

Now fix the polynomials. By Definition 3.3 at accuracy $1/3$ there are constants β_1 and β_2 with $t_0 \leq n^{\beta_1}$ and $\tau \geq n^{-\beta_2}$ for all large n, the number of queries being at most the running time. Since $k = \omega(\log n)$, for all large n we have $k \geq (\beta_1 + 2\beta_2 + 1)\log n$ and $k \geq 9$. For such n, the bound $2^k \geq 2^9 > 288$ gives $2^k - 144 \geq 2^k/2$, so $t_0 \geq \frac{1}{2}2^k\tau^2 \geq \frac{1}{2}n^{\beta_1+2\beta_2+1}n^{-2\beta_2} = \frac{1}{2}n^{\beta_1+1}$. That exceeds n^{β_1} once $n \geq 3$, against $t_0 \leq n^{\beta_1}$.

Hence $f \notin \mathsf{Learn}$. Yet $f \in \mathsf{sFOM}_1$, and f meets the coefficient condition of Definition 2.1 with $q(n) = n$. At $n \geq 16$ every nonzero coefficient is at least $1/(8k) \geq 1/n$, because $k \leq n/8$.

At $n < 16$ the only coefficient is 1. The arity bound does fail: for every constant c, the arity $2k + t + 2 > 2k$ exceeds $c \log n$ at all large n, because $k = \omega(\log n)$. So only the arity bound of Definition 2.1 fails. □

Choosing k against a given arity bound turns that witness into a threshold, which is the next corollary.

Corollary B.5 (The logarithmic arity bound is tight)**.** *Let $a = a(n)$ be an* FOM*-definable function with $a(n) = \omega(\log n)$ and $a(n) \leq n/4$ for every $n \geq 16$. Then there is a presented* FOM*-definable query f with $f \in \mathsf{sFOM}_1$ and $f \notin$ Learn. For every $n \geq 16$ with $a(n) \geq 8$, the function f_n has leap 1, arity at most $a(n)$, a Fourier support of size at most $a(n)^2$, and every nonzero Fourier coefficient of magnitude at least $1/(2a(n))$. By Theorem 4.3, for a constant c and a polynomial q, Algorithm 1 learns every query whose leap is at most one at every length, whose arity is at most $c \log n$ and whose nonzero coefficients have magnitude at least $1/q(n)$, in time $n^{O(1+c)}$. So the two results sit on the two sides of the logarithmic arity bound: every $c \log n$ is on the learnable side, and every* FOM*-definable $\omega(\log n)$ that stays below $n/4$ is on the other.*

Proof of Corollary B.5. The choice of k. Consider first a length n with $a(n) \geq 8$. Let $d(n)$ be the position of the highest one in the binary expansion of $a(n)$, so that $2^{d(n)} \leq a(n) < 2^{d(n)+1}$. Put $t(n) = d(n) - 2$ and $k(n) = 2^{t(n)}$. For a length n with $a(n) < 8$ put instead $t(n) = 1$ and $k(n) = 2$. When $a(n) \geq 8$ we have $d(n) \geq 3$, so $t(n) \geq 1$, and

$$4k(n) = 2^{d(n)} \leq a(n) < 2^{d(n)+1} = 8k(n). \tag{18}$$

The hypotheses of Theorem B.4. The function k is FOM-definable. Its value is a power of two, so it is determined by the one binary digit m at which it has a one, and two alternatives describe that digit. The first is that $a(n)$ has a one in some digit at least 3, has a one in digit $m + 2$, and has no one in any digit above $m + 2$. That alternative says that $a(n) \geq 8$ and $d(n) = m + 2$. The second is that $a(n)$ has no one in any digit at least 3, and that $m = 1$. That alternative says that $a(n) < 8$ and $k(n) = 2$. Both alternatives are first-order over the digit relation of a, which is FOM-definable by hypothesis. For every $n \geq 16$ the value $k(n)$ is $2^{t(n)}$ with $t(n) \geq 1$ an integer, in both cases of the definition.

Next, $k(n) \leq n/8$ for every $n \geq 16$. When $a(n) \geq 8$, Eq. (18) and the hypothesis $a(n) \leq n/4$ give $k(n) \leq a(n)/4 \leq n/16$. When $a(n) < 8$, the value is $k(n) = 2 \leq n/8$. Finally $k(n) = \omega(\log n)$. Since $a(n) = \omega(\log n)$ there is an n_1 with $a(n) \geq 8$ for every $n \geq n_1$. For those n, Eq. (18) gives $k(n) > a(n)/8$, which is $\omega(\log n)$. So Theorem B.4 applies to k and supplies a presented FOM-definable query f with $f \in \mathsf{sFOM}_1$ and $f \notin$ Learn. For every $n \geq 16$ the function f_n has leap 1 and arity $2k(n) + t(n) + 2$. Its Fourier support has $12k(n)^2 + 8$ members, and every nonzero Fourier coefficient has magnitude at least $1/(8k(n))$.

The four bounds. Fix a length $n \geq 16$ with $a(n) \geq 8$, and write $k = k(n)$ and $t = t(n)$ and $a = a(n)$. Then $k \geq 2$ and $t = \log_2 k$, and $4k \leq a < 8k$ by Eq. (18). The leap is 1 as stated. For the arity, first $t + 2 \leq 2k$. This holds at $k = 2$, where the two sides are 3 and 4. It is preserved when k doubles, since the left side then grows by 1 and the right side by $2k \geq 4$. Hence the arity is $2k + t + 2 \leq 4k \leq a$. For the support, $k \leq a/4$ gives $12k^2 + 8 \leq 12(a/4)^2 + 8 = \frac{3}{4}a^2 + 8$. That is at most a^2, because $a^2/4 \geq 16 \geq 8$ when $a \geq 8$. For the coefficients, $k \leq a/4$ gives $1/(8k) \geq 1/(2a)$. By Theorem B.4, f meets every requirement of Definition 2.1 except the arity bound, and $f \notin$ Learn. Its arity $2k + t + 2$ exceeds $2k$, which is greater than $a/4$ by Eq. (18), and $a/4$ is $\omega(\log n)$. So at large n the arity of f_n lies above every $c \log n$ while staying at most $a(n)$.

The statement of Corollary B.5 ends with two sentences about Theorem 4.3. The first of them is that theorem specialized to width $w = 1$. The second reads that theorem together with the witness built here. □

The next proposition is the lower bound the characterization needs, proved in the paper's own model with absolute constants and uniformly over shapes that vary with n, and restated from Section 1.1 with its hypotheses in full.

Proposition B.6 (Uniform correlational leap lower bound, restatement of Proposition 1.2)**.** *Let f be a query whose arity is at most $c \log n$ for a constant c. There is an n_0 depending only on c such that the following holds for every $n \geq n_0$. Write $g = f_n$ for the function at length n, and*

$\ell = \mathrm{Leap}(g)$ *for its leap. Let* R *be the relevant set of* g *and let* $a = |R|$ *be its size, which is the arity of* g*. Let* $P = |\mathrm{supp}(g)|$ *count the sets in the Fourier support, and let* $n' = n - a$ *count the coordinates outside* R*. Suppose* g *is not constant, so that* $\ell \geq 1$*. Suppose a deterministic learner makes at most* t *correlational queries, each of tolerance at least* τ*. Suppose also that for every target* $g' \in \mathcal{O}_n(g)$ *and every oracle legal for* g' *under Definition 3.1 it outputs a hypothesis* h *with* $\mathbf{Pr}_x[h(x) \neq g'(x)] < 2^{-2a}/2$*. Call that last condition the* accuracy hypothesis*. Then*

$$t \geq \frac{1}{2} \cdot \frac{\binom{n'}{\ell}\tau^2}{P}. \tag{19}$$

In particular, if $\tau \geq n^{-\beta}$ *for a constant* β*, then*

$$t \geq \frac{1}{2} n^{-2\beta-c} \left(\frac{n - c\log n}{c\log n}\right)^{\ell}, \tag{20}$$

which is at least $n^{(1-o(1))\ell-2\beta-c}$ *with an* $o(1)$ *that depends on* n *and* c *alone.*

Proof of Proposition B.6. The shape is that of Lemma B.3: a sub-family of the orbit, an adversary whose answers are legal for most of it, and a count of what the learner can then tell apart. The first step splits the support into two parts: the part that a search of width $\ell - 1$ can reach, and the part it cannot. The sub-family comes from placing the coordinates of the second part. Write $w = \ell - 1 \geq 0$ for that width.

Recall that every support set is a subset of R and that the support sets cover R, that is $\mathrm{supp}(g) \subseteq 2^R$ with $\bigcup \mathrm{supp}(g) = R$. By hypothesis $a \leq c\log n$. Every support set lies inside R, so $P \leq 2^a \leq n^c$. The leap satisfies $1 \leq \ell \leq a$, and both bounds are read off Definition 4.1, which applies because a $\{-1, 1\}$-valued g is nonzero. For the lower bound, the support of g contains a nonempty set, since a function whose support is contained in $\{\emptyset\}$ is the constant $\widehat{g}(\emptyset)$ and g is not constant. In any ordering, let S be the first nonempty set listed. Every set listed before it is empty, so the jump of S is $|S| \geq 1$, and the largest jump of every ordering is at least 1. For the upper bound, every support set S lies inside R, so every jump of every ordering is at most $|S| \leq a$. Parseval gives $\sum_S \widehat{g}(S)^2 = 1$, since g is $\{-1, 1\}$-valued.

Take n_0 so large that $c\log n \leq (n - c\log n)/2$ and $c\log n/(n - 2c\log n) \leq 1/2$ for all $n \geq n_0$. The first of these is what the second-moment bound below needs, and the second is what the identification step needs. Both are conditions on n and c alone, and such an n_0 exists because $c\log n = o(n)$. The first condition, rearranged, says $n \geq 3c\log n$, and two consequences of it are used below. First, $n - 2c\log n \geq c\log n \geq a \geq 1$, so the denominator in the second condition is positive. Second, $n' = n - a \geq n - c\log n \geq 2c\log n \geq a$, so $n' \geq a \geq \ell$ and $\binom{n'}{\ell} \geq 1$.

The stuck closure. Let $A_0 = \emptyset$ and $U_0 = \emptyset$, and for $j \geq 0$ let $A_{j+1} = A_j \cup \{S \in \mathrm{supp}(g) : |S \setminus U_j| \leq w\}$ and $U_{j+1} = \bigcup_{S \in A_{j+1}} S$. In words, stage $j + 1$ collects every support set that introduces at most w coordinates beyond those the earlier stages gathered, and U_{j+1} is the set of coordinates the collected sets use. The sets A_j increase with j, since A_{j+1} is A_j together with further sets. They are subsets of the finite set $\mathrm{supp}(g)$, so the indices j with $A_{j+1} \neq A_j$ number at most P, and beyond the last of them the sequence is constant. Let j^* be an index beyond which it is constant, and put $A^* = A_{j^*}$ and $U^* = \bigcup_{S \in A^*} S$, so that $U_{j^*} = U^*$. The pair (A^*, U^*) is a fixed point: the equality $A_{j^*+1} = A_{j^*}$ says that every $S \in \mathrm{supp}(g)$ with $|S \setminus U^*| \leq w$ already lies in A^*. Every member of A^* is a support set and so lies inside R, hence $U^* \subseteq R$. Also, every $S \in A^*$ satisfies $S \subseteq U^*$ by the definition of U^*. The sets in A^* are the ones a search at width w reaches, and U^* holds their coordinates. Put $B = \mathrm{supp}(g) \setminus A^*$. Being a fixed point means exactly that no $S \in B$ has $|S \setminus U^*| \leq w$. Sizes are integers and $w + 1 = \ell$, so

$$|S \setminus U^*| \geq \ell \qquad \text{for every } S \in B. \tag{21}$$

Moreover $B \neq \emptyset$. Otherwise $A^* = \mathrm{supp}(g)$, and the support can be listed by the stage at which each set entered, in any order within a stage. A set S that entered at stage $j + 1$ has $|S \setminus U_j| \leq w$. Every set of A_j entered at an earlier stage, and so is listed before S. Hence the union of the sets listed before S contains U_j. The jump of S is therefore at most w. So the listing by stage is an ordering of $\mathrm{supp}(g)$ with every jump at most $w = \ell - 1$, against $\mathrm{Leap}(g) = \ell$. Take any $S \in B$. It is a support set, so $S \subseteq R$ and hence $S \setminus U^* \subseteq R \setminus U^*$. With Eq. (21) this gives $|R \setminus U^*| \geq |S \setminus U^*| \geq \ell$. Write $b = |R \setminus U^*|$ for its size, so that $\ell \leq b \leq a$.

The sub-family. Fix an injection $\xi\colon U^* \to [n]$, which places the reached coordinates once and for all. Let $\Omega = [n] \setminus \xi(U^*)$ be the positions that placement leaves free. Call an injection $\varrho\colon R \setminus U^* \to \Omega$ a *placement* of the unreached coordinates, and let Λ be the set of placements, so that

$$|\Lambda| = n''(n''-1)\cdots(n''-b+1), \qquad n'' := |\Omega| = n - |U^*| \geq n'. \tag{22}$$

Here $|\Omega| = n - |U^*|$ because ξ is injective, and $n'' \geq n'$ because $U^* \subseteq R$ gives $|U^*| \leq a$. The product in Eq. (22) is positive, since $n'' \geq n' \geq a \geq b$ by the choice of n_0, so Λ is not empty. For a placement ϱ, the map $\xi \cup \varrho$ sends U^* by ξ and $R \setminus U^*$ by ϱ. It is an injection of R into $[n]$: ξ and ϱ are injective, and their images are disjoint because ϱ takes its values in $\Omega = [n] \setminus \xi(U^*)$. Extend it to a permutation σ of $[n]$ by mapping $[n] \setminus R$ bijectively onto $[n] \setminus (\xi \cup \varrho)(R)$, which is possible because both sets have $n - a$ elements. Let $g_\varrho = g^\sigma$, a member of $\mathcal{O}_n(g)$ by Definition 3.2. By the rule for g^σ in Section 2.1, the coefficient of g_ϱ at the moved set $\sigma(S)$ is $\widehat{g_\varrho}(\sigma(S)) = \widehat{g}(S)$, and $\widehat{g_\varrho}$ vanishes at every set not of that form. Written out, that rule is the expansion $g_\varrho = \sum_{S \in \mathrm{supp}(g)} \widehat{g}(S)\chi_{\sigma(S)}$. Every support set of g lies inside R, and σ agrees with $\xi \cup \varrho$ there. So $\sigma(S) = (\xi \cup \varrho)(S)$ for every $S \in \mathrm{supp}(g)$. Hence g_ϱ depends only on $\xi \cup \varrho$ and not on the extension σ. For $S \in \mathrm{supp}(g)$ put $V_S(\varrho) = \xi(S \cap U^*) \cup \varrho(S \setminus U^*) = \sigma(S)$, the image of S once the placement is made.

Splitting a query. Every $S \in A^*$ satisfies $S \subseteq U^*$, so $V_S(\varrho) = \xi(S)$ does not depend on ϱ. Let $\psi\colon \{-1,1\}^n \to [-1,1]$. Expanding both ψ and g_ϱ by Eq. (1) and using orthonormality of the characters, $\mathbb{E}_x[\psi g_\varrho] = \sum_{T \subseteq [n]} \widehat{\psi}(T)\widehat{g_\varrho}(T)$. The coefficient $\widehat{g_\varrho}(T)$ is nonzero only at $T = \sigma(S)$ for $S \in \mathrm{supp}(g)$, where it equals $\widehat{g}(S)$. Distinct S give distinct $\sigma(S)$, because σ is injective. So $\mathbb{E}_x[\psi g_\varrho] = \sum_{S \in \mathrm{supp}(g)} \widehat{g}(S)\widehat{\psi}(V_S(\varrho))$. Hence, splitting the sum into the sets $S \in A^*$, where $V_S(\varrho) = \xi(S)$, and the sets $S \in B$,

$$\begin{aligned} \mathbb{E}_x[\psi g_\varrho] &= c_\psi + D_\varrho(\psi), \qquad c_\psi = \sum_{S \in A^*} \widehat{g}(S)\widehat{\psi}(\xi(S)), \\ D_\varrho(\psi) &= \sum_{S \in B} \widehat{g}(S)\widehat{\psi}(V_S(\varrho)). \end{aligned} \tag{23}$$

Here c_ψ is the part of the correlation that the placement cannot change, the same number for every ϱ, and $D_\varrho(\psi)$ is the part that the placement moves.

The placement count. Fix a set $S \in B$. Put $s = |S \setminus U^*|$, the number of coordinates of S that the placement moves. Then $\ell \leq s$ by Eq. (21), and $s \leq b \leq a$ because $S \setminus U^* \subseteq R \setminus U^*$. Let V be a set of coordinates and ask when $V_S(\varrho) = V$. The reached part of S always lands on $\xi(S \cap U^*)$, a subset of $\xi(U^*)$. The moved part lands on $\varrho(S \setminus U^*)$, a subset of Ω with exactly s elements because ϱ is injective. The sets $\xi(U^*)$ and Ω are disjoint, so $V_S(\varrho) = V$ holds only if $V \cap \xi(U^*) = \xi(S \cap U^*)$ and $V' := V \cap \Omega$ has s elements. When it holds, V is the union of these two parts. Call such a V admissible; there are $\binom{n''}{s}$ of them, one for each s-element subset V' of Ω. Given such a V, the equality asks ϱ to map $S \setminus U^*$ onto V'. The placements that do so number $s!(n''-s)\cdots(n''-b+1)$: there are $s!$ bijections onto V', and then an injection of the remaining $b-s$ elements into the remaining $n''-s$ coordinates. Dividing by Eq. (22), for ϱ uniform on Λ,

$$\begin{aligned} \mathbf{Pr}_\varrho[V_S(\varrho) = V] &= \frac{s!(n''-s)\cdots(n''-b+1)}{n''(n''-1)\cdots(n''-b+1)} \\ &= \frac{s!}{n''(n''-1)\cdots(n''-s+1)} = \frac{s!(n''-s)!}{n''!} = \binom{n''}{s}^{-1} \end{aligned} \tag{24}$$

for each of the $\binom{n''}{s}$ admissible V, and 0 for every other V. The second equality cancels the factors $(n''-s)\cdots(n''-b+1)$ common to the numerator and the denominator, a product that is empty, and equal to 1, when $s = b$.

The second moment of a single query. The Cauchy–Schwarz inequality, in the form $\left(\sum_{S \in B} y_S\right)^2 \leq |B| \sum_{S \in B} y_S^2$ with $y_S = \widehat{g}(S)\widehat{\psi}(V_S(\varrho))$, gives $D_\varrho(\psi)^2 \leq |B| \sum_{S \in B} \widehat{g}(S)^2 \widehat{\psi}(V_S(\varrho))^2$. For a fixed

$S \in B$ with $s = |S \setminus U^*|$, Eq. (24), Parseval's identity for ψ and $\psi(x)^2 \le 1$ give

$$\begin{aligned}\mathbb{E}_\varrho\left[\widehat{\psi}\left(V_S(\varrho)\right)^2\right] &= \binom{n''}{s}^{-1} \sum_{V \text{ admissible}} \widehat{\psi}(V)^2 \\ &\le \binom{n''}{s}^{-1} \sum_{V \subseteq [n]} \widehat{\psi}(V)^2 = \binom{n''}{s}^{-1} \mathbb{E}_x\left[\psi(x)^2\right] \le \binom{n''}{s}^{-1}.\end{aligned} \tag{25}$$

Taking expectations over ϱ in the Cauchy–Schwarz bound, by linearity, and inserting Eq. (25) for each $S \in B$,

$$\mathbb{E}_\varrho\left[D_\varrho(\psi)^2\right] \le |B| \sum_{S \in B} \widehat{g}(S)^2 \binom{n''}{|S \setminus U^*|}^{-1} \le \frac{|B|}{\binom{n''}{\ell}} \sum_{S \in B} \widehat{g}(S)^2 \le \frac{P}{\binom{n'}{\ell}}. \tag{26}$$

The middle inequality uses Eq. (21) together with the monotonicity of the binomial coefficient, namely that $\binom{n''}{\ell} \le \binom{n''}{s}$ whenever $\ell \le s \le n''/2$. That holds because $\binom{m}{r+1}/\binom{m}{r} = (m-r)/(r+1)$, which is at least 1 exactly when $r \le (m-1)/2$. Each step $r \to r+1$ from ℓ up to s has $r+1 \le s \le m/2$ with $m = n''$, hence $r \le m/2 - 1 \le (m-1)/2$. That range covers every s that occurs, since $\ell \le s \le a \le c \log n \le (n - c\log n)/2 \le n''/2$. In that chain, the inequality $c \log n \le (n - c\log n)/2$ holds by the choice of n_0, and the last holds because $n'' \ge n' = n - a \ge n - c\log n$. The last inequality in Eq. (26) uses three facts. First, $\sum_{S \in B} \widehat{g}(S)^2 \le \sum_S \widehat{g}(S)^2 = 1$ by Parseval. Second, $|B| \le P$ since $B \subseteq \operatorname{supp}(g)$. Third, $\binom{n''}{\ell} \ge \binom{n'}{\ell}$, because $n'' \ge n'$ and $m \mapsto \binom{m}{\ell}$ is nondecreasing by Pascal's rule $\binom{m+1}{\ell} = \binom{m}{\ell} + \binom{m}{\ell-1} \ge \binom{m}{\ell}$. Markov's inequality applied to the nonnegative random variable $D_\varrho(\psi)^2$ at the level τ^2, together with Eq. (26), gives for every ψ

$$\mathbf{Pr}_\varrho\left[|D_\varrho(\psi)| > \tau\right] \le \frac{1}{\tau^2}\mathbb{E}_\varrho\left[D_\varrho(\psi)^2\right] \le p := \frac{P}{\binom{n'}{\ell}\tau^2}. \tag{27}$$

The adversary. If $tp \ge 1/2$, the desired bound already holds, so assume $tp < 1/2$. Let the oracle answer c_ψ to every query ψ. Whenever $|D_\varrho(\psi)| \le \tau$, Eq. (23) makes this answer legal for the target g_ϱ at tolerance τ, and hence at every larger tolerance. By Eq. (27) and a union bound, every prefix of at most t queries is legal for at least a fraction $1 - tp > 0$ of placements. The learner must therefore halt with an output within its query bound, since otherwise it fails on a surviving target. The learner is deterministic and receives the same answer to each of its queries whatever ϱ is, so its queries $\psi_1, \dots, \psi_t$ and its output h form one fixed transcript. Let $\Lambda_t = \{\varrho \in \Lambda : |D_\varrho(\psi_j)| \le \tau \text{ for every } j \le t\}$ be the placements against which every answer of that transcript is legal. The queries $\psi_1, \dots, \psi_t$ are fixed before ϱ is drawn, so Eq. (27) applies to each of them. A union bound over the t queries gives $\mathbf{Pr}_\varrho\left[\varrho \notin \Lambda_t\right] \le \sum_{j \le t} \mathbf{Pr}_\varrho\left[|D_\varrho(\psi_j)| > \tau\right] \le tp$, and the bound only improves if the transcript has fewer than t queries. Since ϱ is uniform on Λ, $|\Lambda_t| = |\Lambda|\,\mathbf{Pr}_\varrho\left[\varrho \in \Lambda_t\right]$, and so

$$|\Lambda_t| \ge |\Lambda|\,(1 - tp)\,. \tag{28}$$

Identification. Suppose $g_\varrho = g_{\varrho'}$ for two placements. The two functions then have the same relevant set. The support of g_ϱ is $\{\sigma(S) : S \in \operatorname{supp}(g)\}$, and the union of those sets is $\sigma(R)$ because the support sets of g cover R. So the relevant set of g_ϱ is $(\xi \cup \varrho)(R) = \xi(U^*) \cup \varrho(R \setminus U^*)$, and likewise that of $g_{\varrho'}$ is $\xi(U^*) \cup \varrho'(R \setminus U^*)$. Intersecting both with Ω, which is disjoint from $\xi(U^*)$ and contains the images of ϱ and ϱ', gives $\varrho(R \setminus U^*) = \varrho'(R \setminus U^*)$. So ϱ and ϱ' are bijections from $R \setminus U^*$ onto one and the same set. Hence $\alpha = \varrho^{-1} \circ \varrho'$ is a permutation of $R \setminus U^*$, and $\varrho' = \varrho \circ \alpha$. Each function g_ϱ therefore arises from at most $K := b!$ placements, one for each such permutation.

Suppose $|\Lambda_t| > K$. Then Λ_t contains two placements ϱ and ϱ' with $g_\varrho \ne g_{\varrho'}$. Otherwise, every member of Λ_t would give one and the same function, and Λ_t would have at most K members. Both g_ϱ and $g_{\varrho'}$ depend only on the coordinates in $(\xi \cup \varrho)(R) \cup (\xi \cup \varrho')(R)$, a set of at most $2a$ coordinates. Call that set W, so $|W| \le 2a$. Two distinct functions that both depend only on the coordinates in W differ at some assignment z to those coordinates, and they differ at every input x that agrees with z on W. Those inputs form an event of probability $2^{-|W|} \ge 2^{-2a}$ under the uniform distribution.

For each $\mu \in \{\varrho, \varrho'\}$, let the oracle O_μ answer a query ψ with c_ψ when $|D_\mu(\psi)| \le \tau$ and with $\mathbb{E}_x[\psi g_\mu]$ otherwise. By Eq. (23) both answers lie within τ of $\mathbb{E}_x[\psi g_\mu]$, so O_μ is legal for g_μ at

every tolerance of at least τ. Since $\mu \in \Lambda_t$, the oracle O_μ answers each ψ_j with c_{ψ_j}. By induction on j, the deterministic learner facing g_μ and O_μ asks $\psi_1, \ldots, \psi_t$ in turn and outputs the h of the fixed transcript. The accuracy hypothesis applies to both runs, since $g_\varrho, g_{\varrho'} \in \mathcal{O}_n(g)$, and a point where g_ϱ and $g_{\varrho'}$ differ is a point where h differs from at least one of them, so

$$2^{-2a} \leq \mathbf{Pr}_x \left[g_\varrho \neq g_{\varrho'}\right] \leq \mathbf{Pr}_x \left[h \neq g_\varrho\right] + \mathbf{Pr}_x \left[h \neq g_{\varrho'}\right] < 2^{-2a}, \tag{29}$$

which is false.

So $|\Lambda_t| \leq K$, and with Eq. (28) this gives $|\Lambda|(1 - tp) \leq |\Lambda_t| \leq K$. Dividing by $|\Lambda| > 0$ gives $tp \geq 1 - K/|\Lambda|$. Now bound the ratio $K/|\Lambda|$. Each of the b factors in Eq. (22) is at least $n'' - b + 1$, so $|\Lambda| \geq (n'' - b + 1)^b$. Since $b! \leq b^b$, the ratio is at most $(b/(n'' - b + 1))^b$. Here $b \leq a \leq c \log n$ and $n'' \geq n' \geq n - c \log n$, so $n'' - b + 1 > n - 2c \log n > 0$, the last by the choice of n_0. Hence the fraction inside is at most $c \log n/(n - 2c \log n)$. That fraction is at most $1/2$ by the choice of n_0. A number $x \in [0, 1/2]$ raised to the power $b \geq 1$ is at most x, so $K/|\Lambda| \leq 1/2$. Therefore $tp \geq 1 - 1/2 = 1/2$, and since $p > 0$ we get $t \geq 1/(2p) = \frac{1}{2} \cdot \binom{n'}{\ell} \tau^2 / P$, which is Eq. (19).

The explicit form. Suppose $\tau \geq n^{-\beta}$. For integers $1 \leq \ell \leq m$ we have $\binom{m}{\ell} = \prod_{i=0}^{\ell-1} \frac{m-i}{\ell-i} \geq \left(\frac{m}{\ell}\right)^\ell$. Each factor satisfies $\frac{m-i}{\ell-i} \geq \frac{m}{\ell}$, which rearranges to $i(m - \ell) \geq 0$. Apply this with $m = n'$, which is at least ℓ by the choice of n_0. Then $n' \geq n - c \log n$ while $1 \leq \ell \leq c \log n$, so

$$\binom{n'}{\ell} \geq \left(\frac{n'}{\ell}\right)^\ell \geq \left(\frac{n - c \log n}{c \log n}\right)^\ell. \tag{30}$$

Substituting this, $\tau^2 \geq n^{-2\beta}$ and $P \leq n^c$ into Eq. (19) gives Eq. (20). The logarithm of the base in Eq. (20) is $\log(n - c \log n) - \log(c \log n) = \log n - \log(c \log n) + \log(1 - c \log n/n)$. Write it as $(1 - \delta_1) \log n$, where $\delta_1 = (\log(c \log n) - \log(1 - c \log n/n)) / \log n$. This δ_1 is a function of n and c alone. It tends to 0 as n grows, because $\log(c \log n) = o(\log n)$ and $c \log n/n \to 0$. The remaining factor $\frac{1}{2}$ is $n^{-1/\log n}$, because $n^{1/\log n} = 2^{\log n/\log n} = 2$ for every $n \geq 2$. So the right-hand side of Eq. (20) equals $n^{(1-\delta_1)\ell - 1/\log n - 2\beta - c}$. Since $\ell \geq 1$, we have $1/\log n \leq \ell/\log n$, so that exponent is at least $(1 - \delta)\ell - 2\beta - c$ with $\delta = \delta_1 + 1/\log n$. The quantity δ depends on n and c alone and tends to 0, which gives $n^{(1-o(1))\ell - 2\beta - c}$ as a lower bound. □

The upper bound and the lower bound meet in the next theorem, which is the leap-side reading of the characterization.

Theorem B.7 (Semantic characterization)**.** *Let f be tame with parameters (c, q). Then $f \in$ Learn if and only if $\mathrm{Leap}(f_n) = O(1)$. More precisely, if $\mathrm{Leap}(f_n) \leq \ell$ for every n, then $\mathcal{O}(f)$ is learnable in time $n^{O(\ell+c)}$, at every accuracy. In the other direction, let β be a constant, let $n_0 = n_0(c)$ be the threshold of Proposition B.6, and consider a length $n \geq n_0$ at which f_n is not constant. Every deterministic learner that reaches accuracy $n^{-2c}/4$ on $\mathcal{O}_n(f_n)$ with queries of tolerance at least $n^{-\beta}$ makes at least the number of queries in Eq.* (20)*, with $\ell = \mathrm{Leap}(f_n)$ there.*

Proof of Theorem B.7. The upper bound, the count and the only-if direction are proved in that order. If $\mathrm{Leap}(f_n) \leq \ell$ for every n, then the hypotheses of Theorem 4.3 hold with $w = \ell$. Namely, f is tame with parameters (c, q) and $w \geq \mathrm{Leap}(f_n)$ at every length. That theorem then learns $\mathcal{O}(f)$ in time $n^{O(\ell+c)}$ at every accuracy, with tolerance at least $1/\mathrm{poly}(n)$.

Now suppose $\mathrm{Leap}(f_n) = O(1)$, so that there are constants C and n_1 with $\mathrm{Leap}(f_n) \leq C$ for every $n \geq n_1$. Each of the finitely many f_n with $n < n_1$ has leap at most its arity, so $\ell = \max(C, \max_{n<n_1} \mathrm{Leap}(f_n))$ is a constant with $\mathrm{Leap}(f_n) \leq \ell$ for every n. For that constant ℓ the time $n^{O(\ell+c)}$ is polynomial, because c is a constant too, so $f \in$ Learn.

Turn to the lower bound. Fix β, a length $n \geq n_0$ at which f_n is not constant, and a deterministic learner as in the statement, and write a for the arity of f_n. Tameness gives $a \leq c \log n$ at every length, which is the arity hypothesis of Proposition B.6. It also gives $2^{-2a} \geq 2^{-2c \log n} = n^{-2c}$, hence $2^{-2a}/2 \geq n^{-2c}/2 > n^{-2c}/4$. So the learner, whose error is at most $n^{-2c}/4$ against every target in $\mathcal{O}_n(f_n)$ and every legal oracle, has error strictly below $2^{-2a}/2$ there, which is the accuracy hypothesis of Proposition B.6. Let t be the largest number of queries it makes over those targets and

oracles; if there is no largest number, there is nothing to prove. Each query has tolerance at least $\tau = n^{-\beta}$, so Proposition B.6 applies with this t and this τ, and Eq. (20) is the stated count.

Now suppose $f \in$ Learn and, for contradiction, that $\mathrm{Leap}(f_n)$ is unbounded. Run the learner of Definition 3.3 at accuracy $\varepsilon = n^{-2c}/4$. It is deterministic. By Definition 3.3 there are constants $C \geq 1$ and d such that, at accuracy ε, the learner runs in time at most $C(n/\varepsilon)^d$ and makes queries of tolerance at least $\left(C(n/\varepsilon)^d\right)^{-1}$. At $\varepsilon = n^{-2c}/4$ we have $n/\varepsilon = 4n^{1+2c}$, so $C(n/\varepsilon)^d = C4^d n^{d(1+2c)} \leq n^{\beta'}$ for every $n \geq 2$, with $\beta' = d(1+2c) + \log(C4^d)$. So the learner runs in time at most $n^{\beta'}$ and makes queries of tolerance at least $n^{-\beta}$ with $\beta = \beta'$, both constants.

By the count just proved, at every $n \geq n_0$ at which f_n is not constant it makes at least the number of queries in Eq. (20) with $\ell = \mathrm{Leap}(f_n)$. Put $L = 2(\beta' + 2\beta + c + 1)$. Infinitely many lengths n have $\mathrm{Leap}(f_n) \geq L$: if only finitely many did, the leap would be bounded by the largest of L and their leaps, against the assumption that it is unbounded. At each such length f_n is not constant, since a constant f_n has support $\{\emptyset\}$ and leap $0 < L$. Once n is so large that $n \geq 2c \log n$ and $\sqrt{n} \geq 2c \log n$, a condition on c alone, the base of the power in Eq. (20) satisfies $\frac{n - c\log n}{c \log n} \geq \frac{n/2}{c \log n} \geq \sqrt{n}$.

So at every length with $\mathrm{Leap}(f_n) \geq L$, $n \geq n_0$, $n \geq 3$ and these two conditions, and there are infinitely many, the count there is at least $\frac{1}{2} n^{-2\beta - c} n^{\ell/2}$ with $\ell = \mathrm{Leap}(f_n)$. The exponent $\ell/2 - 2\beta - c$ is at least $\beta' + 1$, because $\ell \geq L$ gives $\ell/2 \geq \beta' + 2\beta + c + 1$. So that count is at least $\frac{1}{2} n^{\beta' + 1}$, which exceeds $n^{\beta'}$ because $n \geq 3$. Each query takes at least one step, so at those lengths the learner makes at most $n^{\beta'}$ queries on every run, while by the count it makes more than $n^{\beta'}$ on some run. That contradiction shows that $\mathrm{Leap}(f_n)$ is bounded, which is $\mathrm{Leap}(f_n) = O(1)$. □

The next lemma is the bridge between the syntax of Definition 5.1 and the leap of Definition 4.1, and both halves of the characterization pass through it.

Lemma B.8 (Bridge). *Let $g\colon \{-1,1\}^n \to \{-1,1\}$ be a Boolean function and let $w \geq 0$ be a width.*

1. *If g has a width-w correlationally visible staircase derivation under the uniform distribution, then* $\mathrm{Leap}(g) \leq w$.
2. *Let q be a positive function of n and suppose that every nonzero Fourier coefficient of g has magnitude at least $1/q(n)$. Then* every *ordering of* $\mathrm{supp}(g)$ *whose jumps are all at most w yields a sequence of features and weights with the properties below. That sequence meets the base, bounded-introduction and readout conditions of Definition 5.1 at width w. It also meets Eq.* (4) *at every step, with* $1/\mathrm{poly}(n)$ *replaced by* $1/q(n)$. *When q is bounded by a polynomial, $1/q(n)$ is the inverse polynomial that Eq.* (4) *asks for, and the sequence is a width-w derivation of g. The features of that sequence are the constant* 1 *followed by the characters of the nonempty sets of the ordering, taken in order, and its weights are the corresponding Fourier coefficients of g. Every visibility margin in Eq.* (4) *is at least* $1/q(n)$. *An ordering of* $\mathrm{supp}(g)$ *whose jumps are all at most w exists whenever* $\mathrm{Leap}(g) \leq w$.

Proof of Lemma B.8. From a derivation to a small leap. Here an ordering of $\mathrm{supp}(g)$ is read off the derivation, and every jump of that ordering is shown to be at most w, which is what $\mathrm{Leap}(g) \leq w$ asks for. The sets of the ordering are the witnesses of the steps that introduce a coordinate, listed in the order those steps occur, with the remaining support sets appended at the end. Let $\psi_0, \ldots, \psi_m$ be a width-w derivation, with C_i, U_i and F_{i+1} as in Definition 5.1. Then $U_0 = \emptyset$. Call a step $i+1$ *productive* when $F_{i+1} \neq \emptyset$. At a productive step let S_{i+1} be a witness supplied by Eq. (4), so that $F_{i+1} \subseteq S_{i+1} \subseteq C_{i+1}$.

Each ψ_j with $j \leq i$ is a function of the coordinates in $C_j \subseteq U_i$, so every element of the span of $\psi_0, \ldots, \psi_i$ is a function of the coordinates in U_i alone. Such a function has its Fourier support inside the subsets of U_i, by the following fact, which is used once more below. Let h be a function of the coordinates in a set $V \subseteq [n]$ alone, and let S be a set that is not contained in V. Pick a coordinate $k \in S \setminus V$. The product $h\chi_{S\setminus\{k\}}$ does not read x_k, because h reads only coordinates in V, which does not contain k, and $\chi_{S\setminus\{k\}}$ reads only coordinates in $S \setminus \{k\}$. Under the uniform distribution x_k is independent of the other coordinates and has mean 0, so

$$\widehat{h}(S) = \mathbb{E}_x\left[h(x)\chi_{S\setminus\{k\}}(x)x_k\right] = \mathbb{E}_x\left[h(x)\chi_{S\setminus\{k\}}(x)\right]\mathbb{E}_x\left[x_k\right] = 0. \tag{31}$$

The projection $\Pi_i g$ lies in the span of $\psi_0, \ldots, \psi_i$, so $\widehat{\Pi_i g}(S) = 0$ for every S that is not contained in U_i. The Fourier coefficient at S is linear in the function, so $\widehat{(g - \Pi_i g)}(S) = \widehat{g}(S) - \widehat{\Pi_i g}(S) = \widehat{g}(S)$ for every S that is not contained in U_i. At a productive step the witness S_{i+1} is a set of that kind, that is, one not contained in U_i. The reason is that F_{i+1} is nonempty, that F_{i+1} is contained in S_{i+1}, and that F_{i+1} is disjoint from U_i. Visibility at step $i+1$, with the witness S_{i+1}, gives $\left|\widehat{(g - \Pi_i g)}(S_{i+1})\right| \geq 1/\mathrm{poly}(n) > 0$. By the previous paragraph that residual coefficient is $\widehat{g}(S_{i+1})$, so $\widehat{g}(S_{i+1}) \neq 0$, that is $S_{i+1} \in \mathrm{supp}(g)$.

Moreover $S_{i+1} \subseteq C_{i+1}$ gives $S_{i+1} \setminus U_i \subseteq C_{i+1} \setminus U_i = F_{i+1}$, and $F_{i+1} \subseteq S_{i+1}$ gives the reverse inclusion, so

$$S_{i+1} \setminus U_i = F_{i+1} \qquad \text{and} \qquad U_{i+1} = U_i \cup C_{i+1} = U_i \cup F_{i+1} = U_i \cup S_{i+1}. \tag{32}$$

A step that is not productive leaves U_i unchanged, because $F_{i+1} = \emptyset$ says $C_{i+1} \subseteq U_i$, so $U_{i+1} = U_i \cup C_{i+1} = U_i$. By induction along the derivation, U_i is exactly the union of the witnesses of the productive steps up to i. At $i = 0$ both sides are empty, since $U_0 = C_0 = \emptyset$ and no step precedes position 0. If the claim holds at i, then at a productive step $i+1$ Eq. (32) adds to U_i exactly the new witness S_{i+1}, and at a step that is not productive neither side changes, so the claim holds at $i+1$.

List those witnesses in the order their steps occur. They are pairwise distinct, since at a productive step $i+1$ the witness is not contained in U_i, while every earlier witness S_{j+1}, with $j+1 \leq i$, lies in $U_{j+1} \subseteq U_i$ by Eq. (32). By Eq. (32) the jump of S_{i+1} against the union of the earlier witnesses is $|S_{i+1} \setminus U_i| = |F_{i+1}| \leq w$. This is an ordering of part of $\mathrm{supp}(g)$ with every jump at most w. It remains to place the rest. Each feature ψ_i reads only coordinates in $C_i \subseteq U_m$, so the weighted sum $\sum_i a_i \psi_i$ reads only coordinates in U_m, and so does its sign, which by the readout condition is g. By Eq. (31), applied with $h = g$ and $V = U_m$, every set in $\mathrm{supp}(g)$ is a subset of U_m, that is, $\mathrm{supp}(g) \subseteq 2^{U_m}$. The set U_m is the union of the listed witnesses. Every unlisted element of $\mathrm{supp}(g)$ is therefore contained in the union of the listed ones and contributes a jump of 0 when appended in any order. The resulting ordering of $\mathrm{supp}(g)$ has all jumps at most w, so $\mathrm{Leap}(g) \leq w$.

From a small leap to a derivation. Here the sequence is written down from the ordering, and the four conditions of Definition 5.1 are checked in turn. Let π be an ordering of $\mathrm{supp}(g)$ with every jump at most w. One such ordering exists when $\mathrm{Leap}(g) \leq w$, by Definition 4.1. Deleting $\emptyset$ from π, if it occurs there, changes no jump, since $\emptyset$ enlarges no union and introduces nothing. Relabel the remaining sets as $S_1, \ldots, S_P$ in the order π gives them, so that every jump is at most w. Put $\psi_0 = 1$, which is $\chi_\emptyset$, and $\psi_i = \chi_{S_i}$ for $i \geq 1$, so $C_i = S_i$. The base condition holds, because ψ_0 is the constant 1, which reads no coordinate, so $C_0 = \emptyset$. Each U_i is the union of $S_1, \ldots, S_i$, so $F_{i+1} = S_{i+1} \setminus U_i$ has size the jump of S_{i+1}. That jump is at most w, so bounded introduction holds.

The features $\psi_0, \ldots, \psi_i$ are the characters of the pairwise distinct sets $\emptyset, S_1, \ldots, S_i$, so they are orthonormal under the uniform distribution. The orthogonal projection onto the span of an orthonormal family sends g to the sum, over the members of the family, of the inner product of g with each member times that member. The inner product of g with a character χ_T is $\mathbb{E}_x\left[g(x)\chi_T(x)\right] = \widehat{g}(T)$, so $\Pi_i g = \widehat{g}(\emptyset) + \sum_{1 \leq j \leq i} \widehat{g}(S_j)\chi_{S_j}$.

Since S_{i+1} is distinct from each of $\emptyset, S_1, \ldots, S_i$, the projection removes nothing at S_{i+1}, and $\widehat{(g - \Pi_i g)}(S_{i+1}) = \widehat{g}(S_{i+1})$. Take $S = S_{i+1}$ as the witness of step $i+1$. Then $F_{i+1} \subseteq S \subseteq C_{i+1}$. The residual coefficient at S is $\widehat{g}(S_{i+1})$, whose magnitude is at least $1/q(n)$ by hypothesis. And $\widehat{\psi_{i+1}}(S) = 1 \neq 0$, because ψ_{i+1} is the character χ_S. So Eq. (4) holds at step $i+1$ with $1/\mathrm{poly}(n)$ replaced by $1/q(n)$, and the visibility margin is at least $1/q(n)$. When q is bounded by a polynomial, $1/q(n)$ is an inverse polynomial, and visibility holds as Definition 5.1 states it.

Finally, the weighted sum of the features is $\widehat{g}(\emptyset) + \sum_{i \geq 1} \widehat{g}(S_i)\chi_{S_i}$, which is the Fourier expansion of g and so equals g. Since g takes values in $\{-1, 1\}$, it is the sign of that sum. Each coefficient is $\widehat{g}(S) = 2^{-n} \sum_x g(x)\chi_S(x)$, and the sum is an integer, since each of its 2^n terms is ± 1. So each weight is an integer multiple of 2^{-n} of absolute value at most 1. Its binary expansion is therefore its sign, its units digit and at most n digits after the binary point, which is $\mathrm{poly}(n)$-bit precision. The readout condition holds. So the sequence meets all four conditions of Definition 5.1, the visibility condition with margin $1/q(n)$, and it is a width-w derivation of g whenever q is bounded by a polynomial. □

Taking the two halves of Lemma B.8 together gives the following corollary, which Section 6 quotes: the least width of a derivation is the leap.

Corollary B.9 (The least width is the leap)**.** *Let $g\colon \{-1,1\}^n \to \{-1,1\}$ be nonconstant. Let q be a polynomial with positive values, and suppose that every nonzero Fourier coefficient of g has magnitude at least $1/q(n)$. Then g has a width-w correlationally visible staircase derivation under the uniform distribution if and only if $w \geq \mathrm{Leap}(g)$. So the least width of a derivation of g is $\mathrm{Leap}(g)$. Moreover a derivation of that width exists in which every visibility margin is at least $1/q(n)$.*

Proof of Corollary B.9. The leap is at least one. Since g is not constant, some $S \neq \emptyset$ has $\widehat{g}(S) \neq 0$. Otherwise Eq. (1) would read $g = \widehat{g}(\emptyset)\chi_\emptyset = \widehat{g}(\emptyset)$, a constant. So $\mathrm{supp}(g)$ contains a nonempty set. In any ordering of $\mathrm{supp}(g)$ let S be the first nonempty set. Every set before S is $\emptyset$, so the union of the earlier sets is empty, and the jump of S in Eq. (2) is $|S| \geq 1$. Every ordering therefore has a jump of at least 1, and $\mathrm{Leap}(g) \geq 1$ by Eq. (3).

No derivation of width below the leap. Let $w \geq 0$. Suppose g has a width-w derivation $\psi_0, \ldots, \psi_m$, with C_i, U_i and F_{i+1} as in Definition 5.1. If $w \geq 1$, then Lemma B.8(1) gives $\mathrm{Leap}(g) \leq w$. If $w = 0$, then bounded introduction forces $F_{i+1} = C_{i+1} \setminus U_i = \emptyset$ for every $i < m$, that is $C_{i+1} \subseteq U_i$. Every U_i and every C_i is then empty, by induction along the sequence. At $i = 0$, the base condition gives $U_0 = C_0 = \emptyset$. If $U_i = \emptyset$, then $C_{i+1} \subseteq U_i = \emptyset$, so $C_{i+1} = \emptyset$ and $U_{i+1} = U_i \cup C_{i+1} = \emptyset$. Each ψ_i is then a function of no coordinates, that is, a constant. So the readout $g = \mathrm{sign}\,(\sum_i a_i\psi_i)$ is constant, against the hypothesis. So a width-w derivation exists only when $w \geq \mathrm{Leap}(g)$.

A derivation at every width from the leap up. Put $w_0 = \mathrm{Leap}(g)$, which is at least 1 by the first step. The minimum in Eq. (3) is taken over the finitely many orderings of $\mathrm{supp}(g)$, so it is attained. So there is an ordering of $\mathrm{supp}(g)$ whose largest jump is w_0, and every jump in it is at most w_0. Lemma B.8(2), applied with the width w_0 and this q, turns it into a width-w_0 derivation of g whose every visibility margin is at least $1/q(n)$. As q is a polynomial, that margin meets the $1/\mathrm{poly}(n)$ that Eq. (4) asks for, so the sequence is a derivation in the sense of Definition 5.1. The hypothesis that q is a polynomial is spent only in that last clause. Everything else uses only that q is positive. A width-w_0 derivation is a width-w derivation for every $w \geq w_0$, since bounded introduction asks for $|F_{i+1}| \leq w$ and the other three conditions do not mention w. Hence a width-w derivation exists exactly when $w \geq \mathrm{Leap}(g)$, the least such w is $\mathrm{Leap}(g)$, and the derivation built at that width has the margin stated. □

Proof of Theorem 6.1. Fix n. Membership $f \in \mathsf{sFOM}_w$ supplies a width-w derivation of f_n under the uniform distribution, since the distribution is dropped from the notation exactly when it is uniform. The function f_n maps $\{-1,1\}^n$ to $\{-1,1\}$, so Lemma B.8(1) applies with $g = f_n$ and gives $\mathrm{Leap}(f_n) \leq w$. This holds at every n. Only the existence of the derivations is used here, not the definability of their features and weights. Since f is tame with parameters (c, q) and $w \geq \mathrm{Leap}(f_n)$ for every n, the hypotheses of Theorem 4.3 are met. So Algorithm 1 learns $\mathcal{O}(f)$ in time $n^{O(w+c)}$. For the second claim, a query in sFOM lies in sFOM_w for some constant w, and c is constant. So the exponent $O(w + c)$ is a constant, and the time bound $n^{O(w+c)}$ is a polynomial in n, hence a polynomial in n and $1/\varepsilon$. Learnability in that time, at every accuracy, is learnability in time $\mathrm{poly}(n, 1/\varepsilon)$ in the sense of Definition 3.3, which is $f \in \mathsf{Learn}$. □

Proof of Theorem 6.2. By Theorem B.7, applied to f, which is tame with parameters (c, q), the membership $f \in \mathsf{Learn}$ forces $\mathrm{Leap}(f_n) = O(1)$. So there are constants C and n_1 with $\mathrm{Leap}(f_n) \leq C$ for every $n \geq n_1$. At the finitely many lengths $n < n_1$ the leap is at most the arity, because every jump is at most the size of one support set and every support set consists of relevant coordinates, and the arity is at most $c \log n < c \log n_1$. Fix a constant $\ell \geq 1$ with $\ell \geq C$ and $\ell \geq c \log n_1$, so that $\mathrm{Leap}(f_n) \leq \ell$ for every n.

Tameness supplies a polynomial q with every nonzero Fourier coefficient of f_n at least $1/q(n)$ in magnitude. The hypotheses of Lemma B.2 hold with $w = \ell$. The query f is presented, tame with parameters (c, q) and FOM-definable; ℓ is a constant; and $\mathrm{Leap}(f_n) \leq \ell$ for every n. So that lemma gives, for every n, an ordering of $\mathrm{supp}(f_n)$ with every jump at most ℓ, whose rank function rk is FOM-definable. Here $\mathrm{rk}(j) = i$ says that S_j occupies position i. Every nonzero Fourier coefficient of f_n has magnitude at least $1/q(n)$, and q is a polynomial. So Lemma B.8(2), applied with $g = f_n$,

width ℓ and this q, gives f_n a width-ℓ derivation with every visibility margin at least $1/q(n)$. Its features are the constant 1 followed by the characters of the nonempty support sets, taken in that order, and its weights are the corresponding Fourier coefficients of f_n.

One demand of Definition 5.1 remains to be checked: the two families must be definable when indexed by the position in the sequence, rather than by the code of Lemma B.1. Positions are numbered from 0, as the rank function rk of Lemma B.2 numbers them. The order built in the proof of that lemma puts $\emptyset$ first, although the statement of the lemma does not record it. That order is by level. The empty set has level 0, since it is reachable and contained in itself. A nonempty support set S has level at least $|S| \geq 1$, since every reachable set containing S has at least $|S|$ elements. So $\emptyset$ precedes every other set and occupies position 0 whenever it lies in the support.

Put $\mathrm{pos}(j) = \mathrm{rk}(j)$ when $\emptyset \in \mathrm{supp}(f_n)$, and $\mathrm{pos}(j) = \mathrm{rk}(j) + 1$ otherwise. This function is FOM-definable, in the sense that the relation "$\mathrm{pos}(j) = i$" is. The relation "$\widehat{f_n}(\emptyset) \neq 0$" is FOM-definable, for the following two reasons. The set $\emptyset$ is S_0, since the code 0 has no binary digit equal to one. A real of the coefficient family of Lemma B.1 is nonzero exactly when one of its digits is one, which is an existential quantification over the digit index of that family's digit relation at the code 0. The relation "$i = r + 1$" on numbers below n^c is FOM-definable from the comparison of c-tuples, since it says $r < i$ and no number lies strictly between them (Section A.5).

So "$\mathrm{pos}(j) = i$" is the disjunction of two definable cases: $\widehat{f_n}(\emptyset) \neq 0$ and $\mathrm{rk}(j) = i$, or $\widehat{f_n}(\emptyset) = 0$ and $\mathrm{rk}(j) = r$ for some r with $i = r + 1$. Either way pos is a bijection from the nonempty support sets onto $\{1, \ldots, P\}$, where P is the number of those sets. If $\emptyset$ lies in the support, rk maps the $P+1$ support sets onto $\{0, \ldots, P\}$ with $\emptyset$ at 0, so it maps the nonempty ones onto $\{1, \ldots, P\}$. If not, rk maps the P nonempty support sets onto $\{0, \ldots, P-1\}$, and adding 1 maps them onto $\{1, \ldots, P\}$. In both cases pos preserves the order, so the set at position $i \geq 1$ is the i-th nonempty set of the ordering, which is the set S_i of the relabeling in the proof of Lemma B.8(2). Every position is at most $P \leq 2^{|R_n|} - 1 \leq n^c - 1$, so a position is a number below n^c and is named by a c-tuple, as Section A.4 prescribes.

Now index the derivation by position. Its first feature is $\psi_0 = 1$, with weight $a_0 = \widehat{f_n}(\emptyset)$. For each nonempty support set S_j, the feature at position $\mathrm{pos}(j)$ is $\psi_{\mathrm{pos}(j)} = \chi_{S_j}$, with weight $a_{\mathrm{pos}(j)} = \widehat{f_n}(S_j)$. So the relation "$\psi_i(x) = 1$" holds in exactly two cases. The first is that $i = 0$, since ψ_0 is the constant 1. The second is that some code j has S_j in the support, $S_j \neq \emptyset$, $\mathrm{pos}(j) = i$ and $\chi_{S_j}(x) = 1$. The weights are handled the same way. The sign relation of the weight family holds of n and i exactly when one of two cases holds. Either $i = 0$ and the sign relation of the coefficient family of Lemma B.1 holds of n and the code 0 of $\emptyset$. Or some code j has S_j in the support, $S_j \neq \emptyset$, $\mathrm{pos}(j) = i$, and the sign relation of the coefficient family holds of n and j. The digit relation of the weight family holds of n, i and m under the same two cases, with the digit relation of the coefficient family at m in place of its sign relation.

Each weight is a Fourier coefficient of the $\{-1, 1\}$-valued f_n, so its absolute value is at most 1, and by Eq. (5) it is an integer multiple of $2^{-|R_n|}$, so its binary expansion terminates. These are the two conditions Section A.4 places on a family of reals. Every relation used here is FOM-definable: "S_j is in the support" because its coefficient is nonzero, "$S_j \neq \emptyset$" because it says $j \neq 0$, "$\mathrm{pos}(j) = i$" as shown above, and the characters and coefficients by Lemma B.1. Each relation displayed is built from these by Boolean connectives and one existential quantification over the c-tuple naming the code j, and FOM is closed under both (Section A.5), so each is FOM-definable.

Since f is itself FOM-definable, $f \in \mathsf{sFOM}_\ell$. The inclusion in the other direction, from sFOM to Learn on these queries, is Theorem 6.1. □

The next proposition is why the accuracy in Definition 3.3 is a parameter rather than a constant.

Proposition B.10 (A constant accuracy would break the characterization)**.** *There is a presented tame* FOM*-definable query f with the following three properties. For every $n \geq 64$ there is an algorithm that makes n correlational queries of tolerance $1/8$ and outputs, for every target in $\mathcal{O}_n(f_n)$ and every legal oracle, a hypothesis of error at most $1/(2n)$. Yet $f \notin$ sFOM. Moreover $f \notin$ Learn, for the following reason. For every polynomial p, every deterministic learner that reaches accuracy $n^{-4}/4$ on $\mathcal{O}_n(f_n)$ with queries each of tolerance at least $1/p(n)$ makes more than $p(n)$ queries for all large n.*

Proof of Proposition B.10. The query. For $n < 64$ let f_n be the constant 1. For $n \geq 64$ put $r = \lceil \log_2 n \rceil$ and $\ell = \lceil \log_2 \log_2 n \rceil$. Let $L = \{2, \ldots, r+1\}$ be the *flag* block, a block of r coordinates. Let $T = \{r+2, \ldots, r+\ell+1\}$ be the next ℓ coordinates. Define

$$f_n(x) = \begin{cases} x_1 \chi_T(x) & \text{if } x_i = 1 \text{ for every } i \in L, \\ x_1 & \text{otherwise.} \end{cases} \tag{33}$$

These coordinates all lie in $[n]$. Indeed $r < \log_2 n + 1$ and $\ell < \log_2 \log_2 n + 1$, since a ceiling exceeds its argument by less than 1, so $1 + r + \ell < \log_2 n + \log_2 \log_2 n + 3$. Write $t = \log_2 n$, so that $t \geq 6$ for $n \geq 64$. At $t = 6$ we have $\log_2 t + 3 = \log_2 6 + 3 < 3 + 3 = t$. The function $t - \log_2 t$ is increasing for $t \geq 2$, since its derivative $1 - 1/(t \ln 2)$ is positive there. So $\log_2 t + 3 \leq t$ for every $t \geq 6$. Hence $1 + r + \ell < 2 \log_2 n$. Finally $2 \log_2 n \leq n$ for $n \geq 64$. Indeed, this holds at $n = 64$, where it reads $12 \leq 64$. The function $n - 2 \log_2 n$ is increasing for $n \geq 3$, since its derivative $1 - 2/(n \ln 2)$ is positive there, so the inequality holds for every $n \geq 64$.

Both r and ℓ are binary lengths. The binary length of an integer $m \geq 1$ is the least b with $m \leq 2^b - 1$, that is, with $m + 1 \leq 2^b$. So the binary length of $n - 1$ is the least b with $n \leq 2^b$, which is $\lceil \log_2 n \rceil = r$. Likewise, since $r \geq 6$, the binary length of $r - 1$ is $\lceil \log_2 r \rceil = \lceil \log_2 \lceil \log_2 n \rceil \rceil$. This is ℓ by the identity $\lceil \log_2 \lceil x \rceil \rceil = \lceil \log_2 x \rceil$ for $x \geq 1$, taken at $x = \log_2 n$. The identity holds because, for every integer $b \geq 0$, the number 2^b is an integer, so $\lceil x \rceil \leq 2^b$ if and only if $x \leq 2^b$, and the least b satisfying the one is the least b satisfying the other. A binary length is read off the BIT predicate. The number b is the binary length of m exactly when digit $b - 1$ of m is one and no digit of m at position b or higher is one. This is a first-order condition. Since $n - 1$ and $r - 1$ are elements of the universe, both r and ℓ are therefore FOM-definable functions of n.

So L and T are definable families, their endpoints being sums of r, ℓ and constants, which is arithmetic on indices. The relevant set of f_n is $[1 + r + \ell]$ for $n \geq 64$, as the spectrum computed below shows, and it is empty for $n < 64$, where f_n is constant. So the relation "i is relevant to f_n" holds exactly when $n \geq 64$ and $i \leq 1 + r + \ell$. This relation is FOM-definable because r and ℓ are, so f is presented. The query is FOM-definable.

For $n \geq 64$, the statement $f_n(x) = 1$ says that either some flag coordinate is -1 and $x_1 = 1$, or every flag coordinate is 1 and $\chi_{\{1\} \cup T}(x) = 1$. The two cases are exhaustive and exclusive, and in the second $f_n(x) = x_1 \chi_T(x) = \chi_{\{1\} \cup T}(x)$ because $1 \notin T$. The first case is the first-order condition $X(1) \wedge \exists i \, (2 \leq i \leq r + 1 \wedge \neg X(i))$, and the condition that every flag coordinate is 1 is $\forall i \, (2 \leq i \leq r + 1 \to X(i))$; both are arithmetic on indices, since r is definable. Put $T'_n = \{1\} \cup T$ for $n \geq 64$ and $T'_n = \emptyset$ for $n < 64$. This is a definable family of sets, so its parity is decided by an FOM-sentence, by Proposition 2.2. The condition $n \geq 64$ is the first-order sentence saying that the universe has 64 distinct elements, and for $n < 64$ the constant 1 is decided by a true sentence. The sentence "$n < 64$, or $n \geq 64$ and one of the two cases above holds" therefore decides f at every length.

The spectrum. Let ν be the indicator that every flag coordinate is 1, so that ν takes the value 1 on those inputs and 0 on all others. Each factor $(1 + x_i)/2$ is 1 when $x_i = 1$ and 0 when $x_i = -1$. So $\nu(x) = \prod_{i \in L} (1 + x_i)/2$. Expanding the product, choosing from each factor either the term 1 or the term x_i, gives its Fourier expansion $\nu = 2^{-r} \sum_{Z \subseteq L} \chi_Z$, the set Z recording the factors from which x_i was chosen. By Eq. (33) this makes $f_n = x_1 \, (1 - \nu + \nu \chi_T)$, since where $\nu = 1$ the right-hand side is $x_1 \chi_T$, and where $\nu = 0$ it is x_1. The sets $\{1\}$, L and T are pairwise disjoint, so $x_1 \chi_Z = \chi_{\{1\} \cup Z}$ and $x_1 \chi_T \chi_Z = \chi_{\{1\} \cup T \cup Z}$ for every $Z \subseteq L$. Multiplying out therefore gives $f_n = \chi_{\{1\}} - 2^{-r} \sum_{Z \subseteq L} \chi_{\{1\} \cup Z} + 2^{-r} \sum_{Z \subseteq L} \chi_{\{1\} \cup T \cup Z}$, and moving the term $Z = \emptyset$ of the middle sum onto $\chi_{\{1\}}$ gives

$$f_n = \left(1 - 2^{-r}\right) \chi_{\{1\}} - 2^{-r} \sum_{\emptyset \neq Z \subseteq L} \chi_{\{1\} \cup Z} + 2^{-r} \sum_{Z \subseteq L} \chi_{\{1\} \cup T \cup Z}, \tag{34}$$

and the 2^{r+1} sets named here are pairwise distinct.

Within the first two terms the map $Z \mapsto \{1\} \cup Z$ is injective because $1 \notin L$, and within the third sum the map $Z \mapsto \{1\} \cup T \cup Z$ is injective because Z is disjoint from $\{1\} \cup T$. Across them, a set of the third sum contains T, which is nonempty because $|T| = \ell \geq 3$ for $n \geq 64$, and no other set does, since the others lie inside $\{1\} \cup L$. By the uniqueness of the expansion in Eq. (1), the coefficients in Eq. (34) are therefore the Fourier coefficients of f_n, and each is nonzero because $1 - 2^{-r} > 0$.

Since $r = \lceil \log_2 n \rceil$, we have $n \le 2^r < 2n$. So the support has $|\mathrm{supp}(f_n)| = 2^{r+1} < 4n \le n^2$ sets. The nonzero coefficients have magnitudes $1 - 2^{-r}$ and 2^{-r}, and $1 - 2^{-r} \ge 2^{-r}$ because $r \ge 1$, so every one of them is at least $2^{-r} > 1/(2n)$.

Every coordinate of $[1 + r + \ell]$ lies in a support set, namely 1 in $\{1\}$, each $i \in L$ in $\{1, i\}$ and each $i \in T$ in $\{1\} \cup T$. No support set contains any other coordinate. So the relevant set of f_n is $[1 + r + \ell]$, and the arity is $1 + r + \ell \le 2 \log_2 n$ by the computation after Eq. (33). For $n < 64$ the function f_n is the constant 1, of arity 0 and with the single coefficient $\widehat{f_n}(\emptyset) = 1 \ge 1/(2n)$. So f is tame with parameters $(2, 2n)$ in the sense of Definition 2.1, that is, with $c = 2$ and $q(n) = 2n$.

The leap. Every support set that meets T contains all of T, by Eq. (34). The sets that meet T are those of the third sum, and there are $2^r \ge 1$ of them. So every ordering of the support has a first set that meets T; no earlier set meets T, and that set contains all of T, so its jump is at least $|T| = \ell$. Every ordering therefore has largest jump at least ℓ, and $\mathrm{Leap}(f_n) \ge \ell$.

For the matching upper bound, order the support in five groups: first $\{1\}$, then the sets $\{1, i\}$ for each $i \in L$, then the remaining sets $\{1\} \cup Z$ with $|Z| \ge 2$, then $\{1\} \cup T$, and finally the sets $\{1\} \cup T \cup Z$ with $Z \ne \emptyset$. These five groups sort the terms of Eq. (34) by $|Z|$ and by whether the set contains T, so together they list the support once each. The set $\{1\}$ introduces the coordinate 1, a jump of 1. Each set $\{1, i\}$ introduces only i, a jump of 1. After the second group every coordinate of $\{1\} \cup L$ has been introduced, so each set of the third group introduces nothing. The set $\{1\} \cup T$ introduces exactly the ℓ coordinates of T. After it every coordinate of $[1 + r + \ell]$ has been introduced, so the last group introduces nothing. The largest jump of this ordering is therefore $\max(1, \ell) = \ell$, since $\ell \ge 3$. So $\mathrm{Leap}(f_n) = \ell = \lceil \log_2 \log_2 n \rceil$, which is unbounded because it is at least $\log_2 \log_2 n$.

Outside sFOM *and outside* Learn. Suppose $f \in$ sFOM, so that $f \in$ sFOM$_w$ for a constant w. We may take $w \ge 1$, since a width-0 derivation meets the bounded-introduction condition of Definition 5.1 at width 1 as well, and the other three conditions do not mention the width. By Definition 5.1 each f_n then has a width-w derivation under the uniform distribution, and f_n is $\{-1, 1\}$-valued. So Lemma B.8(1) applies to $g = f_n$ and gives $\mathrm{Leap}(f_n) \le w$ for every n. That is false because $\mathrm{Leap}(f_n) = \ell$ is unbounded. So $f \notin$ sFOM.

Turn to the query bound. The query f is tame with $c = 2$, so the arity a of f_n satisfies $a \le 2 \log_2 n$. Hence $2^{-2a}/2 \ge n^{-4}/2 > n^{-4}/4$. A learner reaching accuracy $n^{-4}/4$ therefore satisfies the accuracy hypothesis of Proposition B.6. That proposition also requires the target to be nonconstant. The function f_n is nonconstant for $n \ge 64$, because a constant function has no support set other than $\emptyset$, while $\mathrm{supp}(f_n)$ has $2^{r+1} > 1$ sets. Let p be a polynomial. Let β be a constant with $p(n) \le n^\beta$ for all large n, so that each tolerance, being at least $1/p(n)$, is at least $n^{-\beta}$. Eq. (20) of Proposition B.6, with $c = 2$ and $\mathrm{Leap}(f_n) = \ell$, gives a lower bound of $\frac{1}{2} n^{-2\beta-2} \left((n - 2 \log n)/(2 \log n)\right)^\ell$ queries, at every $n \ge 64$ that is at least the threshold n_0 of that proposition for $c = 2$ and at which $p(n) \le n^\beta$. For $n \ge 2^{10}$ the base of the power is at least $n^{1/2}$, because $n - 2 \log n \ge 2n^{1/2} \log n$ there. At $n = 2^{10}$ that inequality reads $1004 \ge 640$, and the difference of its two sides is increasing for $n \ge 2^{10}$, its derivative $1 - 2/(n \ln 2) - (\log n + 2/\ln 2)/n^{1/2}$ being positive there. So the lower bound is at least $\frac{1}{2} n^{-2\beta-2} n^{\ell/2} = \frac{1}{2} n^{\ell/2-2\beta-2}$. Since $\ell = \lceil \log_2 \log_2 n \rceil$ tends to infinity, $\ell \ge 6\beta + 6$ for all large n, and then $\ell/2 - 2\beta - 2 \ge \beta + 1$. The lower bound is then at least $\frac{1}{2} n^{\beta+1}$, which exceeds $n^\beta \ge p(n)$ because $n > 2$. The bound just proved gives $f \notin$ Learn directly.

Suppose $f \in$ Learn, and run the learner of Definition 3.3 at accuracy $\varepsilon = n^{-4}/4$. It is deterministic. Since $1/\varepsilon = 4n^4$ is polynomial in n, there are positive polynomials p_1 and p_2 such that each of its queries has tolerance at least $1/p_1(n)$ and it runs in time at most $p_2(n)$. Hence it makes at most $p_2(n)$ queries. With $p = p_1 + p_2$, each tolerance is at least $1/p(n)$ and the number of queries is at most $p(n)$, which the bound rules out for all large n. The same conclusion is Theorem B.7 applied to f, which is tame with parameters $(2, 2n)$ and has unbounded leap.

The cheap learner. Let $n \ge 64$ and let $g = f_n^\sigma$ be the target. The only singleton in $\mathrm{supp}(f_n)$ is $\{1\}$, because every other set of Eq. (34) contains 1 together with a nonempty Z or with the nonempty set T. Relabeling preserves the sizes of sets, so the only singleton in $\mathrm{supp}(g)$ is $\{\sigma(1)\}$, and its coefficient is $1 - 2^{-r}$. That value comes from Eq. (34) together with the rule that the coefficient of f^σ at $\sigma(S)$ is $\widehat{f}(S)$. Since $n \ge 64$, the binary length r of $n - 1$ is at least 6, so that coefficient is at least $63/64$. Query $\left(\chi_{\{i\}}, 1/8\right)$ for each $i \in [n]$, that is, ask for the correlation of the target with each

single coordinate at tolerance $1/8$. Each $\chi_{\{i\}}$ takes values in $\{-1,1\} \subseteq [-1,1]$, so these are queries in the sense of Definition 3.1, and the answer at i lies within $1/8$ of $\mathbb{E}_x\left[g(x)\chi_{\{i\}}(x)\right] = \widehat{g}(\{i\})$. So the answer at $i = \sigma(1)$ is at least $63/64 - 8/64 = 55/64$, which exceeds $3/8 = 24/64$. At every other i the coefficient $\widehat{g}(\{i\})$ is zero, since $\{\sigma(1)\}$ is the only singleton in $\mathrm{supp}(g)$, so the answer there is at most $1/8 < 3/8$. So $\sigma(1)$ is the unique coordinate whose answer exceeds $3/8$, whatever the legal oracle.

Output $h = \chi_{\{\sigma(1)\}}$. Transporting Eq. (33) by σ, that is, substituting $x_{\sigma(i)}$ for x_i, gives $g(x) = x_{\sigma(1)}\chi_{\sigma(T)}(x)$ if $x_{\sigma(i)} = 1$ for every $i \in L$, and $g(x) = x_{\sigma(1)}$ otherwise. Since $h(x) = x_{\sigma(1)}$ is ± 1, the hypothesis and the target differ at x exactly when $x_{\sigma(i)} = 1$ for every $i \in L$ and $\chi_{\sigma(T)}(x) = -1$. The first condition fixes the r coordinates of $\sigma(L)$, which are distinct because σ is a permutation, so it holds with probability 2^{-r}. The second reads only the coordinates of $\sigma(T)$, which are disjoint from $\sigma(L)$ because T and L are disjoint, so it is independent of the first. It holds with probability $\frac{1}{2}$, because $\sigma(T)$ is nonempty and so $\mathbb{E}_x\left[\chi_{\sigma(T)}(x)\right] = 0$. The disagreement therefore has probability $2^{-r} \cdot \frac{1}{2} = 2^{-r-1}$. Since $2^r \geq n$, because $r = \lceil \log_2 n \rceil$, this is at most $1/(2n)$. □

Proof of Theorem 6.3. The family of sets $([k(n)])_n$ is definable because k is. The condition $i \in [k(n)]$ holds exactly when $i \leq k(n)$. Comparing i with the number $k(n)$ is arithmetic on indices, since the definable digit relation of k supplies the binary digits of $k(n)$. So the query is FOM-definable by Proposition 2.2, which makes the parity χ_T definable for every definable family of sets T. The support of $f_n = \chi_{[k(n)]}$ is the single set $[k(n)]$, so the relevant set of f_n is $[k(n)]$, the same definable family, and f is presented. It has arity $k(n) \leq \log n = 1 \cdot \log n$ and the single nonzero Fourier coefficient $\widehat{f_n}([k(n)]) = 1 \geq 1/1$, so it is tame with parameters $(1,1)$, that is, with $c = 1$ and the constant polynomial $q = 1$. By Example 4.2 its leap is $k(n)$.

Suppose $f \in \mathsf{sFOM}$, so $f \in \mathsf{sFOM}_w$ for a constant w. We may take w to be at least 1, since a width-0 derivation is also a width-1 derivation. By Definition 5.1 each f_n has a width-w derivation under the uniform distribution, and f_n is $\{-1,1\}$-valued, so Lemma B.8(1) applies to $g = f_n$. It gives $\mathrm{Leap}(f_n) = k(n) \leq w$ for every n, contradicting $k(n) = \omega(1)$. Hence $f \notin \mathsf{sFOM}$.

For learnability, $f_n^\sigma = \chi_{\sigma([k(n)])}$ by Section A.2. Every set of size $k(n)$ is $\sigma([k(n)])$ for some permutation σ, so the orbit is $\mathcal{O}_n(f) = \{\chi_S : |S| = k(n)\}$, a family of $\binom{n}{k(n)}$ distinct characters. This is what Lemma B.3 takes as its hypothesis. Suppose $f \in \mathsf{Learn}$, and run the learner of Definition 3.3 at accuracy $\varepsilon = 1/3$. It is deterministic. Since at $\varepsilon = 1/3$ the polynomials of Definition 3.3 depend on n alone, there are constants β and β' such that for all large n each of its queries has tolerance at least $n^{-\beta}$ and it runs in time at most $n^{\beta'}$. It therefore makes $t \leq n^{\beta'}$ queries. Every answer within $\tau = n^{-\beta}$ of the true correlation is legal for each of its queries, so it also succeeds when every tolerance is replaced by τ, which is the setting of Lemma B.3.

Against each target $g \in \mathcal{O}_n(f)$ its hypothesis h is $\{-1,1\}$-valued with $\mathbf{Pr}_x\left[h(x) \neq g(x)\right] \leq 1/3$. So $\mathbb{E}_x\left[h(x)g(x)\right] \geq 1/3$ by Remark 3.4, which is the correlation hypothesis of the lemma with $\gamma = 1/3$. The lemma then gives $t \geq \left(\binom{n}{k(n)} - 9\right) n^{-2\beta}$. Since $k(n) = \omega(1)$, for all large n we have $k(n) \geq 2(\beta' + 2\beta + 1)$. For such $n \geq 16$ we also have $k(n) \leq \log n \leq n^{1/2}$, the last because $\log n = n^{1/2}$ at $n = 16$ and $\log n / n^{1/2}$ decreases for $n \geq 16$. So each factor $(n-j)/(k(n)-j)$ of $\binom{n}{k(n)} = \prod_{j<k(n)}(n-j)/(k(n)-j)$ is at least $n/k(n) \geq n^{1/2}$, and $\binom{n}{k(n)} \geq n^{k(n)/2} \geq 18$. Hence $t \geq \frac{1}{2}n^{k(n)/2}n^{-2\beta} \geq \frac{1}{2}n^{\beta'+1} > n^{\beta'}$, which contradicts $t \leq n^{\beta'}$. Hence $f \notin \mathsf{Learn}$.

Since f is FOM-definable and $f \notin \mathsf{sFOM}$, while $\mathsf{sFOM} \subseteq \mathsf{FOM}$ holds by Definition 5.1, the inclusion is strict.

The example $k(n) = \lceil \log \log n \rceil$ for $n \geq 2$, with $k(1) = 0$, meets the hypotheses. It tends to infinity. It is at most $\log n$, because $\lceil \log t \rceil \leq t$ for every real $t \geq 1$, taken at $t = \log n$. For $1 \leq t \leq 2$ the left side is at most 1. For $2^j < t \leq 2^{j+1}$ with $j \geq 1$ it is $j + 1 \leq 2^j < t$. The example k is FOM-definable, being the binary length of $r - 1$ where r is the binary length of $n - 1$, as in the proof of Proposition B.10. □