hexagon:2611.00133v1

The Price of Bayes-Optimal In-Context Learning

Xiaoyu Li*

Abstract

In-context learning is often explained as implicit Bayesian inference: given labeled examples of a task drawn from a prior, the learner predicts like the posterior. We determine the computational price of this explanation for task families whose labels depend on few bits of the task description, such as quantized models on sparse data, and we compare it with the price of merely learning. The comparison uses the recent refutations of the Exponential Time Hypothesis by OpenAI and Anthropic.

- **Bayesian inference in context is counting.** For κ-local task families with N-bit descriptions under the uniform prior, the posterior predictive and posterior sampling reduce to $\#\kappa$-SAT. Without label noise they run in time $\mathrm{poly}(m) \cdot 2^{o(N)}$ on contexts of every length m when $\kappa = o((\log N)^{1/4}/\sqrt{\log\log N})$, and label noise is handled by error profiles. A two-variable gadget gives contexts that are always consistent and whose posterior predictive is $2\#F/(2\#F+1)$ for a CNF F. So approximating the posterior predictive within $1/4$ needs time $2^{(1-\gamma)N}$ at locality $O(\log N)$ under CNF-SETH, and time $2^{N/\mathrm{polylog}N}$ at locality 3 under a length form of ETH. At constant locality, Bayesian inference in context is SAT parameterized by formula length.
- **Learning in context is cheap.** The Bayes risk of any class with N-bit descriptions is at most $N/(m+1)$ after m examples. For binarized perceptrons and literal disjunctions on sparse data, the very families of the lower bounds, a linear program is a polynomial-time in-context learner with essentially the same sample efficiency. Every κ-local family is linear over $\sum_{j\le\kappa}\binom{N}{j}$ local Fourier features, so regression in context learns it from $N^{O(\kappa)}$ examples, in time polynomial in N^κ and the inverse accuracy.
- **The average case.** For Goldreich's random local functions, the Bayes risk is at most $N/(m+1)$, while every in-context learner yields a distinguisher for Goldreich's generator, with the same running time up to a linear term, whose advantage is $1/2$ minus the risk. Under a local pseudorandom generator hypothesis with polynomial stretch, polynomial-time in-context learners need polynomially more examples than the Bayes predictor. Under its length form at linear stretch, which the new algorithms refute below rate $1/2$, the Bayes predictor of the refutations is optimal up to polylogarithmic factors in the exponent.

For transformers, these bounds limit the length of a chain of thought that computes the posterior predictive, and they separate Bayes-optimal in-context learning from the regression-type in-context learning that transformers are known to implement.

*University of New South Wales, xiaoyulics@gmail.com

Contents

1 Introduction **3**
- 1.1 The model . . . 3
- 1.2 Results . . . 3
- 1.3 Techniques . . . 5
- 1.4 What is new . . . 6
- 1.5 Organization . . . 6

2 Preliminaries **6**
- 2.1 Local classes . . . 7
- 2.2 In-context learning . . . 7
- 2.3 Imported results . . . 8
- 2.4 Hypotheses . . . 9

3 Bayesian inference in context **9**
- 3.1 Algorithms . . . 10
- 3.2 The Bayes risk . . . 10

4 The hardness of Bayesian inference in context **11**
- 4.1 The gadget . . . 11
- 4.2 Lower bounds . . . 12

5 Efficient in-context learning **13**
- 5.1 The hard families are halfspaces . . . 13
- 5.2 Local classes are linear . . . 14

6 The average case: random local functions **15**
- 6.1 Random local functions and Goldreich's generator . . . 15
- 6.2 Three regimes . . . 16

7 Consequences for transformers **17**
- 7.1 Chains of thought . . . 17
- 7.2 Bayesian and algorithmic in-context learning . . . 18

8 Related work **19**

9 Open problems **20**

A Proofs for Sections 2 and 3 **24**
- A.1 The Bayes decision rule . . . 24
- A.2 Algorithms . . . 24
- A.3 The Bayes risk . . . 26

B Proofs for Section 4 **28**
- B.1 The gadgets . . . 28
- B.2 Lower bounds . . . 30

C Proofs for Section 5 **31**
- C.1 Halfspaces in context . . . 31
- C.2 Local Fourier features and regression . . . 32

D Proofs for Sections 6 and 7 **34**

1 Introduction

A transformer that learns in context receives labeled examples of a task and predicts the label of a query, with no update of its weights. A widely used explanation of this ability is Bayesian: pretraining on a distribution of tasks teaches the model to predict like the posterior of that distribution, given the examples in its context [XRLM22, PAG24]. Prior-fitted networks are trained for exactly this purpose [MHA+22]. Other work shows that transformers can implement learning algorithms in context, such as least squares, ridge regression and gradient descent [GTLV22, ASA+23, vONR+23, BCW+23], and experiments find that models trained on diverse task families leave the Bayes estimator of their pretraining prior and follow such algorithms instead [RPCG23].

This paper asks what Bayes-optimality in context costs, and how the cost compares with that of learning in context at all. We answer the question in exact exponents for a natural class of discrete task families. In these families there are 2^N tasks, and the posterior is a sum over all of them. The answer turns on the recent refutations of the Exponential Time Hypothesis (ETH) by OpenAI and Anthropic [Ope26, Cla26]. Both count the satisfying assignments of a k-CNF on n variables in time $2^{n/(\log n)^{1/2-o(1)}}$. The companion paper [Li26] determined what these algorithms mean for proper learning. Here they supply the upper bounds for Bayesian prediction, and the hypotheses that survive them, CNF-SETH and a length form of ETH, supply the lower bounds.

1.1 The model

A class $\mathcal{H} = \{h_w : w \in \{0,1\}^N\}$ is *κ-local* if for each input x the label $h_w(x)$ depends only on at most κ bits of w [Li26]. Quantized models on sparse data are local. A binarized perceptron $x \mapsto \mathbf{1}[\langle \sigma, x\rangle \geq 0]$ with $\sigma \in \{\pm 1\}^N$ is k-local on k-sparse inputs, and so are literal disjunctions, quantized embedding models on bags of k tokens, and Goldreich's random local functions [Gol00].

An *episode* draws a task w uniformly from $\{0,1\}^N$, inputs $x_1, \ldots, x_{m+1}$ independently from a distribution $\mathcal{D}$, and labels $y_i = h_w(x_i)$, each flipped independently with probability $\eta < 1/2$. An *in-context learner* sees the context $S = ((x_1, y_1), \ldots, (x_m, y_m))$ and the query x_{m+1} and predicts y_{m+1}; its *risk* is its error probability. The *Bayes in-context predictor* predicts 1 when the posterior predictive $p_S(x_{m+1}) = \Pr[h_w(x_{m+1}) = 1 \mid S]$ is at least $1/2$. It has the least risk of all in-context learners. We call the computation of $p_S(x)$ for given S and x *Bayesian inference in context*.

1.2 Results

Three theorems make the comparison. Figure 1 summarizes them.

Theorem A (Bayesian inference in context is counting; Theorems 3.1 and 4.4)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions, under the uniform prior.*

(1) Without label noise, the posterior predictive and exact posterior samples can be computed in time $\mathrm{poly}(N, m) \cdot 2^{O(\kappa^2 N \log\log N/\sqrt{\log N})}$ *for contexts of every length m. With a rational noise rate a/b they take time* $\mathrm{poly}(N, m, \log b) \cdot 2^{O(\kappa m/\sqrt{\log m})}$ *for $m \geq \max\{\kappa, 2\}$. Exact Bayesian inference in context is equivalent to #κ-SAT, up to a factor 2 in the locality.*

(2) Under CNF-SETH, for every $\gamma > 0$ there is K such that no algorithm approximates the posterior predictive within $1/4$ in time $O(2^{(1-\gamma)N})$ for binarized perceptrons, or for literal disjunctions, on $(K \log N)$-sparse data. The hard contexts have polynomial size and are consistent.

(3) Under the length form $\mathrm{ETH}^{\mathrm{len}}_C$ of ETH, for literal disjunctions on 3-sparse data no algorithm does so in time $2^{N/(\log N)^{C+\theta}}$ on contexts of $O(N^2)$ examples, for every $\theta > 0$. At

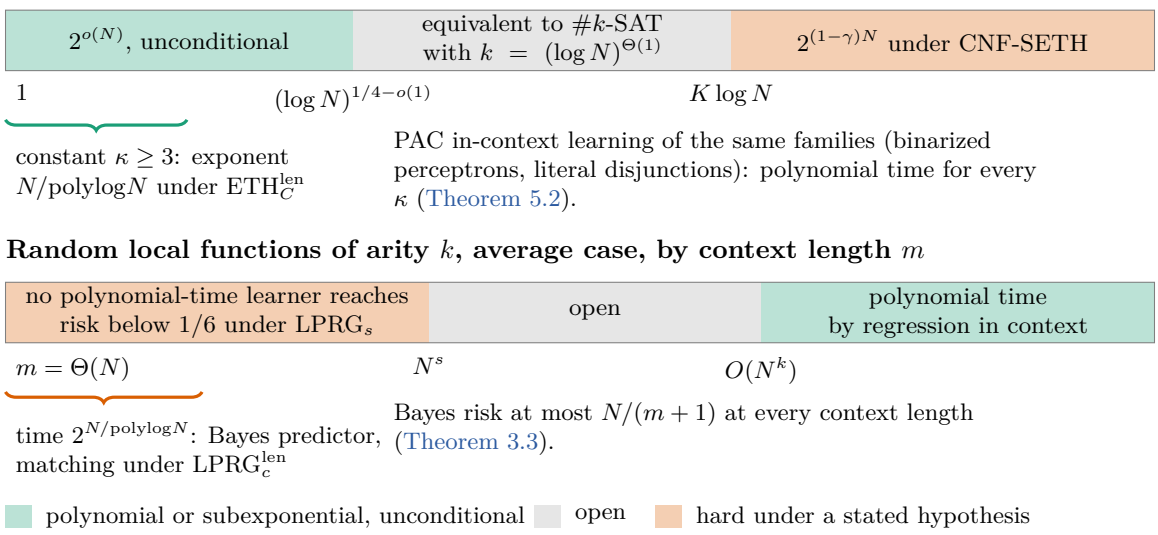


Figure 1: The price of Bayes-optimality. Row 1: exact or approximate (within 1/4) Bayesian inference in context for κ-local families (Theorems 3.1 and 4.4); the middle window is the locality window of [Li26]. Row 2: in-context learning of Goldreich's random local functions (Theorem 6.6 and Corollary 6.7). The regimes are not drawn to scale.

constant locality, Bayesian inference in context is SAT parameterized by formula length, up to polylogarithmic factors in the exponent.

Part (1) is the version-space and error-profile machinery of [Li26] applied to prediction. The lower bounds come from one gadget. For a CNF F, add a variable a to every clause as $\neg a$, and add clauses that set every other variable to 0 when a is false; add one free variable for a true. The satisfying assignments are $2\#F$ assignments with $a = 1$ and one with $a = 0$. As a context, this formula is consistent for every F, and the posterior probability that $a = 1$ is $2\#F/(2\#F+1)$, which is 0 or at least 2/3 according to whether F is satisfiable. So the difficulty is not consistency, which is trivial here, but the posterior.

Theorem B (Learning in context is cheap; Theorems 3.3, 5.2, 5.4 and 5.5)**.** *(1) For every class with N-bit descriptions and every input distribution, the Bayes risk after m examples is at most $N/(m+1)$, and at most $\eta + (1-2\eta)N/((1-\mathrm{h}(\eta))(m+1))$ with label noise η, where* h *is the binary entropy.*

(2) For binarized perceptrons and literal disjunctions on k-sparse data, for every k, a linear program is a polynomial-time in-context learner with error at most ε, with probability at least $1-\delta$, from $O((N\log(1/\varepsilon)+\log(1/\delta))/\varepsilon)$ examples, for every input distribution and every task.

(3) Every κ-local class is linear over $D_\kappa = \sum_{j\leq\kappa}\binom{N}{j}$ local Fourier features. Regression in context learns it, also agnostically, to error ε with probability $1-\delta$, in time $\mathrm{poly}(N, D_\kappa, 1/\varepsilon, \log(1/\delta))$ from $O((D_\kappa + \log(1/\delta))/\varepsilon^2)$ examples.

Parts (2) and Theorem A(2) concern the same families on the same contexts. At locality $K\log N$, the Bayes predictor needs time $2^{(1-\gamma)N}$ under CNF-SETH, while a linear program learns the family in polynomial time with the sample efficiency of the Bayes predictor up to a logarithmic factor. The expensive part of Bayes-optimal in-context learning is the Bayes-optimality, not the learning.

The hard contexts of Theorem A are atypical for the prior: every label in them is positive. The third theorem works in the average case, over the task distribution itself. Goldreich's function with predicate $P\colon \{0,1\}^k \to \{0,1\}$ maps w and random k-tuples $\tau_1, \dots, \tau_m$ of indices to the bits $P(w_{\tau_j})$ [Gol00, App16]. As a task family, its tasks are $w \in \{0,1\}^N$ and its inputs are k-tuples.

Theorem C (Random local functions; Theorem 6.6 and Corollary 6.7)**.** *For the random local functions of a predicate of arity k, under the uniform distribution on k-tuples:*

(1) the Bayes risk is at most $N/(m+1)$, and the Bayes predictor runs in time $\mathrm{poly}(m) \cdot 2^{O(N \log\log N/\sqrt{\log N})}$;

(2) regression in context has risk at most ε from $O(N^k \log(1/\varepsilon)/\varepsilon)$ examples, in polynomial time;

(3) every in-context learner A yields a distinguisher for Goldreich's generator with $m+1$ outputs that runs in the time of A plus $O(mk \log N)$ and has advantage exactly $|1/2 - R_m(A)|$.

So under a local pseudorandom generator hypothesis LPRG_s with stretch N^s, no polynomial-time in-context learner has risk below $1/6$, for large N, at a context length $m(N) \le \lceil N^s \rceil - 1$ computable in polynomial time, although at $m = \lceil N^s \rceil - 1$ the Bayes risk is at most N^{1-s}. Under its length form $\mathrm{LPRG}_c^{\mathrm{len}}$ at linear stretch, an in-context learner needs time more than $2^{N/(\log N)^c - 1}$ to reach risk below $1/6$ from $\lceil \alpha N \rceil - 1$ examples, for a large enough constant α, and the Bayes predictor of (1) reaches risk $1/\alpha$ in time $2^{N/(\log N)^{1/2-o(1)}}$. The hypothesis $\mathrm{LPRG}_c^{\mathrm{len}}$ is false for $c < 1/2$, by the new algorithms, and open for $c \ge 1/2$.

Consequences for transformers. A polynomial-size transformer whose weights are computed in polynomial time, for example by a polynomial-time training procedure, and that writes a chain of thought of length L can be simulated in time polynomial in L and the context. The lower bounds above therefore bound L from below. Under CNF-SETH, such a transformer that outputs the Bayes prediction for every context of binarized perceptrons on $(K \log N)$-sparse data needs $L = 2^{\Omega(N)}$ for infinitely many N, and under $\mathrm{ETH}_C^{\mathrm{len}}$ already locality 3 needs $L = 2^{N/\mathrm{polylog}N}$ for infinitely many N. Under LPRG_s, no such transformer with a polynomial chain of thought reaches risk below $1/6$ on random local functions, for large N, at a context length $m(N) \le \lceil N^s \rceil - 1$ computable in polynomial time (Corollary 7.1). Regression in context, the efficient alternative of Theorem B, is the kind of algorithm that transformers are known to implement [ASA+23, vONR+23, BCW+23]. For discrete task families with local structure, these results separate the Bayesian explanation of in-context learning from the algorithmic one, and they are consistent with the departure from the Bayes estimator that [RPCG23] observe on diverse task families (Section 7).

1.3 Techniques

Prediction as a ratio of counts. Without noise, the posterior is uniform on the version space, and $p_S(x)$ is the number of consistent descriptions that label x positive divided by the number of all consistent descriptions. For local classes both are counts of κ-CNFs [Li26]. With noise, two error profiles, of the context and of the context with $(x, 1)$ added, determine the profile of the descriptions that label x positive, and the posterior predictive is a ratio of two polynomials evaluated at $\eta/(1-\eta)$.

The gadget. The two-variable construction above keeps the context consistent and moves the whole difficulty into the posterior. It needs only one more variable than the formula, and one more for the factor 2, so the width reduction of [Sch05, CIP06] in the form of [Li26] turns it into a $2^{(1-\gamma)N}$ lower bound. For the length form, fresh variables split the clauses, and clauses that force them to 0 when a is false keep the branch $a = 0$ a single assignment.

The Bayes risk. Each mistake of the Bayes predictor at least halves the posterior mass, so there are at most N mistakes in any episode; this is the halving argument [BF72, Lit88] with posterior weights. The Bayes risk does not increase with the context length, because the posterior predictive of a fixed query is a martingale and the error of the Bayes decision is a concave function of it. Together the two facts give $N/(m+1)$. With noise, the cumulative log-loss regret of the Bayes mixture, which is at most N, bounds the mistakes through a Kullback–Leibler divergence.

Local Fourier features. For each input x, the label $h_w(x)$ is a function of the at most κ bits of w in $R(x)$, so its Fourier expansion has at most 2^κ terms, all of degree at most κ. Hence $h_w(x) = \langle \Phi(w), \Psi(x) \rangle$ with $\Phi(w) \in \{\pm 1\}^{D_\kappa}$, and the class is a linear class over the features Ψ. Linear algebra learns it in the realizable case, and ℓ_1 regression [KKMS08] in the agnostic case. For binarized perceptrons and literal disjunctions a halfspace in the original coordinates suffices.

Next-bit prediction. An in-context learner for random local functions predicts the last output of Goldreich's generator from the earlier outputs. By Yao's argument [Yao82], its advantage over $1/2$ is the advantage of a distinguisher.

1.4 What is new

The realizable Bayes risk bound is classical [HKS94], the linearization is the Fourier expansion of a local function, and next-bit prediction is Yao's. The posterior algorithms use only the statements of [Li26], which in turn use the refutations [Ope26, Cla26] as black boxes. New are the gadget and the fine-grained lower bounds for Bayesian inference in context, the comparison with polynomial-time learning of the same families on the same contexts, the noisy Bayes risk bound in this form, the sample bound for regression on local Fourier features, the three regimes for random local functions with the hypothesis $\mathrm{LPRG}_c^{\mathrm{len}}$ and its refutation below rate $1/2$, and the consequences for transformers.

1.5 Organization

Section 2 fixes the model and imports the results of [Li26]. Section 3 gives the algorithms for Bayesian inference in context and the bound on the Bayes risk. Section 4 proves the lower bounds. Section 5 gives the efficient in-context learners. Section 6 treats random local functions, and Section 7 draws the consequences for transformers. Section 8 discusses related work, and Section 9 lists open problems. The appendix contains all proofs in structured form.

AI disclosure. This paper was written with Claude Opus 5.5 (Anthropic), working as an agent in Claude Code. The author asked Claude to propose questions that build on [Li26]. Claude chose one of them, proved the results and wrote the text. The gadgets, the linearization and the risk bounds were checked by brute force on small instances, and separate Claude agents checked the citations and audited the proofs. The author is responsible for the content.

2 Preliminaries

This section fixes the model of in-context learning, imports the results of [Li26] that we use, and states the hypotheses that survive the refutation of ETH.

2.1 Local classes

We write log for $\log_2$, $[n]$ for $\{1, \dots, n\}$, and $w|_R$ for the restriction of $w \in \{0,1\}^N$ to $R \subseteq [N]$. A *clause* is a nonempty set of literals, its *width* is its size, a *CNF* is a finite set of clauses, and a *k-CNF* has clauses of width at most k. The *length* of a CNF is its number of literal occurrences, and $\#F$ is the number of satisfying assignments of F on its declared variables. Time is measured in the word-RAM model.

Definition 2.1 (Local class [Li26, Definition 2.2])**.** Let $N, \kappa \geq 1$. A *κ-local class* $\mathcal{H}$ on a domain $\mathcal{X}$ assigns to every $x \in \mathcal{X}$ a set $R(x) \subseteq [N]$ with $|R(x)| \leq \kappa$ and a predicate $f_x \colon \{0,1\}^{R(x)} \to \{0,1\}$, both computable in time polynomial in N and the length of x. A *description* $w \in \{0,1\}^N$ defines $h_w(x) = f_x(w|_{R(x)})$, and $\mathcal{H} = \{h_w : w \in \{0,1\}^N\}$.

Example 2.2 ([Li26, Example 2.3])**.** Let $\mathcal{X}_k = \{x \in \{-1,0,1\}^n : |\mathrm{supp}(x)| \leq k\}$, and describe a sign vector $\sigma \in \{\pm 1\}^n$ by the bits $w_i = (1+\sigma_i)/2$, so that $N = n$.

(a) *Literal disjunctions* LD_k on $\mathcal{X}_k$: $h_\sigma(x) = 1$ if and only if $x_i = \sigma_i$ for some $i \in \mathrm{supp}(x)$. A positive example x is the clause $\bigvee_{i \in \mathrm{supp}(x)} (\sigma_i = x_i)$, so a sample of positive examples is a k-CNF.

(b) *Binarized perceptrons* BP_k on $\mathcal{X}_k$: $h_\sigma(x) = \mathbf{1}[\langle \sigma, x \rangle \geq 0]$.

Both are k-local with $R(x) = \mathrm{supp}(x)$. Random local functions are introduced in Section 6.

A *sample* or *context* is a sequence $S = ((x_1, y_1), \dots, (x_m, y_m))$ of labeled inputs. Its *version space* is $\mathrm{VS}(S) = \{w : h_w(x_i) = y_i \text{ for all } i\}$, the *error count* of w is $e_S(w) = |\{i : h_w(x_i) \neq y_i\}|$, and the *error profile* is $E_S(V) = \sum_{w \in \{0,1\}^N} V^{e_S(w)}$, a polynomial of degree at most m whose coefficient of V^j counts the descriptions with exactly j errors.

2.2 In-context learning

Definition 2.3 (Episodes, learners and risk)**.** A *task family* is a class $\mathcal{H} = \{h_w : w \in \{0,1\}^N\}$ on $\mathcal{X}$ together with a distribution $\mathcal{D}$ on $\mathcal{X}$; the *noise rate* is some $\eta \in [0, 1/2)$. An *episode* of length m draws a task w uniformly from $\{0,1\}^N$, inputs $x_1, \dots, x_{m+1}$ independently from $\mathcal{D}$, and independent bits $\xi_1, \dots, \xi_{m+1}$ with $\Pr[\xi_i = 1] = \eta$, and sets $y_i = h_w(x_i) \oplus \xi_i$. Its *context* is $S = ((x_i, y_i))_{i \leq m}$, its *query* is x_{m+1} and its *target* is y_{m+1}. An *in-context learner* is an algorithm A that maps a context and a query to a label, possibly using random bits. Its *risk* at context length m is $R_m(A) = \Pr[A(S, x_{m+1}) \neq y_{m+1}]$, the probability over the episode and the random bits of A.

The prior is uniform on descriptions. For quantized models this is the uniform prior on parameters; the induced prior on functions need not be uniform. Let $\mu = \eta/(1-\eta)$, with $0^0 = 1$.

Definition 2.4 (Posterior, Bayes and Gibbs predictors)**.** For a context S, let $\omega_S(w) = \mu^{e_S(w)}$ and $Z_S = \sum_w \omega_S(w)$. If $Z_S > 0$, the *posterior* is $\pi_S(w) = \omega_S(w)/Z_S$, and the *posterior predictive* of a query x is $p_S(x) = \sum_{w : h_w(x) = 1} \pi_S(w)$. The *Bayes in-context predictor* is $\mathsf{B}(S, x) = \mathbf{1}[p_S(x) \geq 1/2]$. The *Gibbs in-context predictor* $\mathsf{G}(S, x)$ draws w from π_S and outputs $h_w(x)$. *Bayesian inference in context* is the problem of computing $p_S(x)$ from S and x, and *β-approximate* Bayesian inference in context asks for a number within β of $p_S(x)$.

For $\eta = 0$ the weight ω_S is the indicator of $\mathrm{VS}(S)$, so $Z_S > 0$ exactly when the context is consistent, and the posterior is uniform on the version space. Contexts that occur in episodes have $Z_S > 0$. In an episode, π_S is the conditional distribution of the task given the context, by Bayes' rule, and the conditional probability that the target is 1 is

$$\hat{p}_S(x) = (1-\eta)\, p_S(x) + \eta\, (1 - p_S(x)). \tag{1}$$

Since $\eta < 1/2$, the predictor B outputs the more likely target. Hence it minimizes the risk.

Lemma 2.5 (Bayes decision rule)**.** *For every task family, noise rate and context length m, every in-context learner A has $R_m(A) \geq R_m(\mathsf{B})$, and $R_m(\mathsf{B}) = \mathbb{E}[\min\{\hat{p}_S(x_{m+1}), 1 - \hat{p}_S(x_{m+1})\}]$.*

We write $R^*_m = R_m(\mathsf{B})$ for the *Bayes risk*. We also use the worst-case guarantee of learning theory.

Definition 2.6 (PAC in-context learners)**.** An in-context learner A for $\mathcal{H}$ is (ε, δ)*-PAC from m examples* if for every distribution $\mathcal{D}$ on $\mathcal{X}$ and every w, with probability at least $1 - \delta$ over a context of m independent examples $(x, h_w(x))$ with $x \sim \mathcal{D}$ and over the random bits of A, it satisfies $\Pr_{x\sim\mathcal{D}}[A(S, x) \neq h_w(x)] \leq \varepsilon$. It is (ε, δ)*-agnostic from m examples* if for every distribution $\mathcal{D}'$ on $\mathcal{X} \times \{0, 1\}$, with probability at least $1 - \delta$ over a context of m independent examples from $\mathcal{D}'$, it satisfies $\Pr_{(x,y)\sim\mathcal{D}'}[A(S, x) \neq y] \leq \mathrm{OPT} + \varepsilon$, where $\mathrm{OPT} = \min_w \Pr_{(x,y)\sim\mathcal{D}'}[h_w(x) \neq y]$.

An (ε, δ)-PAC learner has risk at most $\varepsilon+\delta$ in every noiseless episode, for every input distribution, whatever the prior. An (ε, δ)-agnostic learner has risk at most $\eta + \varepsilon + \delta$ in every episode with noise rate η, because for each fixed task the error of h_w itself is η.

2.3 Imported results

We use the refutations of ETH [Ope26, Cla26] only through the following statements of [Li26]. Let $T_\kappa(N) = \min\{2^N, 2^{C_1\kappa^2 g(N)}\}$, where $g(N) = N \log\log N/\sqrt{\log N}$ for $N \geq 16$, $g(N) = 1$ for smaller N, and C_1 is the absolute constant of the counting algorithm of [Cla26, Main Theorem and Corollary 11].

Imported Theorem 2.7 (Version spaces are κ-CNFs [Li26, Lemma 3.1])**.** *Given a sample S of size m for a κ-local class with N-bit descriptions, one can compute in time $\mathrm{poly}(N, m) \cdot 2^\kappa$ a κ-CNF Γ_S on the variables $w_1, \ldots, w_N$ with at most $m2^\kappa$ clauses such that $\mathrm{VS}(S)$ is the set of satisfying assignments of Γ_S.*

Imported Theorem 2.8 (Counting and sampling version spaces [Li26, Theorem 3.2])**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions, and let S be a sample of size m. Then $|\mathrm{VS}(S)|$ can be computed deterministically in time $\mathrm{poly}(N, m) \cdot 2^\kappa \cdot T_\kappa(N)$. If $\mathrm{VS}(S) \neq \emptyset$, a uniformly random element of $\mathrm{VS}(S)$ can be generated exactly in the same time, given uniform random integers in $\{0, \ldots, M - 1\}$ at cost $\mathrm{poly}(\log M)$.*

Imported Theorem 2.9 (Error profiles [Li26, Lemma 2.8])**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions, let S be a sample of size m, and let $s = \max\{4\kappa m, m + 1, \kappa^2\}$. The error profile E_S can be computed exactly and deterministically in time $\mathrm{poly}(N, m) \cdot 2^{O(s/\sqrt{\log s})}$.*

Imported Theorem 2.10 (Exponential mechanism [Li26, Theorem 4.6])**.** *In the setting of Imported Theorem 2.9, let $\lambda > 0$. One can sample w with probability proportional to $e^{-\lambda e_S(w)}$ in time $\mathrm{poly}(N, m) \cdot 2^{O(s/\sqrt{\log s})}$, given exact samples from explicitly given distributions on $\{0, \ldots, m\}$. Without exact sampling, the output distribution is within statistical distance ν of the target for any $\nu \geq 2^{-\mathrm{poly}(N,m)}$, in the same time.*

Imported Theorem 2.10 is [Li26, Theorem 4.6] with privacy parameter 2λ, since the score there is m times the empirical error, that is $e_S(w)$. Its algorithm samples an error count j with probability proportional to $c_j e^{-\lambda j}$, where c_j is the coefficient of V^j in E_S, and then a uniform description with exactly j errors.

Imported Theorem 2.11 (Perceptron gadget [Li26, Lemma 3.3]). *Let $k \geq 1$ and $p = \max\{1, k-2\}$. Given a k-CNF Γ on $z_1, \ldots, z_n$ with M clauses, none of them tautological, one can compute in polynomial time a sample S_Γ of $M+p$ examples in $\{-1,0,1\}^{n+p}$, all labeled 1 and each with support of size at most $\max\{2k-2, 2\}$, such that a sign vector σ is consistent with S_Γ as a binarized perceptron if and only if $\sigma_{n+1} = \cdots = \sigma_{n+p} = 1$ and the assignment $z_i = \mathbf{1}[\sigma_i = 1]$ satisfies Γ.*

Imported Theorem 2.12 (Width reduction [Li26, Corollary 3.6]). *Under Hypothesis 2.14, for every $\gamma \in (0,1)$ there are K', c' such that k-SAT with $k \leq K' \log n$ cannot be solved in time $O(2^{(1-\gamma)n})$ by a randomized algorithm with error probability at most $1/3$, even on instances with at most $n^{c'}$ clauses. Under Hypothesis 2.15, there are $K', c', \rho > 0$ such that k-SAT with $k \leq K' \log n$ and at most $n^{c'}$ clauses cannot be solved in time $O(2^{\rho n})$.*

We also use the standard sample sizes for classes of VC dimension d, finite or infinite [BEHW89, SSBD14]. There are absolute constants $C_0, C_B \geq 1$ such that, with probability at least $1-\delta$ over m independent examples, the following hold.

Imported Theorem 2.13 (Sample sizes). *Let $\mathcal{G}$ be a class of VC dimension $d \geq 1$.*

(a) If $m \geq m_{\mathrm{VC}}(d, \varepsilon, \delta) := C_B(d \log(2/\varepsilon) + \log(2/\delta))/\varepsilon$ and the examples are labeled by some $g^* \in \mathcal{G}$, every $g \in \mathcal{G}$ consistent with them has error at most ε.

(b) If $m \geq m_{\mathrm{UC}}(d, \varepsilon, \delta) := C_0(d + \ln(1/\delta))/\varepsilon^2$, every $g \in \mathcal{G}$ has empirical error within $\varepsilon/2$ of its error.

2.4 Hypotheses

The new algorithms need bounded clause width or linear total size, and the following hypotheses concern the opposite regime [Li26]. All are stated for randomized algorithms with error probability at most $1/3$.

Hypothesis 2.14 (CNF-SETH). For every $\gamma > 0$ there is $c > 0$ such that satisfiability of CNF formulas with n variables and at most n^c clauses, of arbitrary width, cannot be decided in time $O(2^{(1-\gamma)n})$.

Hypothesis 2.15 (CNF-ETH). There are $c, \rho > 0$ such that satisfiability of CNF formulas with n variables and at most n^c clauses cannot be decided in time $O(2^{\rho n})$.

Hypothesis 2.16 (Length form of ETH at rate C [Li26, Hypothesis 2.14]). For $C > 0$, the hypothesis $\mathrm{ETH}^{\mathrm{len}}_C$ states that no randomized algorithm decides satisfiability of every CNF formula of length L in time $2^{L/(\log L)^C}$, for all sufficiently large L.

The new algorithms decide SAT of length L in time $2^{O(L/\sqrt{\log L})}$, so $\mathrm{ETH}^{\mathrm{len}}_C$ is false for $C < 1/2$ and open for $C \geq 1/2$ [Li26]. It is the instance with an explicit rate of the polylog-slack hypotheses of [CRTY23]. CNF-SETH and CNF-ETH are not contradicted by the new algorithms, and by width reduction their hard instances may be taken to have width $O(\log n)$ (Imported Theorem 2.12). The local pseudorandom generator hypotheses of Section 6 complete the list.

3 Bayesian inference in context

This section shows that the Bayes and Gibbs in-context predictors of a local class are counting problems, and that they are statistically efficient for every class with short descriptions.

3.1 Algorithms

Without noise, the posterior is uniform on the version space, and the posterior predictive is a ratio of two version-space sizes. With noise, it is a ratio of two weighted sums, and two error profiles determine both.

Theorem 3.1 (Bayesian inference in context for local classes)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions, let S be a context of size m, and let x be a query.*

(a) Without noise. *If $\mathrm{VS}(S) \neq \emptyset$, then $p_S(x)$ and $\mathsf{B}(S, x)$ can be computed exactly and deterministically in time $\mathrm{poly}(N, m) \cdot 2^{\kappa} \cdot T_\kappa(N)$, and the Gibbs predictor can be run exactly in the same time, given uniform random integers. This is $\mathrm{poly}(m) \cdot 2^{o(N)}$ when $\kappa = o((\log N)^{1/4}/\sqrt{\log\log N})$.*

(b) With noise. *Let $\eta = a/b$ for integers $0 < a$, $2a < b$, and let $s = \max\{4\kappa(m+1), m+2, \kappa^2\}$. Then $p_S(x)$ and $\mathsf{B}(S, x)$ can be computed exactly in time $\mathrm{poly}(N, m, \log b) \cdot 2^{O(s/\sqrt{\log s})}$, and the Gibbs predictor can be run in the same time, exactly given exact samples from explicitly given distributions on $\{0, \ldots, m\}$, and otherwise within statistical distance ν for any $\nu \geq 2^{-\mathrm{poly}(N,m)}$. For $m \geq \max\{\kappa, 2\}$ the time is $\mathrm{poly}(N, m, \log b) \cdot 2^{O(\kappa m/\sqrt{\log m})}$.*

Proof sketch. For (a), append the example $(x, 1)$ to S to obtain S^+. Then $p_S(x) = |\mathrm{VS}(S^+)|/|\mathrm{VS}(S)|$, and Imported Theorem 2.8 computes both counts; Gibbs sampling is uniform sampling from $\mathrm{VS}(S)$. For (b), write a_j and b_j for the numbers of descriptions with $e_S(w) = j$ and $h_w(x) = 1$, respectively $h_w(x) = 0$. The coefficient of V^j is $a_j + b_j$ in E_S and $a_j + b_{j-1}$ in E_{S^+}, so the two profiles of Imported Theorem 2.9 determine all a_j and b_j recursively, starting from $b_{-1} = 0$. Then $p_S(x) = \sum_j a_j \mu^j / \sum_j (a_j + b_j)\mu^j$. Gibbs sampling is Imported Theorem 2.10 with $e^{-\lambda} = \mu$ (Section A). □

The same reduction runs in the other direction: Section 4 turns #k-SAT into exact Bayesian inference in context. For the upper bound, any algorithm for #κ-SAT can replace Imported Theorem 2.8.

Corollary 3.2 (Bayesian inference from counting)**.** *If #κ-SAT on N variables with M clauses can be solved in time $T(N, M)$, with T nondecreasing, then noiseless Bayesian inference in context for every κ-local class with N-bit descriptions takes time $2T(N, (m+1)2^\kappa) + \mathrm{poly}(N, m) \cdot 2^\kappa$ on consistent contexts of size m.*

Proof sketch. By Imported Theorem 2.7, $\mathrm{VS}(S)$ and $\mathrm{VS}(S^+)$ are the solution sets of κ-CNFs with at most $(m+1)2^\kappa$ clauses (Section A). □

3.2 The Bayes risk

The Bayes in-context predictor learns every class with short descriptions from few examples. The bound needs no locality.

Theorem 3.3 (Bayes risk)**.** *Let $\mathcal{H} = \{h_w : w \in \{0,1\}^N\}$ be any class, $\mathcal{D}$ any input distribution and $\eta \in [0, 1/2)$ the noise rate. Let $\mathrm{h}(\eta) = -\eta \log \eta - (1-\eta)\log(1-\eta)$, with $\mathrm{h}(0) = 0$.*

(a) $R^*_{m+1} \leq R^*_m$ *for every* $m \geq 0$.

(b) $R^*_m \leq \eta + \dfrac{(1-2\eta)N}{(1-\mathrm{h}(\eta))(m+1)}$ *for every* $m \geq 0$. *In particular, without noise* $R^*_m \leq N/(m+1)$.

(c) Without noise, the Gibbs predictor has risk at most $2R^*_m \leq 2N/(m+1)$.

Proof sketch. For (a), fix the query. Its posterior predictive $\hat{p}$ of (1) is a martingale as examples are added to the context, and the Bayes error $\min\{\hat{p}, 1-\hat{p}\}$ is a concave function of $\hat{p}$; Jensen's inequality gives the claim. For (b) without noise, run the Bayes predictor on the examples of an episode in order. Each mistake leaves at most half of the version space, which starts with 2^N elements and always contains the task, so every episode has at most N mistakes among its first $m+1$ rounds. The expected number of these mistakes is $\sum_{t=0}^{m} R_t^* \geq (m+1)R_m^*$ by (a). With noise, the log-loss regret of the Bayes mixture against the task is at most N bits, so the expected Kullback–Leibler divergences between the true label distribution and the predictive sum to at most N. A mistake of B against $h_w(x)$ costs at least $1 - \mathrm{h}(\eta)$ bits of divergence and adds $1-2\eta$ to the error. For (c), the Gibbs error at a query is $2p(1-p) \leq 2\min\{p, 1-p\}$ (Section A). □

Without noise, the halving argument behind (b) is that of [BF72, Lit88], with posterior weights, and bounds of this kind on the Bayes and Gibbs learning curves go back to Haussler, Kearns and Schapire [HKS94]. So the Bayes in-context predictor has risk ε from N/ε examples, for every class with N-bit descriptions and every input distribution, and Theorem 3.1 computes it in time $2^{o(N)}$ for local classes of small locality. The next section shows that, up to the rate of SAT, this computation cannot be avoided.

4 The hardness of Bayesian inference in context

The lower bounds rest on one gadget. It turns a CNF into a context that is consistent for every formula, together with a query whose posterior predictive reveals whether the formula is satisfiable.

4.1 The gadget

Lemma 4.1 (Bayes gadget)**.** *Let $k \geq 1$, and let F be a k-CNF on $z_1, \ldots, z_n$ with M clauses, none of them tautological. Let Γ_F be the CNF on $z_1, \ldots, z_n, a, u$ with the clauses*

$$\neg a \vee C \quad (C \in F), \qquad a \vee \neg z_i \quad (i \in [n]), \qquad a \vee \neg u.$$

Then Γ_F is a $(k+1)$-CNF with $M+n+1$ clauses, none of them tautological. It has $2\#F + 1$ satisfying assignments, and exactly $2\#F$ of them set $a = 1$.

Proof sketch. If $a = 1$, every clause $a \vee \cdot$ holds, the clauses $\neg a \vee C$ reduce to F, and u is free: $2\#F$ assignments. If $a = 0$, the clauses $\neg a \vee C$ hold and the others force $z = 0$ and $u = 0$: one assignment (Section B). □

A positive example of literal disjunctions is a clause (Example 2.2(a)), and Imported Theorem 2.11 turns a CNF into a sample of binarized perceptrons with the same solutions. The query e_a, the unit vector of the coordinate of a, is labeled 1 exactly by the descriptions with $a = 1$, in both classes.

Lemma 4.2 (Bayes contexts)**.** *Let $k \geq 2$, and let F be a k-CNF on n variables with M clauses, none of them tautological. One can compute in polynomial time:*

(a) a context S_F^{LD} of $M+n+1$ positive examples of LD_{k+1} in dimension $n+2$, and
(b) a context S_F^{BP} of $M+n+k$ positive examples of BP_{2k} in dimension $n+k+1$,

both consistent, such that the query e_a has posterior predictive $p_S(e_a) = 2\#F/(2\#F+1)$ in both. In particular $p_S(e_a) = 0$ if F is unsatisfiable, $p_S(e_a) \geq 2/3$ otherwise, and $\#F = p_S(e_a)/(2(1-p_S(e_a)))$.

Proof sketch. For (a), take the positive examples of the clauses of Γ_F. For (b), apply Imported Theorem 2.11 to Γ_F, a $(k+1)$-CNF, with $p = k-1$ padding coordinates. In both, the version space is in bijection with the satisfying assignments of Γ_F, the posterior is uniform on it, and the query is labeled by a (Section B). □

For the length form of ETH we need locality 3 and dimension linear in the length of the formula. Splitting clauses changes the number of solutions, but not whether there is one, and that suffices.

Lemma 4.3 (Bayes contexts by length)**.** *Given a CNF F of length $L \geq 1$, one can compute in polynomial time a consistent context of positive examples of LD_3 in dimension $N \leq 3L+2$, with at most $\min\{5L+1, 36N^2\}$ examples, such that the query e_a has posterior predictive 0 if F is unsatisfiable and at least $2/3$ otherwise. The same holds for BP_4 in dimension $N+1$, with one more example.*

Proof sketch. Split F into an equisatisfiable 3-CNF F' with at most $2L$ variables and at most L clauses. Replace each clause C of F' by $\neg a \vee C$ if $|C| \leq 2$. If $C = \ell_1 \vee \ell_2 \vee \ell_3$, use a fresh variable y_C and the clauses $\neg a \vee \ell_1 \vee y_C$, $\neg y_C \vee \ell_2 \vee \ell_3$ and $a \vee \neg y_C$. Add $a \vee \neg z$ for every variable z of F' and $a \vee \neg u$. If $a = 0$, all variables are forced to 0, which satisfies every clause; if $a = 1$, the clauses for C have a solution in y_C exactly when C holds, and u is free. So the branch $a = 1$ has no assignment or at least two, and the branch $a = 0$ has one (Section B). □

4.2 Lower bounds

Theorem 4.4 (Hardness of Bayesian inference in context)**.** *(a) Assume Hypothesis 2.14. For every $\gamma \in (0,1)$ there are K and c_γ such that no randomized algorithm with error probability at most $1/3$ does the following in time $O(2^{(1-\gamma)N})$: given a consistent context of at most N^{c_γ} positive examples of $\mathsf{BP}_{\lceil K \log N \rceil}$ in dimension N and a query, output a number within $1/4$ of the posterior predictive. The same holds for $\mathsf{LD}_{\lceil K \log N \rceil}$, for computing $\mathsf{B}(S,x)$, and for running the Gibbs predictor exactly or within statistical distance $1/6$. Under Hypothesis 2.15, there are $K, c_\gamma, \rho > 0$ with the same statements for time $O(2^{\rho N})$.*

(b) Assume $\mathrm{ETH}^{\mathrm{len}}_C$, and let $\theta > 0$. No randomized algorithm with error probability at most $1/3$ outputs, for every consistent context of at most $36N^2$ positive examples of LD_3 in dimension N and every query, a number within $1/4$ of the posterior predictive in time $2^{N/(\log N)^{C+\theta}}$. The same holds for BP_4, for computing $\mathsf{B}(S,x)$, and for running the Gibbs predictor exactly or within statistical distance $1/6$.

Proof sketch. For (a), take a k-CNF with $k = \lfloor K' \log n \rfloor$ and at most $n^{c'}$ clauses from Imported Theorem 2.12 for $\gamma/2$, and build S_F^{BP}. Its dimension is $N \leq n + K' \log n + 1$ and its sparsity is $2k \leq K \log N$. A number within $1/4$ of $p_S(e_a)$ is at most $1/4$ if F is unsatisfiable and at least $5/12$ otherwise, and $\mathsf{B}(S, e_a) = 1$ exactly when F is satisfiable. An algorithm in time $O(2^{(1-\gamma)N})$ would decide the formula in time $O(2^{(1-\gamma/2)n})$. For (b), use Lemma 4.3: a time bound $2^{N/(\log N)^{C+\theta}}$ at $N \leq 3L+2$ is at most $2^{L/(\log L)^C}$ for large L (Section B). □

Together with Theorem 3.1, the theorem fixes the exponent of Bayesian inference in context at constant locality, up to polylogarithmic factors.

Corollary 4.5 (Bayesian inference in context is SAT by length)**.** *Let $\kappa \geq 3$ be a constant. Without noise, Bayesian inference in context for κ-local classes takes time $\mathrm{poly}(m) \cdot 2^{O(N \log\log N/\sqrt{\log N})}$ for every context length m. Under $\mathrm{ETH}^{\mathrm{len}}_C$, for every $\theta > 0$, approximating it within $1/4$ needs time $2^{N/(\log N)^{C+\theta}}$ already for LD_3 on contexts of at most $36N^2$ examples.*

Proof. The upper bound is Theorem 3.1(a) with κ constant, and the lower bound is Theorem 4.4(b), since LD_3 is κ-local for every $\kappa \geq 3$. □

Exact Bayesian inference in context computes $\#F$ through Lemma 4.2, so it is equivalent to counting, as consistency is equivalent to satisfiability [Li26, Theorem 3.4].

Theorem 4.6 (Bayesian inference in context is $\#\kappa$-SAT)**.** *Let $k \geq 2$, and let T, T' be nondecreasing time bounds.*

(i) If #k-SAT on n variables with M clauses can be solved in time $T(n, M)$, then noiseless Bayesian inference in context for k-local classes with N-bit descriptions takes time $2T(N, (m+1)2^k) + \mathrm{poly}(N, m) \cdot 2^k$ on consistent contexts of size m.

(ii) If noiseless Bayesian inference in context for BP_{2k}, on consistent contexts of size m in dimension n, can be done exactly in time $T'(n, m)$, then #k-SAT on n variables with M clauses can be solved in time $T'(n + k + 1, M + n + k) + \mathrm{poly}(n, M)$.

Proof. Part (i) is Corollary 3.2. For (ii), delete tautological clauses, which changes no count, build S_F^{BP} by Lemma 4.2, compute $p = p_S(e_a) < 1$, and output $p/(2(1-p))$. □

The hard contexts are consistent, so the lower bounds do not come from deciding consistency, which is trivial for them. They are, however, atypical for the prior: all their labels are positive, and a random task labels them so only with probability $(2\#F + 1)/2^N$. The bounds therefore concern Bayesian inference in context on every context, the guarantee that the Bayesian explanation of in-context learning attributes to a trained model, and not the Bayes risk on typical episodes. Section 6 treats the latter.

5 Efficient in-context learning

This section shows that learning in context is cheap for local classes when Bayes-optimality is not required. For the families of the lower bounds a halfspace suffices, and every local class is a linear class over local Fourier features.

5.1 The hard families are halfspaces

Proposition 5.1 (Halfspace forms)**.** *Let $\sigma \in \{\pm 1\}^n$ and $x \in \{-1, 0, 1\}^n$ with $r = |\mathrm{supp}(x)|$. Then $h_\sigma^{\mathsf{BP}}(x) = \mathbf{1}[\langle \sigma, x \rangle \geq 0]$ and $h_\sigma^{\mathsf{LD}}(x) = \mathbf{1}[\langle \sigma, x \rangle + r - 2 \geq 0]$. If $h_\sigma^{\mathsf{BP}}(x) = 0$ then $\langle \sigma, x \rangle \leq -1$, and if $h_\sigma^{\mathsf{LD}}(x) = 0$ then $\langle \sigma, x \rangle + r - 2 = -2$.*

Proof sketch. The first form is the definition, and $\langle \sigma, x \rangle$ is an integer. For literal disjunctions, if t coordinates of $\mathrm{supp}(x)$ agree with σ, then $\langle \sigma, x \rangle = t - (r - t)$, so $t \geq 1$ if and only if $\langle \sigma, x \rangle + r - 2 \geq 0$, and $t = 0$ gives -2 (Section C). □

Let $c(x) = 0$ for binarized perceptrons and $c(x) = |\mathrm{supp}(x)| - 2$ for literal disjunctions. The in-context learner A_{LP} solves the linear program

$$\langle v, x_i \rangle + c(x_i) \geq 0 \ \text{ if } y_i = 1, \qquad \langle v, x_i \rangle + c(x_i) \leq -1 \ \text{ if } y_i = 0 \qquad (i \leq m) \tag{2}$$

over $v \in \mathbb{R}^n$ and predicts $\mathbf{1}[\langle v, x \rangle + c(x) \geq 0]$; if the program is infeasible it predicts 0.

Theorem 5.2 (Linear programming in context)**.** *For every $k \leq n$, the in-context learner A_{LP} for BP_k and for LD_k runs in time $\mathrm{poly}(n, m)$ and is (ε, δ)-PAC from $m_{\mathrm{VC}}(n + 1, \varepsilon, \delta)$ examples, which is $O((n \log(1/\varepsilon) + \log(1/\delta))/\varepsilon)$. Its risk in every noiseless episode, for every input distribution, is $O(n \log m/m)$.*

Proof sketch. If the context is consistent with σ, then $v = \sigma$ is feasible by Proposition 5.1, and every feasible v classifies the context correctly. The classifiers $x \mapsto \mathbf{1}[\langle v, x\rangle + c(x) \geq 0]$ are homogeneous halfspaces on $(x, c(x)) \in \mathbb{R}^{n+1}$ with last coefficient 1, a class of VC dimension at most $n+1$, so Imported Theorem 2.13(a) applies. For the risk, take $\varepsilon = C_B(n+2)\log(2m)/m$ and $\delta = 1/m$ (Section C). □

On the families of Theorem 4.4(a), the comparison is now complete.

Corollary 5.3 (The price of Bayes-optimality)**.** *Assume Hypothesis 2.14, let $\gamma \in (0,1)$, and let K and c_γ be as in Theorem 4.4(a). Consider $\mathsf{BP}_{\lceil K \log N\rceil}$ in dimension N.*

(a) The learner A_{LP} runs in polynomial time and has risk $O(N \log m/m)$ in every noiseless episode.

(b) The Bayes predictor has risk at most $N/(m+1)$ (Theorem 3.3).

(c) No randomized algorithm computes the Bayes predictor, or a number within $1/4$ of the posterior predictive, on all consistent contexts of size at most N^{c_γ} in time $O(2^{(1-\gamma)N})$.

The same holds for $\mathsf{LD}_{\lceil K \log N\rceil}$, and, with $\mathrm{ETH}^{\mathrm{len}}_C$, time $2^{N/(\log N)^{C+\theta}}$ and contexts of at most $36N^2$ examples in (c), for LD_3.

So for these families a polynomial-time in-context learner matches the sample efficiency of the Bayes predictor up to a factor $\log m$, while the Bayes predictor itself costs time $2^{(1-\gamma)N}$. In the language of [Li26], the locality principle governs proper learning and Bayesian computation, but not improper learning: proper PAC learning of $\mathsf{BP}_{\lceil K \log N\rceil}$ at inverse-polynomial accuracy needs time $2^{(1-\gamma)N}$ under CNF-SETH [Li26, Theorem 3.8], and A_{LP} is improper.

5.2 Local classes are linear

For a general local class no halfspace in the input coordinates need exist, but a linear form in local Fourier features always does. For $T \subseteq [N]$ let $\chi_T(w) = \prod_{i\in T}(-1)^{w_i}$, and let $D_\kappa = D_\kappa(N) = \sum_{j=0}^{\kappa}\binom{N}{j}$.

Theorem 5.4 (Local Fourier features)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions. For $x \in \mathcal{X}$ and $T \subseteq [N]$ with $|T| \leq \kappa$, let*

$$\psi_T(x) = 2^{-|R(x)|} \sum_{\beta \in \{0,1\}^{R(x)}} f_x(\beta)\,\chi_T(\beta) \quad \text{if } T \subseteq R(x), \qquad \psi_T(x) = 0 \quad \text{otherwise,}$$

and let $\Psi(x) = (\psi_T(x))_{|T|\leq\kappa}$ and $\Phi(w) = (\chi_T(w))_{|T|\leq\kappa}$, vectors of dimension D_κ. Then $h_w(x) = \langle \Phi(w), \Psi(x)\rangle$ for all w and x. The vector $\Psi(x)$ has at most $2^{|R(x)|}$ nonzero entries, all in $2^{-\kappa}\mathbb{Z}$, and it can be computed in time $\mathrm{poly}(N, |x|)\cdot\kappa 2^\kappa$. For $1 \leq \kappa \leq N$, $D_\kappa \leq (eN/\kappa)^\kappa$.

Proof sketch. This is the Fourier expansion of f_x on the cube $\{0,1\}^{R(x)}$ [O'D14], and $\chi_T(w|_{R(x)}) = \chi_T(w)$ for $T \subseteq R(x)$. The coefficients come from $2^{|R(x)|}$ evaluations of f_x and a fast Walsh–Hadamard transform (Section C). □

The learner $\mathsf{A}_{\mathrm{lin}}$ solves the linear system $\langle v, \Psi(x_i)\rangle = y_i$ $(i \leq m)$ over $\mathbb{Q}$ by Gaussian elimination and predicts $\mathbf{1}[\langle v, \Psi(x)\rangle \geq 1/2]$; if the system has no solution it predicts 0. The learner A_{ℓ_1} minimizes $\sum_{i\leq m}|\langle v, \Psi(x_i)\rangle - y_i|$ over $v \in \mathbb{R}^{D_\kappa}$ by linear programming, then chooses $\theta \in [0,1]$ to minimize the number of errors of $x \mapsto \mathbf{1}[\langle v, \Psi(x)\rangle \geq \theta]$ on the context, and predicts with that classifier. This is the ℓ_1 regression of [KKMS08].

Theorem 5.5 (Regression in context)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions.*

(a) $\mathsf{A}_{\mathrm{lin}}$ *is* (ε, δ)*-PAC from* $m_{\mathrm{VC}}(D_\kappa + 1, \varepsilon, \delta)$ *examples.*

(b) A_{ℓ_1} *is* (ε, δ)*-agnostic from* $m_{\mathrm{UC}}(D_\kappa + 1, \varepsilon, \delta)$ *examples. In an episode with noise rate* η*, its risk is at most* $\eta + \varepsilon + \delta$*.*

(c) Both run in time $\mathrm{poly}(N, m, D_\kappa) \cdot 2^\kappa$*. This is polynomial for constant* κ*, it is* $N^{O(\log N)}$ *for* $\kappa = O(\log N)$*, and it is* $2^{o(N)}$ *times a polynomial in* m *whenever* $\kappa \log(eN/\kappa) = o(N)$*.*

Proof sketch. For (a), $v = \Phi(w)$ solves the system, every solution is consistent with the context, and the thresholded linear forms over Ψ have VC dimension at most $D_\kappa + 1$. For (b), for a threshold θ uniform in $[0, 1]$ the expected error of $\mathbf{1}[z \geq \theta]$ against $y \in \{0, 1\}$ is at most $|z - y|$, so the chosen classifier has at most as many errors on the context as the best h_w, whose linear form $\Phi(w)$ is feasible for the regression. Uniform convergence over the thresholded linear forms, which contain every h_w, finishes the proof (Section C). □

Remark 5.6 (Improper learning of local classes). Theorem 5.5 uses no algorithm for SAT. Improper learning of local classes was polynomial at constant locality before the refutation of ETH, and it stays quasi-polynomial at logarithmic locality, where proper learning needs time $2^{(1-\gamma)N}$ under CNF-SETH [Li26]. From below, parities on a fixed set of κ description bits form a κ-local class, and agnostic learning them under uniform inputs with constant noise rate is learning parity with noise on κ bits. The best known algorithm for it runs in time $2^{O(\kappa/\log \kappa)}$ [BKW03]. If no algorithm is faster than $2^{\Omega(\kappa/\log \kappa)}$, then for $\kappa = N^{\Omega(1)}$ the time of improper agnostic learning of κ-local classes lies between $2^{\Omega(\kappa/\log \kappa)}$ and $N^{O(\kappa)} = 2^{O(\kappa \log N)}$, so its exponent is κ up to polylogarithmic factors. Improper learning is governed by the locality itself, proper learning and Bayesian inference by the rate of κ-SAT on N variables.

The price of these learners is in the examples: $\mathsf{A}_{\mathrm{lin}}$ needs about $D_\kappa \approx N^\kappa$ of them, where the Bayes predictor needs N. Section 6 shows that for random local functions some such price is unavoidable for every efficient learner.

6 The average case: random local functions

The lower bounds of Section 4 concern atypical contexts. On typical episodes, the question is whether an efficient learner can reach the Bayes risk. For random local functions it cannot, at the context lengths where the Bayes predictor already succeeds, unless Goldreich's generator is insecure.

6.1 Random local functions and Goldreich's generator

Definition 6.1 (Random local functions)**.** Let $k \geq 2$ and $P\colon \{0,1\}^k \to \{0,1\}$. Let $\mathcal{X}_N^{(k)}$ be the set of k-tuples of distinct indices in $[N]$, and let $\mathcal{D}_k$ be the uniform distribution on it. The class RL_P on $\mathcal{X}_N^{(k)}$ has the hypotheses $h_w(\tau) = P(w_{\tau_1}, \ldots, w_{\tau_k})$ for $w \in \{0,1\}^N$. It is k-local with $R(\tau) = \{\tau_1, \ldots, \tau_k\}$. Goldreich's generator with m outputs maps w and $\tau_1, \ldots, \tau_m$ to $G_{P,m}(\tau; w) = (h_w(\tau_1), \ldots, h_w(\tau_m))$ [Gol00].

So an episode of the task family $(\mathsf{RL}_P, \mathcal{D}_k)$ without noise is a run of Goldreich's generator with $m+1$ outputs: the context is the first m outputs with their tuples, and the target is the last output.

Security is asymptotic: a fixed algorithm must fail for all large N. (For a single N, an algorithm that stores the finite image of the generator distinguishes it quickly, so a statement about all algorithms of a given running time at one N would be false.)

Definition 6.2 (Pseudorandomness)**.** Let $m = m(N)$. The *advantage* of a probabilistic algorithm A against $G_{P,m}$ at dimension N is

$$\mathrm{adv}_A(N) = \left|\Pr[A(\tau, G_{P,m}(\tau; w)) = 1] - \Pr[A(\tau, b) = 1]\right|,$$

where τ is uniform in $(\mathcal{X}_N^{(k)})^m$, w is uniform in $\{0,1\}^N$, and b is uniform in $\{0,1\}^m$. For a function $T = T(N)$, the generator $G_{P,m}$ is *T-pseudorandom* if every probabilistic algorithm whose running time on inputs of dimension N is at most $T(N)$ for all sufficiently large N has advantage at most $1/3$ at all sufficiently large N.

A distinguisher for $G_{P,m'}$ with $m' \le m$ yields one for $G_{P,m}$ with the same advantage, because it can ignore the last outputs: the first m' outputs of $G_{P,m}$, with their tuples, are distributed as those of $G_{P,m'}$, and a prefix of a uniform string is uniform. Pseudorandomness of random local functions is a central hypothesis of cryptography in NC^0; see the survey [App16]. At polynomial stretch $m = N^s$ it needs predicates that are uncorrelated with all parities of few inputs, and for a predicate uncorrelated with all parities of fewer than t inputs, sum-of-squares and statistical-query lower bounds support it up to stretch $N^{\Omega(t)}$ [OW14, KMOW17, FPV18]. We use two forms of the hypothesis. The first, for every $s > 1$, is Assumption 1 of Daniely and Vardi [DV21], which they use to prove the hardness of improper PAC learning for DNF formulas, intersections of halfspaces and other classes.

Hypothesis 6.3 (LPRG_s)**.** For a constant $s > 1$: there are $k \ge 2$ and $P\colon \{0,1\}^k \to \{0,1\}$ such that $G_{P,\lceil N^s \rceil}$ is T-pseudorandom for every polynomial T.

Hypothesis 6.4 ($\mathrm{LPRG}_c^{\mathrm{len}}$)**.** For a constant $c > 0$: there are $k \ge 2$, $P\colon \{0,1\}^k \to \{0,1\}$ and $\alpha_0 \ge 2$ such that for every constant $\alpha \ge \alpha_0$, the generator $G_{P,\lceil \alpha N \rceil}$ is $2^{N/(\log N)^c}$-pseudorandom.

The second hypothesis is the analogue of $\mathrm{ETH}_C^{\mathrm{len}}$ for random local functions at linear stretch. Like $\mathrm{ETH}_C^{\mathrm{len}}$, the new algorithms refute it below rate $1/2$.

Proposition 6.5 ($\mathrm{LPRG}_c^{\mathrm{len}}$ fails below rate $1/2$)**.** *For every $k \ge 2$, every P, every constant $\alpha \ge 2$ and every $c < 1/2$, there is a deterministic algorithm that runs in time at most $2^{N/(\log N)^c}$ for large N and has advantage at least $1 - 2^{-N}$ against $G_{P,\lceil \alpha N \rceil}$ at every N. Hence $\mathrm{LPRG}_c^{\mathrm{len}}$ is false for every $c < 1/2$.*

Proof sketch. Given (τ, z), decide whether the context $((\tau_j, z_j))_j$ is consistent with RL_P, by Imported Theorem 2.8 in time $\mathrm{poly}(N) \cdot 2^{O(k^2 N \log\log N/\sqrt{\log N})}$, and output 1 if it is. Outputs of the generator are always consistent. A uniform z is consistent with probability at most $2^N/2^{\lceil \alpha N \rceil} \le 2^{-N}$, since for fixed tuples at most 2^N strings are images (Section D). □

For $c \ge 1/2$ the new algorithms do not contradict $\mathrm{LPRG}_c^{\mathrm{len}}$, and before 2026 the best known distinguishers for suitable predicates at linear stretch took time $2^{\Omega(N)}$.

6.2 Three regimes

Theorem 6.6 (Three regimes for random local functions)**.** *Let $k \ge 2$ be a constant and $P\colon \{0,1\}^k \to \{0,1\}$. Consider the task family $(\mathsf{RL}_P, \mathcal{D}_k)$ without noise.*

(a) The Bayes risk is at most $N/(m+1)$, and the Bayes predictor runs in time $\mathrm{poly}(m) \cdot 2^{O(N \log\log N/\sqrt{\log N})}$.

(b) The learner $\mathsf{A}_{\mathrm{lin}}$ is (ε, δ)-PAC from $m_{\mathrm{VC}}(D_k(N) + 1, \varepsilon, \delta) = O((N^k \log(1/\varepsilon) + \log(1/\delta))/\varepsilon)$ examples and runs in time $\mathrm{poly}(N, m)$.

(c) *There is an absolute constant c_* with the following property. For every in-context learner A and every context length m, there is a probabilistic algorithm A' that runs in time $T_A + c_*(m+1)k \log N$, where T_A is the running time of A on contexts of length m, and has advantage exactly $|1/2 - R_m(A)|$ against $G_{P,m+1}$. If $m = m(N)$ is computable in time polynomial in N, then A' is a single algorithm for all N.*

Proof sketch. Parts (a) and (b) are Theorems 3.1, 3.3 and 5.5 for this k-local class. For (c), A' runs A on the first m outputs and the last tuple and accepts if the prediction equals the last output. On outputs of the generator it accepts with probability $1 - R_m(A)$; on uniform bits the last bit is independent of the rest, so it accepts with probability $1/2$ (Section D). □

Corollary 6.7 (The three regimes under the hypotheses)**.** *(a) Assume LPRG_s, with predicate P of arity k. Let A be a polynomial-time in-context learner for $(\mathsf{RL}_P, \mathcal{D}_k)$ and $m(N) \le \lceil N^s \rceil - 1$ a context length computable in time polynomial in N. Then $R_{m(N)}(A) \ge 1/6$ for all sufficiently large N. At $m = \lceil N^s \rceil - 1$ the Bayes risk is at most N^{1-s}, and $\mathsf{A}_{\mathrm{lin}}$ has risk at most ε from $O(N^k \log(1/\varepsilon)/\varepsilon)$ examples.*

(b) Assume $\mathrm{LPRG}_c^{\mathrm{len}}$, with predicate P of arity k and constant α_0. Let $\alpha \ge \max\{\alpha_0, 7\}$ be a constant and $m(N) = \lceil \alpha N \rceil - 1$. Every in-context learner that runs in time $2^{N/(\log N)^c - 1}$ on contexts of length $m(N)$ has $R_{m(N)}(A) \ge 1/6$ for all sufficiently large N, while the Bayes predictor has risk at most $1/\alpha < 1/6$ and runs in time $2^{O(N \log\log N/\sqrt{\log N})}$.

So, at context length $\lceil \alpha N \rceil - 1$, the time needed for risk below $1/6$ on random local functions has exponent between $N/(\log N)^c$ under $\mathrm{LPRG}_c^{\mathrm{len}}$ and $O(N \log\log N/\sqrt{\log N})$, which is $N/\mathrm{polylog} N$ on both sides when c is a constant at least $1/2$. The Bayes predictor of the refutations is optimal up to polylogarithmic factors in the exponent. Between N^s and N^k examples the complexity is open; the thresholds of refutation and recovery for planted constraint satisfaction problems, near $N^{t/2}$ for predicates uncorrelated with all parities of fewer than t inputs, suggest where polynomial time begins [FPV18, KMOW17].

7 Consequences for transformers

The lower bounds above hold for every algorithm. This section states them for transformers that write a chain of thought, and it compares the two explanations of in-context learning.

7.1 Chains of thought

We model a trained transformer only by its cost. A *sequence model* with p parameters maps a sequence of ℓ tokens to a next token. We assume that this map can be computed in time $(p + \ell)^{c_M}$ for a constant c_M, as it can for a transformer with p parameters of polynomial precision and polynomially many layers and heads. The model answers a context and a query, encoded with $\mathrm{poly}(N)$ tokens per example, after generating a *chain of thought* of at most L tokens. Generating L tokens from a prompt of ℓ tokens takes time at most $(L+1)(p + \ell + L)^{c_M} \le (p + \ell + L)^{c_M + 1}$.

The hypotheses we use are statements about algorithms, so the models must be produced by an algorithm too. A family (M_N) of sequence models is *uniform* if a fixed deterministic algorithm outputs the parameters of M_N in time polynomial in N, for example a polynomial-time training procedure run with a fixed random seed. Let $L(N)$ be the largest length of a chain of thought of M_N on the inputs under consideration.

Corollary 7.1 (Chain-of-thought lengths)**.** *Let (M_N) be a uniform family of sequence models with $p(N) \le N^{O(1)}$ parameters, as above.*

(a) *Assume Hypothesis 2.14, and let γ, K, c_γ be as in Theorem 4.4(a). If M_N outputs $\mathsf{B}(S,x)$ for every consistent context of at most N^{c_γ} examples of $\mathsf{BP}_{\lceil K \log N \rceil}$ in dimension N and every query, then $L(N) > 2^{(1-\gamma)N/(c_M+1)} - N^{O(1)}$ for infinitely many N. So $L(N) = 2^{\Omega(N)}$ for infinitely many N.*

(b) *Assume $\mathrm{ETH}_C^{\mathrm{len}}$ and let $\theta > 0$. If M_N outputs $\mathsf{B}(S,x)$ for every consistent context of at most $36N^2$ examples of LD_3 in dimension N and every query, then $L(N) \geq 2^{N/(\log N)^{C+\theta}}$ for infinitely many N.*

(c) *Assume LPRG_s with predicate P, and let $m(N) \leq \lceil N^s \rceil - 1$ be computable in time polynomial in N. If $L(N) \leq N^{O(1)}$, then M has risk at least $1/6$ on $(\mathsf{RL}_P, \mathcal{D}_k)$ at context length $m(N)$ for all sufficiently large N.*

(d) *Assume $\mathrm{LPRG}_c^{\mathrm{len}}$ with predicate P, and let α and $m(N)$ be as in Corollary 6.7(b). If $L(N) \leq 2^{N/((c_M+2)(\log N)^c)}$ for all sufficiently large N, then M has risk at least $1/6$ on $(\mathsf{RL}_P, \mathcal{D}_k)$ at context length $m(N)$ for all sufficiently large N.*

Proof sketch. Computing the parameters and then simulating M_N is an algorithm of time $N^{O(1)} + (p + \ell + L)^{c_M+1}$, and on the inputs considered the prompt length ℓ is polynomial in N. In (a) and (b), an algorithm that computes $\mathsf{B}(S,x)$ on all these contexts is excluded below the times of Theorem 4.4; solving for L gives the bounds. Parts (c) and (d) are Corollary 6.7 for the simulation (Section D). □

For families with arbitrary trained weights, parts (c) and (d) hold under the versions of the generator hypotheses that allow non-uniform distinguishers, such as circuits of the stated size, and parts (a) and (b) would need non-uniform versions of CNF-SETH and $\mathrm{ETH}_C^{\mathrm{len}}$. Conversely, transformers that generate a chain of thought can simulate Turing machines [PBM21, MS24], so the algorithm of Theorem 3.1 can in principle run in a chain of thought of subexponential length for locality $o((\log N)^{1/4}/\sqrt{\log \log N})$. Before the refutation of ETH, the best known algorithms for the Bayes prediction of LD_3 took time $2^{\Omega(N)}$; under $\mathrm{ETH}_C^{\mathrm{len}}$, a uniform family that computes it needs chains of thought of length $2^{N/(\log N)^{C+\theta}}$ for infinitely many N.

7.2 Bayesian and algorithmic in-context learning

Two explanations of in-context learning are current. The Bayesian one holds that a model trained on a task distribution predicts like its posterior [XRLM22, MHA+22, PAG24]. The algorithmic one holds that the model runs a learning algorithm on its context, such as least squares, ridge regression or gradient descent [GTLV22, ASA+23, vONR+23, BCW+23]. Raventós, Paul, Chen and Ganguli [RPCG23] find both in one setting: trained on few regression tasks, a transformer follows the Bayes estimator of its pretraining distribution; trained on many, it leaves that estimator and follows ridge regression, which lets it solve new tasks.

For discrete task families with local structure, our results give the computational side of this picture. The Bayes estimator over 2^N tasks is a counting problem: computing it on all contexts costs time $2^{N/\mathrm{polylog}N}$ at constant locality and $2^{(1-\gamma)N}$ at logarithmic locality under the surviving hypotheses (Theorem 4.4), and on random local functions no efficient learner reaches the Bayes risk at the context lengths where the Bayes estimator does (Corollary 6.7). The algorithmic alternative is efficient: least squares over local Fourier features learns every κ-local family from $N^{O(\kappa)}$ examples (Theorem 5.5), and for binarized perceptrons and literal disjunctions a linear program learns from $O(N \log(1/\varepsilon)/\varepsilon)$ examples (Theorem 5.2). In-context least squares and ridge regression are known to be implementable by transformers [ASA+23, BCW+23], here with $\Psi(x)$ as the representation of an input. A polynomial-size model produced by a polynomial-time training procedure can therefore

be an efficient in-context learner for such a family, but, unless the hypotheses fail, not a Bayes estimator for it.

The setting of [RPCG23] differs from ours: its discrete pretraining distributions are explicit lists of regression tasks, for which the Bayes estimator is a finite mixture whose cost is linear in the number of tasks. Our results concern exponentially many tasks with local structure, where the cost of the mixture is the cost of counting.

8 Related work

The refutation of ETH and learning. OpenAI [Ope26] and Anthropic [Cla26] refuted ETH by counting the solutions of k-CNFs in time $2^{n/(\log n)^{1/2-o(1)}}$, by polynomial interpolation and by signed sums of tensor networks. The companion paper [Li26] derived the locality principle for proper learning, error-profile algorithms for exact empirical risk minimization, agnostic and private learning, the transitions in locality, accuracy and treewidth, and a ledger of the conditional lower bounds of learning theory; we use its statements as black boxes. Bhattacharjee [Bha26] runs the algorithm of [Ope26] over arbitrary commutative $\mathbb{Q}$-algebras, which gives weighted counting with several linear statistics at once.

In-context learning. Garg, Tsipras, Liang and Valiant [GTLV22] trained transformers to learn simple function classes in context. Akyürek et al. [ASA+23], von Oswald et al. [vONR+23] and Bai et al. [BCW+23] showed that transformers can implement gradient descent, least squares, ridge regression and algorithm selection in context, with statistical guarantees. Xie, Raghunathan, Liang and Ma [XRLM22] explained in-context learning as implicit Bayesian inference, prior-fitted networks [MHA+22] are trained to approximate posterior predictives, and Panwar, Ahuja and Goyal [PAG24] compared trained transformers with Bayes predictors on several task families. Raventós, Paul, Chen and Ganguli [RPCG23] observed the transition from the Bayes estimator of the pretraining distribution to ridge regression as task diversity grows. Bhattamishra et al. [BPBK24] studied in-context learning of discrete function classes. Pérez, Barceló and Marinkovic [PBM21] and Merrill and Sabharwal [MS24] studied the computational power of transformers with intermediate generation. These works concern what in-context learners compute and how well they generalize; the present paper concerns the computational cost of Bayes-optimal in-context prediction.

Bayesian learning curves, halving and version spaces. The halving algorithm goes back to Barzdin and Freivald and to Littlestone [BF72, Lit88]. Haussler, Kearns and Schapire [HKS94] bounded the learning curves of the Bayes and Gibbs algorithms by information gain and VC dimension. The log-loss regret of Bayes mixtures is standard [CBL06]. Sampling by self-reducibility is due to Jerrum, Valiant and Vazirani [JVV86].

Proper and improper learning. Pitt and Valiant [PV88] showed that proper learning can be NP-hard when improper learning is easy, as for k-term DNF and k-CNF; Corollary 5.3 is a fine-grained, exponential-time form of this contrast for Bayes prediction. The ℓ_1 regression algorithm is that of Kalai, Klivans, Mansour and Servedio [KKMS08]. Daniely, Linial and Shalev-Shwartz [DLSS13] showed that more data speeds up improper learning of halfspaces over sparse vectors, which complements the sample price of Theorem 5.5.

Local pseudorandom generators. Goldreich proposed random local functions as one-way functions [Gol00]; Applebaum's survey [App16] covers their use as pseudorandom generators.

O'Donnell and Witmer [OW14] gave evidence for polynomial stretch, and Kothari, Mori, O'Donnell and Witmer [KMOW17] and Feldman, Perkins and Vempala [FPV18] proved sum-of-squares and statistical-query lower bounds for refuting and solving random and planted constraint satisfaction problems. Daniely and Vardi [DV21] derived hardness of improper PAC learning from local pseudorandom generators. Learning parity with noise is solved by the algorithm of Blum, Kalai and Wasserman [BKW03]. The step from prediction to distinguishing is Yao's [Yao82].

9 Open problems

Open Problem 1 (Typical contexts for natural families)**.** The worst-case lower bounds of Theorem 4.4 use atypical contexts, and the average-case bounds of Corollary 6.7 use random local functions. Is Bayesian inference in context hard on typical episodes of binarized perceptrons or literal disjunctions on sparse data, for natural input distributions, at locality $O(\log N)$?

Open Problem 2 (Approximate Bayesian inference)**.** Our lower bounds exclude additive error $1/4$ on all contexts. With label noise, the posterior is a Gibbs measure at inverse temperature $\ln((1-\eta)/\eta)$. Is there a noise rate below which approximating the posterior predictive becomes easy, for example by Markov chain methods, and does the threshold depend on the locality?

Open Problem 3 (Between N^s and N^k examples)**.** For random local functions of arity k, polynomial-time learning needs more than N^s examples under LPRG_s and succeeds from $O(N^k)$. What is the time needed from $N^{1+\beta}$ examples, as a function of β? Spectral refutation trades density for time on random instances; can it be combined with the algorithms of the refutation of ETH to beat $2^{N/\sqrt{\log N}}$ at superlinear context length?

Open Problem 4 (The locality window)**.** By Theorem 4.6, the complexity of exact Bayesian inference in context at locality between $(\log N)^{1/4}$ and $K \log N$ is that of $\#k$-SAT for $k = (\log N)^{\Theta(1)}$, the window of [Li26, Open Problem 1]. Is approximate Bayesian inference in context easier there than exact counting?

Open Problem 5 (Bayes risk without Bayes inference)**.** For binarized perceptrons on $(K \log N)$-sparse data, A_{LP} has risk $O(N \log m/m)$ and the Bayes risk is at most $N/(m+1)$. Can a polynomial-time in-context learner achieve risk $R^*_m + \varepsilon$ for every input distribution and every $\varepsilon > 0$, with m polynomial in N/ε, or is the Bayes risk itself hard to reach for this family?

References

[App16] Benny Applebaum. Cryptographic hardness of random local functions: Survey. *Computational Complexity*, 25(3):667–722, 2016.

[ASA+23] Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? Investigations with linear models. In *11th International Conference on Learning Representations (ICLR)*, 2023.

[BCW+23] Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, 2023.

[BEHW89] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the Vapnik–Chervonenkis dimension. *Journal of the ACM*, 36(4):929–965, 1989.

[BF72] J. M. Barzdin and R. V. Freivald. On the prediction of general recursive functions. *Doklady Akademii Nauk SSSR*, 206(3):521–524, 1972. English translation in Soviet Mathematics Doklady 13:1224–1228.

[Bha26] Deep Bhattacharjee. Subexponential weighted counting and exact optimisation. Hexagon preprint hexagon:2610.00594v1, dated October 29, 2026. https://hexagonmath.org/2610.00594v1, 2026.

[BKW03] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM*, 50(4):506–519, 2003.

[BPBK24] Satwik Bhattamishra, Arkil Patel, Phil Blunsom, and Varun Kanade. Understanding in-context learning in transformers and LLMs by learning to learn discrete functions. In *12th International Conference on Learning Representations (ICLR)*, 2024.

[CBL06] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[CIP06] Chris Calabro, Russell Impagliazzo, and Ramamohan Paturi. A duality between clause width and clause density for SAT. In *21st Annual IEEE Conference on Computational Complexity (CCC)*, pages 252–260, 2006.

[Cla26] Claude (Anthropic). A $2^{o(n)}$ time algorithm for 3-SAT. Hexagon preprint hexagon:2610.00577v1, dated October 26, 2026. https://hexagonmath.org/2610.00577v1. Lean development at https://github.com/anthropics/formal-math/tree/main/eth, 2026.

[CRTY23] Lijie Chen, Ron D. Rothblum, Roei Tell, and Eylon Yogev. On exponential-time hypotheses, derandomization, and circuit lower bounds. *Journal of the ACM*, 70(4):1–62, 2023.

[DLSS13] Amit Daniely, Nati Linial, and Shai Shalev-Shwartz. More data speeds up training time in learning halfspaces over sparse vectors. In *Advances in Neural Information Processing Systems 26 (NIPS)*, 2013.

[DV21] Amit Daniely and Gal Vardi. From local pseudorandom generators to hardness of learning. In *34th Conference on Learning Theory (COLT)*, volume 134 of *PMLR*, pages 1358–1394, 2021.

[FPV18] Vitaly Feldman, Will Perkins, and Santosh Vempala. On the complexity of random satisfiability problems with planted solutions. *SIAM Journal on Computing*, 47(4):1294–1338, 2018.

[Gol00] Oded Goldreich. Candidate one-way functions based on expander graphs. Technical Report TR00-090, Electronic Colloquium on Computational Complexity, 2000.

[GTLV22] Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. What can transformers learn in-context? A case study of simple function classes. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

[HKS94] David Haussler, Michael Kearns, and Robert E. Schapire. Bounds on the sample complexity of Bayesian learning using information theory and the VC dimension. *Machine Learning*, 14(1):83–113, 1994.

[JVV86] Mark R. Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science*, 43:169–188, 1986.

[KKMS08] Adam Tauman Kalai, Adam R. Klivans, Yishay Mansour, and Rocco A. Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

[KMOW17] Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. Sum of squares lower bounds for refuting any CSP. In *49th Annual ACM Symposium on Theory of Computing (STOC)*, pages 132–145, 2017.

[Lam95] Leslie Lamport. How to write a proof. *The American Mathematical Monthly*, 102(7):600–608, 1995.

[Li26] Xiaoyu Li. What does the refutation of the Exponential Time Hypothesis mean for learning theory? Hexagon preprint hexagon:2610.00613v2. https://hexagonmath.org/2610.00613v2, 2026.

[Lit88] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[MHA+22] Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do Bayesian inference. In *10th International Conference on Learning Representations (ICLR)*, 2022.

[MS24] William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. In *12th International Conference on Learning Representations (ICLR)*, 2024.

[O'D14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

[Ope26] OpenAI. The Exponential Time Hypothesis is false. OpenAI Math Release preprint, dated October 17, 2026, revised October 25, 2026, released October 26, 2026. https://github.com/openai/math/blob/main/preprints/The-Exponential-Time-Hypothesis-is-False-October-25-2026/paper.pdf, 2026.

[OW14] Ryan O'Donnell and David Witmer. Goldreich's PRG: Evidence for near-optimal polynomial stretch. In *29th IEEE Conference on Computational Complexity (CCC)*, pages 1–12, 2014.

[PAG24] Madhur Panwar, Kabir Ahuja, and Navin Goyal. In-context learning through the Bayesian prism. In *12th International Conference on Learning Representations (ICLR)*, 2024.

[PBM21] Jorge Pérez, Pablo Barceló, and Javier Marinkovic. Attention is Turing-complete. *Journal of Machine Learning Research*, 22(75):1–35, 2021.

[PV88] Leonard Pitt and Leslie G. Valiant. Computational limitations on learning from examples. *Journal of the ACM*, 35(4):965–984, 1988.

[RPCG23] Allan Raventós, Mansheej Paul, Feng Chen, and Surya Ganguli. Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, 2023.

[Sch05] Rainer Schuler. An algorithm for the satisfiability problem of formulas in conjunctive normal form. *Journal of Algorithms*, 54(1):40–44, 2005.

[SSBD14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[vONR+23] Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *40th International Conference on Machine Learning (ICML)*, volume 202 of *PMLR*, pages 35151–35174, 2023.

[XRLM22] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit Bayesian inference. In *10th International Conference on Learning Representations (ICLR)*, 2022.

[Yao82] Andrew C. Yao. Theory and applications of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 80–91, 1982.

A Proofs for Sections 2 and 3

The proofs are hierarchical in the style of Lamport [Lam95]: each step is a claim with its own proof, and Q.E.D. marks the step that concludes its level.

A.1 The Bayes decision rule

Lemma 2.5 (Bayes decision rule)**.** *For every task family, noise rate and context length m, every in-context learner A has $R_m(A) \geq R_m(\mathsf{B})$, and $R_m(\mathsf{B}) = \mathbb{E}[\min\{\hat{p}_S(x_{m+1}), 1 - \hat{p}_S(x_{m+1})\}]$.*

Proof of Lemma 2.5. ⟨1⟩1. Condition on the inputs $x_1, \ldots, x_{m+1}$ and the labels $y_1, \ldots, y_m$. Then the task has distribution π_S, and $\Pr[y_{m+1} = 1 \mid S, x_{m+1}] = \hat{p}_S(x_{m+1})$.

PROOF: The inputs are independent of the task and of the noise. Given the inputs, the probability of the labels $y_1, \ldots, y_m$ under task w is $\prod_{i \leq m} \eta^{[h_w(x_i) \neq y_i]}(1-\eta)^{[h_w(x_i)=y_i]} = (1-\eta)^m \mu^{e_S(w)}$, and the prior is uniform, so the posterior is proportional to $\omega_S(w)$. The target is $h_w(x_{m+1}) \oplus \xi_{m+1}$ with ξ_{m+1} independent of everything else, so it is 1 with probability $\sum_w \pi_S(w)\,[(1-\eta)h_w(x_{m+1}) + \eta(1 - h_w(x_{m+1}))] = \hat{p}_S(x_{m+1})$.

⟨1⟩2. For every in-context learner A, the conditional error probability given S and x_{m+1} is at least $\min\{\hat{p}, 1-\hat{p}\}$, with $\hat{p} = \hat{p}_S(x_{m+1})$, and B attains it.

PROOF: Given S and x_{m+1}, the output of A depends only on its own random bits, which are independent of the target. If A outputs 1 with conditional probability q, it errs with probability $q(1-\hat{p}) + (1-q)\hat{p} \geq \min\{\hat{p}, 1-\hat{p}\}$. Since $\hat{p} - 1/2 = (1-2\eta)(p_S(x_{m+1}) - 1/2)$ and $\eta < 1/2$, the predictor B outputs 1 exactly when $\hat{p} \geq 1/2$, so its conditional error is $\min\{\hat{p}, 1-\hat{p}\}$.

⟨1⟩3. Q.E.D.

PROOF: Take expectations in ⟨1⟩2.

□

A.2 Algorithms

Theorem 3.1 (Bayesian inference in context for local classes)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions, let S be a context of size m, and let x be a query.*

(a) Without noise. *If $\mathrm{VS}(S) \neq \emptyset$, then $p_S(x)$ and $\mathsf{B}(S, x)$ can be computed exactly and deterministically in time $\mathrm{poly}(N, m) \cdot 2^{\kappa} \cdot T_\kappa(N)$, and the Gibbs predictor can be run exactly in the same time, given uniform random integers. This is $\mathrm{poly}(m) \cdot 2^{o(N)}$ when $\kappa = o((\log N)^{1/4}/\sqrt{\log\log N})$.*

(b) With noise. *Let $\eta = a/b$ for integers $0 < a$, $2a < b$, and let $s = \max\{4\kappa(m+1), m+2, \kappa^2\}$. Then $p_S(x)$ and $\mathsf{B}(S, x)$ can be computed exactly in time $\mathrm{poly}(N, m, \log b) \cdot 2^{O(s/\sqrt{\log s})}$, and the Gibbs predictor can be run in the same time, exactly given exact samples from explicitly given distributions on $\{0, \ldots, m\}$, and otherwise within statistical distance ν for any $\nu \geq 2^{-\mathrm{poly}(N,m)}$. For $m \geq \max\{\kappa, 2\}$ the time is $\mathrm{poly}(N, m, \log b) \cdot 2^{O(\kappa m/\sqrt{\log m})}$.*

Proof of Theorem 3.1. ⟨1⟩1. LET: S^+ be S followed by the example $(x, 1)$. Then $\omega_{S^+}(w) = \omega_S(w)\,\mu^{[h_w(x) \neq 1]}$ for every w.

PROOF: The error count of w on S^+ is $e_S(w) + [h_w(x) \neq 1]$.

⟨1⟩2. Part (a) holds.

⟨2⟩1. $p_S(x) = |\mathrm{VS}(S^+)|/|\mathrm{VS}(S)|$.

PROOF: For $\eta = 0$ the posterior is uniform on $\mathrm{VS}(S)$, and the descriptions in $\mathrm{VS}(S)$ with $h_w(x) = 1$ are exactly those in $\mathrm{VS}(S^+)$.

⟨2⟩2. $p_S(x)$ and $\mathsf{B}(S,x)$ can be computed in time $\mathrm{poly}(N,m)\cdot 2^{\kappa}\cdot T_\kappa(N)$.

PROOF: Imported Theorem 2.8 computes both counts, for samples of sizes m and $m+1$. Both are integers below 2^{N+1}, so the quotient and its comparison with $1/2$ take polynomial time.

⟨2⟩3. The Gibbs predictor runs exactly in the same time.

PROOF: Imported Theorem 2.8 draws a uniform element w of $\mathrm{VS}(S)$, which is a draw from π_S, and $h_w(x)$ is computable in polynomial time.

⟨2⟩4. If $\kappa = o((\log N)^{1/4}/\sqrt{\log\log N})$, the time is $\mathrm{poly}(m)\cdot 2^{o(N)}$.

PROOF: Then $\kappa^2 g(N) = \kappa^2 N\log\log N/\sqrt{\log N} = o(N)$ and $2^\kappa = 2^{o(N)}$, and $\mathrm{poly}(N) = 2^{o(N)}$.

⟨2⟩5. Q.E.D.

⟨1⟩3. Part (b) holds.

⟨2⟩1. LET: , for $j\in\{0,\dots,m\}$, a_j and b_j be the numbers of descriptions w with $e_S(w)=j$ and $h_w(x)=1$, respectively $h_w(x)=0$, and LET: $b_{-1}=a_{m+1}=0$. LET: c_j and c_j^+ be the coefficients of V^j in E_S and E_{S^+}. Then $c_j = a_j + b_j$ for $0\le j\le m$ and $c_j^+ = a_j + b_{j-1}$ for $0\le j\le m+1$.

PROOF: By ⟨1⟩1 with V in place of μ: a description with $h_w(x)=1$ contributes $V^{e_S(w)}$ to E_{S^+}, and one with $h_w(x)=0$ contributes $V^{e_S(w)+1}$.

⟨2⟩2. The numbers a_j, b_j for $0\le j\le m$ are determined by the coefficients: $a_j = c_j^+ - b_{j-1}$ and $b_j = c_j - a_j$, in the order $j=0,1,\dots,m$.

PROOF: By ⟨2⟩1 and induction on j, starting from $b_{-1}=0$.

⟨2⟩3. $p_S(x) = \sum_{j=0}^m a_j\mu^j / \sum_{j=0}^m c_j\mu^j$, and the denominator is positive.

PROOF: $\sum_{w:h_w(x)=1}\omega_S(w) = \sum_j a_j\mu^j$ and $Z_S = \sum_j c_j\mu^j$. Since $\mu>0$ and $\sum_j c_j = 2^N$, $Z_S>0$.

⟨2⟩4. $p_S(x)$ and $\mathsf{B}(S,x)$ can be computed in time $\mathrm{poly}(N,m,\log b)\cdot 2^{O(s/\sqrt{\log s})}$.

PROOF: Imported Theorem 2.9 computes E_S and E_{S^+}; the parameter of the second is s, and that of the first is at most s. With $\mu = a/(b-a)$, multiplying numerator and denominator by $(b-a)^m$ turns them into the integers $\sum_j a_j a^j(b-a)^{m-j}$ and $\sum_j c_j a^j(b-a)^{m-j}$, which have $O(N + m\log b)$ bits, since $a_j, c_j \le 2^N$. Computing them, the quotient and its comparison with $1/2$ takes time $\mathrm{poly}(N,m,\log b)$.

⟨2⟩5. The Gibbs predictor runs in time $\mathrm{poly}(N,m,\log b)\cdot 2^{O(s/\sqrt{\log s})}$, exactly given exact samples from explicitly given distributions on $\{0,\dots,m\}$, and otherwise within statistical distance ν.

PROOF: Apply Imported Theorem 2.10 with $\lambda = \ln(1/\mu) > 0$. It samples w with probability proportional to $e^{-\lambda e_S(w)} = \mu^{e_S(w)} = \omega_S(w)$, that is, from π_S; output $h_w(x)$. The distributions on $\{0,\dots,m\}$ that it samples from have weights $c_j\mu^j$, or, after scaling by $(b-a)^m$, the integers $c_j a^j(b-a)^{m-j}$ with $O(N+m\log b)$ bits, and computing them adds a factor $\mathrm{poly}(N,m,\log b)$.

⟨2⟩6. If $m\ge\max\{\kappa,2\}$ then $s/\sqrt{\log s}\le 8\kappa m/\sqrt{\log m}$.

PROOF: Here $4\kappa(m+1)\le 8\kappa m$, $m+2\le 2m\le 8\kappa m$ and $\kappa^2\le\kappa m$, so $s\le 8\kappa m$, and $s\ge m$ gives $\log s\ge\log m$.

⟨2⟩7. Q.E.D.

⟨1⟩4. Q.E.D.

□

Corollary 3.2 (Bayesian inference from counting)**.** *If #κ-SAT on N variables with M clauses can be solved in time $T(N, M)$, with T nondecreasing, then noiseless Bayesian inference in context for every κ-local class with N-bit descriptions takes time $2T(N, (m+1)2^\kappa) + \mathrm{poly}(N, m) \cdot 2^\kappa$ on consistent contexts of size m.*

Proof of Corollary 3.2. ⟨1⟩1. $|\mathrm{VS}(S)|$ and $|\mathrm{VS}(S^+)|$ are the numbers of satisfying assignments of κ-CNFs Γ_S and Γ_{S^+} on N variables with at most $(m+1)2^\kappa$ clauses each, computable in time $\mathrm{poly}(N, m) \cdot 2^\kappa$.

PROOF: Imported Theorem 2.7 for the samples S and S^+.

⟨1⟩2. Q.E.D.

PROOF: Count both with the assumed algorithm, in time $T(N, (m+1)2^\kappa)$ each because T is nondecreasing, and output the quotient, which is $p_S(x)$ by ⟨2⟩1 of the proof of Theorem 3.1. □

A.3 The Bayes risk

Theorem 3.3 (Bayes risk)**.** *Let $\mathcal{H} = \{h_w : w \in \{0,1\}^N\}$ be any class, $\mathcal{D}$ any input distribution and $\eta \in [0, 1/2)$ the noise rate. Let $\mathrm{h}(\eta) = -\eta \log \eta - (1-\eta)\log(1-\eta)$, with $\mathrm{h}(0) = 0$.*

(a) $R^*_{m+1} \le R^*_m$ for every $m \ge 0$.

(b) $R^*_m \le \eta + \dfrac{(1-2\eta)N}{(1-\mathrm{h}(\eta))(m+1)}$ for every $m \ge 0$. In particular, without noise $R^*_m \le N/(m+1)$.

(c) Without noise, the Gibbs predictor has risk at most $2R^*_m \le 2N/(m+1)$.

Proof of Theorem 3.3. ⟨1⟩1. LET: an infinite episode draw w uniformly, inputs $x_1, x_2, \ldots$ independently from $\mathcal{D}$ and noise bits $\xi_1, \xi_2, \ldots$, with $y_t = h_w(x_t) \oplus \xi_t$, and LET: S_t be its first t examples. Let (x, y) be one more example of the same task, independent of the sequence given w. Then $R^*_t = \Pr[\mathsf{B}(S_t, x_{t+1}) \ne y_{t+1}] = \Pr[\mathsf{B}(S_t, x) \ne y]$ for every $t \ge 0$.

PROOF: $(w, S_t, x_{t+1}, y_{t+1})$ and (w, S_t, x, y) both have the distribution of an episode of length t with its task.

⟨1⟩2. Part (a) holds.

⟨2⟩1. LET: $q_t = \Pr[y = 1 \mid S_t, x]$. Then $q_t = \hat{p}_{S_t}(x)$ and $R^*_t = \mathbb{E}[\min\{q_t, 1-q_t\}]$.

PROOF: By ⟨1⟩1 and Lemma 2.5.

⟨2⟩2. $\mathbb{E}[q_{t+1} \mid S_t, x] = q_t$.

PROOF: The tower property, since (S_t, x) is a function of (S_{t+1}, x).

⟨2⟩3. $\mathbb{E}[\min\{q_{t+1}, 1-q_{t+1}\} \mid S_t, x] \le \min\{q_t, 1-q_t\}$.

PROOF: The function $q \mapsto \min\{q, 1-q\}$ is concave; apply the conditional Jensen inequality and ⟨2⟩2.

⟨2⟩4. Q.E.D.

PROOF: Take expectations in ⟨2⟩3 and use ⟨2⟩1.

⟨1⟩3. Without noise, in every infinite episode the predictions $\hat{y}_t = \mathsf{B}(S_{t-1}, x_t)$ are wrong for at most N indices $t \ge 1$.

⟨2⟩1. $|\mathrm{VS}(S_0)| = 2^N$, $\mathrm{VS}(S_t) \subseteq \mathrm{VS}(S_{t-1})$, and $w \in \mathrm{VS}(S_t)$ for all t.

PROOF: Without noise every label is $h_w(x_t)$.

⟨2⟩2. If $\hat{y}_t \ne y_t$, then $|\mathrm{VS}(S_t)| \le |\mathrm{VS}(S_{t-1})|/2$.

PROOF: Let $p = p_{S_{t-1}}(x_t)$, the fraction of $\mathrm{VS}(S_{t-1})$ that labels x_t positive. If $y_t = 1$ and $\hat{y}_t = 0$, then $p < 1/2$ and $|\mathrm{VS}(S_t)| = p\,|\mathrm{VS}(S_{t-1})|$. If $y_t = 0$ and $\hat{y}_t = 1$, then $p \ge 1/2$ and

$|\mathrm{VS}(S_t)| = (1-p)|\mathrm{VS}(S_{t-1})|$.

⟨2⟩3. Q.E.D.

PROOF: After r mistakes, $1 \le |\mathrm{VS}(S_t)| \le 2^{N-r}$ by ⟨2⟩1 and ⟨2⟩2, so $r \le N$.

⟨1⟩4. With noise rate $\eta > 0$, $\sum_{t=1}^{m+1}(R^*_{t-1} - \eta) \le (1-2\eta)N/(1-\mathrm{h}(\eta))$.

⟨2⟩1. LET: $P_v(y_{1..T}) = \prod_{t \le T}(1-\eta)^{[y_t = h_v(x_t)]}\eta^{[y_t \ne h_v(x_t)]}$ for each description v, given the inputs, and $Q = 2^{-N}\sum_v P_v$. Then $Q(y_{1..t})/Q(y_{1..t-1})$ is $\hat{p}_t$ if $y_t = 1$ and $1-\hat{p}_t$ if $y_t = 0$, where $\hat{p}_t = \hat{p}_{S_{t-1}}(x_t)$.

PROOF: $Q(y_{1..t})/Q(y_{1..t-1})$ is the conditional probability of y_t given the inputs and $y_{1..t-1}$ in the mixture, which is the predictive probability of ⟨1⟩1 of the proof of Lemma 2.5.

⟨2⟩2. $\sum_{t=1}^{m+1} \log \frac{P_w(y_t|x_t)}{Q(y_t|S_{t-1},x_t)} \le N$ in every episode, where $P_w(y_t \mid x_t)$ is the t-th factor of P_w and $Q(y_t \mid S_{t-1}, x_t)$ is the quotient of ⟨2⟩1.

PROOF: The sum telescopes to $\log(P_w(y_{1..m+1})/Q(y_{1..m+1}))$, and $Q \ge 2^{-N}P_w$.

⟨2⟩3. LET: $q_t = (1-\eta)h_w(x_t) + \eta(1-h_w(x_t))$ and $\mathrm{KL}_t = \mathrm{KL}(\mathrm{Bern}(q_t) \,\|\, \mathrm{Bern}(\hat{p}_t))$ in bits. Then $\sum_{t=1}^{m+1}\mathbb{E}[\mathrm{KL}_t] \le N$.

PROOF: Given w, all inputs and $y_{1..t-1}$, the label y_t is 1 with probability q_t, so the conditional expectation of the t-th term of ⟨2⟩2 is KL_t. All terms are finite because $\hat{p}_t \in [\eta, 1-\eta]$.

⟨2⟩4. If $\hat{y}_t \ne h_w(x_t)$, then $\mathrm{KL}_t \ge 1 - \mathrm{h}(\eta)$.

PROOF: For fixed q, the divergence $\mathrm{KL}(\mathrm{Bern}(q)\|\mathrm{Bern}(p))$ decreases in p on $(0, q]$ and increases on $[q, 1)$. At $p = 1/2$ it equals $1 - \mathrm{h}(q)$, and $\mathrm{h}(q) = \mathrm{h}(\eta)$ for $q \in \{\eta, 1-\eta\}$. If $h_w(x_t) = 1$ and $\hat{y}_t = 0$, then $q_t = 1 - \eta > 1/2 > \hat{p}_t$. If $h_w(x_t) = 0$ and $\hat{y}_t = 1$, then $q_t = \eta < 1/2 \le \hat{p}_t$. In both cases $1/2$ lies between q_t and $\hat{p}_t$.

⟨2⟩5. $\Pr[\hat{y}_t \ne y_t] = \eta + (1-2\eta)\Pr[\hat{y}_t \ne h_w(x_t)]$.

PROOF: Given w, S_{t-1} and x_t, the noise bit ξ_t is independent and $\hat{y}_t$ is fixed; the prediction is wrong with probability η if $\hat{y}_t = h_w(x_t)$ and $1-\eta$ otherwise.

⟨2⟩6. Q.E.D.

PROOF: By ⟨1⟩1, $R^*_{t-1} = \Pr[\hat{y}_t \ne y_t]$. By ⟨2⟩5 and ⟨2⟩4, $R^*_{t-1} - \eta \le (1-2\eta)\mathbb{E}[\mathrm{KL}_t]/(1-\mathrm{h}(\eta))$; sum and use ⟨2⟩3.

⟨1⟩5. Part (b) holds.

PROOF: Without noise, ⟨1⟩3 and ⟨1⟩1 give $\sum_{t=1}^{m+1} R^*_{t-1} \le N$, which is the case $\eta = 0$ of the inequality of ⟨1⟩4. In both cases, part (a) gives $R^*_m \le R^*_{t-1}$ for $t \le m+1$, so $(m+1)(R^*_m - \eta) \le (1-2\eta)N/(1-\mathrm{h}(\eta))$.

⟨1⟩6. Part (c) holds.

PROOF: Without noise, let $p = p_S(x)$ for the context S and query x of an episode. Given S and x, the task and the description drawn by G are independent draws from π_S, so G errs with probability $2p(1-p) \le 2\min\{p, 1-p\}$, and B errs with probability $\min\{p, 1-p\}$ by Lemma 2.5. Take expectations and use part (b).

⟨1⟩7. Q.E.D.

□

B Proofs for Section 4

B.1 The gadgets

Lemma 4.1 (Bayes gadget)**.** *Let $k \geq 1$, and let F be a k-CNF on $z_1, \dots, z_n$ with M clauses, none of them tautological. Let Γ_F be the CNF on $z_1, \dots, z_n, a, u$ with the clauses*

$$\neg a \vee C \quad (C \in F), \qquad a \vee \neg z_i \quad (i \in [n]), \qquad a \vee \neg u.$$

Then Γ_F is a $(k+1)$-CNF with $M + n + 1$ clauses, none of them tautological. It has $2\#F + 1$ satisfying assignments, and exactly $2\#F$ of them set $a = 1$.

Proof of Lemma 4.1. ⟨1⟩1. Γ_F is a $(k+1)$-CNF with $M + n + 1$ clauses, none of them tautological.
PROOF: A clause $\neg a \vee C$ has width $|C| + 1 \leq k + 1$ and is tautological only if C is, because a does not occur in C. The clauses $a \vee \neg z_i$ and $a \vee \neg u$ have width $2 \leq k + 1$ and are not tautological. The $M + n + 1$ clauses are distinct.

⟨1⟩2. The satisfying assignments with $a = 1$ are the assignments $(z, 1, u)$ with z satisfying F and $u \in \{0, 1\}$; there are $2\#F$ of them.
PROOF: With $a = 1$ the clauses $a \vee \neg z_i$ and $a \vee \neg u$ hold and $\neg a \vee C$ is equivalent to C.

⟨1⟩3. Exactly one satisfying assignment has $a = 0$, namely $z = 0^n$, $u = 0$.
PROOF: With $a = 0$ the clauses $\neg a \vee C$ hold, $a \vee \neg z_i$ forces $z_i = 0$, and $a \vee \neg u$ forces $u = 0$.

⟨1⟩4. Q.E.D.

□

Lemma 4.2 (Bayes contexts)**.** *Let $k \geq 2$, and let F be a k-CNF on n variables with M clauses, none of them tautological. One can compute in polynomial time:*

(a) a context S_F^{LD} of $M + n + 1$ positive examples of LD_{k+1} in dimension $n + 2$, and
(b) a context S_F^{BP} of $M + n + k$ positive examples of BP_{2k} in dimension $n + k + 1$,

both consistent, such that the query e_a has posterior predictive $p_S(e_a) = 2\#F/(2\#F + 1)$ in both. In particular $p_S(e_a) = 0$ if F is unsatisfiable, $p_S(e_a) \geq 2/3$ otherwise, and $\#F = p_S(e_a)/(2(1 - p_S(e_a)))$.

Proof of Lemma 4.2. ⟨1⟩1. Part (a) holds, with a bijection between $\mathrm{VS}(S_F^{\mathsf{LD}})$ and the satisfying assignments of Γ_F that preserves the value of a.
PROOF: Index the $n + 2$ coordinates by $z_1, \dots, z_n, a, u$. For each clause D of Γ_F, let x_D have entry $+1$ at the variables that occur positively in D, -1 at those that occur negatively, and 0 elsewhere; it is well defined because D is not tautological, and $|\mathrm{supp}(x_D)| = |D| \leq k + 1$. By Example 2.2(a), $h_\sigma(x_D) = 1$ if and only if the assignment with value $(1 + \sigma_i)/2$ at coordinate i satisfies D. So $\sigma \mapsto ((1 + \sigma_i)/2)_i$ maps $\mathrm{VS}(S_F^{\mathsf{LD}})$ bijectively onto the satisfying assignments of Γ_F, and the coordinate of a is $+1$ exactly when $a = 1$.

⟨1⟩2. Part (b) holds, with a bijection between $\mathrm{VS}(S_F^{\mathsf{BP}})$ and the satisfying assignments of Γ_F that preserves the value of a.
PROOF: Γ_F is a $(k+1)$-CNF without tautological clauses on $n + 2$ variables (Lemma 4.1). Imported Theorem 2.11 with $k + 1 \geq 3$ in place of k gives $p = k - 1$ padding coordinates, $M + n + 1 + (k - 1) = M + n + k$ examples in dimension $n + 2 + k - 1 = n + k + 1$, all labeled 1, with supports of size at most $2(k + 1) - 2 = 2k$. A sign vector is consistent if and only if its padding coordinates are 1 and the assignment $\mathbf{1}[\sigma_i = 1]$ of the first $n + 2$ coordinates satisfies Γ_F; this is a bijection, and it reads a off the coordinate of a.

⟨1⟩3. In both contexts, $h(e_a) = 1$ exactly for the descriptions whose assignment has $a = 1$.

PROOF: For literal disjunctions, $\mathrm{supp}(e_a) = \{a\}$, so $h_\sigma(e_a) = 1$ if and only if $\sigma_a = 1$. For binarized perceptrons, $\langle \sigma, e_a \rangle = \sigma_a \geq 0$ if and only if $\sigma_a = 1$.

⟨1⟩4. $p_S(e_a) = 2\#F/(2\#F + 1)$ in both contexts, and both are consistent.

PROOF: Without noise the posterior is uniform on the version space, which by ⟨1⟩1 and ⟨1⟩2 is in bijection with the $2\#F + 1 \geq 1$ satisfying assignments of Γ_F, of which $2\#F$ have $a = 1$ (Lemma 4.1). Apply ⟨1⟩3.

⟨1⟩5. Q.E.D.

PROOF: If $\#F = 0$ the predictive is 0, and if $\#F \geq 1$ it is $2\#F/(2\#F + 1) \geq 2/3$. Solving $p = 2\#F/(2\#F + 1)$ gives $\#F = p/(2(1 - p))$. The constructions take polynomial time.

□

Lemma 4.3 (Bayes contexts by length)**.** *Given a CNF F of length $L \geq 1$, one can compute in polynomial time a consistent context of positive examples of LD_3 in dimension $N \leq 3L + 2$, with at most $\min\{5L + 1, 36N^2\}$ examples, such that the query e_a has posterior predictive 0 if F is unsatisfiable and at least $2/3$ otherwise. The same holds for BP_4 in dimension $N + 1$, with one more example.*

Proof of Lemma 4.3. ⟨1⟩1. One can compute in polynomial time a 3-CNF F' without tautological clauses, with $n \leq 2L$ variables and $M \leq L$ clauses, that is satisfiable if and only if F is.

PROOF: Delete the tautological clauses of F and the variables that no longer occur; this changes neither satisfiability nor the bounds. Replace each clause $\ell_1 \vee \cdots \vee \ell_r$ with $r \geq 4$ by the $r - 2$ clauses $\ell_1 \vee \ell_2 \vee t_1$, $\neg t_1 \vee \ell_3 \vee t_2$, …, $\neg t_{r-3} \vee \ell_{r-1} \vee \ell_r$ with $r - 3$ fresh variables. An assignment that satisfies the clause extends to one that satisfies the chain: set $t_j = 1$ for the indices j before the clause of the chain that contains the first true literal, and $t_j = 0$ after it. If every ℓ_i is false, the chain forces $t_1 = 1$, then $t_2 = 1$, and so on, and its last clause fails. The occurring variables number at most L, the fresh ones at most $\sum_C \max\{0, |C| - 3\} \leq L$, and the clauses at most $\sum_C \max\{1, |C| - 2\} \leq L$. No new clause is tautological.

⟨1⟩2. LET: Γ' be the CNF on the variables z of F', one fresh variable y_C for each clause C of F' of width 3, and two more variables a, u, with the clauses: $\neg a \vee C$ for each clause C of F' of width at most 2; $\neg a \vee \ell_1 \vee y_C$, $\neg y_C \vee \ell_2 \vee \ell_3$ and $a \vee \neg y_C$ for each clause $C = \ell_1 \vee \ell_2 \vee \ell_3$; $a \vee \neg z$ for each variable z of F'; and $a \vee \neg u$. Then Γ' is a 3-CNF without tautological clauses on $N \leq n + M + 2 \leq 3L + 2$ variables with at most $\min\{5L + 1, 36N^2\}$ clauses.

PROOF: The fresh variables are distinct, and F' has no tautological clause. Let F_0 be F without its tautological clauses, $L_0 \leq L$ its length, and $n_0 \leq N$ the number of its variables. The bounds of ⟨1⟩1 hold with L_0 in place of L, so Γ' has at most $3M + n + 1 \leq 5L_0 + 1$ clauses. The clauses of F_0 of width at most 2 are distinct and not tautological, so there are at most $2n_0 + 4\binom{n_0}{2} \leq 2N^2$ of them, of total length at most $4N^2$. A clause of F_0 of width $r \geq 3$ gives Γ' at least $r/3$ variables of its own: one variable y_C if $r = 3$, and $r - 3$ chain variables and $r - 2$ variables y_C if $r \geq 4$. So these clauses have total length at most $3N$. Hence $L_0 \leq 4N^2 + 3N \leq 7N^2$ and $5L_0 + 1 \leq 36N^2$.

⟨1⟩3. Exactly one satisfying assignment of Γ' has $a = 0$.

PROOF: With $a = 0$ the clauses $a \vee \neg z$, $a \vee \neg y_C$ and $a \vee \neg u$ force all other variables to 0. The all-zero assignment of them satisfies $\neg a \vee \ldots$ through $\neg a$ and $\neg y_C \vee \ell_2 \vee \ell_3$ through $\neg y_C$.

⟨1⟩4. The number A of satisfying assignments of Γ' with $a = 1$ is 0 if F' is unsatisfiable and at least 2 otherwise.

PROOF: With $a = 1$, the clauses with the literal a hold, $\neg a \vee C$ reduces to C, and the two

clauses of a clause C of width 3 reduce to $(\ell_1 \vee y_C) \wedge (\neg y_C \vee \ell_2 \vee \ell_3)$. For a fixed z, this has a solution y_C if and only if z satisfies C: if ℓ_1 is true, $y_C = 0$ works; if ℓ_1 is false, y_C must be 1, which works if and only if $\ell_2 \vee \ell_3$ holds. So the assignments with $a = 1$ exist only for z satisfying F', each such z has at least one choice of every y_C, and u is free.

⟨1⟩5. The positive examples of LD_3 for the clauses of Γ' form a consistent context in dimension N with at most $\min\{5L+1, 36N^2\}$ examples, and the query e_a has posterior predictive $A/(A+1)$, which is 0 if F is unsatisfiable and at least $2/3$ otherwise.

PROOF: As in ⟨1⟩1 and ⟨1⟩3–⟨1⟩4 of the proof of Lemma 4.2, the version space is in bijection with the satisfying assignments of Γ', and e_a is labeled by a. Apply ⟨1⟩3, ⟨1⟩4 and ⟨1⟩1.

⟨1⟩6. Q.E.D.

PROOF: For BP_4, apply Imported Theorem 2.11 to Γ' with $k = 3$: one padding coordinate and one more example, supports of size at most 4, and the same bijection, as in ⟨1⟩2 of the proof of Lemma 4.2. All steps take polynomial time.

□

B.2 Lower bounds

Theorem 4.4 (Hardness of Bayesian inference in context)**.** *(a) Assume Hypothesis 2.14. For every $\gamma \in (0,1)$ there are K and c_γ such that no randomized algorithm with error probability at most $1/3$ does the following in time $O(2^{(1-\gamma)N})$: given a consistent context of at most N^{c_γ} positive examples of $\mathsf{BP}_{\lceil K \log N \rceil}$ in dimension N and a query, output a number within $1/4$ of the posterior predictive. The same holds for $\mathsf{LD}_{\lceil K \log N \rceil}$, for computing $\mathsf{B}(S,x)$, and for running the Gibbs predictor exactly or within statistical distance $1/6$. Under Hypothesis 2.15, there are $K, c_\gamma, \rho > 0$ with the same statements for time $O(2^{\rho N})$.*

(b) Assume $\mathrm{ETH}^{\mathrm{len}}_C$, and let $\theta > 0$. No randomized algorithm with error probability at most $1/3$ outputs, for every consistent context of at most $36N^2$ positive examples of LD_3 in dimension N and every query, a number within $1/4$ of the posterior predictive in time $2^{N/(\log N)^{C+\theta}}$. The same holds for BP_4, for computing $\mathsf{B}(S,x)$, and for running the Gibbs predictor exactly or within statistical distance $1/6$.

Proof of Theorem 4.4. ⟨1⟩1. If an algorithm $\mathcal{A}$ outputs a number within $1/4$ of $p_S(e_a)$, with error probability at most $1/3$, on a context S of Lemma 4.2 or Lemma 4.3, then one run of $\mathcal{A}$ decides the satisfiability of F with error probability at most $1/3$. The same holds for an algorithm that computes $\mathsf{B}(S, e_a)$, and for $O(1)$ runs of the Gibbs predictor, exact or within statistical distance $1/6$.

PROOF: The predictive is 0 if F is unsatisfiable and at least $2/3$ otherwise, so a number within $1/4$ of it is at most $1/4$ in the first case and at least $5/12$ in the second; answer "satisfiable" when it exceeds $1/3$. $\mathsf{B}(S, e_a) = 1$ exactly when the predictive is at least $1/2$, that is, when F is satisfiable. A run of the Gibbs predictor outputs 1 with probability at most $1/6$ in the first case and at least $1/2$ in the second; twelve runs and the answer "satisfiable" when at least four of them output 1 err with probability less than $1/7$ in the first case and at most $299/4096 < 1/13$ in the second.

⟨1⟩2. Part (a) holds under Hypothesis 2.14.

⟨2⟩1. LET: K', c' be given by Imported Theorem 2.12 for $\gamma/2$, and LET: $K = 2K'$ and $c_\gamma = c' + 1$.
ASSUME: an algorithm $\mathcal{A}$ that does what part (a) excludes, in time $O(2^{(1-\gamma)N})$.

⟨2⟩2. Given a k-CNF F on n variables with $k \le K' \log n$ and at most $n^{c'}$ clauses, with n large, LET: $k_0 = \max\{2, \lfloor K' \log n \rfloor\}$, delete the tautological clauses of F, and build S_F^{BP} by

Lemma 4.2 with k_0 in place of k. Then S_F^{BP} is a consistent context of at most N^{c_γ} positive examples of $\mathsf{BP}_{\lceil K \log N \rceil}$ in dimension $N = n + k_0 + 1$.

PROOF: F is a k_0-CNF. The supports have size at most $2k_0 \le 2K' \log n \le K \log N$, and the context has at most $n^{c'} + n + k_0 \le N^{c'+1}$ examples for large n.

⟨2⟩3. Running $\mathcal{A}$ on (S_F^{BP}, e_a) decides F in time $O(2^{(1-\gamma/2)n})$ with error probability at most $1/3$.

PROOF: By ⟨1⟩1. The time is $\mathrm{poly}(n) + O(2^{(1-\gamma)(n+K' \log n+1)}) = O(2^{(1-\gamma)n} n^{K'})$, which is $O(2^{(1-\gamma/2)n})$.

⟨2⟩4. The same argument applies to $\mathsf{LD}_{\lceil K \log N \rceil}$ with S_F^{LD}, of dimension $n+2$ and supports of size at most $k_0 + 1 \le K \log N$, and to B and the Gibbs predictor by ⟨1⟩1.

⟨2⟩5. Q.E.D.

PROOF: ⟨2⟩3 contradicts Imported Theorem 2.12.

⟨1⟩3. Part (a) holds under Hypothesis 2.15.

PROOF: Imported Theorem 2.12 gives K', c', ρ' for time $O(2^{\rho' n})$. With $K = 2K'$, $c_\gamma = c' + 1$ and $\rho = \rho'/2$, the construction of ⟨1⟩2 turns an algorithm in time $O(2^{\rho N})$ into one in time $O(2^{\rho n} n^{K'}) = O(2^{\rho' n})$.

⟨1⟩4. Part (b) holds.

⟨2⟩1. ASSUME: an algorithm $\mathcal{A}$ that, for all sufficiently large N, does what part (b) excludes in time $2^{f(N)}$, where $f(N) = N/(\log N)^{C+\theta}$.

⟨2⟩2. Given a CNF F of length L, build the context of Lemma 4.3, of dimension $N_F \le 3L + 2$, and run $\mathcal{A}$. This decides the satisfiability of F with error probability at most $1/3$.

PROOF: By ⟨1⟩1.

⟨2⟩3. The time is at most $2^{L/(\log L)^C}$ for all sufficiently large L.

PROOF: The context has dimension $N_F \le 3L + 2$ and at most $36 N_F^2$ examples, so it is an input of the kind that $\mathcal{A}$ handles. Below any fixed dimension there are only finitely many such contexts, because their size is bounded, so on them $\mathcal{A}$ takes time $O(1)$. The function f is nondecreasing for $N \ge e^{C+\theta}$. For large L, $f(3L+2) \le 4L/(\log L)^{C+\theta} \le L/(2(\log L)^C)$ once $(\log L)^\theta \ge 8$, so the time is $\mathrm{poly}(L) + O(1) + 2^{L/(2(\log L)^C)} \le 2^{L/(\log L)^C}$.

⟨2⟩4. The same holds for BP_4 in dimension $N_F + 1$, and for B and the Gibbs predictor.

PROOF: By Lemma 4.3 and ⟨1⟩1. The BP_4 context has dimension $N_F + 1$ and at most $36N_F^2 + 1 \le 36(N_F + 1)^2$ examples, and the extra dimension changes $f(3L+2)$ to $f(3L+3)$.

⟨2⟩5. Q.E.D.

PROOF: ⟨2⟩2–⟨2⟩4 contradict $\mathrm{ETH}_C^{\mathrm{len}}$.

⟨1⟩5. Q.E.D.

□

C Proofs for Section 5

C.1 Halfspaces in context

Proposition 5.1 (Halfspace forms)**.** *Let $\sigma \in \{\pm 1\}^n$ and $x \in \{-1, 0, 1\}^n$ with $r = |\mathrm{supp}(x)|$. Then $h_\sigma^{\mathsf{BP}}(x) = \mathbf{1}[\langle \sigma, x \rangle \ge 0]$ and $h_\sigma^{\mathsf{LD}}(x) = \mathbf{1}[\langle \sigma, x \rangle + r - 2 \ge 0]$. If $h_\sigma^{\mathsf{BP}}(x) = 0$ then $\langle \sigma, x \rangle \le -1$, and if $h_\sigma^{\mathsf{LD}}(x) = 0$ then $\langle \sigma, x \rangle + r - 2 = -2$.*

Proof of Proposition 5.1. ⟨1⟩1. The statements for BP hold.

PROOF: $h_\sigma^{\mathsf{BP}}(x) = \mathbf{1}[\langle \sigma, x\rangle \geq 0]$ is the definition, and $\langle \sigma, x\rangle$ is an integer, so it is at most -1 when it is negative.

⟨1⟩2. LET: t be the number of $i \in \mathrm{supp}(x)$ with $x_i = \sigma_i$. Then $\langle \sigma, x\rangle = 2t - r$.
PROOF: Each $i \in \mathrm{supp}(x)$ contributes $\sigma_i x_i = +1$ if $x_i = \sigma_i$ and -1 otherwise.

⟨1⟩3. Q.E.D.
PROOF: By definition $h_\sigma^{\mathsf{LD}}(x) = \mathbf{1}[t \geq 1]$. By ⟨1⟩2, $t \geq 1$ if and only if $\langle \sigma, x\rangle + r - 2 = 2t - 2 \geq 0$, and $t = 0$ gives -2.

□

Theorem 5.2 (Linear programming in context)**.** *For every $k \leq n$, the in-context learner A_{LP} for BP_k and for LD_k runs in time $\mathrm{poly}(n, m)$ and is (ε, δ)-PAC from $m_{\mathrm{VC}}(n+1, \varepsilon, \delta)$ examples, which is $O((n \log(1/\varepsilon) + \log(1/\delta))/\varepsilon)$. Its risk in every noiseless episode, for every input distribution, is $O(n \log m/m)$.*

Proof of Theorem 5.2. ⟨1⟩1. If the context is consistent with σ, then $v = \sigma$ satisfies (2).
PROOF: By Proposition 5.1: a positive example has $\langle \sigma, x_i\rangle + c(x_i) \geq 0$, and a negative one has $\langle \sigma, x_i\rangle \leq -1$ with $c = 0$ for BP, and $\langle \sigma, x_i\rangle + c(x_i) = -2$ for LD.

⟨1⟩2. Every solution v of (2) classifies every example of the context correctly.
PROOF: Positive examples get value at least 0 and negative ones at most $-1 < 0$.

⟨1⟩3. LET: $\mathcal{G} = \{x \mapsto \mathbf{1}[\langle v, x\rangle + c(x) \geq 0] : v \in \mathbb{R}^n\}$. It contains every h_σ and has VC dimension at most $n + 1$.
PROOF: It contains h_σ by Proposition 5.1. Every element is $x \mapsto \mathbf{1}[\langle (v, 1), (x, c(x))\rangle \geq 0]$, the restriction of a homogeneous halfspace in $\mathbb{R}^{n+1}$ to the image of $x \mapsto (x, c(x))$, and homogeneous halfspaces in $\mathbb{R}^{n+1}$ have VC dimension $n + 1$.

⟨1⟩4. A_{LP} runs in time $\mathrm{poly}(n, m)$.
PROOF: (2) is a linear program with n variables and m constraints whose coefficients are in $\{-1, 0, 1\}$ and whose right-hand sides are integers of absolute value at most $n + 2$. Linear programming in polynomial time decides feasibility and finds a rational solution.

⟨1⟩5. A_{LP} is (ε, δ)-PAC from $m_{\mathrm{VC}}(n+1, \varepsilon, \delta)$ examples.
PROOF: In the realizable case, ⟨1⟩1 and ⟨1⟩2 show that the output is an element of $\mathcal{G}$ consistent with the context. Apply Imported Theorem 2.13(a) with ⟨1⟩3.

⟨1⟩6. Q.E.D.
PROOF: For the risk, let $\varepsilon_m = C_B(n+2)\log(2m)/m$ and $\delta_m = 1/m$. If $\varepsilon_m \geq 1$ there is nothing to prove. Otherwise $\varepsilon_m \geq 1/m$, so $\log(2/\varepsilon_m) \leq \log(2m)$ and $m_{\mathrm{VC}}(n+1, \varepsilon_m, \delta_m) \leq C_B(n+2)\log(2m)/\varepsilon_m = m$. By ⟨1⟩5 the risk is at most $\varepsilon_m + \delta_m = O(n \log m/m)$.

□

C.2 Local Fourier features and regression

Theorem 5.4 (Local Fourier features)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions. For $x \in \mathcal{X}$ and $T \subseteq [N]$ with $|T| \leq \kappa$, let*

$$\psi_T(x) = 2^{-|R(x)|} \sum_{\beta \in \{0,1\}^{R(x)}} f_x(\beta)\, \chi_T(\beta) \quad \text{if } T \subseteq R(x), \qquad \psi_T(x) = 0 \quad \text{otherwise,}$$

and let $\Psi(x) = (\psi_T(x))_{|T| \leq \kappa}$ and $\Phi(w) = (\chi_T(w))_{|T| \leq \kappa}$, vectors of dimension D_κ. Then $h_w(x) = \langle \Phi(w), \Psi(x)\rangle$ for all w and x. The vector $\Psi(x)$ has at most $2^{|R(x)|}$ nonzero entries, all in $2^{-\kappa}\mathbb{Z}$, and it can be computed in time $\mathrm{poly}(N, |x|) \cdot \kappa 2^\kappa$. For $1 \leq \kappa \leq N$, $D_\kappa \leq (eN/\kappa)^\kappa$.

Proof of Theorem 5.4. ⟨1⟩1. For every x and every $\beta \in \{0,1\}^{R(x)}$, $f_x(\beta) = \sum_{T \subseteq R(x)} \psi_T(x)\chi_T(\beta)$.

PROOF: This is the Fourier expansion of a function on $\{0,1\}^{R(x)}$ [O'D14, Chapter 1]: the characters χ_T, $T \subseteq R(x)$, are orthonormal for the uniform measure, and $\psi_T(x)$ is the coefficient of χ_T.

⟨1⟩2. $h_w(x) = \langle \Phi(w), \Psi(x) \rangle$.

PROOF: For $T \subseteq R(x)$ we have $\chi_T(w|_{R(x)}) = \chi_T(w)$. So by ⟨1⟩1, $h_w(x) = f_x(w|_{R(x)})$ equals $\sum_{T \subseteq R(x)} \psi_T(x)\chi_T(w)$. Every $T \subseteq R(x)$ has $|T| \le \kappa$, and $\psi_T(x) = 0$ for the other T with $|T| \le \kappa$.

⟨1⟩3. $\Psi(x)$ has at most $2^{|R(x)|}$ nonzero entries, all in $2^{-\kappa}\mathbb{Z}$, and it can be computed in time $\mathrm{poly}(N, |x|) \cdot \kappa 2^\kappa$.

PROOF: Only $T \subseteq R(x)$ give nonzero entries. Each entry is $2^{-|R(x)|}$ times an integer. Compute $R(x)$, the $2^{|R(x)|}$ values of f_x, each in time $\mathrm{poly}(N, |x|)$, and the Walsh–Hadamard transform of the value table in $O(|R(x)|2^{|R(x)|})$ operations.

⟨1⟩4. Q.E.D.

PROOF: For $1 \le \kappa \le N$ and $j \le \kappa$ we have $(\kappa/N)^\kappa \le (\kappa/N)^j$, so $(\kappa/N)^\kappa \sum_{j \le \kappa} \binom{N}{j} \le \sum_{j \le N} \binom{N}{j}(\kappa/N)^j = (1 + \kappa/N)^N \le e^\kappa$.

□

Theorem 5.5 (Regression in context)**.** *Let $\mathcal{H}$ be κ-local with N-bit descriptions.*

(a) $\mathsf{A}_{\mathrm{lin}}$ *is (ε, δ)-PAC from $m_{\mathrm{VC}}(D_\kappa + 1, \varepsilon, \delta)$ examples.*

(b) A_{ℓ_1} *is (ε, δ)-agnostic from $m_{\mathrm{UC}}(D_\kappa + 1, \varepsilon, \delta)$ examples. In an episode with noise rate η, its risk is at most $\eta + \varepsilon + \delta$.*

(c) Both run in time $\mathrm{poly}(N, m, D_\kappa) \cdot 2^\kappa$. This is polynomial for constant κ, it is $N^{O(\log N)}$ for $\kappa = O(\log N)$, and it is $2^{o(N)}$ times a polynomial in m whenever $\kappa \log(eN/\kappa) = o(N)$.

Proof of Theorem 5.5. ⟨1⟩1. LET: $\mathcal{G}_\Psi = \{x \mapsto \mathbf{1}[\langle v, \Psi(x) \rangle \ge \theta] : v \in \mathbb{R}^{D_\kappa}, \theta \in \mathbb{R}\}$. It has VC dimension at most $D_\kappa + 1$, and it contains every h_w.

PROOF: Its elements are affine halfspaces in $\mathbb{R}^{D_\kappa}$ composed with Ψ. By Theorem 5.4, $h_w(x) = \langle \Phi(w), \Psi(x) \rangle \in \{0,1\}$, so $h_w(x) = \mathbf{1}[\langle \Phi(w), \Psi(x) \rangle \ge 1/2]$.

⟨1⟩2. Part (a) holds.

PROOF: In the realizable case $v = \Phi(w)$ solves the system, so Gaussian elimination over $\mathbb{Q}$ finds a solution v. Every solution has $\langle v, \Psi(x_i) \rangle = y_i \in \{0,1\}$, so the output $\mathbf{1}[\langle v, \Psi(\cdot) \rangle \ge 1/2] \in \mathcal{G}_\Psi$ is consistent with the context. Apply Imported Theorem 2.13(a) with ⟨1⟩1.

⟨1⟩3. Part (b) holds.

⟨2⟩1. LET: $\mathcal{D}'$ be a distribution on $\mathcal{X} \times \{0,1\}$, let w^* minimize the error of h_w under $\mathcal{D}'$, and let $\hat{v}$ be the regression solution. Then $\sum_i |\langle \hat{v}, \Psi(x_i) \rangle - y_i| \le \sum_i [h_{w^*}(x_i) \ne y_i]$.

PROOF: $\Phi(w^*)$ is a feasible point with $|\langle \Phi(w^*), \Psi(x_i) \rangle - y_i| = [h_{w^*}(x_i) \ne y_i]$, by Theorem 5.4.

⟨2⟩2. Let $z \in \mathbb{R}$, $y \in \{0,1\}$ and $\bar{z} = \min\{\max\{z, 0\}, 1\}$. If θ is uniform in $[0,1]$, then $\Pr_\theta[\mathbf{1}[z \ge \theta] \ne y] = |\bar{z} - y| \le |z - y|$.

PROOF: If $y = 1$, the error event is $\theta > z$; if $y = 0$, it is $\theta \le z$.

⟨2⟩3. The classifier g chosen by A_{ℓ_1} makes at most as many errors on the context as h_{w^*}.

PROOF: By ⟨2⟩2 and ⟨2⟩1, the average over $\theta \in [0,1]$ of the number of errors of $\mathbf{1}[\langle \hat{v}, \Psi(\cdot) \rangle \ge \theta]$ is at most the number of errors of h_{w^*}, so some θ achieves at most that number. The number of errors is a step function of θ that changes only at the values $\langle \hat{v}, \Psi(x_i) \rangle$, and on

each piece its value is attained at the right end of the piece, at 0 or at 1; so the minimum over $[0,1]$ is attained at one of at most $m+2$ candidates, which A_{ℓ_1} tries.

⟨2⟩4. With probability at least $1-\delta$, g has error at most $\mathrm{OPT}+\varepsilon$ under $\mathcal{D}'$.

PROOF: By Imported Theorem 2.13(b) and ⟨1⟩1, with probability at least $1-\delta$ every element of $\mathcal{G}_\Psi$, in particular g and h_{w^*}, has empirical error within $\varepsilon/2$ of its error. Then the error of g is at most its empirical error plus $\varepsilon/2$, which by ⟨2⟩3 is at most the empirical error of h_{w^*} plus $\varepsilon/2$, which is at most $\mathrm{OPT}+\varepsilon$.

⟨2⟩5. Q.E.D.

PROOF: For an episode with noise rate η and a fixed task w, the examples are independent draws from the distribution of $(x, h_w(x)\oplus\xi)$, under which h_w has error η, so $\mathrm{OPT}\le\eta$, and the query with its target is one more draw. By ⟨2⟩4 the risk is at most $\eta+\varepsilon+\delta$.

⟨1⟩4. Part (c) holds.

PROOF: Computing $\Psi(x_1),\dots,\Psi(x_m)$ takes time $m\cdot\mathrm{poly}(N)\cdot\kappa 2^\kappa$ by Theorem 5.4. Gaussian elimination over $\mathbb{Q}$ and linear programming with $D_\kappa+m$ variables and $O(m)$ constraints, with entries in $2^{-\kappa}\mathbb{Z}$ of polynomial size, take time $\mathrm{poly}(D_\kappa,m)$, and the threshold search takes time $\mathrm{poly}(m,D_\kappa)$. Since $D_\kappa\le(eN/\kappa)^\kappa=2^{\kappa\log(eN/\kappa)}$, this is polynomial for constant κ, $N^{O(\log N)}$ for $\kappa=O(\log N)$, and $2^{o(N)}\mathrm{poly}(m)$ when $\kappa\log(eN/\kappa)=o(N)$, since then also $2^\kappa=2^{o(N)}$.

⟨1⟩5. Q.E.D.

□

D Proofs for Sections 6 and 7

Proposition 6.5 ($\mathrm{LPRG}_c^{\mathrm{len}}$ fails below rate 1/2)**.** *For every $k\ge2$, every P, every constant $\alpha\ge2$ and every $c<1/2$, there is a deterministic algorithm that runs in time at most $2^{N/(\log N)^c}$ for large N and has advantage at least $1-2^{-N}$ against $G_{P,\lceil\alpha N\rceil}$ at every N. Hence $\mathrm{LPRG}_c^{\mathrm{len}}$ is false for every $c<1/2$.*

Proof of Proposition 6.5. ⟨1⟩1. LET: $m=\lceil\alpha N\rceil\ge2N$, and LET: the algorithm A, on input (τ,z), form the context $S=((\tau_j,z_j))_{j\le m}$ of RL_P, compute $|\mathrm{VS}(S)|$, and output 1 if and only if $|\mathrm{VS}(S)|>0$. Then A is deterministic and runs in time $\mathrm{poly}(N)\cdot2^{O(N\log\log N/\sqrt{\log N})}$.

PROOF: RL_P is k-local, so Imported Theorem 2.8 takes time $\mathrm{poly}(N,m)\cdot2^k\cdot2^{C_1k^2g(N)}$, with k and α constant.

⟨1⟩2. On outputs of the generator, A outputs 1 with probability 1.

PROOF: The task w lies in $\mathrm{VS}(S)$.

⟨1⟩3. On uniform z, A outputs 1 with probability at most 2^{-N}.

PROOF: For fixed τ, $\mathrm{VS}(S)\ne\emptyset$ means that z is one of the at most 2^N strings $G_{P,m}(\tau;w')$, and z is uniform among $2^m\ge2^{2N}$ strings.

⟨1⟩4. Q.E.D.

PROOF: The advantage of A is at least $1-2^{-N}>1/3$ at every N. For $c<1/2$, the exponent $N\log\log N/\sqrt{\log N}$ is $o(N/(\log N)^c)$, so the time of A is at most $2^{N/(\log N)^c}$ for large N. So $G_{P,\lceil\alpha N\rceil}$ is not $2^{N/(\log N)^c}$-pseudorandom, for any k, P and $\alpha\ge2$, and no choice of α_0 in $\mathrm{LPRG}_c^{\mathrm{len}}$ works.

□

Theorem 6.6 (Three regimes for random local functions)**.** *Let $k\ge2$ be a constant and $P\colon\{0,1\}^k\to\{0,1\}$. Consider the task family $(\mathsf{RL}_P,\mathcal{D}_k)$ without noise.*

(a) *The Bayes risk is at most* $N/(m+1)$*, and the Bayes predictor runs in time* $\mathrm{poly}(m) \cdot 2^{O(N \log\log N/\sqrt{\log N})}$.

(b) *The learner* $\mathsf{A}_{\mathrm{lin}}$ *is* (ε,δ)*-PAC from* $m_{\mathrm{VC}}(D_k(N)+1,\varepsilon,\delta) = O((N^k \log(1/\varepsilon)+\log(1/\delta))/\varepsilon)$ *examples and runs in time* $\mathrm{poly}(N,m)$.

(c) *There is an absolute constant* c_* *with the following property. For every in-context learner* A *and every context length* m*, there is a probabilistic algorithm* A' *that runs in time* $T_A + c_*(m+1)k\log N$*, where* T_A *is the running time of* A *on contexts of length* m*, and has advantage exactly* $|1/2 - R_m(A)|$ *against* $G_{P,m+1}$*. If* $m = m(N)$ *is computable in time polynomial in* N*, then* A' *is a single algorithm for all* N.

Proof of Theorem 6.6. ⟨1⟩1. Part (a) holds.

PROOF: Theorem 3.3(b) without noise, and Theorem 3.1(a) with $\kappa = k$ constant, where $\mathrm{poly}(N,m)\cdot 2^k \cdot T_k(N) = \mathrm{poly}(m)\cdot 2^{O(N\log\log N/\sqrt{\log N})}$.

⟨1⟩2. Part (b) holds.

PROOF: Theorem 5.5(a) with $\kappa = k$, where $D_k(N) \le (eN/k)^k = O(N^k)$, so $m_{\mathrm{VC}}(D_k(N)+1,\varepsilon,\delta) = O((N^k\log(1/\varepsilon)+\log(1/\delta))/\varepsilon)$ and the time is $\mathrm{poly}(N,m)$.

⟨1⟩3. Part (c) holds.

⟨2⟩1. LET: A', on input $(\tau_1,\dots,\tau_{m+1},z_1,\dots,z_{m+1})$, run A on the context $((\tau_j,z_j))_{j\le m}$ and the query τ_{m+1}, and output 1 if and only if the prediction equals z_{m+1}. Then A' runs in time $T_A + c_*(m+1)k\log N$ for an absolute constant c_*, and if $m(N)$ is computable in time polynomial in N, A' is one algorithm for all N.

PROOF: Besides running A, A' copies its input, which has $O((m+1)k\log N)$ bits, and compares two bits. The number m is read off the length of the input.

⟨2⟩2. On outputs of the generator, A' outputs 1 with probability $1 - R_m(A)$.

PROOF: For uniform w and independent uniform tuples, $(\tau, G_{P,m+1}(\tau;w))$ is a noiseless episode of length m of the task family $(\mathsf{RL}_P,\mathcal{D}_k)$, with target z_{m+1}.

⟨2⟩3. On uniform z, A' outputs 1 with probability 1/2.

PROOF: z_{m+1} is uniform and independent of the context, of the query and of the random bits of A.

⟨2⟩4. Q.E.D.

PROOF: The advantage of A' is $|1 - R_m(A) - 1/2| = |1/2 - R_m(A)|$ by ⟨2⟩2 and ⟨2⟩3.

⟨1⟩4. Q.E.D.

□

Corollary 6.7 (The three regimes under the hypotheses)**.** *(a) Assume* LPRG_s*, with predicate* P *of arity* k*. Let* A *be a polynomial-time in-context learner for* $(\mathsf{RL}_P,\mathcal{D}_k)$ *and* $m(N) \le \lceil N^s\rceil - 1$ *a context length computable in time polynomial in* N*. Then* $R_{m(N)}(A) \ge 1/6$ *for all sufficiently large* N*. At* $m = \lceil N^s\rceil - 1$ *the Bayes risk is at most* N^{1-s}*, and* $\mathsf{A}_{\mathrm{lin}}$ *has risk at most* ε *from* $O(N^k\log(1/\varepsilon)/\varepsilon)$ *examples.*

(b) *Assume* $\mathrm{LPRG}_c^{\mathrm{len}}$*, with predicate* P *of arity* k *and constant* α_0*. Let* $\alpha \ge \max\{\alpha_0, 7\}$ *be a constant and* $m(N) = \lceil \alpha N\rceil - 1$*. Every in-context learner that runs in time* $2^{N/(\log N)^c - 1}$ *on contexts of length* $m(N)$ *has* $R_{m(N)}(A) \ge 1/6$ *for all sufficiently large* N*, while the Bayes predictor has risk at most* $1/\alpha < 1/6$ *and runs in time* $2^{O(N\log\log N/\sqrt{\log N})}$.

Proof of Corollary 6.7. ⟨1⟩1. Part (a) holds.

⟨2⟩1. LET: A'', on an input with $\lceil N^s\rceil$ outputs, compute $m(N)$ and run the algorithm A' of

Theorem 6.6(c) on the first $m(N)+1$ outputs and their tuples. Then A'' is one algorithm, it runs in time polynomial in N, and its advantage against $G_{P,\lceil N^s\rceil}$ is $|1/2 - R_{m(N)}(A)|$.
PROOF: A runs in time polynomial in its input, whose length is polynomial in N because $m(N) \le N^s$, and $m(N)$ is computable in polynomial time. The first $m(N)+1$ outputs of $G_{P,\lceil N^s\rceil}$ with their tuples are distributed as $G_{P,m(N)+1}$, and those of a uniform string are uniform, so the advantage is that of A'.

⟨2⟩2. $R_{m(N)}(A) \ge 1/6$ for all sufficiently large N.
PROOF: By LPRG_s the advantage of A'' is at most $1/3$ for all large N, so $|1/2 - R_{m(N)}(A)| \le 1/3$.

⟨2⟩3. Q.E.D.
PROOF: At $m = \lceil N^s\rceil - 1$, Theorem 6.6(a) gives $R^*_m \le N/\lceil N^s\rceil \le N^{1-s}$, and Theorem 6.6(b) with $\delta = \varepsilon/2$ and $\varepsilon/2$ in place of ε gives the claim for $\mathsf{A}_{\mathrm{lin}}$.

⟨1⟩2. Part (b) holds.
PROOF: Here $m(N)+1 = \lceil \alpha N\rceil$ is computable in polynomial time. The algorithm A' of Theorem 6.6(c) for A runs in time $2^{N/(\log N)^{c-1}} + c_*\lceil \alpha N\rceil k \log N \le 2^{N/(\log N)^c}$ for large N, so by $\mathrm{LPRG}^{\mathrm{len}}_c$ with this $\alpha \ge \alpha_0$ its advantage $|1/2 - R_{m(N)}(A)|$ is at most $1/3$ for all large N, and $R_{m(N)}(A) \ge 1/6$. The Bayes risk is at most $N/(m(N)+1) = N/\lceil \alpha N\rceil \le 1/\alpha \le 1/7$, and the time of the Bayes predictor is that of Theorem 6.6(a) with $m = O(N)$.

⟨1⟩3. Q.E.D.

□

Corollary 7.1 (Chain-of-thought lengths)**.** *Let (M_N) be a uniform family of sequence models with $p(N) \le N^{O(1)}$ parameters, as above.*

(a) Assume Hypothesis 2.14, and let γ, K, c_γ be as in Theorem 4.4(a). If M_N outputs $\mathsf{B}(S,x)$ for every consistent context of at most N^{c_γ} examples of $\mathsf{BP}_{\lceil K\log N\rceil}$ in dimension N and every query, then $L(N) > 2^{(1-\gamma)N/(c_M+1)} - N^{O(1)}$ for infinitely many N. So $L(N) = 2^{\Omega(N)}$ for infinitely many N.

(b) Assume $\mathrm{ETH}^{\mathrm{len}}_C$ and let $\theta > 0$. If M_N outputs $\mathsf{B}(S,x)$ for every consistent context of at most $36N^2$ examples of LD_3 in dimension N and every query, then $L(N) \ge 2^{N/(\log N)^{C+\theta}}$ for infinitely many N.

(c) Assume LPRG_s with predicate P, and let $m(N) \le \lceil N^s\rceil - 1$ be computable in time polynomial in N. If $L(N) \le N^{O(1)}$, then M has risk at least $1/6$ on $(\mathsf{RL}_P, \mathcal{D}_k)$ at context length $m(N)$ for all sufficiently large N.

(d) Assume $\mathrm{LPRG}^{\mathrm{len}}_c$ with predicate P, and let α and $m(N)$ be as in Corollary 6.7(b). If $L(N) \le 2^{N/((c_M+2)(\log N)^c)}$ for all sufficiently large N, then M has risk at least $1/6$ on $(\mathsf{RL}_P, \mathcal{D}_k)$ at context length $m(N)$ for all sufficiently large N.

Proof of Corollary 7.1. ⟨1⟩1. LET: $\mathcal{S}$ be the algorithm that, given an input of dimension N, computes the parameters of M_N and generates its chain of thought and answer. On a prompt of length ℓ it runs in time at most $N^{O(1)} + (p+\ell+L(N))^{c_M+1}$, and on the inputs of each part $\ell \le N^{O(1)}$.
PROOF: The family is uniform, each of the at most $L(N)+1$ generated tokens takes time at most $(p+\ell+L(N))^{c_M}$, and in every part the contexts have polynomially many examples: at most N^{c_γ} in (a), at most $36N^2$ in (b), and $m(N) \le N^s$ in (c) and (d).

⟨1⟩2. Part (a) holds.
PROOF: Suppose that $L(N) \le 2^{(1-\gamma)N/(c_M+1)} - p(N) - \ell_{\max}(N)$ for all large N, where $\ell_{\max}(N) \le N^{O(1)}$ is the largest prompt length. Then $\mathcal{S}$ computes $\mathsf{B}(S,x)$ on all the con-

texts of Theorem 4.4(a) in time $N^{O(1)} + 2^{(1-\gamma)N} = O(2^{(1-\gamma)N})$, which that theorem excludes.

⟨1⟩3. Part (b) holds.

PROOF: Suppose that $L(N) < 2^{N/(\log N)^{C+\theta}}$ for all large N. Then the time of $\mathcal{S}$ is at most $N^{O(1)} + (N^{O(1)} + 2^{N/(\log N)^{C+\theta}})^{c_M+1} \leq 2^{(c_M+2)N/(\log N)^{C+\theta}} \leq 2^{N/(\log N)^{C+\theta/2}}$ for large N, which Theorem 4.4(b) with $\theta/2$ in place of θ excludes.

⟨1⟩4. Part (c) holds.

PROOF: With $L(N) \leq N^{O(1)}$, $\mathcal{S}$ is a polynomial-time in-context learner; apply Corollary 6.7(a).

⟨1⟩5. Q.E.D.

PROOF: For part (d), if $L(N) \leq 2^{N/((c_M+2)(\log N)^c)}$ for all large N, the time of $\mathcal{S}$ on contexts of length $m(N) = O(N)$ is at most $N^{O(1)} + (N^{O(1)} + 2^{N/((c_M+2)(\log N)^c)})^{c_M+1}$, which is at most $2^{N/(\log N)^c - 1}$ for large N. Apply Corollary 6.7(b).

□