Learning with Simulators: No Regret in a Computationally Bounded World

Sasha Voitovych* VOITOVYC@MIT.EDU
MIT

Abhishek Shetty* SHETTY@MIT.EDU
MIT

Noah Golowich NZG@CS.UTEXAS.EDU
Microsoft Research

Alexander Rakhlin RAKHLIN@MIT.EDU
MIT

Editors: Steve Hanneke and Tor Lattimore

Abstract

Understanding the minimal assumptions necessary for generalization is the fundamental question in learning theory. Unfortunately, most results rely heavily on independence (or some proxy thereof) of the data-generating process, while results for strongly dependent data are far more limited. Towards addressing this gap, we introduce the framework of *simulatable processes*, where the learner has access to a simulator that approximates the distribution generating the data (which may be an arbitrarily complex and dependent process). Surprisingly, given access to such a simulator, we show that we can recover the same learning guarantees as in the classical setting with independent data, namely, error bounds that depend on the VC dimension. Further, we use this framework to study the power of conditional sampling and show strict statistical and computational advantages in this setting. As a highlight of our framework, we exhibit a single algorithm that *simultaneously* learns any given VC class under all processes samplable in bounded polynomial time, with regret controlled by the time-bounded Kolmogorov complexity of the process. This provides a significant conceptual broadening of the classical PAC model.

Keywords: Online learning, transductive learning, VC dimension, Kolmogorov complexity

1. Introduction

A major focus of learning theory has been to understand what structural properties of data are sufficient for efficient learning. One central assumption that has emerged is that of *independence*, namely, that data points are generated independently from a common population distribution. This has enabled a rich theory of statistical learning, with celebrated results characterizing learnability under independence, such as via the notion of *VC dimension* (Vapnik and Chervonenkis, 1971, 1974; Blumer et al., 1989). Unfortunately, independence is often hard to verify, or even justify, in practice. In fact, many natural data-generating processes exhibit rich dependencies across time: today's weather patterns affect tomorrow's; the distribution of a city's population this year depends on that of last year; the spread of disease depends on past infections.

These examples highlight that independence is not a justifiable assumption, but they also point towards a potential resolution to this quagmire. A tacit assumption in the natural sciences is that

* These authors contributed equally.

natural processes are governed by simple, efficient mechanisms—a belief often articulated as the *extended Church–Turing thesis* (Copeland, 2026; Deutsch, 1985). In addition to this abstract belief, we often have explicit simulators for such processes, developed through domain expertise. This is exemplified by the classical study of simulation-based inference in statistics and physics (Marin et al., 2012; Sisson et al., 2018; Deistler et al., 2025), where complex data-generating processes are modeled via simulatable dynamics. For instance, the evolution of physical systems is often modeled via statistical mechanics processes such as Glauber dynamics (Glauber, 1963) and PDEs; opinion dynamics in social networks can be modeled via stochastic processes (Jackson, 2008). More recently, the success of generative models—large language models (Brown et al., 2020; Chowdhery et al., 2023) and diffusion models (Ho et al., 2020; Ramesh et al., 2022; Saharia et al., 2022)—demonstrates that we can build accurate simulators even for complex sequential data such as text and images. Moreover, the aforementioned belief about efficiently computable nature has been a stated motivation behind recent pushes toward better AI systems (Hassabis, 2025, 2024; Sutskever, 2023; Hutter et al., 2024). This leads to the question:

Can the belief that natural processes are governed by efficient, possibly unknown mechanisms be leveraged for learning? (1)

To set the stage for learning without independence, we consider the most well-studied model for sequential learning, namely, *online learning* (Littlestone, 1988; Cesa-Bianchi and Lugosi, 2006): at each round t the learner observes a data point X_t and predicts its label, with the goal of competing with respect to a prespecified class $\mathcal{F}$ of labeling functions. Online learning permits learning even under adversarially generated data, providing the promise of learning with no independence. However, this robustness incurs a significant cost: even simple hypothesis classes, such as thresholds, which are easy to learn under independence, are not learnable in the fully adversarial model. The issue is further exacerbated by computational barriers indicating that classes that are efficiently learnable under independence may not be efficiently learnable in the online setting (Hazan and Koren, 2016). This cements the folk belief that some notion of independence in the data is necessary for efficient learning to be possible. In particular, this belief suggests that learning should be strictly harder for strongly dependent processes.

In this paper, we introduce a new framework for learning under complex dependencies, which we term *simulatable processes*, and show that this framework suffices to recover learnability at essentially the same statistical and computational complexity as learning under independence. Our work can be viewed as part of a long line of work on beyond-worst-case analysis of online learning, where the adversarial nature of the data is relaxed. A number of proposed relaxations allow learning all Vapnik–Chervonenkis (VC) function classes (which are precisely the classes learnable under independence), often efficiently (Rakhlin et al., 2011a; Block et al., 2022; Haghtalab et al., 2024; Montasser et al., 2025). Our framework differs in that it allows learning under *general* dependent processes, provided the learner has access to an (approximate) simulator for the covariate-generating process. The main takeaway of our work is as follows:

Under simulatable processes, learning has the same statistical and computational complexity as learning under independence.

A highlight of our framework is that it provides a lens to take advantage of the belief that nature is computationally bounded. Under this lens, using a connection to time-bounded Kolmogorov complexity, we show that all VC classes are learnable under *unknown* polynomial-time samplable processes, leading to a significant conceptual broadening of the classical PAC model under the widely

believed extended Church–Turing thesis (Copeland, 2026). This can be seen as a formalization of the statement that, while efficiently computable processes themselves might not be learnable,[1] we can learn to predict under these processes.

Notation. We let $\log$ be the binary logarithm. For an integer n, we let $[n]$ denote the set $\{1, \dots, n\}$. We use boldface notation for sequences. For a sequence $\mathbf{s} = (s_1, \dots, s_n)$, we use $\mathbf{s}_{a:b}$ to denote the subsequence $(s_a, s_{a+1}, \dots, s_b)$; for $I \subset [n]$, we use $\mathbf{s}_I$ to denote the subsequence $(s_i)_{i \in I}$. For a distribution $P \in \Delta(\mathcal{X}^n)$ and subsets $I, J \subset [n]$ of indices, we use P_I to denote the marginal of P on the subsequence $\mathbf{X}_I$; for any $\mathbf{x}_J$, we use $P_I(\cdot \mid \mathbf{x}_J)$ to denote the conditional distribution of the subsequence $\mathbf{X}_I$ given $\mathbf{X}_J = \mathbf{x}_J$. We adopt the standard non-asymptotic big-O notation and write $f \leq O(g)$ or $f \lesssim g$ to denote that $f \leq Cg$ for some universal constant $C > 0$; similarly, we write $f \geq \Omega(g)$ or $f \gtrsim g$ to denote that $f \geq cg$ for some universal constant $c > 0$.

Related Work. Our paper is in a long line of work on sequential prediction and beyond-worst-case analysis (Shetty, 2024). Further, our framework has connections to notions such as learning with predictions, simulators in reinforcement learning, simulation-based inference, and computational perspectives on learning. We defer a thorough discussion to Appendix C.

2. Our Framework and Contributions

We begin by formally introducing our learning setting. We assume the learner and nature play a T-round game. Before the start of the game, nature chooses a covariate-generating distribution $P^\star \in \Delta(\mathcal{X}^T)$ and samples $(X_1, \dots, X_T) \sim P^\star$. Let us re-emphasize that we make no assumptions on the distribution $P^\star$, which can be arbitrarily complex and dependent across time. Then, at each round $t \leq T$, nature reveals X_t to the learner. The learner makes a prediction $\hat{Y}_t \in \{0, 1\}$ for the label of X_t, after which it learns the true label Y_t, and suffers loss 1 if $\hat{Y}_t \neq Y_t$. The objective of the learner is to minimize *expected regret* with respect to a comparator class of labeling functions $\mathcal{F}$:

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) = \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{\hat{Y}_t \neq Y_t\} - \inf_{f \in \mathcal{F}} \sum_{t=1}^{T} \mathbb{1}\{f(X_t) \neq Y_t\}\right].$$

The expectation above is taken with respect to the generation from $P^\star$ and the internal randomness of the learner. Throughout, we operate under the assumption that the labels are chosen *obliviously*.

Assumption 1 *There exists a function $f^\star$ such that, for every $t \in [T]$, $Y_t = f^\star(X_t)$ and $f^\star$ is fixed before the draw of $(X_1, \dots, X_T)$. We refer to $f^\star$ as the ground truth labeling function.*

The assumption formalizes the belief that, while the data may have complex structure and dependencies, the ground truth is a fixed property of the world: it does not change in response to the learner's predictions or the realization of past covariates. While some of our results hold under milder conditions, we defer further discussion of this assumption to Appendix H.

A motivating problem: learning against computationally bounded nature. Consider the most ambitious instantiation of the motivating question in (1): can we hope to learn against *any* unknown nature, provided only that nature is computationally bounded? Concretely, assume the covariates

1. This is indeed the case under widely held beliefs in computational complexity, e.g., existence of one-way functions.

$X_1, \ldots, X_T$ are produced by an unknown distribution $P^\star$ samplable in time $p(T)$ for some polynomial p. Can we learn efficiently in this setting, *simultaneously* against every such $P^\star$ and every labeling function $f^\star$? Our first headline contribution answers in the affirmative, as long as $f^\star \in \mathcal{F}$ and $\mathcal{F}$ has VC dimension in $O(1)$.

Informal Theorem 2.1 (Theorem 5.4) *Let $\mathcal{F}$ be a class of VC dimension d. Then, there exists a single algorithm that, given a realizable ERM oracle for $\mathcal{F}$, runs in time $\mathrm{poly}(T^d, p(T))$ and, for every p-time samplable $P^\star$ and every $f^\star \in \mathcal{F}$, achieves $o(T)$ regret, where the rate depends only on $P^\star$ and d.*

The above can be viewed as a learning-theoretic analog of Levin search (Levin, 1973). Indeed, just as Levin search pays only a bounded overhead relative to the fastest program for any task, our algorithm pays only a bounded overhead relative to the sampling complexity of $P^\star$ for any p-time process, without any knowledge of $P^\star$ in advance.

The rate in Informal Theorem 2.1 depends on the *time-bounded Kolmogorov complexity of $P^\star$*. The proof leverages results from the Kolmogorov complexity literature (Antunes et al., 2006; Lu et al., 2022; Hirahara and Nanashima, 2023) and constructs a *single* distribution P, which we refer to as the *simulator*, that can be sampled from efficiently, and, in a certain sense, *dominates* every other distribution samplable in p-time. To show that access to such simulators suffices to recover learnability of VC classes, we introduce our framework of learning under simulatable processes.

Learning under simulatable processes. A key defining feature of our *simulatable process* framework is that we assume access to a sampling oracle for a distribution P that we will tacitly assume is close to $P^\star$.

Assumption 2 (Simulator access) *The learner has access to a simulatable process P that approximates $P^\star$. That is, the learner has access to an oracle $\mathcal{O}_P$ that, when queried, returns an independent sample (a trajectory $(X_1, \ldots, X_T)$) from P. We refer to P simply as the simulator.*

Other than having simulation access to the distribution of covariates, it is important to note that we are making no further restriction on the distributions P and $P^\star$. In particular, this setting generalizes the standard computational learning theory setting of learning under a known, product distribution such as the uniform distribution or Gaussian distribution. Further, we note here that we only assume sampling access and not access to the density. We separate *realizable* ($f^\star \in \mathcal{F}$) and *agnostic* ($f^\star$ unconstrained) learning settings. Our first contribution is then showing that all VC classes are learnable under simulatable processes, even given an approximate simulator, provided $f^\star$ belongs to the class $\mathcal{F}$.

Informal Theorem 2.2 (Theorem 3.1) *For any class of VC dimension d, given access to N samples from an approximate simulator P, and any $f^\star \in \mathcal{F}$, there is an algorithm that achieves*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim d\log(T) + \frac{dT}{N} + d \cdot \mathsf{KL}(P^\star \parallel P) + \mathsf{KL}(P^\star \parallel P)\left(\frac{T}{\log\left(1 + N/T\right)}\right).$$

The first term above corresponds to the optimal error of learning under independence, which is achieved by empirical risk minimization (ERM). This already hints at a key characteristic of our setting. In our setting, ERM can fail drastically as it does not exploit the simulator, and thus, standard lower bounds for online learning with ERM apply. The remaining three terms above account for the price of sampling (finite N) and the price of approximation ($\mathsf{KL}(P^\star \parallel P) > 0$) in

our setting. Note that the Kullback–Leibler (KL) divergence is, perhaps, the most natural metric on distributions over sequences due to the tensorization property; it is also typically smaller than other measures considered in the literature, such as coverage (i.e., uniform density ratio bounds). Further, KL divergence has practical significance in the modern context of generative models, where LLMs are typically trained with the log loss for perhaps similar reasons.

Two aspects of the result above might seem unsatisfactory at first glance: the number of samples needed (with approximate simulator access) and the realizability requirement. Unfortunately, both these aspects are inherent to this setting. In Theorem 3.3, we show that the tradeoff between the number of samples and the quality of approximation in Informal Theorem 2.2 is essentially tight. Further, even when the simulator is exact, in Theorem 3.5, we show that agnostic learning is statistically impossible even for simple classes with VC dimension 1. This indicates that Informal Theorem 2.2 is the strongest possible result for algorithms in the setting discussed so far.

A second motivating problem: learning without mixing. While Informal Theorem 2.1 is a very general learnability statement, this generality comes at a cost: the result requires realizability of $f^\star$, and the running time has exponential dependence on the VC dimension. To explore whether more structure can alleviate these limitations, we consider an example of learning under Markov chains, which is a well-studied problem (Aldous and Vazirani, 1995; Bartlett et al., 1994). As concrete examples, in this paper we consider two families of processes motivated by evolutions of physical systems: linear dynamical systems (LDS), and Glauber dynamics. Here, we focus on the setting of LDS and defer the discussion of Glauber dynamics to Appendix B. In particular, suppose $X_1, \ldots, X_T \in \mathbb{R}^n$ evolve according to a, possibly unknown, LDS, described as follows:

$$X_{t+1} = A^\star X_t + \eta_t, \quad \text{where} \quad \eta_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2 I). \tag{2}$$

The above model exhibits non-independence if the chain is non-mixing (which happens when the spectral radius of $A^\star$ is close to 1), disallowing approaches that extract independence structure from mixing (Yu, 1994; Mohri and Rostamizadeh, 2010; Kuznetsov and Mohri, 2017; Gamarnik, 1999). Similarly, naive applications of oracle-efficient methods from smoothed analysis (Haghtalab et al., 2024) yield bounds with exponential dependence on the dimension.[2] However, the process has only at most $O(n^2)$ free parameters, so one could hope for complexity that scales with $\mathrm{poly}(n)$.

Informal Theorem 2.3 (Theorem 5.5) *Let $B > 0$ be arbitrary, and let $\rho(\cdot)$ denote the spectral radius of a matrix. Then, for any class $\mathcal{F}$ of VC dimension d, there is a $\mathrm{poly}(n, T)$-time algorithm that, given an agnostic ERM oracle for $\mathcal{F}$, achieves sublinear expected regret in the agnostic setting for all $P^\star$ evolving according to Eq. (2) with $A^\star \in [-B, B]^{n \times n}$ with $\rho(A^\star) \le 1$.*

We highlight that the above result has *no dependence on the mixing time*. The proof of the above result leverages the fact that all distributions in the LDS family can be simultaneously and efficiently simulated in a *conditional sense*. This motivates the second part of our framework.

Efficient algorithms with conditional simulators. Broadly inspired both by autoregressive generative models and simulations for physical processes, we consider the setting when the learner has access to a *conditional* sampling oracle for P.

2. The reason is that a Gaussian density with unknown mean can only be dominated by a fixed density up to a constant that scales exponentially in n.

Setting		Unconditional samples	Conditional samples	Refs.
Realizable, exact	Regret	$\tilde{O}(d)$	$O(\sqrt{dT})$	Thms. 3.1, 4.1
	Runtime	$\exp(\Omega(d))$	$\mathrm{poly}(T, d)$	Thms. 3.4, 4.1
Agnostic, exact	Regret	$\Omega(T)$	$O(\sqrt{dT})$	Thms. 3.5, 4.1
	Runtime	—	$\mathrm{poly}(T, d)$	Thm. 4.1
Realizable, approximate	Regret	$\tilde{O}(d + d \cdot \mathsf{KL})$	$O(\sqrt{dT(1+\mathsf{KL})})$	Thms. 3.1, 4.1
	Runtime	$\exp(\Omega(T))$	$\mathrm{poly}(T, d)$	Thms. 3.3, 4.1
Agnostic, approximate	Regret	$\Omega(T)$	$O(\sqrt{dT(1+\mathsf{KL})})$	Thms. 3.5, 4.1
	Runtime	—	$\mathrm{poly}(T, d)$	Thm. 4.1

Table 1: Summary of regret and runtime bounds for algorithms for classes of VC dimension d, with access to unconditional or conditional samples from the exact or approximate simulator P, and with $\mathsf{KL} := \mathsf{KL}(P^\star \,\|\, P)$. Any upper bound listed under *Unconditional samples* also applies under *Conditional samples*, since the former oracle is strictly weaker.

Assumption 3 (Conditional Simulator access) *The learner has access to a conditionally simulatable process P that approximates $P^\star$. That is, the learner has access to an oracle $\mathcal{O}_P^{\mathsf{cond}}$ that, when queried on a prefix $\mathbf{x}_{1:t} \in \mathcal{X}^t$, returns an independent sample from $P(\cdot \mid \mathbf{x}_{1:t})$.*

Perhaps surprisingly, this additional power allows us to circumvent the statistical and computational barriers discussed above. In particular, both realizable and agnostic learning become statistically tractable, and admit oracle-efficient algorithms.

Informal Theorem 2.4 (Theorem 4.1) *Assume conditional sampling access to simulator P. Then, given access to an agnostic ERM oracle for the class $\mathcal{F}$ of VC dimension d, there exists an algorithm running in $\mathrm{poly}(T)$ time, with $\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT(1 + \mathsf{KL}(P^\star \,\|\, P))}$ in the agnostic setting.*

The above theorem matches PAC-style learnability guarantees, both statistically (regret controlled by the VC dimension) and computationally (ERM oracle suffices). This result is based on relaxations with random playout (Rakhlin et al., 2012). Specifically, we extend the results of Rakhlin and Sridharan (2015) to the setting where only approximate conditional simulator access is available.

Trade-off: conditional vs. unconditional simulators. Taken together, the results above suggest a trade-off: unconditional samples are often easy to obtain but do not yield efficient algorithms in general, while conditional samples can be more costly but enable $\mathrm{poly}(T, d)$ algorithms. The trade-off is already visible in Informal Theorem 2.1: the universal simulator P is hard to sample from conditionally, assuming the existence of one-way functions (see Section 5.1), whereas structured families such as Glauber dynamics and LDS admit efficient conditional simulators. This trade-off is further exemplified in Table 1.

3. Learning with Unconditional Simulators

3.1. VC Classes Are Learnable under Simulatable Processes

Our results depend on the VC dimension of the class $\mathcal{F}$, which is a canonical complexity measure in learning theory (Vapnik, 1998). In particular, bounded VC is equivalent to learnability under independent processes. Thus, VC is precisely the "correct" complexity when learning under independence. We formally define it in Definition D.1. Throughout, we reserve d to denote the VC dimension of the class at hand. The main result of this section is that bounded VC dimension suffices to recover logarithmic regret in our setting, even with an approximate simulator. A priori, this result is surprising, since the process $P^\star$ can be arbitrarily complex and dependent.

Theorem 3.1 *Let $\mathcal{F}$ be a VC dimension d class, and let $P^\star$ be an arbitrary distribution in $\Delta(\mathcal{X}^T)$. Let $N \in \mathbb{N}$ be arbitrary. Then, there exists an algorithm that makes at most N oracle calls to $\mathcal{O}_P$, at most $N^{O(d)}$ calls to a realizable ERM oracle w.r.t. $\mathcal{F}$, and achieves regret*

$$\mathbb{E}\,\mathbf{Reg} \lesssim d \log T + \frac{dT}{N} + d \cdot \mathsf{KL}(P^\star \parallel P) + \frac{T}{\log(1 + N/T)} \cdot \mathsf{KL}(P^\star \parallel P).$$

The bound quantitatively captures the tradeoff between the quality of the simulator, the number of samples, and the regret. Specifically, the second term, dT/N, captures the *price of sampling* and is present since even with an exact simulator $P = P^\star$, the learner is restricted to N samples from P. The third term, $d \cdot \mathsf{KL}(P^\star \parallel P)$, captures the *price of approximation* and quantifies the regret that an inexact simulator introduces even with infinitely many samples. Finally, the last term is the *price of approximate sampling* that quantifies the additional error from having both an inexact simulator and a finite sample budget. Somewhat surprisingly, the upper bound in Theorem 3.1 is essentially tight, and we can show that all the error terms are indeed necessary (see Theorem 3.3).

3.2. Proof Sketch for Theorem 3.1

The regret bound in Theorem 3.1 is attained by Algorithm 1. The first phase of the algorithm is based on the following idea: by drawing samples from the uniform mixture of the marginals of P, $\bar{P} := \frac{1}{T}\sum_{t=1}^{T} P_t$, we can identify a small subset of functions that represents the behavior of any $f^\star$ on the draw $\mathbf{X} \sim P$ well. Concretely, for each $\ell \in [L]$, our algorithm draws N_ℓ i.i.d. samples $\mathbf{Z}^\ell := (Z_1^\ell, \ldots, Z_{N_\ell}^\ell) \sim \bar{P}^{\otimes N_\ell}$. Then, it constructs an *improper cover* $\mathcal{G}_\ell$ by iterating through all labelings of $\mathbf{Z}^\ell$ realized by $\mathcal{F}$ (a set we denote by $\mathcal{F}|_{\mathbf{Z}^\ell}$) and running an optimal PAC learner on each labeling. Specifically, we use the Majority-of-Three learner of Aden-Ali et al. (2024). Then, using the population error guarantee for the PAC learner, we can show that any fixed function $f \in \mathcal{F}$ is well-represented with respect to the distribution $\bar{P}$, and hence, on average under a draw $\mathbf{X} \sim P$.

Formally, let $\mathsf{Alg}_{\mathsf{M3}}(S) \in \{0,1\}^{\mathcal{X}}$ be the output of the Majority-of-Three algorithm on a labeled sample S; additionally, for functions f, g, and a sequence $\mathbf{X} \in \mathcal{X}^T$ of covariates, let $\|f - g\|_{\mathbf{X}} := \frac{1}{T}\sum_{t=1}^{T} |f(X_t) - g(X_t)|$ be the average disagreement between f and g on $\mathbf{X}$. Then, the preceding discussion is formalized in the following statement.

Lemma 3.2 *Let $\mathbf{Z} = (Z_1, \ldots, Z_N) \sim \bar{P}^{\otimes N}$, and $\mathcal{G}(\mathbf{Z})$ be an improper cover of $\mathcal{F}$ constructed as follows*

$$\mathcal{G}(\mathbf{Z}) := \left\{ \mathsf{Alg}_{\mathsf{M3}}\left((Z_i, y_i)_{i\in[N]}\right) : (y_i)_{i\in[N]} \in \mathcal{F}|_{\mathbf{Z}} \right\}. \tag{3}$$

Then, for any $f \in \mathcal{F}$, we have $\mathbb{E}_{\mathbf{Z}}\,\mathbb{E}_{\mathbf{X}\sim P} \min_{g\in\mathcal{G}(\mathbf{Z})} \|f - g\|_{\mathbf{X}} \lesssim \frac{d}{N}$.

Algorithm 1: MultiCover

input: Sampling oracle $\mathcal{O}_P$ for P; realizable ERM oracle w.r.t. $\mathcal{F}$

parameters: Learning rate $\eta > 0$; number of levels L; number of samples per level $\{N_\ell\}_{\ell \in [L]}$

/* Phase 1: Build multi-scale covers */

for $\ell \leftarrow 1$ **to** L **do**

  Sample $\mathbf{Z}^\ell \sim \bar{P}^{\otimes N_\ell}$ with $\bar{P} := \frac{1}{T}\sum_{t=1}^{T} P_t$ // N_ℓ queries to $\mathcal{O}_P$

  Let $\mathcal{G}_\ell := \mathcal{G}(\mathbf{Z}^\ell)$ be an improper cover of $\mathcal{F}$ on $\mathbf{Z}^\ell$ as in Eq. (3) // $N_\ell^{O(d)}$ ERM queries

end

$\mathcal{G} \leftarrow \bigcup_{\ell=1}^{L} \mathcal{G}_\ell$

/* Phase 2: Exponential weights over $\mathcal{G}$ */

For $\ell \in [L]$, $g \in \mathcal{G}_\ell$, set $w_1(g) \leftarrow \frac{1}{L|\mathcal{G}_\ell|}$ // Initialize the weights

for $t \leftarrow 1$ **to** T **do**

  Sample g_t according to the probability vector $\{w_t(g)\}_{g \in \mathcal{G}}$

  Observe X_t and play $\hat{Y}_t \leftarrow g_t(X_t)$

  Observe Y_t and suffer loss $\mathbb{1}\left\{Y_t \neq \hat{Y}_t\right\}$

  Update the weights for each g: $w_{t+1}(g) \leftarrow \frac{w_t(g)\exp(-\eta \mathbb{1}\{g(X_t) \neq Y_t\})}{\sum_{g' \in \mathcal{G}} w_t(g')\exp(-\eta \mathbb{1}\{g'(X_t) \neq Y_t\})}$

end

We note that cover $\mathcal{G}(\mathbf{Z})$ can be constructed in time $N^{O(d)}$ with access to a realizable ERM oracle. Indeed, from the celebrated Sauer–Shelah lemma, the number of distinct realizable labelings in $\mathcal{F}|_{\mathbf{Z}}$ is on the order $N^{O(d)}$ and can be enumerated in time $N^{O(d)}$ (Blumer et al., 1989). Also, the Majority-of-Three learner requires $O(1)$ realizable ERM queries per labeling.

Having obtained such a cover, a natural next step is to run exponential weights over it. A natural approach is to apply a change of measure argument to Lemma 3.2 directly by controlling the tail behavior of $\mathrm{d}P^\star/\mathrm{d}P$ (as done in similar contexts, e.g., Block and Polyanskiy (2023)). However, a tighter argument in our setting exploits the multi-scale nature of Algorithm 1, which allows the regret to depend on the loss of the "correct" expert with respect to the *realized* density ratio. In particular, choosing $\{N_\ell\}_{\ell \in [L]}$ appropriately ensures that, with high probability, there is a random level $\ell^\star$ with $\log(N_{\ell^\star}/T) \approx \log(\mathrm{d}P^\star/\mathrm{d}P)$. For this level,

$$\mathbb{E}_{\mathbf{X}\sim P^\star} \min_{g \in \mathcal{G}_{\ell^\star}} \|f - g\|_{\mathbf{X}} \lesssim \sum_{\ell \in [L]} \frac{N_\ell}{T}\, \mathbb{E}_{\mathbf{X}\sim P} \min_{g \in \mathcal{G}_\ell} \|f - g\|_{\mathbf{X}} \lesssim \frac{dL}{T},$$

where the last step follows from Lemma 3.2. With this control of the approximation term, we apply standard results for exponential weights with non-uniform priors (Line 6), and get:

$$\begin{aligned}
\mathbf{Reg}(\mathcal{F}, T) &\lesssim \log\left(L\,|\mathcal{G}_{\ell^\star}|\right) + T \cdot \min_{g \in \mathcal{G}_{\ell^\star}} \|f - g\|_{\mathbf{X}} \\
&\lesssim d \log\left(N_{\ell^\star}\right) + T \cdot \min_{g \in \mathcal{G}_{\ell^\star}} \|f - g\|_{\mathbf{X}} \\
&\approx d \log\left(T \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right) + T \cdot \min_{g \in \mathcal{G}_{\ell^\star}} \|f - g\|_{\mathbf{X}},
\end{aligned}$$

where the first transition uses the Sauer–Shelah lemma. The first term becomes a KL term after taking expectations, and the second term is bounded in expectation by the derivation above.

3.3. Limitations of Unconditional Samples

We now demonstrate that the limitations of Algorithm 1 are inherent. First, we show that the trade-off between the quality of simulator and number of samples in Theorem 3.1 is tight.

Theorem 3.3 *For any $d \in \mathbb{N}$, $N \geq 2$, and $D \in \mathbb{N} \cup \{0\}$, there exists a class $\mathcal{F}$ of VC dimension d, such that, for any learning algorithm that draws at most N unconditional samples from $\mathcal{O}_P$, there exists a choice of distributions $(P, P^\star)$ with $\mathsf{KL}(P^\star \,\|\, P) \leq D$ and $f^\star \in \mathcal{F}$ such that*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \gtrsim \left(d + \frac{dT}{N} + D + D\frac{T}{\log N}\right) \wedge T$$

In particular, the lower bound above implies that even in the moderate KL regime (for instance, $\mathsf{KL}(P^\star \,\|\, P) \leq 0.1$), an *exponential* in T number of samples is necessary to achieve sub-polynomial regret for all VC classes. As mentioned above, with exact simulator access, T unconditional samples suffice, and Algorithm 1 runs in $T^{O(d)}$ time, which is polynomial in the regime $d \leq O(1)$. A natural question to ask is whether $\mathrm{poly}(T, d)$ runtime is also possible. We answer this in the negative. The following result can be viewed as an extension of Hazan and Koren (2016) to our setting.

Theorem 3.4 *There is a sufficiently small constant c so that the following holds. For any $d, N, T \in \mathbb{N}$ satisfying $TN \leq 2^{cd}$, and any learning algorithm Alg that draws at most N unconditional samples from $P^\star$ and makes at most N agnostic ERM oracle queries, there exists a class $\mathcal{F}$ of VC dimension at most d and a distribution $P^\star$ so that Alg suffers regret $\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \gtrsim T$.*

Together, Theorems 3.3 and 3.4 highlight computational challenges in learning with unconditional samples, and show that Algorithm 1 is essentially statistically and computationally optimal in the realizable setting with unconditional samples. We now turn our attention to agnostic learning. Typically, agnostic learning and realizable learning are qualitatively similar in statistical complexity, though the rates may differ. Surprisingly, this is not the case in our setting: we show that we can force linear regret in the agnostic setting on the class of thresholds for any learner that draws finitely many unconditional samples in the agnostic learning setting.

Theorem 3.5 *Let $\mathcal{F}$ be the class of thresholds on the $[0, 1]$ interval. For any $N \geq 1$, and for any learning algorithm that draws at most N unconditional samples from $P^\star$, there exists a choice of a distribution $P^\star$ and $f^\star \notin \mathcal{F}$ such that $\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \gtrsim T$.*

4. Learning with Conditional Simulators

Thus far, we have established that all VC classes are learnable under simulatable processes in the realizable setting. In order to extend these results beyond the realizable setting, we consider a stronger notion of simulatable processes that allows *conditional* sampling. Specifically, given a conditional sampling oracle for P that approximates $P^\star$, we show that agnostic learning of all VC classes is possible. The main result of this section follows. It is achieved by Algorithm 2.

Theorem 4.1 *Let $\mathcal{F}$ be a VC dimension d class. Then, there exists an algorithm that runs in $\mathrm{poly}(T)$ time and, given access to an agnostic ERM oracle over $\mathcal{F}$ and a conditional sampling oracle $\mathcal{O}_P^{\mathsf{cond}}$, achieves the following expected regret bound in the agnostic setting:*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT(1 + \mathsf{KL}(P^\star \,\|\, P))}.$$

When $P = P^\star$ (i.e., exact conditional sampling access), it is known that one can achieve the optimal $O(\sqrt{dT})$ regret for VC dimension d classes using *relaxations with random playout* (Rakhlin et al., 2012; Rakhlin and Sridharan, 2015). Our contribution is to extend these results to a setting where the playout distribution P is only a coarse approximation for $P^\star$ (that is, positive KL). The dependence on $\mathsf{KL}(P^\star \,\|\, P)$ above is also qualitatively tight: the result can be complemented with a $\sqrt{T \cdot \mathsf{KL}(P^\star \,\|\, P)}$ lower bound for the agnostic setting against even inefficient algorithms (Proposition F.4). Additionally, $\sqrt{T}$-type regret is known to be optimal for oracle-efficient algorithms even in the realizable transductive setting (Hazan and Koren, 2016).

Relaxations. A convenient way to design oracle-efficient algorithms is via the *relaxation* framework (Rakhlin et al., 2012). Intuitively, relaxations maintain at each time t an upper bound on the "remaining difficulty" of the problem. Here, we consider what we call a *log-MGF* relaxation, which can be derived as an upper bound for the classical conditional Rademacher complexity relaxation (Rakhlin et al., 2012; Rakhlin and Sridharan, 2015). The relaxation is parameterized by an inverse temperature parameter $\eta > 0$. Formally, for a dataset $\mathbf{S}_{1:t} := (\mathbf{X}_{1:t}, \mathbf{Y}_{1:t})$, we let $L_t(f) := \sum_{s=1}^{t} \mathbb{1}\{f(X_s) \neq Y_s\}$, and define:

$$\mathbf{Rel}(\mathbf{S}_{1:t}) := \frac{1}{\eta} \log \mathbb{E}_{\substack{\mathbf{X}_{t+1:T} \sim P(\cdot|\mathbf{X}_{1:t}), \\ \boldsymbol{\varepsilon}_{t+1:T} \sim \mathsf{Unif}(\{\pm 1\}^{T-t})}} \exp\left(\eta \sup_{f \in \mathcal{F}} \left[\sum_{s=t+1}^{T} \varepsilon_s \frac{2f(X_s) - 1}{2} - L_t(f)\right]\right). \tag{4}$$

The strategy of the learner is defined as a solution to the following minimax problem:

$$q_t(\mathbf{S}_{1:t-1}, X_t) = \operatorname*{argmin}_{q \in \Delta([0,1])} \sup_{Y_t \in \{0,1\}} \left[\mathbb{E}_{\hat{Y} \sim q} \mathbb{1}\{\hat{Y} \neq Y_t\} + \mathbf{Rel}(\mathbf{S}_{1:t})\right] \tag{5}$$

The advantage of the relaxation in Eq. (4) over classical approaches (Rakhlin et al., 2012; Rakhlin and Sridharan, 2015) is that it allows us to control the *exponential moment* of regret, as opposed to just the expectation. This becomes crucial when the simulator at hand is imperfect ($P \neq P^\star$), since concentration properties of regret can be leveraged for sharper change-of-measure arguments. Concretely, using *Donsker–Varadhan* variational formula for KL (Polyanskiy and Wu, 2025), we have:

$$\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) \leq \frac{1}{\eta} \log \mathbb{E}_P \exp(\eta \mathbf{Reg}(\mathcal{F}, T)) + \frac{1}{\eta} \mathsf{KL}(P^\star \,\|\, P). \tag{6}$$

The first term in the above upper bound can be controlled by considering the following potential:

$$\Phi_t(\mathbf{S}_{1:t}) := \exp\left(\eta \bar{L}_t + \eta \mathbf{Rel}(\mathbf{S}_{1:t})\right), \tag{7}$$

where $\bar{L}_t := \sum_{s \leq t} \mathbb{E}_{\hat{Y}_s \sim q_s} \mathbb{1}\{\hat{Y}_s \neq Y_s\}$ is the predictable loss process. Then, it can be shown that Eq. (5) makes Φ_t a *supermartingale* (see Lemma F.1). Since $\Phi_T = \exp(\eta \bar{L}_T - \inf_f L_T(f))$ and $\mathbb{E}\,\Phi_T \leq \Phi_0$ by the supermartingale property, we have the following lemma.

Lemma 4.2 *Let $\mathcal{F}$ be a VC dimension d class. Then, for every $\eta > 0$, an algorithm that plays according to Eq. (5) achieves the following expected regret bound in the agnostic setting:*

$$\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT} + \frac{1}{\eta} \mathsf{KL}(P^\star \,\|\, P) + \eta T.$$

Note that it is non-trivial to compute Eq. (5) from conditional samples from P and agnostic ERM oracle calls, which is why it is the *idealized* strategy of the learner. We now show how the rule in Eq. (5) can be implemented efficiently.

Efficient implementation. We first express the solution to Eq. (5) in closed form in terms of exponential moments of a certain stochastic process. For notational brevity, let $\mathbf{W}^t = (\mathbf{X}_{t+1:T}, \boldsymbol{\varepsilon}_{t+1:T})$ be the random variable summarizing the randomness left after X_t is revealed.

Proposition 4.3 *For a history $\mathbf{S}_{1:t}$ and a covariate X_t, let*

$$\mathfrak{G}_y(\mathbf{W}^t) := \sup_{f \in \mathcal{F}} \left[\sum_{s=t+1}^{T} \varepsilon_s \frac{2f(X_s) - 1}{2} - L_{t-1}(f) - \mathbb{1}\{f(X_t) \neq y\} \right]. \tag{8}$$

Then, q_t as in Eq. (5) can be written as $q_t(\mathbf{S}_{1:t-1}, X_t) = \frac{1}{2} + \frac{1}{2\eta} \log\left(\frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\right)$

We now estimate q_t using polynomially many conditional samples from P. At first sight, this is challenging: $\exp(\eta\mathfrak{G}_y(\mathbf{W}^t))$ can be as large as $\exp(\eta T)$, so estimating its expectation directly may require exponentially many samples. As we show below, this is not an issue due to the concentration properties of this quantity and cancellation effects in the ratio of expectations. Concretely, we consider a natural plug-in estimator for q_t: for i.i.d. samples $\widehat{\mathbf{W}}_1^t, \ldots, \widehat{\mathbf{W}}_N^t$, we let:

$$\hat{q}_t(\mathbf{S}_{1:t-1}, X_t) = \frac{1}{2} + \frac{1}{2\eta} \log\left(\frac{\frac{1}{N}\sum_{j=1}^{N} \exp(\eta\mathfrak{G}_1(\widehat{\mathbf{W}}_j^t))}{\frac{1}{N}\sum_{j=1}^{N} \exp(\eta\mathfrak{G}_0(\widehat{\mathbf{W}}_j^t))} \right). \tag{9}$$

The estimator can be computed using N conditional oracle queries to P and $O(N)$ agnostic ERM oracle calls. Its error is controlled by the normalized second-moment of $\exp(\eta\mathfrak{G}_y)$ (Lemma F.2):

$$\mathbb{E}(\hat{q}_t - q_t)^2 \lesssim \frac{1}{N} \max_{y \in \{0,1\}} \frac{\mathbb{E}\exp(2\eta\mathfrak{G}_y(\mathbf{W}^t))}{\left(\mathbb{E}\exp(\eta\mathfrak{G}_y(\mathbf{W}^t))\right)^2}.$$

Intuitively, both $\mathfrak{G}_1$ and $\mathfrak{G}_0$ have a form of a Rademacher process with an offset of $L_{t-1}(f)$. Since the second moment is normalized, the impact of the offset cancels out, and we can upper bound the above using concentration of the exponential of a Rademacher process (Lemma F.3).

Lemma 4.4 *Let $\mathcal{F}$ be a VC dimension d class. Then, an algorithm that plays according to Eq. (9) with N samples per step and inverse temperature η achieves the following regret bound*

$$\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT} + \eta T + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P) + T\sqrt{\frac{\exp(O(\eta\sqrt{dT} + \eta^2 T))}{N}}.$$

Setting $\eta \asymp 1/\sqrt{dT}$ and $N = \mathrm{poly}(T)$ large enough, the above gives: $\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT}(1 + \mathsf{KL}(P^\star \parallel P))$. This is short of the regret bound claimed in Theorem 4.1, due to the linear dependence on $\mathsf{KL}(P^\star \parallel P)$. In particular, the above is non-vacuous only when $\mathsf{KL}(P^\star \parallel P) \leq o(\sqrt{T})$. This rate can be further sharpened using epoching.

Epoching. The bound in Lemma 4.4 is obtained by performing rollouts from the distribution $P(\cdot \mid \mathbf{X}_{1:t})$ all the way to horizon T. When $P \neq P^\star$, this can be problematic: long rollouts from P may quickly drift away from the typical future under $P^\star$, making the simulated continuation uninformative. Epoching is a natural strategy for mitigating compounding errors and also appears in works on predictable sequences (Raman and Tewari, 2024). Let $B_1, \ldots, B_K$ be a partition of $[T]$ into K consecutive epochs of equal length $L := T/K$, and let $H_k := \bigcup_{i<k} B_i$. The key point is that

KL *tensorizes* across time blocks and thus, the average KL discrepancy across epochs will be small. Concretely, we can run the same relaxation-based update within each epoch, using rollouts from P that only extend to the end of the current epoch. Applying Lemma 4.4 to each epoch separately gives the following upper bound on expected regret:

$$\sum_{k=1}^{K} \sqrt{dL} \cdot \left(1 + \mathbb{E}\, \mathsf{KL}\big(P^{\star}_{B_k}(\cdot \mid \mathbf{X}_{H_k}) \;\big\|\; P_{B_k}(\cdot \mid \mathbf{X}_{H_k})\big)\right) = \sqrt{dTK} + \sqrt{dT/K} \cdot \mathsf{KL}(P^{\star} \,\|\, P).$$

It remains to optimize over K, which can be done without the knowledge of $\mathsf{KL}(P^{\star} \,\|\, P)$ by running exponential weights over different choices of K. Combining these ideas yields a prediction strategy given in Algorithm 2 (see Appendix A).

5. Learning under Families of Processes

Given our results thus far, the next natural question is: What if no natural simulator for $P^{\star}$ is available? Learning under all $P^{\star}$ simultaneously is impossible (it reduces to classical online learning), but in most settings some information about $P^{\star}$ is available. We model this by assuming $P^{\star}$ lies in some family $\mathcal{P}$ of processes. Then, the learner can *construct* a simulator P for $\mathcal{P}$ and run either Algorithm 1 or 2. A natural choice of P is the (approximate) solution to the following minimax optimization problem:

$$\mathcal{R}(\mathcal{P}) := \min_{P} \max_{P^{\star} \in \mathcal{P}} \mathsf{KL}(P^{\star} \,\|\, P). \tag{10}$$

This quantity is known as *redundancy* in the information theory literature (Polyanskiy and Wu, 2025), and it characterizes the universal compression rate with sources coming from the family $\mathcal{P}$. This reconceptualizes our results through the lens of compression:

Learning is possible under universally compressible families of sources.

The quantity $\mathcal{R}(\mathcal{P})$ has well-known closed forms for many families of parametric processes (Polyanskiy and Wu, 2025; Atteson, 1999). Combining Theorem 4.1 and Lemma E.2, we have the following learnability result in terms of $\mathcal{R}(\mathcal{P})$.

Corollary 5.1 *Let $\mathcal{P}$ be a family of processes with $\mathcal{R}(\mathcal{P}) \le o(T)$, and $\mathcal{F}$ a bounded VC dimension class. Then, there exists a (possibly inefficient) algorithm that achieves sublinear agnostic regret for every $P^{\star} \in \mathcal{P}$. Moreover, for every $D \in \mathbb{N}$, there exists $\mathcal{P}$ with $\mathcal{R}(\mathcal{P}) \le D \wedge T$ and a VC class of dimension 1, such that any algorithm suffers regret $\Omega(D \wedge T)$ even in the realizable setting.*

To achieve sublinear regret in the above, it suffices to instantiate Algorithm 2 (see Appendix A) with the simulator P taken to be the (approximate) minimizer of Eq. (10); in the realizable case, we may instead run Algorithm 1 with P as the simulator. However, note that our framework requires efficient sampling (either conditional or unconditional) from P. As we show below, this requirement can be satisfied for many common families.

5.1. Learning Against Computationally Bounded Nature

Consider an instantiation of the above problem when $\mathcal{P}$ consists of all polynomial-time samplable processes. A priori this class seems too large to say anything interesting about. Surprisingly, we show that this is not the case. Specifically, in this section we show that, for any bound on nature's

running time, there exists an algorithm that, for any fixed VC dimension d, runs in time polynomial in T and in nature's running time, and achieves an optimal bound with respect to all such strategies of nature. To accomplish this, we use complexity-theoretic analogues of coding theorems that establish existence of universal compressors for time-bounded polynomial processes.

Specifically, consider an online learning game over a countable domain $\mathcal{X}$, which we take to be the set of all finite binary strings (i.e., $\mathcal{X} = \{0,1\}^*$). We let $\Sigma := \{0,1,\#\}$ be the alphabet, where $\#$ is a special delimiter symbol. In this section, we encode elements $\mathbf{x} \in \mathcal{X}^\infty$ as delimiter-separated infinite binary strings. Formally, for $x \in \mathcal{X}$, let $\langle x \rangle := x\#$, and, for any $\mathbf{x} \in \mathcal{X}^\infty$, let $\langle \mathbf{x} \rangle := \langle x_1 \rangle \langle x_2 \rangle \dots$ be the encoding. Accordingly, we view any distribution $P \in \Delta(\mathcal{X}^\infty)$ as a distribution on Σ^∞ and let $P_{1:t}$ denote the distribution of the encoded length-t prefix $\langle \mathbf{X}_{1:t} \rangle$.

The complexity-theoretic language we follow is similar to one used in Hirahara and Nanashima (2023). We begin by defining a notion of polytime samplable distributions. Let 1^t denote an all-1 string of length t. Intuitively, a distribution P is p-time samplable, if the first t elements can be sampled in $p(t)$ time. In the remainder, fix a universal prefix-free Turing machine U that has a write-only tape where the output is recorded. For a program $\pi \in \{0,1\}^*$ (with $|\pi| \geq 1$), and input $s \in \Sigma^*$, let $U^t(\pi, s)$ denote the contents of the output tape of U after simulating π on s for t steps.

Definition 5.2 (Time-bounded samplable distribution) *Let $p\colon \mathbb{N} \to \mathbb{N}$ be a time bound. Then, distribution $P \in \Delta(\Sigma^\infty)$ is p-time samplable iff there exists a program π such that, for every $t \geq 1$ and $s \sim \mathsf{Unif}(\{0,1\}^{p(t)})$, we have $U^{p(t)}(\pi, 1^t\#s) \sim P_{1:t}$. For a distribution $P \in \Delta(\Sigma^\infty)$, let*

$$K^p(P) = \min\left\{|\pi|\colon U^{p(t)}(\pi, 1^t\#s) \sim P_{1:t} \text{ for every } t \geq 1, \text{ where } s \sim \mathsf{Unif}(\{0,1\}^{p(t)})\right\}$$

be the (time-bounded) Kolmogorov complexity of P if P is p-time samplable.

It is well-appreciated in the Kolmogorov complexity literature that there exist *universal* distributions that dominate every other time-bounded samplable distribution (Antunes et al., 2006; Antunes and Fortnow, 2009; Lu et al., 2022). In particular, we prove a version of such a statement below, with Lemma 6.9 from Hirahara and Nanashima (2023) being the most direct analog (a similar statement is also implicit in Impagliazzo and Levin (1990)).

Theorem 5.3 (Universal samplable distribution) *Let $p\colon \mathbb{N} \to \mathbb{N}$ be a time-constructible bound. There exists a time bound $p'(t) = \mathrm{poly}(p(t))$ and a sequence of measures $\{\mu^t\}_{t\geq 1}$, with each $\mu^t \in \Delta(\Sigma^*)$ samplable in time $p'(t)$ that satisfy the following. For any p-time samplable distribution P and any $\mathbf{x} \in \mathcal{X}^t$ we have*

$$\log \frac{P_{1:t}(\langle \mathbf{x} \rangle)}{\mu^t(\langle \mathbf{x} \rangle)} \leq K^p(P) + 2\log K^p(P) + O(1).$$

For every T, the distributions μ^T defined above can be used as an approximate simulator for *all* nature's strategies that run in $p(T)$-time to generate $\mathbf{X}_{1:T}$. Note that this simulator controls the worst-case density ratio with respect to any time-bounded process, which is stronger than the bounded-KL assumption of Section 3. This can be further leveraged for efficiency. As such, it suffices to run Algorithm 1 with μ^T as the simulator and only a *single layer* ($L = 1$) to obtain the following.

Theorem 5.4 *Suppose the distribution $P^\star$ of nature is p-time samplable (as per Definition 5.2). Let $\mathcal{F}$ be any class of VC dimension d. Then, there exists an algorithm that, given access to a realizable ERM oracle w.r.t. $\mathcal{F}$, runs in time $\mathrm{poly}(T^d, p(T))$ and for any $f^\star \in \mathcal{F}$, achieves regret*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim d\log(T) + d \cdot 2^{K^p(P^\star) + 2\log K^p(P^\star)}.$$

We note that Theorems 3.1 and 5.3 also imply the existence of an $\exp(O(T))$-time algorithm that achieves regret $\mathbb{E}\,\mathbf{Reg} \lesssim d\log(T) + dK^p(P^\star)$; the key point of Theorem 5.4 is that its running time depends only polynomially on nature's running time. One might hope that conditional samplers (Theorem 4.1) could further improve the dependence on VC dimension, but this would require conditional sampling from μ, which is computationally hard under standard cryptographic assumptions since $\log \mu$ closely approximates time-bounded probabilistic Kolmogorov complexity (Goldberg et al., 2022; Hirahara and Nanashima, 2023; Liu and Pass, 2020).

5.2. Learning under Markov Chains: Linear Dynamical Systems

To exploit favorable runtime of algorithms that rely on conditional simulators from Section 4, we now consider examples where the family $\mathcal{P}$ has additional structure. Markov chains are a natural and ubiquitous class of structured processes. Here, we focus on learning under *linear dynamical systems*; in Appendix B we show that similar results can be established for Glauber dynamics. Suppose the family $\mathcal{P}$ consists of all processes that describe the evolution of an unknown linear dynamical system. More concretely, suppose $X_1, \dots, X_T \in \mathbb{R}^n$ evolve as

$$X_{t+1} = AX_t + \eta_t, \qquad \eta_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2 I_n),$$

with unknown parameter $A \in \Theta := [-B, B]^{n\times n}$, starting point $X_0 \in \mathcal{B}(0, R) := \{x \in \mathbb{R}^n : \|x\|_2 \le R\}$, and known noise scale $\sigma > 0$. We assume that X_0 is known; if it is not, we can treat X_1 as the starting point at a cost of one extra loss and a polynomial blow-up in $\|X_1\|_2$. Other than the basic entry-wise boundedness of A, we assume *marginal stability*, namely $\rho(A) \le 1$, where $\rho(\cdot)$ denotes the spectral radius. This includes the regime $\rho(A) = 1$, in which the chain has no stationary distribution and the mixing time is undefined (Simchowitz et al., 2018). Let P^A denote the law of $\mathbf{X}_{1:T}$ under these dynamics, and let $\pi = \mathcal{N}(0, B^2)^{n\times n}$ be the i.i.d. Gaussian prior. We instantiate our simulator as $P(\cdot \mid X_0) := \mathbb{E}_{A\sim\pi} P^A(\cdot \mid X_0)$. Then, it can be shown (see Lemma G.1 and Proposition G.3) that the above distribution controls the worst-case KL within the family of LDS described above, and, moreover, that conditional sampling from P can be implemented efficiently.

Theorem 5.5 *Let $\mathcal{F}$ be a class of VC dimension d, and let $R, B \ge 1$, $\sigma > 0$ and the initial state $X_0 \in \mathcal{B}(0, R)$ be arbitrary. There is a $\mathrm{poly}(n, T)$-time algorithm that, given access to an agnostic ERM oracle for $\mathcal{F}$, the noise scale σ, and the magnitude bound B, achieves expected regret*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT}\left(1 + \sqrt{n^3 \log\left(nBT(R/\sigma + 1)\right)}\right)$$

in the agnostic setting, simultaneously for every $P^\star = P^{A^\star}$ with $A^\star \in \Theta$ satisfying $\rho(A^\star) \le 1$.

Acknowledgments

We acknowledge support from NSF through awards DMS-2031883 and PHY-2019786, the DARPA AIQ program, and AFOSR FA9550-25-1-0375. SV acknowledges support from Amazon AI Research Innovation Fellowship. AS thanks Ankur Moitra for valuable discussions.

LLM usage. We used Claude Opus (4.6–4.8) to polish the exposition and to check the technical arguments for errors. In addition, some technical claims in Section 4 (in particular Lemmas F.2 and F.3) were developed through iterative work with the model.

References

Yasin Abbasi-Yadkori and Csaba Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 1–26, 2011.

Ishaq Aden-Ali, Mikael Møller Høgsgaard, Kasper Green Larsen, and Nikita Zhivotovskiy. Majority-of-three: The simplest optimal learner? In *The Thirty Seventh Annual Conference on Learning Theory*, pages 22–45. PMLR, 2024.

David Aldous and Umesh Vazirani. A markovian extension of valiant's learning model. *Information and Computation*, 117(2):181–186, 1995.

Jason M Altschuler and Kunal Talwar. Resolving the mixing time of the langevin algorithm to its stationary distribution for log-concave sampling. *arXiv preprint arXiv:2210.08448*, 2022.

Luis Antunes and Lance Fortnow. Worst-case running times for average-case algorithms. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 298–303. IEEE, 2009.

Luis Antunes, Lance Fortnow, Dieter Van Melkebeek, and N Variyam Vinodchandran. Computational depth: concept and applications. *Theoretical Computer Science*, 354(3):391–404, 2006.

Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(1):121–164, 2012.

Jan Arpe and Elchanan Mossel. Agnostically learning juntas from random walks. *arXiv preprint arXiv:0806.4210*, 2008.

Kevin Atteson. The asymptotic redundancy of bayes rules for markov chains. *IEEE Transactions on Information Theory*, 45(6):2104–2109, 1999.

Peter L Bartlett, Paul Fischer, and Klaus-Uwe Höffgen. Exploiting random walks for learning. In *Proceedings of the Seventh Annual Conference on Computational Learning Theory*, pages 318–327, 1994.

Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

Shai Ben-David, Eyal Kushilevitz, and Yishay Mansour. Online learning versus offline learning. *Machine Learning*, 29(1):45–63, 1997.

Shai Ben-David, Dávid Pál, and Shai Shalev-Shwartz. Agnostic online learning. In *Conference on Learning Theory*, volume 3, page 1, 2009.

Aditya Bhaskara, Ashok Cutkosky, Ravi Kumar, and Manish Purohit. Online learning with imperfect hints. In *International Conference on Machine Learning*, pages 822–831. PMLR, 2020.

Alankrita Bhatt, Nika Haghtalab, and Abhishek Shetty. Smoothed analysis of sequential probability assignment. *Advances in Neural Information Processing Systems*, 36:79808–79831, 2023.

Moïse Blanchard. Agnostic smoothed online learning. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 1997–2006, 2025.

Adam Block and Yury Polyanskiy. The sample complexity of approximate rejection sampling with applications to smoothed online learning. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 228–273. PMLR, 2023.

Adam Block, Yuval Dagan, Noah Golowich, and Alexander Rakhlin. Smoothed online learning is as easy as statistical learning. In *Conference on Learning Theory*, pages 1716–1786. PMLR, 2022.

Adam Block, Alexander Rakhlin, and Abhishek Shetty. On the performance of empirical risk minimization with smoothed data. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 596–629. PMLR, 2024.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

Guy Bresler, David Gamarnik, and Devavrat Shah. Learning graphical models from the glauber dynamics. *IEEE Transactions on Information Theory*, 64(6):4072–4080, 2017.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

Nader H Bshouty, Elchanan Mossel, Ryan O'Donnell, and Rocco A Servedio. Learning dnf from random walks. *Journal of Computer and System Sciences*, 71(3):250–265, 2005.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Gautam Chandrasekaran, Jason Gaitonde, Ankur Moitra, and Arsen Vasilyan. Learning AC^0 under graphical models. *arXiv preprint arXiv:2604.06109*, 2026.

Sinho Chewi. Log-concave sampling. *Book draft, available at* https://chewisinho.github.io, 2024.

Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In *Conference on Learning Theory*, pages 6.1–6.20. JMLR Workshop and Conference Proceedings, 2012.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023.

Alon Cohen, Avinatan Hassidim, Tomer Koren, Nevena Lazic, Yishay Mansour, and Kunal Talwar. Online linear quadratic control. In *International Conference on Machine Learning*, pages 1029–1038. PMLR, 2018.

B. Jack Copeland. The Church-Turing Thesis. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2026 edition, 2026.

Elisabetta Cornacchia, Dan Mikulincer, and Elchanan Mossel. The benefits of temporal correlations: Sgd learns k-juntas from random walks efficiently. *arXiv preprint arXiv:2605.10237*, 2026.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

Yuval Dagan, Constantinos Daskalakis, Nishanth Dikkala, and Siddhartha Jayanti. Learning from weakly dependent data under dobrushin's condition. In *Conference on Learning Theory*, pages 914–928. PMLR, 2019.

Michael Deistler, Jan Boelts, Peter Steinbach, Guy Moss, Thomas Moreau, Manuel Gloeckler, Pedro LC Rodrigues, Julia Linhart, Janne K Lappalainen, Benjamin Kurt Miller, et al. Simulation-based inference: A practical guide. *arXiv preprint arXiv:2508.12939*, 2025.

David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818): 97–117, 1985.

Mohamad Kazem Shirani Faradonbeh, Ambuj Tewari, and George Michailidis. Finite time identification in unstable linear systems. *Automatica*, 96:342–353, 2018.

Jason Gaitonde and Elchanan Mossel. A unified approach to learning ising models: Beyond independence and bounded width. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 503–514, 2024.

Jason Gaitonde, Ankur Moitra, and Elchanan Mossel. Better models and algorithms for learning Ising models from dynamics. *arXiv preprint arXiv:2507.15173*, 2025.

David Gamarnik. Extension of the pac framework to finite and countable markov chains. In *Proceedings of the twelfth annual conference on Computational learning theory*, pages 308–317, 1999.

Roy J Glauber. Time-dependent statistics of the Ising model. *Journal of Mathematical Physics*, 4 (2):294–307, 1963.

Surbhi Goel, Steve Hanneke, Shay Moran, and Abhishek Shetty. Adversarial resilience in sequential prediction via abstention. In *Advances in Neural Information Processing Systems*, volume 36, pages 8027–8047, 2023.

Halley Goldberg, Valentine Kabanets, Zhenjian Lu, and Igor C Oliveira. Probabilistic kolmogorov complexity with applications to average-case complexity. In *37th Computational Complexity Conference (CCC 2022)*, pages 16.1–16.60. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2022.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, pages 2404–2414. PMLR, 2019.

Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis of online and differentially private learning. *Advances in Neural Information Processing Systems*, 33:9203–9215, 2020.

Nika Haghtalab, Yanjun Han, Abhishek Shetty, and Kunhe Yang. Oracle-efficient online learning for beyond worst-case adversaries. *arXiv preprint arXiv:2202.08549*, 2022.

Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis with adaptive adversaries. *Journal of the ACM*, 71(3):1–34, 2024.

Demis Hassabis. Accelerating scientific discovery with AI. https://www.youtube.com/watch?v=UX8uIW9oIZk, 2024.

Demis Hassabis. Future of AI, simulating reality, physics and video games. https://www.youtube.com/watch?v=-HzgcbRXUK8, 2025.

Elad Hazan and Tomer Koren. The computational power of optimization in online learning. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 128–141, 2016.

Elad Hazan, Karan Singh, and Cyril Zhang. Learning linear dynamical systems via spectral filtering. In *Advances in Neural Information Processing Systems*, pages 6702–6712, 2017.

Elad Hazan, Holden Lee, Karan Singh, Cyril Zhang, and Yi Zhang. Spectral filtering for general linear dynamical systems. In *Advances in Neural Information Processing Systems*, pages 4639–4648, 2018.

Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free MCMC with amortized approximate ratio estimators. In *International Conference on Machine Learning*, pages 4239–4248. PMLR, 2020.

Shuichi Hirahara and Mikito Nanashima. Learning in pessiland via inductive inference. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 447–457. IEEE, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

Marcus Hutter, David Quarel, and Elliot Catt. *An introduction to universal artificial intelligence*. Chapman and Hall/CRC, 2024.

Russell Impagliazzo and Leonid A Levin. No better ways to generate hard np instances than picking uniformly at random. In *FOCS*, 1990.

Matthew O Jackson. *Social and economic networks*. Princeton university press, 2008.

Ali Jadbabaie, Alexander Rakhlin, Shahin Shahrampour, and Karthik Sridharan. Online optimization: Competing with dynamic comparators. In *International Conference on Artificial Intelligence and Statistics*, pages 398–406. PMLR, 2015.

Sham Kakade and Adam T Kalai. From batch to transductive online learning. *Advances in Neural Information Processing Systems*, 18, 2005.

Varun Kanade and Elchanan Mossel. MCMC learning. In *Conference on Learning Theory*, pages 1101–1128. PMLR, 2015.

Vitaly Kuznetsov and Mehryar Mohri. Generalization bounds for non-stationary mixing processes. *Machine Learning*, 106(1):93–117, 2017.

Leonid A Levin. Universal sequential search problems. *Problems of Information Transmission*, 9 (3):265–266, 1973.

Leonid A Levin. Randomness conservation inequalities; information and independence in mathematical theories. *Information and Control*, 61(1):15–37, 1984.

Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Texts in Computer Science. Springer, 4th edition, 2019.

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

Yanyi Liu and Rafael Pass. On one-way functions and Kolmogorov complexity. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1243–1254. IEEE, 2020.

Zhenjian Lu, Igor C Oliveira, and Marius Zimand. Optimal coding theorems in time-bounded Kolmogorov complexity. *arXiv preprint arXiv:2204.08312*, 2022.

Jan-Matthis Lueckmann, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Haipeng Luo and Robert E Schapire. Achieving all with no parameters: AdaNormalHedge. In *Conference on Learning Theory*, pages 1286–1304. PMLR, 2015.

Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *Journal of the ACM (JACM)*, 68(4):1–25, 2021.

Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012.

Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328, 2003.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. *Communications of the ACM*, 65(7):33–35, 2022.

Mehryar Mohri and Afshin Rostamizadeh. Stability bounds for stationary φ-mixing and β-mixing processes. *Journal of Machine Learning Research*, 11:789–814, 2010.

Omar Montasser, Abhishek Shetty, and Nikita Zhivotovskiy. Beyond worst-case online classification: Vc-based regret bounds for relaxed benchmarks. In *The Thirty Eighth Annual Conference on Learning Theory*, pages 4168–4202. PMLR, 2025.

George Papamakarios and Iain Murray. Fast ε-free inference of simulation models with Bayesian conditional density estimation. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Yury Polyanskiy and Yihong Wu. *Information theory: From coding to learning*. Cambridge university press, 2025.

Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Alexander Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In *Advances in Neural Information Processing Systems*, pages 3066–3074, 2013a.

Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Conference on Learning Theory*, pages 993–1019. PMLR, 2013b.

Alexander Rakhlin and Karthik Sridharan. Hierarchies of relaxations for online prediction problems with evolving constraints. In *Conference on Learning Theory*, pages 1457–1479. PMLR, 2015.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Beyond regret. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 559–594. JMLR Workshop and Conference Proceedings, 2011a.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Stochastic, constrained, and smoothed adversaries. *Advances in Neural Information Processing Systems*, 24, 2011b.

Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Relax and randomize: From value to algorithms. *Advances in Neural Information Processing Systems*, 25, 2012.

Vinod Raman and Ambuj Tewari. Online classification with predictions. *Advances in Neural Information Processing Systems*, 37:55884–55914, 2024.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022.

Sébastien Roch. On learning thresholds of parities and unions of rectangles in random walk models. *Random Structures & Algorithms*, 31(4):406–417, 2007.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, volume 35, pages 36479–36494, 2022.

Narayana P Santhanam and Martin J Wainwright. Information-theoretic limits of selecting binary graphical models in high dimensions. *IEEE Transactions on Information Theory*, 58(7):4117–4134, 2012.

Tuhin Sarkar and Alexander Rakhlin. Near optimal finite time identification of arbitrary linear dynamical systems. In *International Conference on Machine Learning*, pages 5610–5618. PMLR, 2019.

Abhishek Vasantha Shetty. *Learning in a Changing World: Covariate Shift, Subset Selection and Optimal PAC Bounds*. PhD thesis, University of California, Berkeley, 2024.

Max Simchowitz, Horia Mania, Stephen Tu, Michael I Jordan, and Benjamin Recht. Learning without mixing: Towards a sharp analysis of linear system identification. In *Conference On Learning Theory*, pages 439–473. PMLR, 2018.

Michael Sipser. A complexity theoretic approach to randomness. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pages 330–335, 1983.

Scott A Sisson, Yanan Fan, and Mark Beaumont. *Handbook of approximate Bayesian computation*. Chapman and Hall/CRC, 2018.

Ilya Sutskever. An observation on generalization. https://www.youtube.com/watch?v=AKMuA_TVz3A, 2023.

Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E. Schapire. Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems*, pages 2989–2997, 2015.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30. IEEE, 2017.

Vladimir Vapnik. *Statistical learning theory*, volume 3. Wiley, New York, 1998.

Vladimir Vapnik and Alexey Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

Vladimir Vapnik and Alexey Chervonenkis. Theory of pattern recognition, 1974.

Bin Yu. Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, 22(1):94–116, 1994.

Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 737–744. IEEE, 2020.

Appendix A. Relaxation-Based Algorithm With Epoching

Here, we formally present an oracle-efficient algorithm that attains the regret bound in Theorem 4.1. As discussed in Section 4, the algorithm combines the relaxation from Eq. (4) with epoching. Specifically, the algorithm partitions the horizon into K consecutive epochs of equal length $L := T/K$. On each round t, the learner uses the conditional simulator $\mathcal{O}_P^{\mathsf{cond}}$ to draw empirical samples of future covariates up to the end of the current epoch, and uses those for the estimation task in Eq. (9).

Algorithm 2: Log-MGF relaxations with epoching

input: Conditional sampling oracle $\mathcal{O}_P^{\mathsf{cond}}$, number of epochs K, inverse temperature $\eta > 0$

Let $L \leftarrow T/K$ be the length of each epoch // WLOG, assume T is a multiple of K

for $t \leftarrow 1$ **to** T **do**

  Let $k \leftarrow \lceil t/L \rceil$ // Index of the current epoch

  Observe X_t

  Sample $\widehat{\mathbf{W}}_i^t \sim P_{t+1:kL}(\cdot \mid \mathbf{X}_{1:t}) \otimes \mathsf{Unif}(\{\pm 1\}^{kL-t})$ for $i \in [N]$ // N calls to $\mathcal{O}_P^{\mathsf{cond}}$

  Compute $\mathfrak{G}_1(\widehat{\mathbf{W}}_i^t)$ and $\mathfrak{G}_0(\widehat{\mathbf{W}}_i^t)$ for $i \in [N]$ as in Eq. (8) for epoch k // $2N$ ERM calls

  Compute $\hat{q}_t$ as in Eq. (9) and predict $\hat{Y}_t \sim \hat{q}_t$

  Observe Y_t and suffer loss $\mathbb{1}\left[\hat{Y}_t \neq Y_t\right]$

end

Appendix B. Learning under Markov chains: Glauber dynamics

We consider the family of Glauber dynamics for the Ising model on n vertices (Glauber, 1963), which can be defined as follows. The Glauber dynamics trajectory consists of an initial configuration $X_0 \in \{0,1\}^n$ and a sequence of pairs $(\mathbf{I}, \mathbf{X}) \in ([n] \times \{0,1\}^n)^T$, generated by

$$\begin{aligned} I_t &\sim \mathsf{Unif}([n]), \\ X_{t,I_t} &\sim \mathsf{Ber}(\sigma(u_{I_t}^\theta(X_{t-1}))), \\ X_{t,j} &= X_{t-1,j} \quad \text{for } j \neq I_t. \end{aligned}$$

Here A is a symmetric, zero-diagonal $n \times n$ matrix and $h \in \mathbb{R}^n$, so that $\theta = (A, h) \in \Theta := [-B, B]^p$ with $p = \binom{n}{2} + n$. The local field is $u_i^\theta(x) = \sum_{j \neq i} A_{ij} x_j + h_i$. As in Section 5.2, we can, without loss of generality, assume that the initial state X_0 is known. Let P^θ denote the joint law of $(\mathbf{I}, \mathbf{X})$ under these dynamics. Following the large body of literature on learning under Glauber dynamics (Bresler et al., 2017), we will assume that the transition site I_t is observed even when the state is unchanged ($X_t = X_{t-1}$). Let π be the uniform distribution on Θ, and define the *simulator*

$$P := \mathbb{E}_{\theta \sim \pi} P^\theta(\cdot \mid X_0).$$

As in the LDS case, the above distribution controls the worst-case KL within the family of Glauber dynamics, and conditional sampling from P can be implemented efficiently via log-concave sampling (Chewi, 2024); see Lemma G.4 and Proposition G.5 for the proof.

Theorem B.1 *Let $\mathcal{F}$ be a class of VC dimension d, and let $B \geq 1$. There is a $\mathrm{poly}(n, T, B)$-time algorithm that, given access to an agnostic ERM oracle for $\mathcal{F}$, achieves expected regret*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT} \cdot \left(1 + \sqrt{n^2 \log(BTn)}\right)$$

in the agnostic setting, simultaneously for every $P^\star = P^{\theta^\star}$ with $\theta^\star \in \Theta$ and initial state $X_0 \in \{0,1\}^n$.

To our knowledge, this work is the first to consider learning arbitrary VC function classes when covariates evolve according to unknown Glauber dynamics. A conceptually similar work is that of Chandrasekaran et al. (2026), which studies learning under covariates generated by a graphical model; there, however, the examples are drawn i.i.d., whereas in our setting the covariates form a single, temporally dependent trajectory.

We recognize that the site observability assumption (observability of I_t) is slightly unnatural in our context. While the latest work on learning under Glauber dynamics removes it (Gaitonde et al., 2025), the sample complexity of parameter identification still has exponential dependence on the so-called ℓ_1-width (see Appendix C for more details). It is an interesting open question whether it is possible to establish a similar guarantee to Theorem B.1 without the site observability assumption.

Appendix C. Related Work

C.1. Beyond-Worst-Case Online Learning

Our work is broadly related to the line of work on beyond-worst-case analysis of sequential decision making (Shetty, 2024), notably smoothed analysis of online learning (Rakhlin et al., 2011b; Haghtalab et al., 2020, 2022; Bhatt et al., 2023; Haghtalab et al., 2024; Block et al., 2022, 2024; Blanchard, 2025). In smoothed online learning, the distribution of the adversarial covariates is restricted to have bounded density (w.r.t. some potentially unknown reference measure). Under this assumption, it is shown that all VC classes are learnable, often efficiently, in the same ERM oracle model as ours. The key conceptual difference is that smoothed online learning imposes a per-round constraint on the adversary via the density bound, whereas our model imposes a global constraint via the existence of an approximate simulator. In this sense, smoothed online learning should be thought of as *an approximation to independence*, while our framework allows for strongly dependent processes. On the other hand, smoothed online learning allows for adversarial labels, which our results generally do not (see Appendix H for a detailed discussion). Overall, the two models are complementary, capturing different ways of relaxing the adversarial nature of online learning. A similar comparison can be made with other recent models, such as learning with abstention (Goel et al., 2023) and learning with relaxed benchmarks (Montasser et al., 2025).

C.2. Learning with Side Information

A number of settings leverage *side information*, most notably transductive online learning, predictable sequences, and online algorithms with predictions. A common denominator of these settings is that the learner is typically given *deterministic* side information about the future, whereas the key technical challenge in our setting is that the learner must account for the *stochasticity* of the covariate sequence, which requires new algorithmic ideas based on the relaxation framework (Rakhlin et al., 2012; Rakhlin and Sridharan, 2015). This challenge arises from the belief that it is often

easier to find a simulator that captures the distribution of a process than to predict its individual future elements, which is the broad principle behind probabilistic modeling from statistical mechanics (Glauber, 1963; Metropolis et al., 1953) to generative AI (Brown et al., 2020; Ho et al., 2020).

Transductive online learning. Our framework can be viewed as a distributional generalization of transductive online learning (Ben-David et al., 1997; Kakade and Kalai, 2005). In classical transductive online learning, the learner receives the entire sequence of unlabeled covariates $(X_1, \dots, X_T)$ at the start of the game and then makes predictions sequentially as labels are revealed. This additional information allows all VC classes to be learned with regret $O(d \log T)$, circumventing the Littlestone-dimension barrier of adversarial online learning (Littlestone, 1988). Our setting generalizes this by replacing the deterministic sequence with *distributional* access: rather than knowing the exact sequence, the learner has access to a simulator for the covariate-generating distribution. Classical transductive learning corresponds to the degenerate case in which the simulator places all its mass on a single sequence.

Predictable sequences. Online learning under the assumption that there is some side information about future instances is typically studied under the name of predictable sequences. These arise naturally in settings such as learning in games and optimization (Rakhlin and Sridharan, 2013b,a; Chiang et al., 2012; Syrgkanis et al., 2015; Jadbabaie et al., 2015; Raman and Tewari, 2024; Bhaskara et al., 2020). Although conceptually related, these settings do not capture learning under simulatable processes, since they typically assume that *the realization* of future inputs is predictable in some sense, while we assume only *distributional* access. Furthermore, most results focus on improving the dependence of regret on the horizon T; in our setting, it is not obvious a priori which function classes are *learnable*, and our key contribution is showing that bounded VC dimension suffices for learnability.

Online algorithms with predictions. A closely related area is the emerging literature on online algorithms with predictions (Mitzenmacher and Vassilvitskii, 2022; Lykouris and Vassilvitskii, 2021; Purohit et al., 2018). In this setting, the algorithm is given (possibly imperfect) predictions about future inputs, and the goal is to design algorithms that perform well when predictions are accurate while maintaining worst-case guarantees. The key difference is that these predictions are typically point estimates of future inputs, whereas in our setting the learner has access to a distributional simulator. Furthermore, this line typically targets competitive-ratio guarantees for optimization problems, whereas we seek learning-theoretic regret bounds for classification.

C.3. Learning under Stochastic Processes

Dynamical systems and control. Learning under dynamical processes is a rich area with close ties to reinforcement learning and control (Abbasi-Yadkori and Szepesvári, 2011; Cohen et al., 2018). A notable direction is the work on learning under linear dynamics (Hazan et al., 2017, 2018), which studies online prediction of the outputs of a latent LDS. We instead focus on the classification setting with general VC classes, and assume access to a simulator rather than imposing structural assumptions on the dynamics. A related use of simulators is sim-to-real transfer (Tobin et al., 2017; Zhao et al., 2020), where simulators are used to train policies that are then deployed in the real world. The central challenge there is the "reality gap" between simulator and reality, which our framework quantifies via KL divergence.

Learning under mixing processes. A long line of work instead learns under processes that satisfy strong mixing properties (Yu, 1994; Mohri and Rostamizadeh, 2010; Kuznetsov and Mohri, 2017; Gamarnik, 1999). The key idea is that if the process mixes sufficiently fast, the dependencies between observations decay, and one can recover generalization bounds similar to the i.i.d. setting. For example, under β-mixing or ϕ-mixing assumptions, PAC-style bounds can be established with rates depending on the mixing coefficients (Mohri and Rostamizadeh, 2010). Dagan et al. (2019) relax mixing via a Dobrushin-style condition, which is another way to quantify weak dependence in the data by bounding pairwise influences of the samples in the dataset. Our framework differs significantly: the data can have arbitrary dependence structure and we impose no mixing assumptions.

Learning without mixing. A complementary line of work learns from a single trajectory of a structured (and potentially non-mixing) process, such as a random walk or Markov chain over the instance space, typically for specific function classes such as DNF or juntas (Aldous and Vazirani, 1995; Bartlett et al., 1994; Bshouty et al., 2005; Roch, 2007; Arpe and Mossel, 2008; Kanade and Mossel, 2015; Cornacchia et al., 2026). Notably, in this line of work, the temporal dependence is often a *resource* rather than an obstacle (e.g., for k-juntas, consecutive covariates that differ in one bit can reveal the relevant coordinates). Our work differs in scope: rather than a fixed class such as DNF or juntas under a specific process, we compete with an arbitrary VC class under a general simulatable process.

Our results for learning under linear dynamical systems (Theorem 5.5) are related to, although distinct from, the line of work of Hazan et al. (2017, 2018) on online prediction in linear dynamical systems. There, the learner observes a stream of inputs and predicts the outputs of a latent LDS whose hidden state evolves linearly, competing with the best such system. In our setting the state X_t is instead observed, and the learner competes with the best predictor in any given VC class. As such, the resulting guarantees are incomparable.

Relatedly, a large literature studies *estimation* of parameters of structured processes, such as Glauber dynamics (Bresler et al., 2017; Gaitonde et al., 2025; Gaitonde and Mossel, 2024) and linear dynamical systems (Simchowitz et al., 2018; Sarkar and Rakhlin, 2019; Faradonbeh et al., 2018); our results are orthogonal, since they concern learning a labeling function that acts *on top of* structured dynamics. Nevertheless, it is interesting to observe that our results for these settings permit learning even when the process parameters are hard to estimate. Indeed, prior works that estimate the parameters of the LDS rely on additional identifiability conditions (Simchowitz et al., 2018; Sarkar and Rakhlin, 2019; Faradonbeh et al., 2018), e.g., requiring the least eigenvalue of the system's Gramian or the least singular value of $A^\star$ to be bounded away from zero, whereas our result requires marginal stability of the system (i.e., $\rho(A^\star) \leq 1$). Similarly, approaches to estimating the parameters of Glauber dynamics have exponential dependence on the ℓ_1-width, defined as $\max_k \sum_{i \neq k} |A_{ik}| + |h_k|$ (Bresler et al., 2017; Gaitonde et al., 2025; Gaitonde and Mossel, 2024); such dependence is moreover information theoretically necessary for estimation (Santhanam and Wainwright, 2012). In contrast, our results in Theorem B.1 depend only *logarithmically* on the magnitudes of entries of A and h.

C.4. Learning and Kolmogorov Complexity

The connection between learning and Kolmogorov complexity has a rich history (Li and Vitányi, 2019), dating back to Kolmogorov and Solomonoff. Solomonoff introduced his method, *Solomonoff induction*, which uses a prior based on (unbounded) Kolmogorov complexity, and argued that it can

be used to learn computable processes. For a detailed exposition of these methods, see (Hutter et al., 2024; Li and Vitányi, 2019; Lu et al., 2022). Though these techniques are conceptually related to the approach we take, a key difference is that methods based on Solomonoff induction typically require *evaluating* Kolmogorov complexity, which can be computationally prohibitive (in the time-bounded version (Liu and Pass, 2020)) or even uncomputable (in the unbounded version). We circumvent these issues using the fact that we only need to *approximately sample* from the (time-bounded) universal distribution as opposed to computing its density. In fact, computing the density of universal distributions is closely related to cryptography: roughly speaking, being efficiently computable is equivalent to the nonexistence of one-way functions (Liu and Pass, 2020).

In this work, we use the time-bounded variant of Kolmogorov complexity (Sipser, 1983; Levin, 1984). Hirahara and Nanashima (2023) made the link between time-bounded Kolmogorov complexity and learning explicit, showing that under "Pessiland" assumptions (most notably, that one-way functions *do not* exist) arbitrary sequences generated by time-bounded processes are learnable. Their proof crucially relies on the nonexistence of one-way functions to obtain an efficient conditional sampler. As discussed above, under cryptographic assumptions, such a conditional sampler is provably infeasible.

C.5. Simulation-Based Inference

Simulation-based inference (SBI), also known as likelihood-free inference, is a classical area in statistics concerned with performing inference when the likelihood function is intractable but a simulator for the data-generating process is available (Marin et al., 2012; Sisson et al., 2018; Cranmer et al., 2020; Deistler et al., 2025). The canonical example is Approximate Bayesian Computation (ABC) (Beaumont et al., 2002; Marjoram et al., 2003), which performs approximate posterior inference by comparing simulated data to observed data via summary statistics. More recently, neural-network-based approaches have emerged that learn density estimators or likelihood ratios from simulated data (Papamakarios and Murray, 2016; Lueckmann et al., 2017; Greenberg et al., 2019; Hermans et al., 2020). Our work shares SBI's core assumption, which is access to a simulator but not the likelihood, and the KL divergence in our bounds can be seen as analogous to the "simulation gap" in SBI. However, the goals in the two settings are different: SBI performs parameter inference, whereas we perform online prediction with the objective of minimizing regret.

Appendix D. Further Preliminaries

D.1. Learning-Theoretic Preliminaries

We begin by recalling the VC dimension, a central combinatorial complexity measure in statistical learning theory (Vapnik, 1998).

Definition D.1 (VC dimension) *For a binary class $\mathcal{F} \subset \{0,1\}^{\mathcal{X}}$, the VC dimension of $\mathcal{F}$ is the largest integer d for which there exists a subset $\{x_1, \dots, x_d\} \subset \mathcal{X}$ such that, for any labels $y_1, \dots, y_d \in \{0,1\}$, there exists $f \in \mathcal{F}$ such that $f(x_t) = y_t$ for every $t \in [d]$.*

A classical theorem states that PAC learnability is equivalent to finiteness of the VC dimension. For our purposes, a useful consequence of bounded VC dimension is the celebrated Sauer–Shelah lemma. It upper bounds the number of realizable labelings on a subset of a domain in terms of the VC dimension.

Lemma D.2 (Sauer–Shelah Lemma) *For a binary class $\mathcal{F}$ with VC dimension d, and any finite set $S \subset \mathcal{X}$, the number of distinct labelings of S by functions in $\mathcal{F}$ is at most*

$$\sum_{i=0}^{d} \binom{|S|}{i} \leq \left(\frac{e|S|}{d}\right)^{d}.$$

We now turn to the online learning setting and recall the Littlestone dimension, an online analogue of the VC dimension. A binary tree $\mathbf{x}$ of depth d on the domain $\mathcal{X}$ is a sequence of mappings $\mathbf{x}_t \colon \{\pm 1\}^{t-1} \to \mathcal{X}$ for $t \in [d]$.

Definition D.3 (Littlestone Dimension) *Let $\mathcal{F} \subset \{0,1\}^{\mathcal{X}}$ be a binary class of functions. The Littlestone dimension is the largest integer d such that the following holds. There exists a tree $\mathbf{x}$ such that, for any choice of signs $\varepsilon_1, \ldots, \varepsilon_d \in \{\pm 1\}$ there exists a function $f \in \mathcal{F}$ such that*

$$f(\mathbf{x}_t(\varepsilon_1, \ldots, \varepsilon_{t-1})) = \frac{\varepsilon_t + 1}{2},$$

for every $t \in [d]$.

Finite Littlestone dimension characterizes learnability in classical online learning. Conversely, when the Littlestone dimension is infinite, one can embed arbitrarily deep Littlestone trees and force any learner to incur linear regret; we record a standard lower bound in the realizable case.

Proposition D.4 *Let $\mathcal{F} \subset \{0,1\}^{\mathcal{X}}$ be a class of infinite Littlestone dimension, and let $\mathbf{x}$ be a depth-T Littlestone tree. Consider nature's strategy that first samples $\varepsilon_t \sim \mathsf{Unif}(\{\pm 1\})$ independently for each $t \in [T]$. Then, at step $t \in [T]$, nature presents the learner with $X_t = \mathbf{x}_t(\varepsilon_1, \ldots, \varepsilon_{t-1})$ labeled by $Y_t := (\varepsilon_t + 1)/2$. Then, any learner suffers $T/2$ regret in this setting.*

The proof is based on the realizability of every root-to-leaf path in a Littlestone tree: for each path, there exists a function in $\mathcal{F}$ that matches the induced labels along that path. We will refer to such a function as being *consistent* with the path.

Definition D.5 *Let $\mathbf{x}$ be a depth-T tree. We say that a function f is consistent with the path $(\varepsilon_1, \ldots, \varepsilon_T)$ in $\mathbf{x}$ iff*

$$f(\mathbf{x}_t(\varepsilon_1, \ldots, \varepsilon_{t-1})) = \frac{\varepsilon_t + 1}{2}.$$

D.2. Learning with ERM Oracles

We adopt an ERM oracle computation model inspired by Hazan and Koren (2016). We distinguish between realizable and agnostic ERM oracles. Both oracles accept as input a labeled dataset, $S \subset \mathcal{X} \times \{0,1\}$. When called on S, the realizable ERM oracle returns a function $f \in \mathcal{F}$ that realizes S (that is, $f(x) = y$ for every $(x,y) \in S$) with ties broken adversarially; if no such function exists, the oracle returns "not realizable." When called on S, the agnostic ERM oracle returns a function f that makes the least number of mistakes, that is, any function in the set:

$$\operatorname*{argmin}_{f \in \mathcal{F}} \sum_{(x,y) \in S} \mathbb{1}\{f(x) \neq y\}.$$

We assume both oracles run in unit time, but it costs time to construct the input set S. We also assume that, given a function class $\mathcal{F}$, any $f \in \mathcal{F}$ can be evaluated efficiently on any $x \in \mathcal{X}$.

D.3. Miscellaneous Lemmata

We collect a few technical inequalities for KL divergence that will be used repeatedly. The first is a simple change of measure bound (Example 7.3 of Polyanskiy and Wu (2025)).

Lemma D.6 *For any $P, Q \in \Delta([0,1])$, we have*

$$\mathbb{E}_Q X \le 3\,\mathbb{E}_P X + 4\mathsf{KL}(P \,\|\, Q).$$

Next, we prove a Markov-type inequality for the density ratio, with a bound expressed in terms of $\mathsf{KL}(P \,\|\, Q)$.

Lemma D.7 *For any P, Q, $\lambda > e$ we have*

$$\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} > \lambda\right] \le \frac{\mathsf{KL}(P \,\|\, Q)}{\log(\lambda/e)}$$

Proof By definition of KL, we have

$$\begin{aligned}
\mathsf{KL}(P \,\|\, Q) &= \mathbb{E}_P \log \frac{\mathrm{d}P}{\mathrm{d}Q} \\
&= \mathbb{E}_P \log \frac{\mathrm{d}P}{\mathrm{d}Q} \mathbb{1}\left\{\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right\} + \mathbb{E}_P \log \frac{\mathrm{d}P}{\mathrm{d}Q} \mathbb{1}\left\{\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right\} \\
&\ge \log(\lambda)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right] + \mathbb{E}_P \log \frac{\mathrm{d}P}{\mathrm{d}Q} \mathbb{1}\left\{\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right\}. \qquad (11)
\end{aligned}$$

It remains to lower bound the second term above. By change of measure, and using $u\log(u) \ge \log(e)(u-1)$, we have:

$$\begin{aligned}
\mathbb{E}_P \log \frac{\mathrm{d}P}{\mathrm{d}Q} \mathbb{1}\left\{\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right\} &= \mathbb{E}_Q \frac{\mathrm{d}P}{\mathrm{d}Q} \log \frac{\mathrm{d}P}{\mathrm{d}Q} \mathbb{1}\left\{\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right\} \\
&\ge \mathbb{E}_Q \log(e)\left(\frac{\mathrm{d}P}{\mathrm{d}Q} - 1\right) \mathbb{1}\left\{\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right\} \\
&= \log(e)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right] - \log(e)\mathbb{P}_Q\left[\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right] \\
&= \log(e) - \log(e)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right] - \log(e)\mathbb{P}_Q\left[\frac{\mathrm{d}P}{\mathrm{d}Q} < \lambda\right] \\
&\ge -\log(e)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right].
\end{aligned}$$

Plugging this into Eq. (11), we have

$$\begin{aligned}
\mathsf{KL}(P \,\|\, Q) &\ge \log(\lambda)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right] - \log(e)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right] \\
&= \log(\lambda/e)\mathbb{P}_P\left[\frac{\mathrm{d}P}{\mathrm{d}Q} \ge \lambda\right].
\end{aligned}$$

Rearranging the above concludes the proof. ■

Finally, we bound the expectation of the positive part of the log density ratio by a KL term.

Lemma D.8 *For any P, Q we have*

$$\mathbb{E}_P \log^+ \left(\frac{\mathrm{d}P}{\mathrm{d}Q} \right) \leq \mathsf{KL}(P \parallel Q) + 1.$$

Proof By definition of KL, we have

$$\mathbb{E}_P \log^+ \left(\frac{\mathrm{d}P}{\mathrm{d}Q} \right) = \mathsf{KL}(P \parallel Q) - \mathbb{E}_P \log \left(\frac{\mathrm{d}P}{\mathrm{d}Q} \right) \mathbb{1} \left\{ \frac{\mathrm{d}P}{\mathrm{d}Q} \leq 1 \right\}. \tag{12}$$

Thus, it suffices to upper bound the second term. By a change of measure argument and using $u \log(u) \geq -\log(e)/e$ for any $u \geq 0$, we have

$$\begin{aligned} \mathbb{E}_P \log \left(\frac{\mathrm{d}P}{\mathrm{d}Q} \right) \mathbb{1} \left\{ \frac{\mathrm{d}P}{\mathrm{d}Q} \leq 1 \right\} &= \mathbb{E}_Q \frac{\mathrm{d}P}{\mathrm{d}Q} \log \left(\frac{\mathrm{d}P}{\mathrm{d}Q} \right) \mathbb{1} \left\{ \frac{\mathrm{d}P}{\mathrm{d}Q} \leq 1 \right\} \\ &\geq -\frac{\log(e)}{e}. \end{aligned}$$

Plugging this into Eq. (12), we obtain:

$$\begin{aligned} \mathbb{E}_P \log^+ \left(\frac{\mathrm{d}P}{\mathrm{d}Q} \right) &\leq \mathsf{KL}(P \parallel Q) + \frac{\log(e)}{e} \\ &\leq \mathsf{KL}(P \parallel Q) + 1, \end{aligned}$$

as desired. ■

Appendix E. Proofs from Section 3

E.1. Proof of Theorem 3.1

We begin by proving Lemma 3.2.

Lemma 3.2 *Let $\mathbf{Z} = (Z_1, \ldots, Z_N) \sim \bar{P}^{\otimes N}$, and $\mathcal{G}(\mathbf{Z})$ be an improper cover of $\mathcal{F}$ constructed as follows*

$$\mathcal{G}(\mathbf{Z}) := \left\{ \mathsf{Alg}_{\mathsf{M3}} \left((Z_i, y_i)_{i \in [N]} \right) : (y_i)_{i \in [N]} \in \mathcal{F}|_{\mathbf{Z}} \right\}. \tag{3}$$

Then, for any $f \in \mathcal{F}$, we have $\mathbb{E}_{\mathbf{Z}} \mathbb{E}_{\mathbf{X} \sim P} \min_{g \in \mathcal{G}(\mathbf{Z})} \|f - g\|_{\mathbf{X}} \lesssim \frac{d}{N}$.

Proof Let $f \in \mathcal{F}$ be arbitrary. From the construction of $\mathcal{G}(\mathbf{Z})$, it contains a function:

$$g := \mathsf{Alg}_{\mathsf{M3}} \left((Z_i, f(Z_i))_{i \in [N]} \right).$$

Using the fact that $\mathbf{Z}$ is sampled in an i.i.d. fashion from $\bar{P}$, and using a PAC-learning guarantee for the Majority-of-Three learner from Aden-Ali et al. (2024), we have:

$$\mathbb{E}_{\mathbf{Z}} \mathbb{P}_{X \sim \bar{P}} \left[f(X) \neq g(X) \right] \lesssim \frac{d}{N}.$$

Thus,

$$\mathbb{E}_{\mathbf{Z}} \min_{g \in \mathcal{G}(\mathbf{Z})} \mathbb{P}_{X \sim \bar{P}} \left[f(X) \neq g(X) \right] \lesssim \frac{d}{N}.$$

It remains to note that

$$\begin{aligned}\mathbb{E}_{\mathbf{Z}} \min_{g\in\mathcal{G}(\mathbf{Z})} \mathbb{E}_{\mathbf{X}\sim P} \|f-g\|_{\mathbf{X}} &= \mathbb{E}_{\mathbf{Z}} \min_{g\in\mathcal{G}(\mathbf{Z})} \mathbb{E}_{\mathbf{X}\sim P} \frac{1}{T}\sum_{t=1}^{T} |f(X_t)-g(X_t)| \\ &= \mathbb{E}_{\mathbf{Z}} \min_{g\in\mathcal{G}(\mathbf{Z})} \mathbb{E}_{X\sim \bar{P}} |f(X)-g(X)| \\ &= \mathbb{E}_{\mathbf{Z}} \min_{g\in\mathcal{G}(\mathbf{Z})} \mathbb{P}_{X\sim \bar{P}} [f(X)\neq g(X)] \\ &\lesssim \frac{d}{N}.\end{aligned}$$

The proof is concluded via Jensen's inequality:

$$\mathbb{E}_{\mathbf{Z}}\, \mathbb{E}_{\mathbf{X}\sim P} \min_{g\in\mathcal{G}(\mathbf{Z})} \|f-g\|_{\mathbf{X}} \le \mathbb{E}_{\mathbf{Z}} \min_{g\in\mathcal{G}(\mathbf{Z})} \mathbb{E}_{\mathbf{X}\sim P} \|f-g\|_{\mathbf{X}} \lesssim \frac{d}{N}.$$

■

The proof of Theorem 3.1 follows.

Theorem 3.1 *Let $\mathcal{F}$ be a VC dimension d class, and let $P^\star$ be an arbitrary distribution in $\Delta(\mathcal{X}^T)$. Let $N\in\mathbb{N}$ be arbitrary. Then, there exists an algorithm that makes at most N oracle calls to $\mathcal{O}_P$, at most $N^{O(d)}$ calls to a realizable ERM oracle w.r.t. $\mathcal{F}$, and achieves regret*

$$\mathbb{E}\,\mathbf{Reg} \lesssim d\log T + \frac{dT}{N} + d\cdot \mathsf{KL}(P^\star \,\|\, P) + \frac{T}{\log(1+N/T)}\cdot \mathsf{KL}(P^\star \,\|\, P).$$

Proof Note that we may WLOG assume that $N\le 2^{O(T)}$ and N is larger than some prespecified constant. Indeed, for $N\le O(1)$, the theorem statement is vacuous, and for N larger than $2^{O(T)}$, the statement follows from $N = T2^T$. Set $L := \lfloor \log\log(N/2)\rfloor$, and let

$$N_\ell = \lfloor (N/2)^{\frac{1}{2^{L-\ell}}}\rfloor,$$

for each $\ell\in[L]$. Then, $L\lesssim \log(T)$. Also, as long as N is larger than some constant, we have the following properties: (i) $\sum_{\ell\in[L]} N_\ell \le N$, (ii) $N_L \asymp N$, (iii) $N_1 \asymp 1$, and (iv) $N_{\ell+1}\lesssim N_\ell^2$. Let $\{E_\ell\}_{\ell\in[L+1]}$ be the following collection of events that partition the probability space. Set

$$E_1 := \left\{\frac{\mathrm{d}P^\star}{\mathrm{d}P} \le \frac{N_1}{T}\right\},$$

for each $2\le\ell\le L$, set

$$E_\ell := \left\{\frac{N_{\ell-1}}{T} < \frac{\mathrm{d}P^\star}{\mathrm{d}P} \le \frac{N_\ell}{T}\right\},$$

and let

$$E_{L+1} := \left\{\frac{N_L}{T} < \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right\}.$$

We use small-loss bounds for expert algorithms with a non-uniform prior. Recall that we initialize the weights in Line 6 of Algorithm 1 as $w_1(g) = 1/(L|\mathcal{G}_\ell|)$ for each $g\in\mathcal{G}_\ell$. Then, by Theorem 2.4

in Arora et al. (2012), for $\eta = 1/2$, the regret of our algorithm is upper bounded with probability 1 by

$$\begin{aligned}
\mathbf{Reg} &\lesssim \min_{g\in\mathcal{G}} \left(\log(1/w_1(g)) + \sum_{t=1}^{T} \mathbb{1}\left\{f^\star(X_t) \neq g(X_t)\right\} \right) \\
&= \min_{\ell\in[L]} \min_{g\in\mathcal{G}_\ell} \left(\log(1/w_1(g)) + \sum_{t=1}^{T} \mathbb{1}\left\{f^\star(X_t) \neq g(X_t)\right\} \right) \\
&= \min_{\ell\in[L]} \min_{g\in\mathcal{G}_\ell} \left(\log(L|\mathcal{G}_\ell|) + \sum_{t=1}^{T} \mathbb{1}\left\{f^\star(X_t) \neq g(X_t)\right\} \right) \\
&\lesssim \min_{\ell\in[L]} \min_{g\in\mathcal{G}_\ell} \left(\log(LN_\ell^d) + \sum_{t=1}^{T} \mathbb{1}\left\{f^\star(X_t) \neq g(X_t)\right\} \right) \\
&= \min_{\ell\in[L]} \left(\log(LN_\ell^d) + \min_{g\in\mathcal{G}_\ell} \sum_{t=1}^{T} \mathbb{1}\left\{f^\star(X_t) \neq g(X_t)\right\} \right) \\
&\leq \min_{\ell\in[L]} \left(d\log(N_\ell) + T \cdot \min_{g\in\mathcal{G}_\ell} \|f^\star - g\|_{\mathbf{X}} \right) + \log L
\end{aligned}$$

where the second inequality uses the Sauer–Shelah lemma (Lemma D.2). Now, since $\{E_\ell\}_{\ell\in[L+1]}$ partition the probability space, we have:

$$\begin{aligned}
\mathbf{Reg} &\leq \sum_{\ell\in[L+1]} \left(d\log(N_{\ell\wedge L}) + T \cdot \min_{g\in\mathcal{G}_{\ell\wedge L}} \|f^\star - g\|_{\mathbf{X}} \right) \mathbb{1}\{E_\ell\} + \log L \\
&\leq \sum_{\ell\in[L+1]} d\log(N_{\ell\wedge L})\mathbb{1}\{E_\ell\} + T \sum_{\ell\in[L+1]} \min_{g\in\mathcal{G}_{\ell\wedge L}} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\{E_\ell\} + \log L. \qquad (13)
\end{aligned}$$

Recall from Line 2 of Algorithm 1 that Z_i^ℓ are sampled from $\bar{P}$ independently for each $\ell \in [L]$ and $i \in [N_\ell]$. Let $N_{\mathsf{tot}} := \sum_{\ell\in[L]} N_\ell \leq N$, and let $\mathbf{Z} := \{Z_i^\ell\}_{\ell\in[L], i\in[N_\ell]}$. Then, $\mathbf{Z} \sim \bar{P}^{\otimes N_{\mathsf{tot}}}$, and $\mathbf{Z}^\ell \sim \bar{P}^{\otimes N_\ell}$ for each $\ell \in [L]$. Using Lemma 3.2, for any $\ell \in [L]$, we have

$$\mathbb{E}_{\mathbf{Z}^\ell} \mathbb{E}_{\mathbf{X}\sim P} \min_{g\in\mathcal{G}_\ell} \|f^\star - g\|_{\mathbf{X}} \lesssim \frac{d}{N_\ell}.$$

In the remainder, we suppress the expectation under $\mathbf{Z}$, and write $\mathbb{E}_{P^\star}$ and $\mathbb{E}_P$ to denote expectations over $(\mathbf{X}, \mathbf{Z}) \sim P^\star \otimes \bar{P}^{\otimes N_{\mathsf{tot}}}$ and $(\mathbf{X}, \mathbf{Z}) \sim P \otimes \bar{P}^{\otimes N_{\mathsf{tot}}}$ respectively for notational clarity. Then, note that, for all $\ell \in [L]$, we have

$$\begin{aligned}
\mathbb{E}_{P^\star} \min_{g\in\mathcal{G}_\ell} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\{E_\ell\} &\leq \frac{N_\ell}{T} \mathbb{E}_P \min_{g\in\mathcal{G}_\ell} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\{E_\ell\} \\
&\leq \frac{N_\ell}{T} \mathbb{E}_P \min_{g\in\mathcal{G}_\ell} \|f^\star - g\|_{\mathbf{X}} \\
&\lesssim \frac{N_\ell}{T} \cdot \frac{d}{N_\ell} \\
&= \frac{d}{T}.
\end{aligned}$$

For the final approximation term, we have by Lemma D.7, for any $\lambda > e$

$$\begin{aligned}
\mathbb{E}_{P^\star} \min_{g \in \mathcal{G}_L} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\{E_{L+1}\} &\le \mathbb{E}_{P^\star} \min_{g \in \mathcal{G}_L} \|f^\star - g\|_{\mathbf{X}} \\
&\le \mathbb{E}_{P^\star} \min_{g \in \mathcal{G}_L} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\left\{\frac{\mathrm{d}P^\star}{\mathrm{d}P} \le \lambda\right\} + \mathbb{P}_{P^\star}\left[\frac{\mathrm{d}P^\star}{\mathrm{d}P} > \lambda\right] \\
&\le \lambda \, \mathbb{E}_P \min_{g \in \mathcal{G}_L} \|f^\star - g\|_{\mathbf{X}} + \frac{\mathsf{KL}(P^\star \,\|\, P)}{\log(\lambda/e)} \\
&\lesssim \lambda \frac{d}{N_L} + \frac{\mathsf{KL}(P^\star \,\|\, P)}{\log(\lambda/e)}.
\end{aligned}$$

Set $\lambda = (N_L/T) + e$. Then,

$$\mathbb{E}_{P^\star} \min_{g \in \mathcal{G}_L} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\{E_{L+1}\} \lesssim \frac{d}{T} + \frac{d}{N_L} + \frac{\mathsf{KL}(P^\star \,\|\, P)}{\log\left(1 + \frac{N_L}{eT}\right)}$$

Thus, the total approximation term is upper bounded as:

$$\begin{aligned}
\sum_{\ell \in [L+1]} \mathbb{E}_{P^\star} \min_{g \in \mathcal{G}_{\ell \wedge L}} \|f^\star - g\|_{\mathbf{X}} \mathbb{1}\{E_\ell\} &\lesssim \frac{dL}{T} + \frac{d}{N_L} + \frac{\mathsf{KL}(P^\star \,\|\, P)}{\log\left(1 + \frac{N_L}{eT}\right)} \\
&\lesssim \frac{dL}{T} + \frac{d}{N} + \frac{\mathsf{KL}(P^\star \,\|\, P)}{\log\left(1 + \frac{N}{T}\right)},
\end{aligned} \tag{14}$$

where in the last line we used $N_L \asymp N$. Recall that, from the way N_ℓ are selected, we have $N_\ell \lesssim N_{\ell-1}^2$ for every $2 \le \ell \le L$. Thus, on event E_ℓ for $2 \le \ell \le L$, we have

$$\log(N_\ell) \lesssim \log(N_{\ell-1}) + 1 \le \log\left(T \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right) + 1$$

Also, on event E_{L+1}, we have

$$\log N_L \le \log\left(T \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right).$$

Thus, for every $\ell \ge 2$, we have

$$d \log(N_{\ell \wedge L}) \mathbb{1}\{E_\ell\} \lesssim d \log\left(T \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right) \mathbb{1}\{E_\ell\} + d\mathbb{1}\{E_\ell\}$$

For $\ell = 1$, we simply have:

$$d \log(N_1) \mathbb{1}\{E_1\} \lesssim d\mathbb{1}\{E_1\}.$$

Then, summing over all ℓ, we have:

$$\begin{aligned}
\sum_{\ell \in [L+1]} d \log(N_{\ell \wedge L}) \mathbb{1}\{E_\ell\} &\lesssim d \sum_{\ell \ge 1} \mathbb{1}\{E_\ell\} + d \log\left(T \cdot \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right) \mathbb{1}\left\{\bigcup_{\ell \ge 2} E_\ell\right\} \\
&\lesssim d + d \log^+\left(T \cdot \frac{\mathrm{d}P^\star}{\mathrm{d}P}\right)
\end{aligned}$$

$$\lesssim d\log(T) + d\log^+\left(\frac{\mathrm{d}P^\star}{\mathrm{d}P}\right).$$

Taking expectations, we have

$$\begin{aligned}\mathbb{E}_{P^\star}\sum_{\ell\in[L+1]} d\log(N_{\ell\wedge L})\mathbb{1}\{E_\ell\} &\lesssim d\log(T) + d\,\mathbb{E}_{P^\star}\log^+\left(\frac{\mathrm{d}P^\star}{\mathrm{d}P}\right)\\ &\lesssim d\log(T) + d\mathsf{KL}(P^\star \,\|\, P), \end{aligned} \tag{15}$$

where in the second step we used Lemma D.8. To conclude the proof, we combine Eqs. (13) to (15):

$$\begin{aligned}\mathbb{E}_{P^\star}\mathbf{Reg}(\mathcal{F},T) &\lesssim d\log T + d\mathsf{KL}(P^\star \,\|\, P) + dL + \frac{dT}{N} + T\frac{\mathsf{KL}(P^\star \,\|\, P)}{\log\left(1+\frac{N}{T}\right)}\\ &\lesssim d\log T + d\mathsf{KL}(P^\star \,\|\, P) + \frac{dT}{N} + T\frac{\mathsf{KL}(P^\star \,\|\, P)}{\log\left(1+\frac{N}{T}\right)},\end{aligned}$$

where the second inequality holds since $L \lesssim \log(T)$. This concludes the proof. ■

E.2. Proof of Theorem 3.3

Theorem 3.3 *For any $d \in \mathbb{N}$, $N \geq 2$, and $D \in \mathbb{N}\cup\{0\}$, there exists a class $\mathcal{F}$ of VC dimension d, such that, for any learning algorithm that draws at most N unconditional samples from $\mathcal{O}_P$, there exists a choice of distributions $(P, P^\star)$ with $\mathsf{KL}(P^\star \,\|\, P) \leq D$ and $f^\star \in \mathcal{F}$ such that*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F},T) \gtrsim \left(d + \frac{dT}{N} + D + D\frac{T}{\log N}\right) \wedge T$$

The $\Omega(d)$ lower bound is trivial from the definition of VC dimension. It suffices to prove the remaining lower bounds separately. Throughout, we work with d-dimensional thresholds, defined as follows. We let $\mathcal{X}_d := [0,1]\times[d]$ be the domain, and let

$$\mathcal{F}_d := \{(x,i) \mapsto \mathbb{1}\{x \geq \theta_i\}, (\theta_1,\ldots,\theta_d) \in [0,1]^d\}. \tag{16}$$

It is easy to see that the VC dimension of the above class is exactly d. We proceed with the proof of the "price of sampling" lower bound of dT/N. It holds even in the case $P^\star = P$.

Lemma E.1 *Let $\mathcal{F}_d$ be the class as defined in Eq. (16). For any $N \in \mathbb{N}$ and for any learning algorithm that draws at most N unconditional samples from $P^\star$, there exists a choice of $P^\star$ and $f^\star \in \mathcal{F}_d$ such that*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}_d,T) \gtrsim \frac{T(d\wedge N)}{N}.$$

Proof For each $i \in [d]$, let $\mathbf{z}_i$ denote a depth-T Littlestone tree in coordinate i. Then, let ρ_i be a distribution that samples a uniform path in $\mathbf{z}_i$. Let ρ be a uniform mixture of ρ_i as $i \sim \mathsf{Unif}([d])$.

Note that we may assume WLOG that $N \geq d$ (otherwise, we can consider $\mathcal{F}_N$ instead of $\mathcal{F}_d$ and prove a linear lower bound on regret). To define $P^\star$, we first sample d paths $\mathbf{x}^i := (x_1^i,\ldots,x_T^i) \sim \rho_i$ for $i \in [d]$; then, we let

$$P^\star = \frac{1}{2N}\sum_{i\in[d]}\delta_{\mathbf{x}^i} + \left(1 - \frac{d}{2N}\right)\rho.$$

Let $f^\star$ be a labeling function that is consistent with each of the paths $\mathbf{x}^i$ for $i \in [d]$ (see Definition D.5). Note that, WLOG, we may assume that the learner draws all N samples before the first round of the online learning game. With probability at least $d/2N$, the distribution $P^\star$ samples a path $\mathbf{x}^{i_\star}$ for some $i_\star \in [d]$. Since the learner only draws N samples, with probability at least:

$$\left(1 - \frac{1}{2N}\right)^N \geq \frac{1}{2},$$

none of the sampled paths are equal to $\mathbf{x}^{i_\star}$ and are either sampled from ρ, or are equal to $\mathbf{x}^j$ for some $j \neq i_\star$. Condition on this constant probability event. Since the randomness used in the construction of $\mathbf{x}^{i_\star}$ is independent of other paths and of samples from ρ, the learner will incur expected regret of at least $T/2$ on this path by standard Littlestone tree hardness argument (see Proposition D.4). Thus, total expected regret of the learner can be lower bounded as:

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}_d, T) \geq \frac{d}{2N} \cdot \frac{1}{2} \cdot \frac{T}{2} = \frac{dT}{8N}.$$

This concludes the proof. ■

Next, we prove a "price of approximation" lower bound: even with full knowledge of P, the regret of any algorithm must scale linearly with $\mathsf{KL}(P^\star \,\|\, P)$.

Lemma E.2 *Let $\mathcal{F}_1$ be as in Eq. (16). For every $D \in \mathbb{N}$, there exist a family $\mathcal{P}_D$ of distributions over $\mathcal{X}^T$ and a simulator P such that:*

(i) $\max_{P^\star \in \mathcal{P}_D} \mathsf{KL}(P^\star \,\|\, P) \leq D \wedge T$.

(ii) For any learning algorithm, even one granted full knowledge of P, there exists $P^\star \in \mathcal{P}_D$ and a labeling $f^\star \in \mathcal{F}_1$ for which

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}_1, T) \geq \frac{D \wedge T}{2}.$$

Proof Let $m := D \wedge T$. Consider a Littlestone tree of depth m, and let P be a distribution that samples a uniform path in that Littlestone tree, padding the remaining $T - m$ rounds with a fixed $x^0 \in \mathcal{X}$. Let $\mathcal{P}_D$ be the family of all such distributions, that is, $P^\star \in \mathcal{P}_D$ iff $P^\star = \delta_{\mathbf{x}}$ for some leaf $\mathbf{x}$ (with the same padding). We note that $P(\mathbf{x}) = \frac{1}{2^m}$, and thus, for any $P^\star \in \mathcal{P}_D$,

$$\mathsf{KL}(P^\star \,\|\, P) = \log(2^m) = m \leq D,$$

which establishes (i). For (ii), we construct $P^\star$ randomly by selecting a path $\mathbf{x}$ in the Littlestone tree uniformly at random and setting $P^\star = \delta_{\mathbf{x}}$, with the same padding. Set $f^\star \in \mathcal{F}_1$ to be a threshold function consistent with the path $\mathbf{x}$ (see Definition D.5). Since the construction of $P^\star$ is independent of P, the setup is equivalent to the standard Littlestone hardness construction (see Proposition D.4). Thus, any learner must suffer $m/2 = (D \wedge T)/2$ expected regret, which concludes the proof. ■

Next, we quantify how access to a bounded number of samples from P worsens the price of approximation.

Lemma E.3 *Let $\mathcal{F}_1$ be as in Eq. (16). For any $N \geq 2$, $D \in \mathbb{N}$, and for any learning algorithm that draws at most N unconditional samples from P, there exists a choice of $(P, P^\star)$ with $\mathsf{KL}(P^\star \,\|\, P) \leq D$ and $f^\star \in \mathcal{F}_1$ such that*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}_1, T) \gtrsim T\left(\frac{D}{\log(N)} \wedge 1\right)$$

Proof Consider a Littlestone tree of depth T. Consider a distribution ρ that samples a path in the Littlestone tree at random, and let $\mathbf{x} \sim \rho$ be a random path in the Littlestone tree. Then, we define $P, P^\star$ as mixtures:

$$P^\star = \alpha\delta_{\mathbf{x}} + (1-\alpha)\rho, \qquad P = \beta\delta_{\mathbf{x}} + (1-\beta)\rho,$$

for some $1 \geq \alpha > \beta > 0$ to be specified later. Then, we have

$$\mathsf{KL}(P^\star \,\|\, P) = \mathsf{KL}(\mathsf{Ber}(P^\star(\mathbf{x})) \,\|\, \mathsf{Ber}(P(\mathbf{x}))),$$

since $P^\star$ and P have a constant likelihood ratio $(1-\alpha)/(1-\beta)$ on the support off $\mathbf{x}$, so the full KL collapses to the binary KL between the Bernoulli marginals $\mathsf{Ber}(P^\star(\mathbf{x}))$ and $\mathsf{Ber}(P(\mathbf{x}))$. Furthermore, a simple computation shows that $p \mapsto \mathsf{KL}(\mathsf{Ber}(\alpha + (1-\alpha)p) \,\|\, \mathsf{Ber}(\beta + (1-\beta)p))$ is non-increasing for $p \in [0,1]$, therefore,

$$\mathsf{KL}(\mathsf{Ber}(P^\star(\mathbf{x})) \,\|\, \mathsf{Ber}(P(\mathbf{x}))) \leq \mathsf{KL}(\mathsf{Ber}(\alpha) \,\|\, \mathsf{Ber}(\beta)) \leq \alpha \log(1/\beta).$$

Together, these inequalities give:

$$\mathsf{KL}(P^\star \,\|\, P) \leq \alpha \log(1/\beta).$$

Let $f^\star$ be the labeling function consistent with the path $\mathbf{x}$ (see Definition D.5). Recall that N is the almost sure upper bound on the number of samples drawn by the algorithm. Let $\beta = 1/N$. Then, with at least constant probability, all of the samples drawn by the algorithm will come from ρ. Now, let $\alpha = (D/\log(N)) \wedge 1$, which gives $\mathsf{KL}(P^\star \,\|\, P) \leq D$.

Now, note that, with probability at least α, the learner will be presented with $\mathbf{x}$ at runtime, on which it incurs linear regret by standard Littlestone hardness (see Proposition D.4). Thus,

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}_1, T) \gtrsim \alpha T = T\left(\frac{D}{\log(N)} \wedge 1\right).$$

This concludes the proof. ■

Proof [Proof of Theorem 3.3] The bound follows by combining the four lower bounds established above, each holding against *any* algorithm drawing at most N samples. The $d \wedge T$ term is the trivial lower bound from the definition of VC dimension. The "price of sampling" bound (Lemma E.1) forces regret $\gtrsim \frac{T(d \wedge N)}{N} = \frac{dT}{N} \wedge T$. The "price of approximation" bound (Lemma E.2) forces regret $\gtrsim D \wedge T$. Finally, "price of approximate sampling" (Lemma E.3) forces regret $\gtrsim T\left(\frac{D}{\log N} \wedge 1\right) = \frac{DT}{\log N} \wedge T$. Each of these constructions satisfies $\mathsf{KL}(P^\star \,\|\, P) \leq D$ and can be realized within $\mathcal{F}_d$ (Eq. (16)), which has VC dimension d. Indeed, the one-dimensional constructions ($\mathcal{F}_1$) can be trivially embedded into $\mathcal{F}_d$.

Note that each of the four bounds holds against any algorithm. Then, after fixing the algorithm at hand, we may adversarially choose the construction that yields the largest lower bound. Thus, writing $R_{\mathsf{max}} = \max\left\{d, \frac{dT}{N}, D, \frac{DT}{\log N}\right\}$, the corresponding construction forces

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}_d, T) \gtrsim R_{\mathsf{max}} \wedge T \geq \frac{1}{4}\left(d + \frac{dT}{N} + D + \frac{DT}{\log N}\right) \wedge T,$$

for class $\mathcal{F}_d$ of VC dimension d and some $(P^\star, P)$ with $\mathsf{KL}(P^\star \,\|\, P) \leq D$, which gives the desired claim. ■

E.3. Proof of Theorem 3.5

Theorem 3.5 *Let $\mathcal{F}$ be the class of thresholds on the $[0,1]$ interval. For any $N \geq 1$, and for any learning algorithm that draws at most N unconditional samples from $P^\star$, there exists a choice of a distribution $P^\star$ and $f^\star \notin \mathcal{F}$ such that $\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \gtrsim T$.*

Proof WLOG, assume $N \geq 2$. Divide the interval $[0,1]$ into N sub-intervals $\{A_i\}_{i\in[N]}$, and consider N Littlestone trees in each respective interval. For each $i \in [N]$, let ρ_i be a distribution that samples a random path in the Littlestone tree in interval A_i. For each $i \in [N]$, sample a path $\mathbf{x}^i \sim \rho_i$. Now, let $P^\star$ and $f^\star$ be defined in the following way. Set

$$P^\star := \frac{1}{N}\sum_{i\in[N]} \delta_{\mathbf{x}^i}.$$

On each interval A_i, $i \in [N]$, let f_i be a threshold function consistent with the path $\mathbf{x}^i$ (see Definition D.5). Then, we let

$$f^\star(x) := \sum_{i\in[N]} \mathbb{1}\{x \in A_i\} f_i(x).$$

Then, with probability 1 over the draw from $\mathbf{X} \sim P^\star$, the labels of $f^\star$ are realizable w.r.t. $\mathcal{F}$ on $\mathbf{X}$, that is, with probability 1:

$$\inf_{f\in\mathcal{F}} \sum_{t=1}^{T} \mathbb{1}\{f^\star(X_t) \neq f(X_t)\} = 0.$$

Let $i_\star \in [N]$ be a random index such that $\mathbf{X} = \mathbf{x}^{i_\star}$. Then, with probability at least

$$\left(1 - \frac{1}{N}\right)^N \geq \frac{1}{4},$$

none of the samples the learner draws are equal to $\mathbf{x}^{i_\star}$. Let us condition on this event. Then, for any $t \in [T]$ the randomness of $\mathbf{x}^{i_\star}$ is independent of the randomness of the learner in round t, that is, learner suffers expected regret of $T/2$ by standard Littlestone hardness result (see Proposition D.4). Thus,

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{f^\star(X_t) \neq \hat{y}_t\} - \inf_{f\in\mathcal{F}} \sum_{t=1}^{T} \mathbb{1}\{f^\star(X_t) \neq f(X_t)\}\right] \geq \frac{1}{4}\cdot\frac{T}{2} = \frac{T}{8},$$

as desired. ■

E.4. Proof of Theorem 3.4

Below we show that in the realizable setting, if the learning algorithm only receives unconditional samples from $P^\star$, then any sublinear-regret learning algorithm requires a number of ERM oracle queries which is at least exponential in the VC dimension. We assume that when making queries to the ERM oracle, the learning algorithm must choose covariates that belong to either (a) the set of previous covariates revealed by nature (i.e., $X_1, \dots, X_t$) or (b) the set of covariates amongst the unconditional samples drawn from $P^\star$. This assumption is essentially without loss of generality, as one may enlarge the domain $\mathcal{X}$ with a large number of "dummy" points; to keep the notation simple, however, we explicitly make this assumption in this section. Moreover, we assume that the learning algorithm Alg must succeed for *any* valid choice of ERM oracle; in particular, the lower bound below establishes the existence of *some* ERM oracle with respect to which Alg suffers linear regret.

Theorem 3.4 *There is a sufficiently small constant c so that the following holds. For any $d, N, T \in \mathbb{N}$ satisfying $TN \le 2^{cd}$, and any learning algorithm* Alg *that draws at most N unconditional samples from $P^\star$ and makes at most N agnostic ERM oracle queries, there exists a class $\mathcal{F}$ of VC dimension at most d and a distribution $P^\star$ so that* Alg *suffers regret $\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \gtrsim T$.*

Proof Fix $T, N \in \mathbb{N}$, and let $M = 10N$. We define $\mathcal{X} := [M] \times [T]$, and for $i \in [M]$, $\mathcal{X}_i := \{i\} \times [T]$. Write $K = CT^4N^{10}$ for a sufficiently large constant C (to be specified below). We construct a random class $\mathcal{F}$, as follows:

- First, we generate $K+1$ random functions, which we denote by $h_1, \dots, h_{K+1}$ (i.e., for each such h_k, $h_k(x) \in \{0,1\}$ is drawn uniformly at random for each $x \in \mathcal{X}$).
- We set $f^\star = h_{K+1}$.
- For each $i \in [M]$, $t \in [T]$, $k_0 \in [K+1]$, $k_1 \in [K]$, and $b \in \{0,1\}$, we define the function g_{t,i,b,k_0,k_1} as follows:

$$g_{t,i,b,k_0,k_1}(x) = \begin{cases} h_{k_0}(x) & : x \in \{(i,1), \dots, (i,t-1)\} \\ b & : x = (i,t) \\ h_{k_1}(x) & : \text{otherwise.} \end{cases} \tag{17}$$

- We set $\mathcal{F} = \{f^\star\} \cup \bigcup_{t\in[T], i\in[M], k_0\in[K+1], k_1\in[K]} \{g_{t,i,b,k_0,k_1}\}$.

It is straightforward to see that realizability holds (i.e., $f^\star \in \mathcal{F}$). Note that $|\mathcal{F}| \le O(MTK^2)$, and so the VC dimension of $\mathcal{F}$ is $O(\log MTK) \le O(\log TN)$. This is bounded above by d as long as $TN \le 2^{cd}$ for a sufficiently small constant c. The distribution $P^\star$ is defined as $\mathrm{Unif}\{((i,1), \dots, (i,T))\}_{i\in[M]}$.

Now consider any algorithm Alg, which functions as follows: first, it draws N samples from $P^\star$, which we denote by $((I_n, t))_{t\in[T]}$, for $n \in [N]$; in particular, $I_n \sim \mathsf{Unif}([M])$ independently for each $n \in [N]$. Moreover, nature chooses $I^\star \sim \mathsf{Unif}([M])$, and the sequence of covariates that Alg observes is $(X_1, \dots, X_T)$, where $X_t = (I^\star, t)$. In particular, the following procedure repeats for T time steps:

1. At each time step t, write $\mathcal{W}_t := \bigcup_{n\in[N], t\in[T]} \{(I_n, t)\} \cup \bigcup_{s\in[t]} \{(I^\star, s)\}$.

2. At time step t, Alg makes some number of queries to the ERM oracle, each of the form

$$\operatorname*{argmin}_{f \in \mathcal{F}} \sum_{x \in \mathcal{W}_t} y_{j,x} \cdot f(x), \tag{18}$$

where $y_{j,x} \in \mathbb{R}$ are real numbers which can depend on the results of previous queries and the labels $Y_1, \dots, Y_{t-1}$. Here we use j to denote a *global* index over all queries, i.e., Eq. (18) denotes the jth ERM query over the entire execution of the algorithm. If there are multiple minimizers to Eq. (18), we assume that the ERM oracle picks one in $\widetilde{\mathcal{F}}_t$ (defined below), if possible; conditioned on this, it returns one uniformly at random. With a slight abuse of notation, we use the expression in Eq. (18) (in particular, the use of argmin) to denote the result returned by this ERM oracle.

After making each such query, which returns some $f^{(j)} \in \mathcal{F}$, we allow Alg to query any of the functions $f^{(j')}$, $j' \le j$, at any number of points in $\mathcal{X}$ before making the subsequent query.

3. Alg then predicts a label $\hat{Y}_t$ for X_t, and observes the true label $Y_t = f^\star(X_t)$.

Let $t^{(j)} \in [T]$ denote the time step at which the jth oracle query is made, and let J denote the total number of oracle queries (we have $J \le N$). Next, we define (random) subclasses $\widetilde{\mathcal{F}}_t \subset \mathcal{F}$ as follows:

$$\begin{aligned}\mathcal{F} \backslash \widetilde{\mathcal{F}}_t :=& \{f^\star\} \cup \{g_{s,I^\star,b,K+1,k_1} : s \ge t+1, b \in \{0,1\}, k_1 \in [K]\} \\ &\cup \{g_{s,i,b,K+1,k_1} : s \in [T], i \ne I^\star, b \in \{0,1\}, k_1 \in [K]\}.\end{aligned}$$

In words, functions in $\mathcal{F} \backslash \widetilde{\mathcal{F}}_t$ are those which reveal information about $f^\star$ which has not already been observed (as of step t) in the form of the labels $Y_1, \dots, Y_{t-1}$. Finally, we let $\widetilde{\mathsf{Alg}}$ denote the algorithm whose execution is identical to that of Alg, except that for the jth oracle call, the minimization in Eq. (18) is performed over $f \in \widetilde{\mathcal{F}}_t$.

Lemma E.4 *Let $\tilde{Y}_1, \dots, \tilde{Y}_T$ denote the predictions of $\widetilde{\mathsf{Alg}}$. Then it holds that*

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{\tilde{Y}_t \ne Y_t\}\right] = \frac{T}{2},$$

where the expectation is over the randomness of $\widetilde{\mathsf{Alg}}$ and the choice of $\mathcal{F}$.

Proof The claim follows immediately from the observation that the random variable $\tilde{Y}_t$ is independent of Y_t (which is distributed as $\mathsf{Unif}\{0,1\}$), due to the definition of $\widetilde{\mathsf{Alg}}$ and the fact that $f^\star : \mathcal{X} \to \{0,1\}$ is chosen to be a uniformly random function. ■

The next lemma shows that, with high probability over the joint probability space including all random variables defined above, Alg and $\widetilde{\mathsf{Alg}}$ behave identically.

Lemma E.5 *With probability at least $1 - 6J^{3/2}TN/K^{1/4} - N/M$ (over the execution of Alg, $\widetilde{\mathsf{Alg}}$, and the choice of $\mathcal{F}$), the execution of Alg is identical to that of $\widetilde{\mathsf{Alg}}$.*

Proof Note that $I^\star \notin \{I_1, \ldots, I_N\}$ with probability at least $1 - \frac{N}{M}$. Let us henceforth condition on this event. We now consider the execution of $\widetilde{\mathsf{Alg}}$: we will show that with high probability, performing the minimization in Eq. (18) over $\mathcal{F}$ as opposed to $\widetilde{\mathcal{F}}_t$ at each oracle query $j \in [J]$, would not change any of the answers returned by the ERM oracle. Thus, on this high probability event, the execution of Alg agrees with that of $\widetilde{\mathsf{Alg}}$.

Recall that $\widetilde{\mathsf{Alg}}$ makes a total of J ERM oracle queries; we let the response of the jth ERM oracle query be denoted $f^{(j)} \in \mathcal{F}$. For each $j \in [J]$, suppose that $f^{(j)}$ is of the form $g_{t,i,b,k_0^{(j)},k_1^{(j)}}$ for some $k_0^{(j)}, k_1^{(j)} \in [K+1]$. Write $\mathcal{K}^{(j)} := \{k_0^{(1)}, \ldots, k_0^{(j-1)}, k_1^{(1)}, \ldots, k_1^{(j-1)}\}$. Let $\mathscr{F}^{(j)}$ denote the sigma-algebra generated by $I_1, \ldots, I_N, I^\star, f^{(1)}, \ldots, f^{(j-1)}, \mathcal{K}^{(j)}, Y_1, \ldots, Y_{t^{(j)}-1}$. Note that the transcript of responses that $\widetilde{\mathsf{Alg}}$ has received from the ERM oracle queries prior to the jth one is $\mathscr{F}^{(j)}$-measurable. We state the following lemma, whose proof is deferred to the end of the section.

Lemma E.6 *Fix any subset $\mathcal{K} \subset [K]$ and step $j \in [J]$, and set $\varepsilon = \frac{2j|\mathcal{K}|}{K}$. Then with probability at least $1 - \sqrt{\varepsilon}$ over the execution of $\widetilde{\mathsf{Alg}}$ up to step j and the choice of $\mathcal{F}$, the distribution of the tuple of functions $(h_k)_{k\in\mathcal{K}}$ conditioned on $\mathscr{F}^{(j)}$ is $\sqrt{\varepsilon}$-close to uniform in total variation distance.*

Let us write $\bar{\varepsilon} := \frac{2J}{\sqrt{K}}$. At any step j of the execution of $\widetilde{\mathsf{Alg}}$, note that $f^\star(\mathcal{W}_{t^{(j)}} \backslash \{X_1, \ldots, X_{t^{(j)}-1}\})$ is uniformly random conditioned on $\mathscr{F}^{(j)}$. (Indeed, the functions in $\tilde{\mathcal{F}}_1, \ldots, \tilde{\mathcal{F}}_{t^{(j)}}$ contain no information about $f^\star$ on $\mathcal{W}_{t^{(j)}} \backslash \{X_1, \ldots, X_{t^{(j)}-1}\}$.) By Lemma E.6, the distribution of $(h_1, \ldots, h_{\sqrt{K}})$ is $\sqrt{\bar{\varepsilon}}$-close to uniform in total variation distance, under some event $\mathcal{E}_j$ that occurs with probability $1 - \sqrt{\bar{\varepsilon}}$. Thus, under $\mathcal{E}_j$, the joint distribution of $f^\star, h_1, \ldots, h_{\sqrt{K}}$ is $\sqrt{\bar{\varepsilon}}$-close to uniform on $\mathcal{W}_{t^{(j)}} \backslash \{X_1, \ldots, X_{t^{(j)}-1}\}$ in total variation distance. Under the event $\mathcal{E}_j$, we have

$$\begin{aligned} &\mathbb{P}\left(\operatorname*{argmin}_{f\in\mathcal{F}} \sum_{x\in\mathcal{W}_{t^{(j)}}} y_{j,x} \cdot f(x) = f^\star \middle| \mathscr{F}^{(j)} \right) \qquad (19) \\ \le &\mathbb{P}\left(\sum_{x\in\mathcal{W}_{t^{(j)}}} y_{j,x} \cdot f^\star(x) < \min_{\substack{k\le\sqrt{K},\\ b\in\{0,1\}}} \sum_{x\in\mathcal{W}_{t^{(j)}}} y_{j,x} \cdot g_{t^{(j)},I^\star,b,K+1,k}(x) \middle| \mathscr{F}^{(j)} \right) \le \sqrt{\bar{\varepsilon}} + \frac{1}{\sqrt{K}}. \end{aligned}$$

In the first inequality above, we have used that Alg will only return $f^\star$ if none of $g_{t^{(j)},I^\star,b,K+1,k}$, $k \le \sqrt{K}, b \in \{0,1\}$ achieve the minimum (as all of the latter functions belong to $\widetilde{\mathcal{F}}_{t^{(j)}}$). The second inequality above uses that, $f^\star$ and $g_{t^{(j)},I^\star,b,K+1,k}$ ($k \le \sqrt{K}$) agree on $\{X_1, \ldots, X_{t^{(j)}-1}\}$, and the joint distribution of these $\sqrt{K}+1$ functions, over $b \sim \mathsf{Unif}\{0,1\}$, is $\sqrt{\bar{\varepsilon}}$ close to uniform on $\mathcal{W}_{t^{(j)}} \backslash \{X_1, \ldots, X_{t^{(j)}-1}\}$ (conditioned on $\mathscr{F}^{(j)}$, under the event $\mathcal{E}_j$).

Similarly, we have that under $\mathcal{E}_j$, for any $t \in [T]$ and $I \in \{I_1, \ldots, I_N\}$,

$$\begin{aligned} &\mathbb{P}\left(\operatorname*{argmin}_{f\in\mathcal{F}} \sum_{x\in\mathcal{W}_{t^{(j)}}} y_{j,x} \cdot f(x) \in \{g_{t,I,b,K+1,k_1} : k_1 \in [K], b \in \{0,1\}\} \middle| \mathscr{F}^{(j)} \right) \qquad (20) \\ \le &\mathbb{P}\left(\min_{\substack{k_1\in[K],\\ b\in\{0,1\}}} \sum_{x\in\mathcal{W}_{t^{(j)}}} y_{j,x} \cdot g_{t,I,b,K+1,k_1}(x) < \min_{\substack{k_1\in[K],\\ k_0\in[\sqrt{K}],\\ b\in\{0,1\}}} \sum_{x\in\mathcal{W}_{t^{(j)}}} y_{j,x} \cdot g_{t,I,b,k_0,k_1}(x) \middle| \mathscr{F}^{(j)} \right) \end{aligned}$$

$$\leq \sqrt{\bar{\varepsilon}} + \frac{1}{\sqrt{K}}.$$

Again, in the first inequality, we have used that Alg will only return one of $g_{t,I,b,K+1,k_1}$ if none of g_{t,I,b,k_0,k_1}, $k_1 \in [K], k_0 \in [\sqrt{K}], b \in \{0,1\}$ achieve the minimum, as all of the latter functions belong to $\widetilde{\mathcal{F}}_{t^{(j)}}$. In the second inequality, we have used that $g_{t,I,b,K+1,k_1}$ and g_{t,I,b,k_0,k_1} differ only on $(I,1),\dots,(I,t-1)$, where their values are given by those of h_{K+1} and h_{k_0}, respectively. The second inequality then follows from the fact that under $\mathcal{E}_j$, conditioned on $\mathscr{F}^{(j)}$, the functions $h_1,\dots,h_{\sqrt{K}},h_{K+1}$, restricted to $(I,1),\dots,(I,t-1)$, have distribution which is $\sqrt{\bar{\varepsilon}}$-close to uniform in total variation distance (as argued above using Lemma E.6).

Finally, note that $g_{s,I^\star,b,K+1,k_1}$, for any $s \geq t+1$, agrees with $g_{t,I^\star,f^\star(X_t),K+1,k_1}$ on all of $\mathcal{W}_t$, meaning that the ERM oracle will never return $g_{s,I^\star,b,K+1,k_1}$ for any $s \geq t+1, b \in \{0,1\}, k_1 \in [K]$. Using this fact as well as Eqs. (19) and (20), we see that under $\bigcap_{j\in[J]} \mathcal{E}_j$, with probability at least $1 - 2JTN \cdot (\sqrt{\bar{\varepsilon}} + 1/\sqrt{K})$ under the execution of $\widetilde{\mathsf{Alg}}$, for all J ERM oracle queries $j \in [J]$, the oracle response in Eq. (18) is the same if the minimization is done over $\mathcal{F}$ as if it is done over $\widetilde{\mathcal{F}}_{t^{(j)}}$. Since $\bigcap_{j\in[J]} \mathcal{E}_j$ occurs with probability at least $1 - J\sqrt{\bar{\varepsilon}}$, the result follows. ■

We may now conclude the proof of Theorem 3.4. First, Lemma E.4 gives that

$$\mathbb{P}_{\widetilde{\mathsf{Alg}}}\left(\mathbf{Reg}(\mathcal{F},T) \geq T/2\right) \geq 1/2.$$

As long as we choose $M \geq 10N$ and $K \geq (600J^{3/2}TN)^4$, Lemma E.5 gives that the execution of Alg and $\widetilde{\mathsf{Alg}}$ have total variation distance at most $1/5$. It follows that $\mathbb{P}_{\mathsf{Alg}}(\mathbf{Reg}(\mathcal{F},T) \geq T/2) \geq 3/10$, i.e., $\mathbb{E}_{\mathsf{Alg}}[\mathbf{Reg}(\mathcal{F},T)] \geq \Omega(T)$. ■

Finally, it remains to prove Lemma E.6.

Proof [Proof of Lemma E.6] We couple the execution of $\widetilde{\mathsf{Alg}}$ to that of a modified algorithm that differs as follows: the ERM oracle returns functions g'_{t,i,b,k_0,k_1} defined as follows (contrast with Eq. (17)):

$$g'_{t,i,b,k_0,k_1}(x) = \begin{cases} h'_{k_0}(x) & : x \in \{(i,1),\dots,(i,t-1)\} \\ b & : x = (i,t) \\ h'_{k_1}(x) & : \text{otherwise,} \end{cases} \tag{21}$$

where $h_k = h'_k$ if $k \notin \mathcal{K}$, and h'_k is a fresh uniformly random function for $k \in \mathcal{K}$. We denote the resulting algorithm by Alg'.

Under the execution of Alg', with probability 1, conditioned on $\mathcal{K}^{(j)}$, the distribution of $(h_k)_{k\in\mathcal{K}}$ is uniformly random (as the functions $h_k, k \in \mathcal{K}$ are never used by the ERM oracle). The probability that the two executions differ is bounded above by the probability that $\mathcal{K} \cap \mathcal{K}^{(j)} \neq \emptyset$, which is at most $\varepsilon = \frac{2j|\mathcal{K}|}{K}$, by symmetry. (In particular, at each oracle call j of $\widetilde{\mathsf{Alg}}$, conditioned on $\mathcal{K} \cap \mathcal{K}^{(j)} = \emptyset$, there is at most a $2|\mathcal{K}|/K$ chance that the ERM oracle returns g_{t,i,b,k_0,k_1} with $k_0 \in \mathcal{K}$ or $k_1 \in \mathcal{K}$.)

In particular, the distribution of the execution of Alg' is ε-close to the distribution of the execution of Alg in total variational distance. The conclusion follows by the general fact that for random variables $(X,Y),(X',Y')$, we have $\mathsf{TV}\left((X,Y),\ (X',Y')\right) \geq \mathbb{E}_X[\mathsf{TV}\left(Y|X,\ Y'|X\right)]$; in

particular, we use this fact with X, X' being the random variables generating $\mathscr{F}^{(j)}$ under $\widetilde{\mathsf{Alg}}, \mathsf{Alg}'$, respectively, and Y, Y' being the random variables $(h_k)_{k\in\mathcal{K}}$ under $\widetilde{\mathsf{Alg}}, \mathsf{Alg}'$, respectively. ■

Appendix F. Proofs from Section 4

F.1. Proof of Theorem 4.1

Proposition 4.3 *For a history $\mathbf{S}_{1:t}$ and a covariate X_t, let*

$$\mathfrak{G}_y(\mathbf{W}^t) := \sup_{f\in\mathcal{F}}\left[\sum_{s=t+1}^{T}\varepsilon_s\frac{2f(X_s)-1}{2} - L_{t-1}(f) - \mathbb{1}\{f(X_t)\neq y\}\right]. \tag{8}$$

Then, q_t as in Eq. (5) can be written as $q_t(\mathbf{S}_{1:t-1}, X_t) = \frac{1}{2} + \frac{1}{2\eta}\log\left(\frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\right)$

Proof Writing $\mathbf{S}_{1:t} = (\mathbf{S}_{1:t-1}, (X_t, Y_t))$, the objective of Eq. (5) is minimized at the unconstrained minimizer

$$\frac{1}{2} + \frac{\mathbf{Rel}(\mathbf{S}_{1:t-1}, (X_t, 1)) - \mathbf{Rel}(\mathbf{S}_{1:t-1}, (X_t, 0))}{2}. \tag{22}$$

Since $L_t(f) = L_{t-1}(f) + \mathbb{1}\{f(X_t)\neq y\}$ when $Y_t = y$, the relaxation Eq. (4) and the offset process $\mathfrak{G}_y$ in Eq. (8) satisfy

$$\mathbf{Rel}(\mathbf{S}_{1:t-1}, (X_t, y)) = \frac{1}{\eta}\log\mathbb{E}\exp(\eta\mathfrak{G}_y(\mathbf{W}^t)),$$

so that the unconstrained minimizer from Eq. (22) can be equivalently rewritten as

$$\frac{1}{2} + \frac{1}{2\eta}\log\left(\frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\right), \tag{23}$$

as claimed. It remains to show that this value lies in $[0, 1]$. First, note that, with probability 1:

$$\left|\sup_{f\in\mathcal{F}}\left[\sum_{s=t+1}^{T}\varepsilon_s\frac{2f(X_s)-1}{2} - L_{t-1}(f) - \mathbb{1}\{f(X_t)\neq 0\}\right]\right.$$
$$\left. - \sup_{f\in\mathcal{F}}\left[\sum_{s=t+1}^{T}\varepsilon_s\frac{2f(X_s)-1}{2} - L_{t-1}(f) - \mathbb{1}\{f(X_t)\neq 1\}\right]\right| \leq 1,$$

and thus, for each $y\in\{0,1\}$

$$\exp(\eta\mathfrak{G}_y(\mathbf{W}^t) \leq \exp(\eta)\cdot\exp(\eta\mathfrak{G}_{1-y}(\mathbf{W}^t)).$$

Taking expectations of both sides gives:

$$\mathbb{E}\exp(\eta\mathfrak{G}_y(\mathbf{W}^t)) \leq \exp(\eta)\,\mathbb{E}\exp(\eta\mathfrak{G}_{1-y}(\mathbf{W}^t)).$$

This yields:

$$-\eta \leq \log\left(\frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\right) \leq \eta,$$

which implies that Eq. (23) is bounded in $[0,1]$. Thus,

$$q_t(\mathbf{S}_{t-1}, X_t) = \frac{1}{2} + \frac{1}{2\eta} \log\left(\frac{\mathbb{E}\exp(\eta \mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta \mathfrak{G}_0(\mathbf{W}^t))}\right),$$

as desired. ■

Lemma F.1 *Let $\mathcal{F}$ be an arbitrary function class, and consider any $t \le T$. Suppose the simulator is exact, that is $P = P^\star$, and let q_t be as in Eq. (5). Then, for any history $\mathbf{S}_{1:t-1}$, we have*

$$\mathbb{E}_{X_t \sim P_t^\star(\cdot|\mathbf{X}_{1:t-1})} \exp\left(\sup_{Y_t}\left[\eta\, \mathbb{E}_{\hat{Y}_t \sim q_t}\, \mathbb{1}\{\hat{Y}_t \neq Y_t\} + \eta \mathbf{Rel}(\mathbf{S}_{1:t})\right]\right) \le \exp\left(\eta \mathbf{Rel}(\mathbf{S}_{1:t-1})\right).$$

Thus, conditioned on $\mathbf{S}_{1:t-1}$, $\mathbb{E}_{X_t} \sup_{Y_t} \Phi_t \le \Phi_{t-1}$, and therefore, $\{\Phi_t\}_{t\le T}$ is a supermartingale.

Proof In the remainder, we let $\mathbf{X}_{t:T} \sim P(\cdot \mid \mathbf{X}_{1:t-1})$ and $\boldsymbol{\varepsilon}_{t+1:T} \sim \mathsf{Unif}(\{\pm 1\}^{T-t})$ and suppress the distributions from the expectations for notational brevity. Using the closed-form expression for q_t from Proposition 4.3 (namely, from Eq. (22)), it suffices to show

$$\mathbb{E}_{X_t} \exp\left(\frac{\eta}{2} + \frac{\eta}{2}\left[\mathbf{Rel}(\mathbf{S}_{1:t-1}, (X_t, 1)) + \mathbf{Rel}(\mathbf{S}_{1:t-1}, (X_t, 0))\right]\right) \le \exp\left(\eta \mathbf{Rel}(\mathbf{S}_{1:t-1})\right),$$

or, equivalently,

$$\mathbb{E}_{X_t} \exp\left(\eta\, \mathbb{E}_{\varepsilon_t}\left[\mathbf{Rel}\left(\mathbf{S}_{1:t-1}, \left(X_t, \frac{1+\varepsilon_t}{2}\right)\right) + \frac{1}{2}\right]\right) \le \exp\left(\eta \mathbf{Rel}(\mathbf{S}_{1:t-1})\right). \tag{24}$$

For every $\varepsilon_t \in \{\pm 1\}$, we have

$$\begin{aligned}
&\mathbf{Rel}\left(\mathbf{S}_{1:t-1}, \left(X_t, \frac{1+\varepsilon_t}{2}\right)\right) + \frac{1}{2} \\
&= \frac{1}{\eta} \log \mathbb{E}_{\substack{\mathbf{X}_{t+1:T} \\ \boldsymbol{\varepsilon}_{t+1:T}}} \exp\left(\eta \sup_{f\in\mathcal{F}}\left[\sum_{s=t+1}^{T} \varepsilon_s \frac{2f(X_s)-1}{2} - \left(\mathbb{1}\left\{f(X_t) \neq \frac{1+\varepsilon_t}{2}\right\} - \frac{1}{2}\right) - L_{t-1}(f)\right]\right) \\
&= \frac{1}{\eta} \log \mathbb{E}_{\substack{\mathbf{X}_{t+1:T}, \\ \boldsymbol{\varepsilon}_{t+1:T}}} \exp\left(\eta \sup_{f\in\mathcal{F}}\left[\sum_{s=t+1}^{T} \varepsilon_s \frac{2f(X_s)-1}{2} + \varepsilon_t \frac{2f(X_t)-1}{2} - L_{t-1}(f)\right]\right) \\
&= \frac{1}{\eta} \log \mathbb{E}_{\substack{\mathbf{X}_{t+1:T}, \\ \boldsymbol{\varepsilon}_{t+1:T}}} \exp\left(\eta \sup_{f\in\mathcal{F}}\left[\sum_{s=t}^{T} \varepsilon_s \frac{2f(X_s)-1}{2} - L_{t-1}(f)\right]\right).
\end{aligned}$$

Then, Eq. (24) can be restated as

$$\begin{aligned}
&\mathbb{E}_{X_t} \exp\left(\mathbb{E}_{\varepsilon_t} \log \mathbb{E}_{\substack{\mathbf{X}_{t+1:T}, \\ \boldsymbol{\varepsilon}_{t+1:T}}} \exp\left(\eta \sup_{f\in\mathcal{F}}\left[\sum_{s=t}^{T} \varepsilon_s \frac{2f(X_s)-1}{2} - L_{t-1}(f)\right]\right)\right) \\
&\qquad \le \mathbb{E}_{\substack{\mathbf{X}_{t:T}, \\ \boldsymbol{\varepsilon}_{t:T}}} \exp\left(\eta \sup_{f\in\mathcal{F}}\left[\sum_{s=t}^{T} \varepsilon_s \frac{2f(X_s)-1}{2} - L_{t-1}(f)\right]\right).
\end{aligned}$$

The proof is concluded using Jensen's inequality $\mathbb{E}_{\varepsilon_t} \log \le \log \mathbb{E}_{\varepsilon_t}$. ■

Lemma 4.2 *Let $\mathcal{F}$ be a VC dimension d class. Then, for every $\eta > 0$, an algorithm that plays according to Eq. (5) achieves the following expected regret bound in the agnostic setting:*

$$\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT} + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P) + \eta T.$$

Proof Recall the predictable loss process $\bar{L}_t$ and the potential Φ_t from Eq. (7). By Lemma F.1, $\{\Phi_t\}_{t \le T}$ is a supermartingale, and moreover

$$\Phi_T = \exp\left(\eta\left(\bar{L}_T - \inf_f L_T(f)\right)\right).$$

Thus, using the tower rule and the Donsker–Varadhan formula (Polyanskiy and Wu, 2025), we have

$$\begin{aligned}
\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) &= \mathbb{E}_{P^\star}\left(\bar{L}_T - \inf_f L_T(f)\right) \\
&\le \frac{1}{\eta}\log \mathbb{E}_P \exp\left(\eta\left(\bar{L}_T - \inf_f L_T(f)\right)\right) + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P) \\
&= \frac{1}{\eta}\log \mathbb{E}_P\, \Phi_T + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P) \\
&\le \frac{1}{\eta}\log \Phi_0 + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P).
\end{aligned}$$

Now, we upper bound $\frac{1}{\eta}\log \Phi_0$. By Eq. (7) and the definition of the relaxation in Eq. (4), we have

$$\frac{1}{\eta}\log \Phi_0 = \frac{1}{\eta}\log \mathbb{E}_{\substack{\mathbf{X}\sim P,\\ \boldsymbol{\varepsilon}\sim\{\pm 1\}^T}} \exp\left(\eta \sup_{f\in\mathcal{F}} \sum_{s=1}^T \varepsilon_s \frac{2f(X_s)-1}{2}\right).$$

Note that, conditional on $\mathbf{X}$, the function

$$\boldsymbol{\varepsilon} \mapsto \sup_f \sum_{s=1}^T \varepsilon_s \frac{2f(X_s)-1}{2}$$

satisfies the bounded differences condition. Thus, conditional on $\mathbf{X}$, we have

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\varepsilon}\sim\{\pm 1\}^T} \exp\left(\eta \sup_{f\in\mathcal{F}} \sum_{s=1}^T \varepsilon_s \frac{2f(X_s)-1}{2}\right) &\lesssim \exp\left(O(\eta^2 T) + \eta\, \mathbb{E}_{\boldsymbol{\varepsilon}} \sup_{f\in\mathcal{F}} \sum_{s=1}^T \varepsilon_s \frac{2f(X_s)-1}{2}\right) \\
&\lesssim \exp\left(O(\eta^2 T + \eta\sqrt{dT})\right).
\end{aligned}$$

Thus, we have

$$\frac{1}{\eta}\log \Phi_0 \lesssim \sqrt{dT} + \eta T.$$

Putting everything together, we have

$$\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT} + \eta T + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P).$$

■

Lemma F.2 *For an arbitrary history $\mathbf{S}_{1:t-1}$, an arbitrary covariate X_t, and $\eta \le 1$, we have*

$$\mathbb{E}\left[(\hat{q}_t - q_t)^2\right] \lesssim \frac{v_t}{N}, \quad \textit{where} \quad v_t := \max_{y\in\{0,1\}} \frac{\mathbb{E}\exp(2\eta\mathfrak{G}_y(\mathbf{W}^t))}{\left(\mathbb{E}\exp(\eta\mathfrak{G}_y(\mathbf{W}^t))\right)^2},$$

is the normalized variance of the exponential moment.

Proof Throughout, q_t is as in Proposition 4.3 and $\hat{q}_t$ is the plug-in estimator in Eq. (9). Note that, for any j, we have

$$\exp(-\eta) \le \frac{\exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))}{\exp(\eta\mathfrak{G}_1(\mathbf{W}_j^t))} \le \exp(\eta),$$

and

$$\exp(-\eta) \le \frac{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))} \le \exp(\eta).$$

Also, $\log$ is $\exp(\eta)$-Lipschitz on $[\exp(-\eta), \exp(\eta)]$, thus

$$\begin{aligned}
(\hat{q}_t - q_t)^2 &= \left(\frac{1}{2\eta}\log\left(\frac{\frac{1}{N}\sum_{j=1}^N \exp(\eta\mathfrak{G}_1(\mathbf{W}_j^t))}{\frac{1}{N}\sum_{j=1}^N \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))}\right) - \frac{1}{2\eta}\log\left(\frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\right)\right)^2 \\
&\le \frac{\exp(2\eta)}{4\eta^2}\left(\frac{\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_1(\mathbf{W}_j^t))}{\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))} - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\right)^2 \\
&= \frac{\exp(2\eta)}{4\eta^2}\left(\frac{\frac{1}{N}\sum_j \left(\exp(\eta\mathfrak{G}_1(\mathbf{W}_j^t)) - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))}\exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))\right)}{\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))}\right)^2.
\end{aligned}$$

Let

$$\mathcal{E} := \left\{\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \ge \frac{1}{2}\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))\right\}.$$

Note that

$$\mathbb{E}\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) = \mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t)),$$

and

$$\begin{aligned}
\mathsf{Var}\left(\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))\right) &\le \frac{1}{N}\mathbb{E}\left(\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))\right)^2 \\
&\le \frac{v_t}{N}\left(\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))\right)^2.
\end{aligned}$$

Thus, by Chebyshev's inequality, we have

$$\mathbb{P}[\mathcal{E}^c] \le \frac{\mathsf{Var}\left(\frac{1}{N}\sum_j \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))\right)}{\left(\frac{1}{2}\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))\right)^2} \le \frac{4v_t}{N}.$$

On $\mathcal{E}$, we have

$$(\hat{q}_t - q_t)^2 \mathbb{1}\{\mathcal{E}\} \le \frac{\exp(2\eta)}{4\eta^2} \left(\frac{\frac{1}{N}\sum_j \left(\exp(\eta \mathfrak{G}_1(\mathbf{W}_j^t)) - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \right)}{\frac{1}{2}\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \right)^2$$
$$= \frac{\exp(2\eta)}{\eta^2} \left(\frac{\frac{1}{N}\sum_j \left(\exp(\eta \mathfrak{G}_1(\mathbf{W}_j^t)) - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \right)}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \right)^2.$$

Consider the numerator in the above. The summands are i.i.d., mean zero, and each is bounded as

$$\left| \exp(\eta \mathfrak{G}_1(\mathbf{W}_j^t)) - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \right|$$
$$= \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \left| \frac{\exp(\eta\mathfrak{G}_1(\mathbf{W}_j^t))}{\exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t))} - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \right|$$
$$\le (\exp(\eta) - \exp(-\eta)) \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)),$$

almost surely, since both ratios lie in $[\exp(-\eta), \exp(\eta)]$. Thus,

$$\mathbb{E}\left(\exp(\eta \mathfrak{G}_1(\mathbf{W}_j^t)) - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \right)^2$$
$$\le (\exp(\eta) - \exp(-\eta))^2 \mathbb{E}\exp(2\eta\mathfrak{G}_0(\mathbf{W}^t))$$
$$\le (\exp(\eta) - \exp(-\eta))^2 \cdot v_t \left(\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))\right)^2.$$

Using that the samples $\mathbf{W}_1^t, \ldots, \mathbf{W}_N^t$ are i.i.d., this implies

$$\mathbb{E}[(\hat{q}_t - q_t)^2 \mathbb{1}\{\mathcal{E}\}] \le \frac{\exp(2\eta)}{\eta^2} \frac{\frac{1}{N^2}\sum_j \mathbb{E}\left(\exp(\eta \mathfrak{G}_1(\mathbf{W}_j^t)) - \frac{\mathbb{E}\exp(\eta\mathfrak{G}_1(\mathbf{W}^t))}{\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))} \exp(\eta\mathfrak{G}_0(\mathbf{W}_j^t)) \right)^2}{\left(\mathbb{E}\exp(\eta\mathfrak{G}_0(\mathbf{W}^t))\right)^2}$$
$$\le \frac{\exp(2\eta)}{\eta^2} \cdot \frac{v_t(\exp(\eta) - \exp(-\eta))^2}{N}$$
$$\lesssim \frac{v_t}{N},$$

where the last step uses $\exp(\eta) - \exp(-\eta) \lesssim \eta$ and $\exp(2\eta) \lesssim 1$ for $\eta \le 1$. On $\mathcal{E}^c$, we simply use that $\hat{q}_t, q_t \in [0,1]$, and get:

$$\mathbb{E}[(\hat{q}_t - q_t)^2 \mathbb{1}\{\mathcal{E}^c\}] \le \mathbb{P}[\mathcal{E}^c] \lesssim \frac{v_t}{N}.$$

Summing the two bounds concludes the proof. ■

Lemma F.3 *For an arbitrary history $\mathbf{S}_{1:t-1}$, an arbitrary covariate X_t, and $\eta \le 1$, we have*

$$v_t \le \exp(2\eta)\, \mathbb{E}\exp\left(2\eta \sup_f \sum_{s=t+1}^{T} \varepsilon_s \frac{2f(X_s) - 1}{2} \right) \lesssim \exp\left(O(\eta\sqrt{dT} + \eta^2 T) \right).$$

Proof Fix $y \in \{0,1\}$, and set $\mathfrak{R}(\mathbf{W}^t) := \sup_f \sum_{s\ge t+1} \varepsilon_s \frac{2f(X_s)-1}{2}$ and $L^\star_{t-1} := \min_f L_{t-1}(f)$. With $\mathfrak{G}_y$ as in Eq. (8), we have

$$\mathfrak{G}_y(\mathbf{W}^t) + L^\star_{t-1} \le \mathfrak{R}(\mathbf{W}^t), \qquad \mathbb{E}_{\mathbf{W}^t}[\mathfrak{G}_y(\mathbf{W}^t)] + L^\star_{t-1} \ge -1.$$

Thus,

$$\begin{aligned}\mathbb{E}\exp(2\eta\mathfrak{G}_y(\mathbf{W}^t)) &= \exp(-2\eta L^\star_{t-1})\,\mathbb{E}\exp(2\eta(\mathfrak{G}_y(\mathbf{W}^t) + L^\star_{t-1})) \\ &\le \exp(-2\eta L^\star_{t-1})\,\mathbb{E}\exp(2\eta\mathfrak{R}(\mathbf{W}^t)).\end{aligned}$$

Also,

$$\begin{aligned}\mathbb{E}\exp(\eta\mathfrak{G}_y(\mathbf{W}^t)) &= \exp(-\eta L^\star_{t-1})\,\mathbb{E}\exp(\eta(\mathfrak{G}_y(\mathbf{W}^t) + L^\star_{t-1})) \\ &\ge \exp(-\eta L^\star_{t-1})\exp(\mathbb{E}\,\eta(\mathfrak{G}_y(\mathbf{W}^t) + L^\star_{t-1})) \\ &\ge \exp(-\eta L^\star_{t-1})\exp(-\eta),\end{aligned}$$

so the common factor $\exp(-2\eta L^\star_{t-1})$ cancels and

$$\frac{\mathbb{E}\exp(2\eta\mathfrak{G}_y(\mathbf{W}^t))}{(\mathbb{E}\exp(\eta\mathfrak{G}_y(\mathbf{W}^t)))^2} \le \exp(2\eta)\cdot\mathbb{E}\exp(2\eta\mathfrak{R}(\mathbf{W}^t)).$$

It remains to control the exponential moment of $\mathfrak{R}(\mathbf{W}^t)$. Note that, conditional on $\mathbf{X}_{t+1:T}$, $\mathfrak{R}(\mathbf{W}^t)$ satisfies the bounded differences condition w.r.t. the independent signs $\boldsymbol{\varepsilon}_{t+1:T}$. Thus,

$$\log\mathbb{E}\left[\exp\left(2\eta\mathfrak{R}(\mathbf{W}^t)\right) \mid \mathbf{X}_{t+1:T}\right] \lesssim \eta\,\mathbb{E}\left[\mathfrak{R}(\mathbf{W}^t) \mid \mathbf{X}_{t+1:T}\right] + \eta^2 T.$$

At the same time, almost surely,

$$\mathbb{E}\left[\mathfrak{R}(\mathbf{W}^t) \mid \mathbf{X}_{t+1:T}\right] \lesssim \sqrt{dT}.$$

Therefore,

$$\log\mathbb{E}\exp\left(2\eta\mathfrak{R}(\mathbf{W}^t)\right) \lesssim \eta\sqrt{dT} + \eta^2 T.$$

This implies

$$\log v_t \lesssim \eta\sqrt{dT} + \eta^2 T,$$

as desired. ∎

Lemma 4.4 *Let $\mathcal{F}$ be a VC dimension d class. Then, an algorithm that plays according to Eq. (9) with N samples per step and inverse temperature η achieves the following regret bound*

$$\mathbb{E}_{P^\star}\mathbf{Reg}(\mathcal{F},T) \lesssim \sqrt{dT} + \eta T + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P) + T\sqrt{\frac{\exp(O(\eta\sqrt{dT}+\eta^2 T))}{N}}.$$

Proof Let $\mathbf{Reg}^\dagger(\mathcal{F},T)$ be the regret of the idealized algorithm that plays Eq. (5). From Lemma 4.2, we have

$$\mathbb{E}_{P^\star}\mathbf{Reg}^\dagger(\mathcal{F},T) \lesssim \sqrt{dT} + \eta T + \frac{1}{\eta}\mathsf{KL}(P^\star \parallel P).$$

Combining Lemmas F.2 and F.3, we have

$$\mathbb{E}\,|\hat{q}_t - q_t| \lesssim \sqrt{\frac{\exp(O(\eta\sqrt{dT} + \eta^2 T))}{N}}.$$

Using the fact that regret is Lipschitz in the learner's strategy, we have

$$\begin{aligned}\mathbb{E}_{P^\star}\,\mathbf{Reg}(\mathcal{F}, T) &\lesssim \mathbb{E}_{P^\star}\,\mathbf{Reg}^\dagger(\mathcal{F}, T) + T\sqrt{\frac{\exp(O(\eta\sqrt{dT} + \eta^2 T))}{N}} \\ &\lesssim \sqrt{dT} + \eta T + \frac{1}{\eta}\mathsf{KL}(P^\star \,\|\, P) + T\sqrt{\frac{\exp(O(\eta\sqrt{dT} + \eta^2 T))}{N}},\end{aligned}$$

as desired. ■

Theorem 4.1 *Let $\mathcal{F}$ be a VC dimension d class. Then, there exists an algorithm that runs in $\mathrm{poly}(T)$ time and, given access to an agnostic ERM oracle over $\mathcal{F}$ and a conditional sampling oracle $\mathcal{O}_P^{\mathsf{cond}}$, achieves the following expected regret bound in the agnostic setting:*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT(1 + \mathsf{KL}(P^\star \,\|\, P))}.$$

Proof Consider Algorithm 2 with some $K \in [T]$, and set $\eta = 1/\sqrt{dL}$ and $N = L$. For every $k \in [K]$, let $B_k := [(k-1)L+1, kL]$ be the rounds in k^{th} epoch, and let $H_k = \bigcup_{i<k} B_i$. Let $\mathbf{Reg}_k$ be the regret of the algorithm in epoch k:

$$\mathbf{Reg}_k(\mathcal{F}, L) = \sum_{t\in B_k} \mathbb{1}\{\hat{Y}_t \neq f^\star(X_t)\} - \inf_f \sum_{t\in B_k} \mathbb{1}\{f(X_t) \neq f^\star(X_t)\}.$$

Then, we have, with probability 1,

$$\mathbf{Reg}(\mathcal{F}, T) \le \sum_{k=1}^{K} \mathbf{Reg}_k(\mathcal{F}, L). \tag{25}$$

Conditionally on X_{H_k}, $\mathbf{Reg}_k$ is upper bounded by Lemma 4.4 as:

$$\mathbb{E}_{P^\star(\cdot|X_{H_k})}\,\mathbf{Reg}_k(\mathcal{F}, L) \lesssim \sqrt{dL} + \sqrt{dL}\cdot\mathsf{KL}\big(P^\star_{B_k}(\cdot \mid X_{H_k}) \,\big\|\, P_{B_k}(\cdot \mid X_{H_k})\big)$$

Thus, averaging out X_{H_k}, we have

$$\mathbb{E}_{P^\star}\,\mathbf{Reg}_k(\mathcal{F}, L) \lesssim \sqrt{dL} + \sqrt{dL}\cdot\mathbb{E}_{P^\star}\,\mathsf{KL}\big(P^\star_{B_k}(\cdot \mid X_{H_k}) \,\big\|\, P_{B_k}(\cdot \mid X_{H_k})\big)$$

Plugging this into Eq. (25), we have

$$\begin{aligned}\mathbb{E}_{P^\star}\,\mathbf{Reg}(\mathcal{F}, T) &\le \sum_{k=1}^{K} \mathbb{E}_{P^\star}\,\mathbf{Reg}_k(\mathcal{F}, L) \\ &\le \sum_{k=1}^{K} \Big[\sqrt{dL} + \sqrt{dL}\cdot\mathbb{E}_{P^\star}\,\mathsf{KL}\big(P^\star_{B_k}(\cdot \mid X_{H_k}) \,\big\|\, P_{B_k}(\cdot \mid X_{H_k})\big)\Big]\end{aligned}$$

$$= K \cdot \sqrt{dL} + \sqrt{dL}\mathsf{KL}(P^\star \,\|\, P)$$
$$= \sqrt{dTK} + \sqrt{d \cdot \frac{T}{K}} \cdot \mathsf{KL}(P^\star \,\|\, P), \tag{26}$$

where in the penultimate equality we used the chain rule for KL.

Now, to tune the value of K in the above, we will run an adaptive variant of exponential weights algorithm over choices of $K = 2^j$, for $j \in \{0, 1, \ldots, \lfloor \log T \rfloor\}$ with initial weights $w_K \propto 2^{-K}$ for each K (so that $\log(1/w_K) = O(K)$). In particular, we use AdaNormalHedge from Luo and Schapire (2015). Note that this results in at most $\log T$ factor of computational overhead. Then, from Eq. (26) and Luo and Schapire (2015), we get:

$$\begin{aligned}
\mathbb{E}_{P^\star} \mathbf{Reg}(\mathcal{F}, T) &\lesssim \min_K \left\{ \sqrt{dTK} + \sqrt{d \cdot \frac{T}{K}} \cdot \mathsf{KL}(P^\star \,\|\, P) + \sqrt{T \log(1/w_K)} \right\} \\
&\lesssim \min_K \left\{ \sqrt{dTK} + \sqrt{d \cdot \frac{T}{K}} \cdot \mathsf{KL}(P^\star \,\|\, P) + \sqrt{TK} \right\} \\
&\lesssim \min_K \left\{ \sqrt{dTK} + \sqrt{d \cdot \frac{T}{K}} \cdot \mathsf{KL}(P^\star \,\|\, P) \right\} \\
&\lesssim \sqrt{dT(1 + \mathsf{KL}(P^\star \,\|\, P))}.
\end{aligned}$$

This concludes the regret bound proof. It remains to observe that the algorithm runs in $\mathrm{poly}(T)$ time in the ERM oracle model. ■

F.2. Proof of the Agnostic Lower Bound with Conditional Sampling Access

Proposition F.4 *Consider the class $\mathcal{F}$ of thresholds on the unit interval $[0, 1]$. Then, for any $D \in [T]$, and for any algorithm, there exists a pair $(P, P^\star)$ such that $\mathsf{KL}(P^\star \,\|\, P) = D$, and the algorithm suffers regret $\geq \sqrt{DT}$ in the agnostic setting.*

Proof The construction below is similar to the standard Littlestone dimension lower bound in agnostic learning (Ben-David et al., 2009). Consider a depth-D Littlestone tree $\mathbf{z}$, and let $N = \lfloor T/D \rfloor$. For each $\boldsymbol{\varepsilon} \in \{\pm 1\}^{\leq D}$, create N distinct copies of the point $\mathbf{z}(\boldsymbol{\varepsilon})$. Specifically, let $\mathbf{z}^i(\boldsymbol{\varepsilon})$ for $i \in [N]$ lie in a δ-neighborhood of $\mathbf{z}(\boldsymbol{\varepsilon})$, with δ small enough that relative order is preserved across all copies, meaning

$$\mathbf{z}(\boldsymbol{\varepsilon}) \leq \mathbf{z}(\boldsymbol{\sigma}) \iff \mathbf{z}^i(\boldsymbol{\varepsilon}) \leq \mathbf{z}^j(\boldsymbol{\sigma}) \text{ for all } i, j \in [N], \boldsymbol{\sigma}, \boldsymbol{\varepsilon} \in \{\pm 1\}^{\leq D}.$$

Let $f^\star$ assign independent uniformly random labels to the points $\{\mathbf{z}^i(\boldsymbol{\varepsilon}) : \boldsymbol{\varepsilon} \in \{\pm 1\}^{\leq D}, i \in [N]\}$. Define P and $P^\star$ as follows. Under P, the sequence is generated in D consecutive blocks: first output $\mathbf{z}^1(\emptyset), \ldots, \mathbf{z}^N(\emptyset)$, then sample $\varepsilon_1 \sim \{0, 1\}$ uniformly and output $\mathbf{z}^1(\varepsilon_1), \ldots, \mathbf{z}^N(\varepsilon_1)$, and continue analogously for D levels. Under $P^\star$, the process is identical except that after outputting a block it chooses the next child by majority vote of the labels in the current block: after outputting $\mathbf{z}^1(\emptyset), \ldots, \mathbf{z}^N(\emptyset)$, it sets $\varepsilon_1' \in \{0, 1\}$ to be the majority label among $f^\star(\mathbf{z}^1(\emptyset)), \ldots, f^\star(\mathbf{z}^N(\emptyset))$ and then outputs $\mathbf{z}^1(\varepsilon_1'), \ldots, \mathbf{z}^N(\varepsilon_1')$, and so on. The two distributions differ only in the D branch choices, hence

$$\mathsf{KL}(P^\star \,\|\, P) = D.$$

Since the labels are independent unbiased coins, any learner suffers expected loss $T/2$. On the other hand, by the Littlestone-tree property there exists a threshold $f^\dagger \in \mathcal{F}$ that realizes the branch labels chosen along the $P^\star$ path, so within each visited block it predicts the majority label. If $S \sim \mathsf{Bin}\,(N, 1/2)$ denotes the number of ones in a block, then the number of mistakes made by $f^\dagger$ on that block is $\min\{S, N-S\}$, and therefore

$$\mathbb{E}\left[\min\{S, N-S\}\right] = \frac{N}{2} - \mathbb{E}\left[\left|S - \frac{N}{2}\right|\right] \le \frac{N}{2} - \Omega\left(\sqrt{N}\right).$$

Summing over the D blocks yields

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left\{f^\dagger(X_t) \neq f^\star(X_t)\right\}\right] \le \frac{DN}{2} - \Omega\left(D\sqrt{N}\right) = \frac{T}{2} - \Omega\left(D\sqrt{N}\right).$$

The learner suffers expected loss $T/2$, since the labels at each new point presented are uniformly random conditioned on the history. Hence, the expected regret is at least $\Omega(D\sqrt{N}) = \Omega(\sqrt{DT})$, which concludes the proof. ■

Appendix G. Proofs from Section 5

G.1. Proof from Section 5.1

Theorem 5.3 (Universal samplable distribution) *Let $p\colon \mathbb{N} \to \mathbb{N}$ be a time-constructible bound. There exists a time bound $p'(t) = \mathrm{poly}(p(t))$ and a sequence of measures $\{\mu^t\}_{t\ge 1}$, with each $\mu^t \in \Delta(\Sigma^*)$ samplable in time $p'(t)$ that satisfy the following. For any p-time samplable distribution P and any $\mathbf{x} \in \mathcal{X}^t$ we have*

$$\log \frac{P_{1:t}(\langle \mathbf{x}\rangle)}{\mu^t(\langle \mathbf{x}\rangle)} \le K^p(P) + 2\log K^p(P) + O(1).$$

Proof Suppose 1^t is given as input. Then, we implement sampling from μ^t as follows. First, we describe sampling from an auxiliary distribution $\tilde{\mu}^t$ supported on encodings of elements of $\mathcal{X}^t$. We sample a random program length n from the distribution supported on $[3p(t)]$ with probabilities proportional to $1/n^2$, sample $\tilde{\pi} \sim \mathsf{Unif}(\{0,1\}^n)$, and sample a seed $\tilde{s} \sim \mathsf{Unif}(\{0,1\}^{p(t)})$. Then, we execute U on $(\tilde{\pi}, 1^t \# \tilde{s})$ for $p(t)$ steps. If the output is a valid encoding of a string in $\mathcal{X}^t$ (that is, it is a delimiter-separated sequence of t binary strings), we output whatever it outputs. Otherwise, we output an encoding of an arbitrary prespecified $\mathbf{x}^0 \in \mathcal{X}^t$. Let $\tilde{\mu}^t$ be the distribution over the output. Also, let ν^t be the uniform distribution on strings in $\Sigma^{\le p(t)}$ with exactly t delimiters ending in a delimiter. Note that any such string is a valid encoding of an element of $\mathcal{X}^t$. Then, we let

$$\mu^t := \frac{1}{2}\tilde{\mu}^t + \frac{1}{2}\nu^t.$$

Note that sampling from μ^t can be implemented in $\mathrm{poly}(p(t))$ time.

Now, we show the desired density ratio upper bound. Note that, for any $\mathbf{x}$ such that $\langle \mathbf{x}\rangle \in \mathrm{supp}(P_{1:t})$, we have $\langle \mathbf{x}\rangle \in \Sigma^{\le p(t)}$ (since it takes time $\le p(t)$ to output $\mathbf{x}$) and $\langle \mathbf{x}\rangle$ has exactly t delimiters. Consider two cases. First, if $K^p(P) \ge 3p(t)$, we have

$$\log \frac{P_{1:t}(\langle \mathbf{x}\rangle)}{\mu^t(\langle \mathbf{x}\rangle)} \le \log \frac{1}{\nu^t(\langle \mathbf{x}\rangle)} + O(1) \le \log|\mathrm{supp}(\nu^t)| + O(1).$$

Using elementary counting, we have:

$$|\operatorname{supp}(\nu^t)| = \sum_{\ell=t}^{p(t)} \binom{\ell-1}{t-1} 2^{\ell-t} \le p(t) 2^{2p(t)} \le 2^{3p(t)}.$$

Thus,

$$\log \frac{P_{1:t}(\langle \mathbf{x} \rangle)}{\mu^t(\langle \mathbf{x} \rangle)} \le 3p(t) + O(1) \le K^p(P) + O(1),$$

which concludes the proof in this case. Now, consider the case $K^p(P) < 3p(t)$. Since P is p-time samplable, by Definition 5.2, there exists a program $\pi^\star$ with $|\pi^\star| = K^p(P)$ such that $U^{p(t)}(\pi^\star, 1^t \# s) \sim P_{1:t}$ when $s \sim \mathsf{Unif}(\{0,1\}^{p(t)})$. Recall that the program $\tilde{\pi}$ and the seed $\tilde{s}$ are drawn independently in the construction of $\tilde{\mu}^t$, and since $|\pi^\star| < 3p(t)$, we have

$$\mathbb{P}[\tilde{\pi} = \pi^\star] \gtrsim \frac{1}{|\pi^\star|^2 \cdot 2^{|\pi^\star|}} = \frac{1}{(K^p(P))^2 \cdot 2^{K^p(P)}}.$$

Conditioned on the event $\{\tilde{\pi} = \pi^\star\}$, the output of the construction is exactly $U^{p(t)}(\pi^\star, 1^t \# \tilde{s})$ with $\tilde{s} \sim \mathsf{Unif}(\{0,1\}^{p(t)})$, which by the above is distributed as $P_{1:t}$. Consequently, for any $\langle \mathbf{x} \rangle \in \operatorname{supp}(P_{1:t})$ we have $\tilde{\mu}^t(\langle \mathbf{x} \rangle) \ge \mathbb{P}[\tilde{\pi} = \pi^\star] \cdot P_{1:t}(\langle \mathbf{x} \rangle)$, and since $\mu^t \ge \frac{1}{2}\tilde{\mu}^t$,

$$\log \frac{P_{1:t}(\langle \mathbf{x} \rangle)}{\mu^t(\langle \mathbf{x} \rangle)} \le \log \frac{2}{\mathbb{P}[\tilde{\pi} = \pi^\star]} \le K^p(P) + 2 \log K^p(P) + O(1),$$

as desired. Moreover, it is easy to see that μ^t is only supported on valid encodings of $\mathcal{X}^t$. This concludes the proof. ■

Theorem 5.4 *Suppose the distribution $P^\star$ of nature is p-time samplable (as per Definition 5.2). Let $\mathcal{F}$ be any class of VC dimension d. Then, there exists an algorithm that, given access to a realizable ERM oracle w.r.t. $\mathcal{F}$, runs in time $\mathrm{poly}(T^d, p(T))$ and for any $f^\star \in \mathcal{F}$, achieves regret*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim d \log(T) + d \cdot 2^{K^p(P^\star) + 2\log K^p(P^\star)}.$$

Proof We run Algorithm 1 with number of layers $L = 1$, using μ^T as the simulator, and the total number of samples $N_1 = T$. Then, from Theorem 5.3, it takes time $O(T \cdot p'(T))$ to draw T samples from μ^T, and it takes time $T^{O(d)}$ to construct a cover $\mathcal{G}_1$ given access to a realizable ERM oracle w.r.t. $\mathcal{F}$. The rest of the proof is concluded similarly to Theorem 3.1. By Lemma 3.2, we have:

$$\mathbb{E}_{\mathbf{Z}}\, \mathbb{E}_{\mathbf{X} \sim \mu^T} \min_{g \in \mathcal{G}_1} \|f - g\|_{\mathbf{X}} \lesssim \frac{d}{N}. \tag{27}$$

Letting $\eta = 1$ be the learning rate, we then have the following regret guarantee from Corollary 2.3 in Cesa-Bianchi and Lugosi (2006):

$$\begin{aligned} \mathbf{Reg} &\lesssim \log |\mathcal{G}_1| + T \cdot \min_{g \in \mathcal{G}_1} \|f - g\|_{\mathbf{X}} \\ &\lesssim d \log T + T \cdot \min_{g \in \mathcal{G}_1} \|f - g\|_{\mathbf{X}}. \end{aligned}$$

Thus,

$$\mathbb{E}_{\mathbf{X}\sim P^\star_{1:T}} \mathbf{Reg} \lesssim d\log T + T\cdot \mathbb{E}_{\mathbf{X}\sim P^\star_{1:T}} \min_{g\in\mathcal{G}_1} \|f-g\|_{\mathbf{X}}\,.$$

Theorem 5.3 tells us that log density ratio between $P^\star$ and μ can be upper bounded uniformly:

$$\log \frac{\mathrm{d}P^\star_{1:T}}{\mathrm{d}\mu^T} \le K^p(P^\star) + 2\log K^p(P^\star) + O(1).$$

Let

$$C_{P^\star} := 2^{K^p(P^\star)+2\log K^p(P^\star)}.$$

Then, using Eq. (27) and a change of measure from μ^T to $P^\star_{1:T}$, we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{X}\sim P^\star_{1:T}} \mathbf{Reg} &\lesssim d\log T + C_{P^\star}\cdot T\cdot \mathbb{E}_{\mathbf{X}\sim \mu^T} \min_{g\in\mathcal{G}_1} \|f-g\|_{\mathbf{X}} \\
&\lesssim d\log T + C_{P^\star}\cdot T\cdot \frac{d}{N} \\
&= d\log T + d\cdot C_{P^\star},
\end{aligned}$$

as desired. Note that it takes time $O(|\mathcal{G}|) \le \mathrm{poly}(T^d)$ per round to run the exponential weights update. Thus, the algorithm runs in time $\mathrm{poly}(T^d, p(T))$. This concludes the proof. ■

G.2. Proofs from Section 5.2

Throughout this subsection, we use the notation introduced in Section 5.2. We begin by analyzing the worst-case KL divergence between our simulator and the class of LDS.

Lemma G.1 *Let P be defined as:*

$$P(\cdot \mid X_0) := \mathbb{E}_{A\sim\pi} P^A(\cdot \mid X_0). \tag{28}$$

Then, for any $A^\star \in \Theta$ with $\rho(A^\star) \le 1$ and $X_0 \in \mathcal{B}(0,R)$, we have

$$\mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) \le O\left(n^3 \log\left(nBT(R/\sigma+1)\right)\right).$$

Proof Fix $A^\star \in \Theta$ with $\rho(A^\star) \le 1$ and $X_0 \in \mathcal{B}(0,R)$ arbitrarily, and let $\varepsilon \in (0, B]$. Define the localized neighborhood

$$\mathcal{B}_\infty(A^\star, \varepsilon) := \{A \in \mathbb{R}^{n\times n} \colon \|A - A^\star\|_\infty \le \varepsilon\}.$$

Under the Gaussian prior $\pi = \mathcal{N}(0, B^2)^{n\times n}$, each coordinate A_{ij} is independent and its density on $[A^\star_{ij}-\varepsilon, A^\star_{ij}+\varepsilon]$ is lower-bounded by $\frac{1}{\sqrt{2\pi}B}\exp\left(-\frac{(|A^\star_{ij}|+\varepsilon)^2}{2B^2}\right)$, so

$$\pi(\mathcal{B}_\infty(A^\star,\varepsilon)) \ge \prod_{i,j} \frac{2\varepsilon}{\sqrt{2\pi}B}\exp\left(-\frac{(|A^\star_{ij}|+\varepsilon)^2}{2B^2}\right) = \left(\frac{2\varepsilon}{\sqrt{2\pi}B}\right)^{n^2}\exp\left(-\frac{1}{2B^2}\sum_{i,j}(|A^\star_{ij}|+\varepsilon)^2\right).$$

Since $A^\star \in [-B, B]^{n\times n}$ and $\varepsilon \le B$, $(|A^\star_{ij}| + \varepsilon)^2 \le 4B^2$, so the exponent is at most $2n^2$. Hence

$$\log \frac{1}{\pi(\mathcal{B}_\infty(A^\star, \varepsilon))} \le n^2 \log(B/\varepsilon) + O(n^2).$$

By a standard localization argument (e.g., (Polyanskiy and Wu, 2025)),

$$\mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) \le n^2 \log(B/\varepsilon) + O(n^2) + \sup_{A\in\mathcal{B}_\infty(A^\star,\varepsilon)} \mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P^{A}(\cdot \mid X_0)\Big).$$

For the supremum, since the initial state X_0 is fixed under both laws, the chain rule for KL together with the closed form

$$\mathsf{KL}\big(\mathcal{N}(\mu_1, \sigma^2 I_n) \,\big\|\, \mathcal{N}(\mu_2, \sigma^2 I_n)\big) = \|\mu_1 - \mu_2\|_2^2 / (2\sigma^2)$$

gives

$$\mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P^{A}(\cdot \mid X_0)\Big) = \frac{1}{2\sigma^2} \sum_{t=1}^{T} \mathbb{E}_{P^{A^\star}(\cdot|X_0)} \|(A^\star - A)X_{t-1}\|_2^2 . \tag{29}$$

For $A \in \mathcal{B}_\infty(A^\star, \varepsilon)$ we have $\|A^\star - A\|_F \le \varepsilon n$, so $\|(A^\star - A)X_{t-1}\|_2^2 \le \varepsilon^2 n^2 \|X_{t-1}\|_2^2$, and it remains to bound $\sum_{t=1}^{T} \mathbb{E}\|X_{t-1}\|_2^2$. It is well-known that the operator norm of powers of marginally stable matrices grows polynomially (Horn and Johnson, 2012; Sarkar and Rakhlin, 2019; Simchowitz et al., 2018). In particular, we will use the following statement, the proof of which is deferred to Lemma G.2 for self-containedness.

$$\|(A^\star)^s\|_{\mathrm{op}} \le n \cdot (\|A^\star\|_{\mathrm{op}} + 1)^{n-1} \cdot s^{n-1}.$$

Now, since $\|A^\star\|_{\mathrm{op}} \le \|A^\star\|_F \le Bn$, the above gives

$$\|(A^\star)^s\|_{\mathrm{op}} \le n(Bn+1)^{n-1} s^{n-1} =: C_n \cdot s^{n-1},$$

where $C_n := n(Bn+1)^{n-1}$ depends only on B and n. Therefore $\|(A^\star)^s\|_F^2 \le n\,\|(A^\star)^s\|_{\mathrm{op}}^2 \le nC_n^2 s^{2(n-1)}$. Decomposing $X_t = (A^\star)^t X_0 + \sum_{s=0}^{t-1} (A^\star)^s \eta_{t-1-s}$ and using independence of the η_s,

$$\mathbb{E}\|X_t\|_2^2 = \big\|(A^\star)^t X_0\big\|_2^2 + \sigma^2 \sum_{s=0}^{t-1} \|(A^\star)^s\|_F^2 \le C_n^2 \cdot t^{2(n-1)} R^2 + \sigma^2 n C_n^2 \cdot t^{2n-1}.$$

Summing $\sum_{t=1}^{T-1} t^{2(n-1)} \le T^{2n-1}$ and $\sum_{t=1}^{T-1} t^{2n-1} \le T^{2n}$, and adding $\|X_0\|_2^2 \le R^2$, yields

$$\sum_{t=1}^{T} \mathbb{E}\|X_{t-1}\|_2^2 \le 2C_n^2 T^{2n}(R^2 + \sigma^2 n). \tag{30}$$

Combining Eqs. (29) and (30), we get

$$\sup_{A\in\mathcal{B}_\infty(A^\star,\varepsilon)} \mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P^{A}(\cdot \mid X_0)\Big) \le \frac{\varepsilon^2 n^2 C_n^2 T^{2n}(R^2 + \sigma^2 n)}{\sigma^2}.$$

Thus,

$$\mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) \le n^2 \log(B/\varepsilon) + O(n^2) + \frac{\varepsilon^2 n^2 C_n^2 T^{2n}(R^2 + \sigma^2 n)}{\sigma^2}.$$

Balancing in the above by choosing $\varepsilon^2 = \sigma^2/(n^2 C_n^2 T^{2n}(R^2 + \sigma^2 n))$ gives:

$$\mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) = O\left(n^2 \log\left(\frac{BC_n T^n (R + \sigma\sqrt{n})}{\sigma}\right)\right) = O\left(n^3 \log\left(nBT(R/\sigma + 1)\right)\right),$$

as desired. ■

Lemma G.2 *For any $A \in \mathbb{R}^{n\times n}$ with $\rho(A) \le 1$ and any $s \ge 1$,*

$$\|A^s\|_{\mathrm{op}} \le n \cdot (\|A\|_{\mathrm{op}} + 1)^{n-1} \cdot s^{n-1}.$$

Proof We consider the Schur decomposition of A, namely, $A = QUQ^*$, where $Q \in \mathbb{C}^{n\times n}$ is unitary and $U \in \mathbb{C}^{n\times n}$ is upper triangular with the eigenvalues of A on its diagonal. Since Q is unitary, $\|A^s\|_{\mathrm{op}} = \|U^s\|_{\mathrm{op}}$ and $\|U\|_{\mathrm{op}} = \|A\|_{\mathrm{op}}$. Decompose $U = D + N$ where D is the diagonal of U and N is strictly upper triangular. Since $\rho(A) \le 1$, the diagonal entries of D have magnitude at most 1, so $\|D\|_{\mathrm{op}} \le 1$ and $\|N\|_{\mathrm{op}} \le \|U\|_{\mathrm{op}} + \|D\|_{\mathrm{op}} \le \|A\|_{\mathrm{op}} + 1$.

Now, we expand $U^s = (D+N)^s$ as follows:

$$U^s = \sum_{j=0}^{s} \sum_{\substack{a_0,\ldots,a_j \ge 0 \\ a_0+\cdots+a_j = s-j}} D^{a_0} N D^{a_1} N \cdots N D^{a_j}.$$

We observe that any term above corresponding to $j \ge n$ vanishes. Indeed, since N is strictly upper triangular, and D is diagonal, the product of $\ge n$ matrices N and any number of matrices D in any order is equal to zero. Thus, the above sum can be rewritten in the following form:

$$U^s = \sum_{j=0}^{n-1} \sum_{\substack{a_0,\ldots,a_j \ge 0 \\ a_0+\cdots+a_j = s-j}} D^{a_0} N D^{a_1} N \cdots N D^{a_j}.$$

Now, each summand above has operator norm at most $\|D\|_{\mathrm{op}}^{s-j} \|N\|_{\mathrm{op}}^{j} \le (\|A\|_{\mathrm{op}}+1)^j$. The number of compositions $(a_0, \ldots, a_j)$ with sum $s-j$ is $\binom{s}{j}$, and $\binom{s}{j} \le s^j \le s^{n-1}$ for $j \le n-1$. Therefore

$$\|U^s\|_{\mathrm{op}} \le \sum_{j=0}^{n-1} \binom{s}{j} (\|A\|_{\mathrm{op}} + 1)^j \le s^{n-1} \sum_{j=0}^{n-1} (\|A\|_{\mathrm{op}} + 1)^j \le n \cdot (\|A\|_{\mathrm{op}} + 1)^{n-1} \cdot s^{n-1},$$

which concludes the proof. ■

Proposition G.3 *For any $X_0, \mathbf{X}_{1:t}$, exact conditional sampling from $P(\cdot \mid \mathbf{X}_{1:t}, X_0)$, where P is as in Lemma G.1, can be implemented in $\mathrm{poly}(n, T)$ time per sample.*

Proof We want to sample $\mathbf{X}_{t+1:T} \sim P(\cdot \mid \mathbf{X}_{1:t}, X_0)$. Recall that P is defined as a mixture over the prior π:

$$P(\cdot \mid X_0) = \mathbb{E}_{A\sim\pi}\, P^A(\cdot \mid X_0)$$

Then,

$$P(\cdot \mid \mathbf{X}_{1:t}, X_0) = \mathbb{E}_{A\sim\pi(\cdot\mid X_{1:t})}\, P^A(\cdot \mid \mathbf{X}_{1:t}, X_0).$$

That is, to sample $\mathbf{X}_{t+1:T} \sim P(\cdot \mid \mathbf{X}_{1:t}, X_0)$, it suffices to sample A from the posterior $\pi(\cdot \mid X_{1:t})$, and subsequently sample from $P^A(\cdot \mid \mathbf{X}_{1:t}, X_0)$. The second part can clearly be done efficiently. Thus, the problem reduces to sampling from the posterior $\pi(\cdot \mid X_{1:t})$.

By the Markov property and the Gaussian transition $P^A(X_{s+1} \in \cdot \mid X_s = x) = \mathcal{N}(Ax, \sigma^2 I_n)$, we have

$$\log \pi(A \mid \mathbf{X}_{1:t}, X_0) = \log \pi(A) - \frac{1}{2\sigma^2} \sum_{s=0}^{t-1} \|X_{s+1} - AX_s\|_2^2 + \text{const}.$$

Both terms on the right-hand side are quadratic in A, so the posterior is itself a Gaussian on $\mathbb{R}^{n\times n}$ with mean and covariance computable in closed form, and exact sampling is trivial. ■

Theorem 5.5 *Let $\mathcal{F}$ be a class of VC dimension d, and let $R, B \geq 1$, $\sigma > 0$ and the initial state $X_0 \in \mathcal{B}(0, R)$ be arbitrary. There is a $\mathrm{poly}(n, T)$-time algorithm that, given access to an agnostic ERM oracle for $\mathcal{F}$, the noise scale σ, and the magnitude bound B, achieves expected regret*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT}\left(1 + \sqrt{n^3 \log\left(nBT(R/\sigma + 1)\right)}\right)$$

in the agnostic setting, simultaneously for every $P^\star = P^{A^\star}$ with $A^\star \in \Theta$ satisfying $\rho(A^\star) \leq 1$.

Proof We instantiate Theorem 4.1 with the simulator P defined in Section 5.2. By Lemma G.1,

$$\sup_{\substack{A^\star\in\Theta,\,\rho(A^\star)\leq 1,\\ X_0\in\mathcal{B}(0,R)}} \mathsf{KL}\Big(P^{A^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) = O\left(n^3 \log\left(nBT(R/\sigma + 1)\right)\right),$$

and by Proposition G.3 the conditional sampler runs in $\mathrm{poly}(n, T)$ time per round. Substituting into Theorem 4.1 gives expected regret

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT}\left(1 + \sqrt{n^3 \log\left(nBT(R/\sigma + 1)\right)}\right)$$

in $\mathrm{poly}(n, T)$ time per round, as claimed. ■

G.3. Proofs from Appendix B

Throughout this subsection, we use the notation introduced in Appendix B.

Lemma G.4 *For any $T \geq 2$, $\theta^\star \in \Theta$, and $X_0 \in \{0, 1\}^n$, we have*

$$\mathsf{KL}\Big(P^{\theta^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) \lesssim p \log(BTn).$$

Proof Fix $\theta^\star \in \Theta$ and $X_0 \in \{0,1\}^n$ arbitrarily, and let $\varepsilon \in (0, B]$. By the Markov property, the trajectory likelihood factors as

$$P^\theta(\mathbf{I}, \mathbf{X} \mid X_0) = \prod_{t=1}^{T} P^\theta(I_t, X_t \mid X_{t-1}).$$

Define the localized neighborhood $\mathcal{B}_\infty(\theta^\star, \varepsilon) := \{\theta \in \mathbb{R}^p \colon \|\theta - \theta^\star\|_\infty \le \varepsilon\}$, so $\pi(\mathcal{B}_\infty(\theta^\star, \varepsilon) \cap \Theta) \ge (\varepsilon/2B)^p$. By a standard localization argument (e.g., (Polyanskiy and Wu, 2025)),

$$\begin{aligned}\mathsf{KL}\Big(P^{\theta^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) &\le \log \frac{1}{\pi(\mathcal{B}_\infty(\theta^\star, \varepsilon) \cap \Theta)} + \sup_{\substack{\theta \in \mathcal{B}_\infty(\theta^\star, \varepsilon) \cap \Theta, \\ (\mathbf{I}, \mathbf{X})}} \log \frac{P^{\theta^\star}(\mathbf{I}, \mathbf{X} \mid X_0)}{P^\theta(\mathbf{I}, \mathbf{X} \mid X_0)} \\ &\le p \log(2B/\varepsilon) + \sup_{\substack{\theta \in \mathcal{B}_\infty(\theta^\star, \varepsilon) \cap \Theta, \\ (\mathbf{I}, \mathbf{X})}} \log \frac{P^{\theta^\star}(\mathbf{I}, \mathbf{X} \mid X_0)}{P^\theta(\mathbf{I}, \mathbf{X} \mid X_0)}.\end{aligned}$$

For the last term, by the Markov factorization and additivity of $\log$,

$$\log \frac{P^{\theta^\star}(\mathbf{I}, \mathbf{X} \mid X_0)}{P^\theta(\mathbf{I}, \mathbf{X} \mid X_0)} = \sum_{t=1}^{T} \log \frac{P^{\theta^\star}(I_t, X_t \mid X_{t-1})}{P^\theta(I_t, X_t \mid X_{t-1})}.$$

The one-step kernel can be written as

$$P^\theta(I_t, X_t \mid X_{t-1}) = \frac{1}{n} \cdot \sigma(u^\theta_{I_t}(X_{t-1}))^{X_{t,I_t}} \Big(1 - \sigma(u^\theta_{I_t}(X_{t-1}))\Big)^{1 - X_{t,I_t}} \cdot \mathbb{1}\{X_{t,j} = X_{t-1,j} \; \forall j \ne I_t\}.$$

Then, after some algebra, the per-step log-ratio can be written as:

$$\begin{aligned}\log \frac{P^{\theta^\star}(I_t, X_t \mid X_{t-1})}{P^\theta(I_t, X_t \mid X_{t-1})} &= X_{t,I_t} \Big(u^{\theta^\star}_{I_t}(X_{t-1}) - u^\theta_{I_t}(X_{t-1})\Big) \\ &\quad - \log\Big(1 + \exp(u^{\theta^\star}_{I_t}(X_{t-1}))\Big) + \log\Big(1 + \exp(u^\theta_{I_t}(X_{t-1}))\Big).\end{aligned}$$

Summing over t and bounding by the supremum over (i, x', x),

$$\log \frac{P^{\theta^\star}(\mathbf{I}, \mathbf{X} \mid X_0)}{P^\theta(\mathbf{I}, \mathbf{X} \mid X_0)} \le \sum_{t=1}^{T} \sup_{(i,x',x)} \Big[x'_i(u^{\theta^\star}_i(x) - u^\theta_i(x)) - \log(1 + \exp(u^{\theta^\star}_i(x))) + \log(1 + \exp(u^\theta_i(x)))\Big].$$

Since $\lambda \mapsto \log(1 + \exp(\lambda))$ is 1-Lipschitz, $x'_i \in \{0,1\}$, and $|u^{\theta^\star}_i(x) - u^\theta_i(x)| \le n \|\theta^\star - \theta\|_\infty \le n\varepsilon$ for $\theta \in \mathcal{B}_\infty(\theta^\star, \varepsilon)$, each per-step term is $\lesssim n\varepsilon$, so the sum is $\lesssim Tn\varepsilon$. Combining,

$$\mathsf{KL}\Big(P^{\theta^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) \lesssim p \log(B/\varepsilon) + Tn\varepsilon.$$

We now choose $\varepsilon = B \wedge p/(Tn)$. If $p/(Tn) \le B$, then $\varepsilon = p/(Tn)$ and the bound becomes $p \log(BTn/p) + p \lesssim p \log(BTn)$. Otherwise $\varepsilon = B$, in which case the first term vanishes and $Tn\varepsilon = TnB \le p \lesssim p \log(BTn)$ (using $p/(Tn) > B$). In either case we obtain the desired bound. ■

Proposition G.5 *For any $X_0 \in \{0,1\}^n$, conditional sampling from $P(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)$, where P is as in Lemma G.4, can be implemented in $\mathrm{poly}(n, T, B, 1/\delta)$ time per sample, to within total-variation error δ.*

Proof We want to sample $(\mathbf{I}_{t+1:T}, \mathbf{X}_{t+1:T}) \sim P(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)$. Recall that P is defined as a mixture over the prior π:

$$P(\cdot \mid X_0) = \mathbb{E}_{\theta\sim\pi} P^\theta(\cdot \mid X_0).$$

Then,

$$P(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0) = \mathbb{E}_{\theta\sim\pi(\cdot\mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)} P^\theta(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0).$$

That is, to sample $(\mathbf{I}_{t+1:T}, \mathbf{X}_{t+1:T}) \sim P(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)$, it suffices to sample θ from the posterior $\pi(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)$, and subsequently sample from $P^\theta(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)$. The second part can clearly be done efficiently by forward-simulating Glauber dynamics from X_t for $T-t$ steps, taking $O((T-t)n)$ time. Thus, the problem reduces to sampling from the posterior $\pi(\cdot \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0)$.

Writing out the log-density of the posterior on Θ, we have

$$\log \pi(\theta \mid \mathbf{I}_{1:t}, \mathbf{X}_{1:t}, X_0) = \sum_{s=1}^{t} X_{s,I_s} \log \sigma(u^\theta_{I_s}(X_{s-1})) + (1 - X_{s,I_s}) \log\Big(1 - \sigma(u^\theta_{I_s}(X_{s-1}))\Big) + \text{const.}$$

The local field $u^\theta_i(x) = \sum_{j\neq i} A_{ij} x_j + h_i$ is linear in $\theta = (A, h)$, so we can write $u^\theta_{I_s}(X_{s-1}) = \langle \phi_s, \theta\rangle$ where $\phi_s \in \{0,1\}^p$ has at most n non-zero entries (in particular, $\|\phi_s\|_2 \le \sqrt{n}$). Using $\frac{\mathrm{d}}{\mathrm{d}\lambda} \log\sigma(\lambda) = 1 - \sigma(\lambda)$ and $\frac{\mathrm{d}}{\mathrm{d}\lambda} \log(1-\sigma(\lambda)) = -\sigma(\lambda)$, the gradient and Hessian of the log-posterior are

$$\nabla_\theta \log\pi = \sum_{s=1}^{t} \left(X_{s,I_s} - \sigma(\langle\phi_s,\theta\rangle)\right)\phi_s, \qquad \nabla^2_\theta \log\pi = -\sum_{s=1}^{t} \sigma(\langle\phi_s,\theta\rangle)\left(1-\sigma(\langle\phi_s,\theta\rangle)\right)\phi_s\phi_s^\top.$$

Since $\sigma(1-\sigma) \le 1/4$ and $\phi_s\phi_s^\top \preceq \|\phi_s\|_2^2 \cdot I_p \le n \cdot I_p$, the triangle inequality for the operator norm gives

$$\left\|\nabla^2_\theta \log\pi\right\|_{\mathrm{op}} \le \sum_{s=1}^{t} \frac{1}{4}\|\phi_s\|_2^2 \le \frac{Tn}{4}.$$

In particular, $\nabla^2_\theta \log\pi \preceq 0$, so the log-posterior is concave on $\mathbb{R}^p$ and the posterior is log-concave restricted to the bounded convex body $\Theta = [-B, B]^p$ (of Euclidean diameter at most $2B\sqrt{p}$). Thus, results for log-concave sampling with smooth densities (Altschuler and Talwar, 2022) imply that we can sample from the above up to error δ in $\mathrm{poly}(p, B, T, 1/\delta)$ time, as desired. ■

Theorem B.1 *Let $\mathcal{F}$ be a class of VC dimension d, and let $B \ge 1$. There is a $\mathrm{poly}(n, T, B)$-time algorithm that, given access to an agnostic ERM oracle for $\mathcal{F}$, achieves expected regret*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT} \cdot \left(1 + \sqrt{n^2 \log(BTn)}\right)$$

in the agnostic setting, simultaneously for every $P^\star = P^{\theta^\star}$ with $\theta^\star \in \Theta$ and initial state $X_0 \in \{0,1\}^n$.

Proof For $T = 1$ the regret is trivially bounded by 1, so assume $T \geq 2$. We instantiate Theorem 4.1 with the simulator P defined in Appendix B. By Lemma G.4,

$$\sup_{\theta^\star \in \Theta,\, X_0 \in \{0,1\}^n} \mathsf{KL}\Big(P^{\theta^\star}(\cdot \mid X_0) \,\Big\|\, P(\cdot \mid X_0)\Big) \lesssim p \log(BTn) \lesssim n^2 \log(BTn).$$

By Proposition G.5, for any $\delta > 0$ there is a conditional sampler running in $\mathrm{poly}(n, T, B, 1/\delta)$ time per call that samples to within total-variation distance δ. The algorithm of Theorem 4.1 issues $M = \mathrm{poly}(T)$ such calls over the horizon; coupling each approximate draw with an exact one, the two executions agree except with probability at most $M\delta$, so the expected regret changes by at most $2TM\delta$. Taking $\delta = 1/\mathrm{poly}(T)$ small enough makes this $O(1)$ while keeping the per-call running time $\mathrm{poly}(n, T, B)$. Substituting into Theorem 4.1, we obtain:

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT}\left(1 + \sqrt{n^2 \log(BTn)}\right),$$

in $\mathrm{poly}(n, T, B)$ time per round, as claimed. ■

Appendix H. Learning with Adaptive Labels

H.1. Discussion on Assumption 1

Assumption 1, intuitively, formalizes the following belief: nature's play may be complex, but it is not adversarial. More concretely, we assume that the data generated by nature may be non-i.i.d., but the "ground truth" $f^\star$ does not adversarially change over time or based on actions of the learner. From this point of view Assumption 1 can be seen as a modeling assumption that is suitable for many online learning use cases.

On the other hand, Assumption 1 is motivated by the fact that Assumption 2 alone is not sufficient to recover learnability of VC classes if the labels are allowed to depend on the realized covariates $(X_1, \ldots, X_T)$ in an arbitrary manner. Concretely, even with exact knowledge of the distribution of $(X_1, \ldots, X_T)$, it is possible to replicate the standard Littlestone tree hardness construction for learning thresholds. It suffices to let $P^\star$ pick out the uniform path in the Littlestone tree, and let $Y_t = 1$ if X_{t+1} is the right child of X_t and 0 otherwise. This can be seen to force linear regret for the learner.

At the same time, milder forms of dependence of Y_t on $X_{1:T}$ can be allowed if we have conditional sampling access. In particular, all of our positive results for this setting can be extended to the scenario where Y_t is allowed to depend on $X_{1:t}$ but not on $X_{t+1:T}$.

In this section, we keep Assumption 2 and discuss alternative assumptions about the dependence of labels on the realized covariates. First, we show that allowing *arbitrary* dependence of labels on the covariate realization still suffers from classical impossibility results in the online learning literature, and essentially demands bounded Littlestone dimension for sublinear regret.

H.2. Impossibility of Learning with Arbitrary Dependence

Suppose that each label Y_t is allowed to depend in an arbitrary fashion on the realization of $\mathbf{X} \sim P^\star$. Then, any class with infinite Littlestone dimension is not learnable.

Proposition H.1 *Consider any class $\mathcal{F}$ with infinite Littlestone dimension, and consider the Littlestone tree of depth T. Let $P^\star$ be a distribution that samples a uniform path $\mathbf{X}$ in the Littlestone tree, and, for each $t \le T-1$, set $Y_t := 1$ iff X_{t+1} is the right child of its parent (and sample Y_T randomly). This strategy of nature forces any learner to have $\ge T/2$ regret.*

Proof We observe that the sampling procedure used by nature is equivalent to the one in Proposition D.4. Thus, any learner suffers $\ge T/2$ expected regret. ■

We point out that, since we fix $P^\star$ in the hardness construction, the learner "knows" $P^\star$, so Assumption 2 and Assumption 3 both hold.

H.3. Labels Dependent on the History

Another form of dependence we consider is allowing the labels to depend on *past* covariates only, and, possibly, the actions of the learner. More formally, letting

$$\mathcal{H}_t := (\mathbf{X}_{1:t}, \mathbf{Y}_{1:t-1}, \hat{\mathbf{Y}}_{1:t-1})$$

denote the history up to time t, we allow the distribution Y_t to depend on $\mathcal{H}_t$ but not on future covariates. Equivalently, the conditional law of future covariates must be conditionally independent from the realization of Y_t. We formalize this assumption as follows.

Assumption 4 *For every $t \in [T]$, the distribution $\mathbf{X}_{t+1:T} \mid (\mathcal{H}_t, Y_t, \hat{Y}_t)$ is given by $P^\star(\cdot \mid \mathbf{X}_{1:t})$.*

In this situation, the picture is more complex. It turns out, even with $P^\star = P$, learning with unconditional samples is *statistically impossible* here. However, with conditional sampling access to P, our positive results in Theorem 4.1 still hold.

Proposition H.2 *Let $\mathcal{F}$ be the class of thresholds on the $[0,1]$ interval. For any $N \ge 1$, and for any learning algorithm that draws at most N unconditional samples, there exists a choice of a distribution $P^\star$ and a choice of realizable labels satisfying Assumption 4 such that*

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{Y_t \neq \hat{Y}_t\} - \inf_{f\in\mathcal{F}} \sum_{t=1}^{T} \mathbb{1}\{Y_t \neq f(X_t)\}\right] \ge T/8.$$

Proof The hardness construction we give here is the same as the one given in Theorem 3.5. WLOG, assume $N \ge 2$. Divide the interval $[0,1]$ into N sub-intervals $\{A_i\}_{i\in[N]}$, and consider N Littlestone trees in each respective interval. For each $i \in [N]$, let ρ_i be a distribution that samples a random path in the Littlestone tree in interval A_i. For each $i \in [N]$, sample a path $\mathbf{x}^i \sim \rho_i$. Now, let $P^\star$ be defined in the following way. Set

$$P^\star := \frac{1}{N} \sum_{i\in[N]} \delta_{\mathbf{x}^i}.$$

On each interval A_i, $i \in [N]$, let f_i be a threshold function consistent with the path $\mathbf{x}^i$ (see Definition D.5). Let $i_\star$ be a random index such that $\mathbf{X} = \mathbf{x}^{i_\star}$. Note that $i_\star$ is X_1-measurable. For each $t \ge 1$, set $Y_t := f_{i_\star}(X_t)$. Then, with probability 1 over the draw from $\mathbf{X} \sim P^\star$, the labels $\mathbf{Y}_{1:T}$ are realizable w.r.t. $\mathcal{F}$ on $\mathbf{X}$, that is, with probability 1:

$$\inf_{f\in\mathcal{F}} \sum_{t=1}^{T} \mathbb{1}\{Y_t \neq f(X_t)\} = 0.$$

At the same time, with probability at least

$$\left(1 - \frac{1}{N}\right)^N \geq \frac{1}{4},$$

none of the samples the learner draws are equal to $\mathbf{x}^{i_\star}$. Let us condition on this event. Then, for any $t \in [T]$ the randomness of $\mathbf{x}^{i_\star}$ is independent of the randomness of the learner in round t, that is, the learner suffers expected regret of $T/2$ by the standard Littlestone hardness result (see Proposition D.4). Thus,

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{Y_t \neq \hat{Y}_t\} - \inf_{f \in \mathcal{F}} \sum_{t=1}^{T} \mathbb{1}\{Y_t \neq f(X_t)\}\right] \geq \frac{1}{4} \cdot \frac{T}{2} = \frac{T}{8},$$

as desired. ■

In contrast, all positive results from Section 4 still hold in this setting. Indeed, since the proof of Theorem 4.1 is phrased in terms of relaxations, it permits the dependence of Y_t on $\mathcal{H}_t$. Thus, the following result still holds in this setting.

Theorem H.3 *Let $\mathcal{F}$ be a class of VC dimension d. Suppose the labels satisfy Assumption 4. Then, there exists an algorithm that, given access to an agnostic ERM oracle over $\mathcal{F}$ and a conditional sampling oracle $\mathcal{O}_P^{\mathsf{cond}}$, achieves the expected regret bound*

$$\mathbb{E}\,\mathbf{Reg}(\mathcal{F}, T) \lesssim \sqrt{dT(1 + \mathsf{KL}(P^\star \,\|\, P))},$$

in both realizable and agnostic settings. Moreover, the algorithm runs in $\mathrm{poly}(T)$ time.