

The Complexity of Sequential Prediction in Dynamical Systems

Vinod Raman*

University of Michigan, Ann Arbor, 48104

VKRAMAN@UMICH.EDU

Unique Subedi*

University of Michigan, Ann Arbor, 48104

SUBEDI@UMICH.EDU

Ambuj Tewari

University of Michigan, Ann Arbor, 48104

TEWARIA@UMICH.EDU

Editors: N. Ozay, L. Balzano, D. Panagou, A. Abate

Abstract

We study the problem of learning to predict the next state of a dynamical system when the underlying evolution function is unknown. Unlike previous work, we place no parametric assumptions on the dynamical system, and study the problem from a learning theory perspective. We define new combinatorial measures and dimensions and show that they quantify the optimal mistake and regret bounds in the realizable and agnostic settings respectively. By doing so, we find that in the realizable setting, the total number of mistakes can grow according to *any* increasing function of the time horizon T . In contrast, we show that in the agnostic setting under the commonly studied notion of Markovian regret, the only possible rates are $\Theta(T)$ and $\tilde{\Theta}(\sqrt{T})$.¹

Keywords: Discrete-time Dynamical Systems, Online Learnability

1. Introduction

A discrete-time dynamical system is a mathematical model that describes the evolution of a system over discrete time steps. Formally, a discrete-time dynamical system is a tuple $(\mathbb{N}, \mathcal{X}, f)$, where \mathbb{N} is the set of natural numbers that denote the timesteps, \mathcal{X} is a non-empty set called the state space, and $f : \mathcal{X} \rightarrow \mathcal{X}$ is a deterministic map that describes the evolution of the state. Dynamical systems have been widely used in practice due to their ability to accurately model natural phenomena. For instance, boolean networks are an important class of discrete-time, discrete-space dynamical systems with widespread applicability to genetic modeling (Kauffman, 1969; Shmulevich et al., 2002). In a boolean network, the state space is $\mathcal{X} = \{0, 1\}^n$ with $|\mathcal{X}|^{\mathcal{X}} = (2^n)^{2^n}$ possible evolution functions. For genetic modeling, n is taken to be the number of genes and $x \in \mathcal{X}$ indicates the expression of all n genes under consideration. As an example, “1” could represent the gene with a high concentration of a certain protein, and “0” could represent the gene with a low concentration. With such formulation, one can study how the state of these genes evolves over time under certain medical interventions. Beyond genetics, dynamical systems have been used in control (Li et al., 2019), computer vision (Doretto et al., 2003), and natural language processing (Sutskever et al., 2014; Belanger and Kakade, 2015).

In this work, we consider the problem of predicting the next state of a dynamical system when the underlying evolution function is unknown (Ghai et al., 2020). To capture the sequential nature of dynamical systems, we consider the model where the learner plays a sequential game with nature

* Equal Contribution

1. The full paper is available at: <https://arxiv.org/abs/2402.06614>.

over T rounds. At the beginning of the game, nature reveals the initial state $x_0 \in \mathcal{X}$. In each round $t \in [T]$, the learner makes its prediction of the next state $\hat{x}_t \in \mathcal{X}$, nature reveals the true next state $x_t \in \mathcal{X}$, and the learner suffers loss $\mathbb{1}\{\hat{x}_t \neq x_t\}$. Given an evolution class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, the goal of the learner is to output predictions of the next state such that its regret, the difference between its cumulative mistakes and the cumulative mistakes of the best-fixed evolution in hindsight (defined formally in Section 2.2), is small. The class \mathcal{F} is said to be learnable if there exists a learning algorithm whose regret is a sublinear function of the time horizon T .

Although we allow the state space \mathcal{X} to be arbitrary, we only consider the 0-1 loss, which may be more appropriate for *discrete-space* dynamical systems. However, even discrete-space dynamical systems can be very expressive, capturing complex processes like cellular automata (Hoekstra et al., 2010; Wolfram, 1986) and language modeling (Elman, 1995). For example, let \mathcal{V} be a countable token space and $\mathcal{X} = \mathcal{V}^*$ to be the state space containing all finite sequences of elements from \mathcal{V} . Consider a function class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ with the following property: for every $f \in \mathcal{F}$ and any input $x \in \mathcal{X}$, $f(x) = x \circ v$, where we use \circ to represent the concatenation operator. Then, \mathcal{F} is a set of auto-regressive models which given a sequence of tokens $x \in \mathcal{X}$, predicts the next token $v \in \mathcal{V}$ in the sequence. In particular, \mathcal{F} could be a class of language models which use deterministic decoding strategies (e.g. greedy decoding) to output the next token (Chorowski and Jaitly, 2016). For such a function class \mathcal{F} , one might be interested in understanding the optimal number of mistakes for next-token prediction when the true sequence of tokens is generated by some unknown $f \in \mathcal{F}$.

Given any learning problem $(\mathcal{X}, \mathcal{F})$, we aim to find necessary and sufficient conditions for the learnability of \mathcal{F} while also quantifying the minimax rates under two notions of expected regret: Markovian and Flow regret (Equations 1 and 2 respectively). To that end, our main contributions are summarized below.

- (i) We provide a quantitative characterization of learnability in the realizable setting, when there exists an evolution function in the class \mathcal{F} that generates the sequence of states. Our characterization is in terms of a new combinatorial complexity measure we call the Evolution complexity. Using this characterization, we show that all rates are possible for the minimax expected mistakes when learning dynamical systems in the realizable setting. This is in contrast to online multiclass classification where only two rates are possible. Finally, we compare realizable learnability of dynamical systems to realizable learnability in PAC and online classification.
- (ii) In the agnostic setting, we lower and upper bound the minimax Markovian regret in terms of the Littlestone dimension. This result shows that the finiteness of the Littlestone dimension characterizes learnability under Markovian regret. As a corollary, we establish a separation between realizable and agnostic learnability under Markovian regret. We show this separation between realizable and agnostic learnability continues to hold when considering Flow regret. However, if the evolution class has uniformly bounded projections, we show that realizable and agnostic learnability under Flow regret are equivalent.

Our characterization of realizable learnability in terms of the Evolution complexity follows from standard techniques in online classification. However, our results showing all possible rates requires a careful construction of a family of classes which have not been studied in learning theory. Likewise, our comparisons of realizable learnability require a careful construction of non-trivial classes, and computing their combinatorial dimensions.

In the agnostic setting, our upper bound on the minimax Markovian regret in terms of the Littlestone dimension results from reducing learning dynamical systems to online multiclass classification. However, both of the lower bounds $\Omega(\sqrt{T})$ and $\Omega(L(\mathcal{H}))$ in Theorem 15 are not standard. The lower bound of $\Omega(L(\mathcal{H}))$ requires constructing a hard stream by traversing down a Littlestone tree *skipping* certain levels. The lower bound of \sqrt{T} requires constructing a hard stream using two different evolution functions f_1, f_2 by: (a) setting x_0 to be a state they differ on (b) generating their trajectories starting from x_0 and finally (c) picking states from each trajectory in an alternating fashion. These arguments are different from the typical lower bound construction for online classification. In addition, the construction in Theorem 18 showing the separation between realizable and agnostic learnability under Flow regret is non-trivial. It involves constructing an $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ so that on a large subset of states in \mathcal{X} , every function $f \in \mathcal{F}$ effectively reveals itself.

1.1. Related Works

There has been a long line of work studying prediction and regret minimization when learning unknown dynamical systems (Hazan et al., 2017, 2018; Ghai et al., 2020; Lee, 2022; Rashidinejad et al., 2020; Kozdoba et al., 2019; Tsiamis and Pappas, 2022; Lale et al., 2020). However, these works focus on prediction for fully/partially-observed linear dynamical systems under various data-generating processes. Moreover, there is also a line of work on regret minimization for linear dynamical systems for online control problems (Abbasi-Yadkori and Szepesvári, 2011; Cohen et al., 2018; Agarwal et al., 2019). For non-linear dynamical systems, there has been some applied work studying data-driven approaches to prediction (Wang et al., 2016; Korda and Mezić, 2018; Ghadami and Epureanu, 2022). Regret minimization for non-linear dynamical systems has mainly been studied in the context of control (Kakade et al., 2020; Muthirayan and Khargonekar, 2021). Another related line of work is online learning for time series forecasting, where autoregressive models are used to predict the next state (Anava et al., 2013, 2015; Liu et al., 2016; Yang et al., 2018).

Another important line of work is that of system identification and parameter estimation (Åström and Eykhoff, 1971; Ljung, 1999). Here, the goal is to recover and estimate the parameters of the underlying evolution function from the observed sequence of states. There is a long history of work studying system identification in both the batch (Campi and Weyer, 2002; Vidyasagar and Karandikar, 2006; Foster et al., 2020; Sattar and Oymak, 2022; Bahmani and Romberg, 2020) and streaming settings (Kowshik et al., 2021a,b; Giannakis et al., 2023; Jain et al., 2021). Several works have also considered the problem of learning the unknown evolution rule of dynamical systems defined over discrete state spaces. For example, Wulff and Hertz (1992) trains a neural network on a sequence of observed states to approximate the unknown evolution rule of a cellular automaton. Grattarola et al. (2021) extends this work to learning the unknown evolution rule of a graph cellular automaton, a generalization of a regular cellular automaton, using graph neural networks. (Qiu et al.; Rosenkrantz et al., 2022) also consider the problem of PAC learning a discrete-time, finite-space dynamical system defined over a (un)directed graph. We also note the related work of Berry and Das (2023, 2025), who study the problem of learning dynamics observed through a continuous embedding and derive learning guarantees based on various structural properties of the underlying system. Finally, our work builds on a rich tradition in learning theory that characterizes learnability through complexity measures and combinatorial dimensions (Vapnik and Chervonenkis, 1971; Littlestone, 1987; Bartlett and Mendelson, 2002; Daniely et al., 2011).

2. Preliminaries

2.1. Discrete-time Dynamical Systems

A discrete-time dynamical system is a tuple $(\mathbb{N}, \mathcal{X}, f)$, where \mathbb{N} is the set of natural numbers denoting the time steps, \mathcal{X} is a non-empty set called the *state space*. In this work, we make no assumption on the cardinality of \mathcal{X} , so it can be unbounded and perhaps even uncountable. The function $f : \mathcal{X} \rightarrow \mathcal{X}$ is a deterministic map that defines the evolution of the dynamical system. That is, the $(t+1)$ -th iterate of the dynamics can be expressed in terms of t -th iterate using the relation $x_{t+1} = f(x_t)$. Define f^t to be the t -fold composition of f . That is, $f^2 = f \circ f$, $f^3 = f \circ f \circ f$, and so forth. Given an initial state $x_0 \in \mathcal{X}$, the sequence $\{f^t(x_0)\}_{t \in \mathbb{N}}$ is called the *flow* of the dynamical system through x_0 . Finally, let $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ denote a class of evolution functions on the state space \mathcal{X} and $\mathcal{F}(x) = \{f(x) \mid f \in \mathcal{F}\} \subseteq \mathcal{X}$ to be the projection of \mathcal{F} onto $x \in \mathcal{X}$.

2.2. Learning-to-Predict in Dynamical Systems

When learning-to-predict in dynamical systems, nature plays a sequential game with the learner over T rounds. At the beginning of the game, nature reveals the initial state $x_0 \in \mathcal{X}$. In each round $t \in [T]$, the learner \mathcal{A} uses the observed sequence of states $x_{<t} := (x_0, \dots, x_{t-1})$ to predict the next state $\mathcal{A}(x_{<t}) \in \mathcal{X}$. Nature then reveals the true state $x_t \in \mathcal{X}$, and the learner suffers the loss $\mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\}$. Given a class of evolution functions $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, the goal of the learner is to make predictions such that its *regret*, defined as a difference between cumulative loss of the learner and the best possible cumulative loss over evolution functions in \mathcal{F} , is small.

Formally, given $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, the expected Markovian regret of an algorithm \mathcal{A} is defined as

$$\text{MR}_{\mathcal{A}}(T, \mathcal{F}) := \sup_{(x_0, x_1, \dots, x_T)} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] - \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\}, \quad (1)$$

where the expectation is taken with respect to the randomness of the learner \mathcal{A} . Given this definition of regret, we define agnostic learnability of an evolution class.

Definition 1 (Agnostic Learnability under Markovian Regret) *An evolution class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ is learnable in the agnostic setting if and only if $\inf_{\mathcal{A}} \text{MR}_{\mathcal{A}}(T, \mathcal{F}) = o(T)$ ².*

Perhaps a more natural definition of expected regret in the agnostic setting is to compare the prediction of the learner to the prediction of the best-fixed *trajectory* generated by functions in our evolution class. To that end, define

$$\text{FR}_{\mathcal{A}}(T, \mathcal{F}) := \sup_{(x_0, x_1, \dots, x_T)} \left(\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] - \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f^t(x_0) \neq x_t\} \right). \quad (2)$$

as the expected *Flow regret*. An analogous definition of agnostic learnability follows.

Definition 2 (Agnostic Learnability under Flow Regret) *An evolution class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ is learnable in the agnostic setting under Flow regret if and only if $\inf_{\mathcal{A}} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) = o(T)$.*

2. $o(T)$ refers to any sublinear function of T .

A sequence of states x_0, x_1, \dots, x_T is said to be *realizable* by \mathcal{F} if there exists an evolution function $f \in \mathcal{F}$ such that $f(x_{t-1}) = x_t$ for all $t \in [T]$. In the realizable setting, the cumulative loss of the best-fixed function is 0, and the goal of the learner is to minimize its expected cumulative mistakes

$$M_{\mathcal{A}}(T, \mathcal{F}) := \sup_{x_0} \sup_{f \in \mathcal{F}} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{t-1}) \neq f(x_{t-1})\} \right].$$

Analogously, we define the realizable learnability of \mathcal{F} .

Definition 3 (Realizable Learnability) *An evolution class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ is learnable in the realizable setting if and only if $\inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) = o(T)$.*

2.3. Complexity Measures

In sequential learning tasks, complexity measures are often defined in terms of *trees*, a basic unit that captures temporal dependence. In this paper, we use complete binary trees to define a new combinatorial object called a trajectory tree. In the remainder of this section and Section 2.4, we use trajectory trees to define complexity measures and combinatorial dimensions for evolution classes.

Definition 4 (Trajectory tree) *A trajectory tree of depth d is a complete binary tree of depth d where internal nodes are labeled by states in \mathcal{X} .*

Given a trajectory tree \mathcal{T} of depth d , a root-to-leaf path down \mathcal{T} is defined by a string $\sigma \in \{-1, 1\}^d$ indicating whether to go left ($\sigma_t = -1$) or to go right ($\sigma_t = +1$) at each depth $t \in [d]$. A path $\sigma \in \{-1, 1\}^d$ down \mathcal{T} gives a trajectory $\{x_t\}_{t=0}^d$, where x_0 denotes the instance labeling the root node and x_t is the instance labeling the edge following the prefix $(\sigma_1, \dots, \sigma_t)$ down the tree. A path $\sigma \in \{-1, 1\}^d$ down \mathcal{T} is *shattered* by \mathcal{F} if there exists a $f \in \mathcal{F}$ such that $f(x_{t-1}) = x_t$ for all $t \in [d]$, where $\{x_t\}_{t=0}^d$ is the corresponding trajectory obtained by traversing \mathcal{T} according to σ . If every path down \mathcal{T} is shattered by \mathcal{F} , we say that \mathcal{T} is shattered by \mathcal{F} .

To make this more rigorous, we define a trajectory tree \mathcal{T} of depth d as a sequence $(\mathcal{T}_0, \mathcal{T}_1, \dots, \mathcal{T}_d)$ of node-labeling functions $\mathcal{T}_t : \{-1, 1\}^t \rightarrow \mathcal{X}$, which provide the labels for each internal node. Then, $\mathcal{T}_t(\sigma_1, \dots, \sigma_t)$ gives the label of the node by following the prefix $(\sigma_1, \dots, \sigma_t)$ and \mathcal{T}_0 denotes the instance labeling the root node. For brevity, we define $\sigma_{\leq t} = (\sigma_1, \dots, \sigma_t)$ and write $\mathcal{T}_t(\sigma_1, \dots, \sigma_t) = \mathcal{T}_t(\sigma_{\leq t})$. Analogously, we let $\sigma_{< t} = (\sigma_1, \dots, \sigma_{t-1})$. Using this notation, a trajectory tree \mathcal{T} of depth d is shattered by the evolution function class \mathcal{F} if $\forall \sigma \in \{-1, 1\}^d$, there exists a $f_{\sigma} \in \mathcal{F}$ such that $f_{\sigma}(\mathcal{T}_{t-1}(\sigma_{< t})) = \mathcal{T}_t(\sigma_{\leq t})$ for all $t \in [d]$. Moreover, we use this notation to define the *Branching factor* of a trajectory tree.

Definition 5 (Branching factor) *The Branching factor of a trajectory tree \mathcal{T} of depth d is*

$$B(\mathcal{T}) := \min_{\sigma \in \{-1, 1\}^d} \sum_{t=1}^d \mathbb{1}\{\mathcal{T}_t((\sigma_{< t}, -1)) \neq \mathcal{T}_t((\sigma_{< t}, +1))\}.$$

The branching factor of a path $\sigma \in \{-1, 1\}^d$ captures the distinctness of states labeling the two children of internal nodes in this path. In particular, it counts the number of nodes in the path whose two children are labeled by distinct states. The branching factor of a trajectory tree is just the smallest branching factor across all paths. Using the notion of shattering and Definition 5, we define a new complexity measure, termed the Evolution complexity, of a function class \mathcal{F} .

Definition 6 (Evolution complexity) Let $\mathcal{S}(\mathcal{F}, d)$ be the set of all trajectory trees of depth $d \in \mathbb{N}$ shattered by \mathcal{F} . Then, the Evolution complexity of \mathcal{F} at depth d is defined as $C_d(\mathcal{F}) := \sup_{\mathcal{T} \in \mathcal{S}(\mathcal{F}, d)} B(\mathcal{T})$.

In Section 3, we show that the Evolution complexity exactly (up to a factor of 2) captures the minimax expected mistakes in the realizable setting. We provide some examples of classes \mathcal{F} and their evolution complexities in Theorem 14. We note that there is an existing notion of complexity for dynamical systems, termed topological entropy, that quantifies the complexity of a particular evolution function $f \in \mathcal{F}$ (Adler et al., 1965). However, topological entropy does not characterize learnability as $\mathcal{F} = \{f\}$ is trivially learnable when f has infinite topological entropy.

2.4. Combinatorial dimensions

In addition to complexity measures, combinatorial dimensions play an important role in providing crisp quantitative characterizations of learnability. For example, the Daniely Shalev-Shwartz dimension (DSdim), originally proposed by Daniely and Shalev-Shwartz (2014) and formally defined below, was recently shown by Brukhim et al. (2022) to provide a tight quantitative characterization of multiclass PAC learnability. In Section 3.2, we use the DSdim to relate the realizable learnability of dynamical systems to multiclass PAC learnability of \mathcal{F} .

Definition 7 (DS dimension (Daniely and Shalev-Shwartz, 2014)) We say that $A \subseteq \mathcal{X}$ is DS-shattered by \mathcal{F} if there exists a finite $\mathcal{H} \subset \mathcal{F}$ such that for every $x \in A$ and $h \in \mathcal{H}$, there exists a $g \in \mathcal{H}$ such that $g(x) \neq h(x)$ and $g(z) = h(z)$ for all $z \in A \setminus \{x\}$. The DS dimension of \mathcal{F} , denoted $DS(\mathcal{F})$, is the largest $d \in \mathbb{N}$ such that there exists a shattered set $A \subset \mathcal{X}$ with cardinality d . If there are arbitrarily large sets $A \subseteq \mathcal{X}$ that are shattered by \mathcal{F} , then we say that $DS(\mathcal{F}) = \infty$.

Analogously, for online multiclass classification, the Littlestone dimension (Ldim), originally proposed by Littlestone (1987) for binary classification and later extended to multiclass classification by Daniely et al. (2011), provides a tight quantitative characterization of learnability (Hanneke et al., 2023).

Definition 8 (Littlestone dimension (Littlestone, 1987; Daniely et al., 2011)) Let \mathcal{T} be a complete binary tree of depth d whose internal nodes are labeled by a sequence $(\mathcal{T}_0, \dots, \mathcal{T}_{d-1})$ of node-labeling functions $\mathcal{T}_{t-1} : \{-1, 1\}^{t-1} \rightarrow \mathcal{X}$. The tree \mathcal{T} is shattered by $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ if there exists a sequence (Y_1, \dots, Y_d) of edge-labeling functions $Y_t : \{-1, 1\}^t \rightarrow \mathcal{X}$ such that for every path $\sigma = (\sigma_1, \dots, \sigma_d) \in \{-1, 1\}^d$, there exists a function $f_{\sigma} \in \mathcal{F}$ such that for all $t \in [d]$, $f_{\sigma}(\mathcal{T}_{t-1}(\sigma_{\leq t})) = Y_t(\sigma_{\leq t})$ and $Y_t((\sigma_{\leq t}, -1)) \neq Y_t((\sigma_{\leq t}, +1))$. The Littlestone dimension of \mathcal{F} , denoted $L(\mathcal{F})$, is the maximal depth of a tree \mathcal{T} that is shattered by \mathcal{F} . If there exists shattered trees of arbitrarily large depth, we say $L(\mathcal{F}) = \infty$.

3. Warmup: Realizable Learnability

In this section, we provide qualitative and quantitative characterizations of realizable learnability in terms of the Evolution complexity. Our main result in this section is Theorem 9, which provides bounds on the minimax expected number of mistakes.

Theorem 9 (Minimax Expected Mistakes) *For any $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, we have $\frac{1}{2} C_T(\mathcal{F}) \leq \inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) \leq C_T(\mathcal{F})$. Moreover, the upper bound is achieved constructively by a deterministic learner.*

The factor of $\frac{1}{2}$ in the lower bound is due to randomized learners, and the lower bound of $C_T(\mathcal{F})$ can be obtained if the learner is restricted only to deterministic learning rules.

We now describe our high-level proof strategy and defer the full proof of Theorem 9 to Appendix A. Our lower bound involves picking the worst-case shattered tree with the largest branching factor and traversing down this tree uniformly at random to generate the sequence of states. For such a sequence, the learner can do no better than random guessing at nodes where the branching occurred, yielding the lower bound $C_T(\mathcal{F})/2$. Next, for our upper bound, we first define a localized complexity measure $C_T(\mathcal{F}, x_0)$, where we only consider shattered trees rooted at the revealed initial state x_0 . Our minimax learner is then a version space algorithm that predicts the state that will result in the largest reduction in the complexity measure if a mistake occurs. This learner is a generalization of the celebrated Standard Optimal Algorithm due to Littlestone (1987).

3.1. Minimax Rates in the Realizable Setting

While Theorem 9 provides a quantitative and qualitative characterization of realizable learnability, it does not shed light on how $\inf_{\mathcal{A}} M_{\mathcal{A}}(\mathcal{F}, T)$ may depend on the time horizon T . In online classification with the 0-1 loss, the seminal work by Littlestone (1987) and Daniely et al. (2011) show that only two rates are possible: $\Theta(T)$ and $\Theta(1)$. That is, if a hypothesis class is online learnable in the realizable setting, then it is learnable with a constant mistake bound (i.e. the Littlestone dimension). Perhaps surprisingly, this is not the case for learning dynamical systems in a strong sense: *every* rate is possible.

Theorem 10 *For every $S \subset \mathbb{N} \cup \{0\}$, there exists $\mathcal{F}_S \subseteq \mathbb{Z}^{\mathbb{Z}}$ such that $C_T(\mathcal{F}_S) = \sup_{n \in \mathbb{N} \cup \{0\}} |S \cap \{n, n+1, \dots, n+T-1\}|$.*

Theorem 10, proved in Appendix B, along with Theorem 9 implies that any minimax rate in the realizable setting is possible. As an example, suppose we would like to achieve the rate $\Theta(T^\alpha)$ for some $\alpha < 1$. Then, picking $S = \{\lfloor t^{\frac{1}{\alpha}} \rfloor : t \in \mathbb{N} \cup \{0\}\}$ suffices since $\sup_{x \in \mathbb{N} \cup \{0\}} |S \cap \{n, n+1, \dots, n+T-1\}| = |\{\lfloor t^{\frac{1}{\alpha}} \rfloor : t \in \mathbb{N} \cup \{0\}\} \cap \{0, 1, \dots, T-1\}| = \Theta(T^\alpha)$. Likewise, one can get logarithmic rates by picking $S = \{2^t : t \in \mathbb{N} \cup \{0\}\}$ and constant rates by picking $S \subset \mathbb{N} \cup \{0\}$ such that $|S| < \infty$. In light of Theorem 10 and the fact that only constant mistake bounds are possible for online multiclass classification, it is natural to ask *when* one can achieve constant mistake bounds for learning dynamical systems. To answer this question, we introduce a new combinatorial dimension termed the Branching dimension.

Definition 11 (Branching dimension) *The Branching dimension, denoted $Bd(\mathcal{F})$, is the smallest natural number $d \in \mathbb{N}$ such that for every shattered trajectory tree \mathcal{T} , we have $B(\mathcal{T}) \leq d$. If for every $d \in \mathbb{N}$, there exists a shattered trajectory tree \mathcal{T} with $B(\mathcal{T}) > d$, we say $Bd(\mathcal{F}) = \infty$.*

Theorem 12, proved via non-constructive arguments in Appendix C, shows that $\inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) = \Theta(1)$ if and and only if $Bd(\mathcal{F}) < \infty$.

Theorem 12 (Constant Minimax Expected Mistakes) *For any $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, we have (i) $\inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) \leq Bd(\mathcal{F})$ and (ii) if $Bd(\mathcal{F}) = \infty$, then $\inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) = \omega(1)$ ³.*

3. Recall that $f(n) = \omega(1)$ if for all $k > 0$, there exists n_k , such that for all $n > n_k$ we have $f(n) > k$.

3.2. Relations to PAC and Online Multiclass Classification

By studying abstract state spaces \mathcal{X} and evolutions function classes $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, we can also compare realizable learnability of dynamical systems to existing notions of realizable learnability in the well-known PAC and online classification settings. Our main result relates the evolution complexity to the DS and Littlestone dimensions, which characterize PAC and online classification respectively.

Theorem 13 (Relations to the DS and Littlestone dimension) *The following statements are true.*

- (i) *There exists $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ such that $\text{DS}(\mathcal{F}) = \infty$ but $\text{C}_T(\mathcal{F}) = \Theta(\log(T))$.*
- (ii) *There exists $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ such that $\text{DS}(\mathcal{F}) = 1$ but $\text{C}_T(\mathcal{F}) = T$.*
- (iii) *For any $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$, we have that $\text{C}_T(\mathcal{F}) \leq \text{L}(\mathcal{F})$.*
- (iv) *There exists $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ such that $\text{L}(\mathcal{F}) = \infty$ but $\text{C}_T(\mathcal{F}) = 1$.*

The proof of Theorem 13 is in Appendix D. Parts (i) and (ii) show that the finiteness of the DSdim is neither necessary nor sufficient for learning dynamical systems in the realizable setting. On the other hand, parts (iii) and (iv) show that finite Ldim is sufficient but not necessary for learning dynamical systems in the realizable setting. Overall, learning dynamical systems is always easier than online multiclass classification, but can be both easier and harder than multiclass PAC classification. The proof of (i), (ii), and (iv) are combinatorial in nature, while the proof of (iii) involves reducing learning-to-predict in dynamical systems to online multiclass classification.

3.3. Examples

In this section, we establish the minimax rates for discrete linear systems and linear Boolean networks. The proof of Theorem 14 is in Appendix E.

Theorem 14 (Linear Systems)

- (i) *Let $\mathcal{X} = \mathbb{Z}^n$ and $r < n$. For $\mathcal{F} = \{x \mapsto Wx : W \in \mathbb{Z}^{n \times n}, \text{rank}(W) \leq r\}$ and $T > r$, we have $\text{C}_T(\mathcal{F}) = r + 1$.*
- (ii) *Let $\mathcal{X} = \{0, 1\}^n$ and $T \geq n$. For $\mathcal{F} = \{x \mapsto Wx \pmod{2} : W \in \mathbb{Z}^{n \times n}\}$, we have $\text{C}_T(\mathcal{F}) = n$.*
- (iii) *Let $\mathcal{X} = \{0, 1\}^n$ and $T \geq n$. For $\mathcal{F} = \{x \mapsto \mathbb{1}\{Wx > 0\} : W \in \{0, 1\}^{n \times n}\}$, we have $n \leq \text{C}_T(\mathcal{F}) \leq n^2$.*

Thresholded Boolean networks have been used to model genetic regulatory dynamics (Mendoza and Alvarez-Buylla, 1998) and social networks (Kempe et al., 2003). Modulo Boolean networks have been studied by Chandrasekhar et al. (2023) in the context of stability.

4. Agnostic Learnability

4.1. Markovian Regret

In this section, we go beyond the realizable setting and consider the case where nature may reveal a trajectory that is not consistent with any evolution function in the class. Our main result in this section establishes bounds on the minimax expected Markovian regret.

Theorem 15 (Minimax Expected Markovian Regret) *For any $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$,*

$$\max \left\{ \frac{L(\mathcal{F})}{18}, \frac{\sqrt{T}}{16\sqrt{3}} \right\} \leq \inf_{\mathcal{A}} \text{MR}_{\mathcal{A}}(T, \mathcal{F}) \leq L(\mathcal{F}) + \sqrt{T L(\mathcal{F}) \log T}.$$

Theorem 15 shows that the finiteness of the Littlestone dimension of \mathcal{F} is both necessary and sufficient for agnostic learnability. This is in contrast to Theorem 13, which shows that the finiteness of the Littlestone dimension of \mathcal{F} is sufficient but *not necessary* for realizable learnability. Thus, Theorem 15 and 13 imply that realizable and agnostic learnability are not equivalent.

Corollary 16 (Realizable Learnability \neq Agnostic Learnability under Markovian Regret) *There exists a class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ such that \mathcal{F} is learnable in the realizable setting but not in the agnostic setting under Markovian regret.*

One such class exhibiting the separation is the thresholds $\mathcal{F} = \{x \mapsto \mathbb{1}\{x \geq a\} : a \in (0, 1)\}$ used in the proof of part (iv) in Theorem 13. Beyond the qualitative separation of realizable and agnostic learnability under Markovian regret, we also observe a quantitative separation in terms of possible minimax rates. Recall that Theorem 10 shows that every rate is possible in the realizable setting. However, Theorem 15 shows that only two types of rates are possible in the agnostic setting: $\Theta(T)$ whenever $L(\mathcal{F}) = \infty$ and $\tilde{\Theta}(\sqrt{T})$ whenever $L(\mathcal{F}) < \infty$. More precisely, when $\sqrt{T} \geq L(\mathcal{F})$, we have a lower bound of $\Omega(\sqrt{T})$ and an upper bound of $O(\sqrt{T L(\mathcal{F}) \log T})$. This raises the natural question of what the right minimax rate is. In Appendix G, we provide an evolution class and establish a lower bound of $\Omega(\sqrt{T L(\mathcal{F})})$, showing that the upper bound is tight up to $\sqrt{\log T}$.

Our proof of the upper bound in Theorem 15 reduces learning dynamical systems to online multiclass classification and uses a result due to Hanneke et al. (2023). To prove the lower bound $\frac{L(\mathcal{F})}{18}$, we construct a hard stream by carefully sampling a random path down a Littlestone tree of depth $L(\mathcal{F})$. To prove the lower bound of $\frac{\sqrt{T}}{16\sqrt{3}}$, we construct a hard randomized stream using just two different evolution functions in \mathcal{F} . The full proof is in Appendix F.

4.2. Flow Regret

The necessity of the Littlestone dimension for agnostic learnability under Markovian regret is quite restrictive. For example, a simple (but unnatural) class like one-dimensional thresholds $\mathcal{F} = \{x \mapsto \mathbb{1}\{x \leq a\} : a \in (0, 1)\}$ has $L(\mathcal{F}) = \infty$ but $C_T(\mathcal{F}) = 1$. The key idea in the lower bound of Theorem 17 is that the adversary can simulate the online multiclass classification game, where finiteness of the Littlestone dimension is necessary, by “giving up” every other round. This is possible because of the definition of Markovian regret. In particular, the evaluation of the “best-fixed evolution function in hindsight” under Markovian regret is only penalized on one-step prediction error but not on long-term consistency of the generated dynamics starting from the initial state x_0 .

This motivates the following natural question. Which evolution classes $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ are agnostic learnable under Flow regret? Is the finiteness of Ldim still necessary? Theorem 17, whose proof can be found in Appendix H, provides partial answers to these question by bounding the minimax expected Flow regret for classes where $\sup_{x \in \mathcal{X}} |\mathcal{F}(x)|$ is uniformly bounded.

Theorem 17 (Minimax Expected Flow Regret) *For any ordered set \mathcal{X} and $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$,*

$$\frac{C_T(\mathcal{F})}{2} \leq \inf_{\mathcal{A}} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) \leq C_T(\mathcal{F}) + \sqrt{C_T(\mathcal{F}) T \ln\left(\frac{T K_{\mathcal{F}}}{C_T(\mathcal{F})}\right)}.$$

where $K_{\mathcal{F}} = \sup_{x \in \mathcal{X}} |\mathcal{F}(x)|$. Moreover, both the lower- and upper bound can be tight.

Note that the upper bound becomes vacuous when $\sup_{x \in \mathcal{X}} |\mathcal{F}(x)| = \infty$, and finding a general characterization of Flow regret learnability remains an open question. Unlike agnostic learnability under Markovian regret, the finiteness of the Ldim is not necessary for agnostic learnability under Flow regret. Indeed, while the class of one-dimensional thresholds is not agnostic learnable under Markovian regret, it is agnostic learnable under Flow regret. Although Markovian learnability is not necessary for Flow regret learnability, when $K_{\mathcal{F}} < \infty$, learnability under Markovian regret is sufficient learnability under flow regret. To see this, recall that Theorem 15 states that learnability under Markovian regret implies $L(\mathcal{F}) < \infty$. Then, since part (iii) of Theorem 13 states $C_T(\mathcal{F}) \leq L(\mathcal{F})$, we can use Theorem 17 to infer that \mathcal{F} is also learnable under flow regret with regret $\leq L(\mathcal{F}) + \sqrt{L(\mathcal{F}) T \ln(T K_{\mathcal{F}})}$.

Theorem 17 shows that realizable and agnostic learnability under Flow regret are equivalent as long as $\sup_{x \in \mathcal{X}} |\mathcal{F}(x)| < \infty$. But this equivalence breaks down when the projection sizes are unbounded, as shown by Theorem 18.

Theorem 18 (Realizable learnability $\not\equiv$ Agnostic Learnability under Flow Regret) *There exists an ordered set \mathcal{X} and $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ such that (i) $\inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) \leq 3$ but (ii) $\inf_{\mathcal{A}} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) \geq \frac{T}{6}$.*

To prove Theorem 18 (see Appendix I), we construct a class $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ such that on large subset of states in $\mathcal{X}' \subset \mathcal{X}$, every function $f \in \mathcal{F}$ effectively reveals its identity.

5. Discussion and Future directions

In this work, we studied the problem of learning-to-predict in discrete-time dynamical systems under the 0-1 loss. A natural extension is to consider continuous state spaces with real-valued losses. For example, one can take \mathcal{X} to be a bounded subset of a Hilbert space and consider the squared norm as the loss function. Another natural extension is to consider learnability under partial observability, where the learner only observes some transformation $\phi(x_t)$ instead of the true state x_t . Such feedback model is standard in prediction for linear dynamical systems (Hazan et al., 2018). It is also natural to study the learnability of function classes where the output of the evolution rules $f : \mathcal{X}^p \rightarrow \mathcal{X}$, depend on the previous $p > 1$ states (e.g. the p -th order VAR model). Lastly, the learning algorithms in this work are *improper*: they use evolution functions that may not lie in \mathcal{F} to make predictions. This might be undesirable as improper learning algorithms may be incompatible with downstream system identification and control tasks. To this end, characterizing *proper* learnability of dynamical systems is an important future direction.

Acknowledgments

AT and US acknowledge the support of NSF via grant DMS-2413089. US also acknowledges the support of Rackham International Student Fellowship. VR acknowledges the support of the NSF GRFP. We thank Chinmaya Kaushik for pointing out a technical result, which helped us to prove part (i) Theorem 14.

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. JMLR Workshop and Conference Proceedings, 2011.

Roy L Adler, Alan G Konheim, and M Harry McAndrew. Topological entropy. *Transactions of the American Mathematical Society*, 114(2):309–319, 1965.

Naman Agarwal, Brian Bullins, Elad Hazan, Sham Kakade, and Karan Singh. Online control with adversarial disturbances. In *International Conference on Machine Learning*, pages 111–119. PMLR, 2019.

Oren Anava, Elad Hazan, Shie Mannor, and Ohad Shamir. Online learning for time series prediction. In *Conference on learning theory*, pages 172–184. PMLR, 2013.

Oren Anava, Elad Hazan, and Assaf Zeevi. Online time series prediction with missing data. In *International conference on machine learning*, pages 2191–2199. PMLR, 2015.

Karl Johan Åström and Peter Eykhoff. System identification—a survey. *Automatica*, 7(2):123–162, 1971.

Sohail Bahmani and Justin Romberg. Convex programming for estimation in nonlinear recurrent models. *The Journal of Machine Learning Research*, 21(1):9563–9582, 2020.

Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

David Belanger and Sham Kakade. A linear dynamical system model for text. In *International Conference on Machine Learning*, pages 833–842. PMLR, 2015.

Tyrus Berry and Sudhhasattwa Das. Learning theory for dynamical systems. *SIAM Journal on Applied Dynamical Systems*, 22(3):2082–2122, 2023.

Tyrus Berry and Sudhhasattwa Das. Limits of learning dynamical systems. *SIAM Review*, 67(1):107–137, 2025.

Nataly Brukhim, Daniel Carmon, Irit Dinur, Shay Moran, and Amir Yehudayoff. A characterization of multiclass learnability, 2022. URL <https://arxiv.org/abs/2203.01550>.

Marco C Campi and Erik Weyer. Finite sample properties of system identification methods. *IEEE Transactions on Automatic Control*, 47(8):1329–1334, 2002.

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

Karthik Chandrasekhar, Claus Kadelka, Reinhard Laubenbacher, and David Murrugarra. Stability of linear boolean networks. *Physica D: Nonlinear Phenomena*, 451:133775, 2023.

Jan Chorowski and Navdeep Jaitly. Towards better decoding and language model integration in sequence to sequence models. *arXiv preprint arXiv:1612.02695*, 2016.

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

Amit Daniely and Shai Shalev-Shwartz. Optimal learners for multiclass problems. In *Conference on Learning Theory*, pages 287–316. PMLR, 2014.

Amit Daniely, Sivan Sabato, Shai Ben-David, and Shai Shalev-Shwartz. Multiclass learnability and the erm principle. In Sham M. Kakade and Ulrike von Luxburg, editors, *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pages 207–232, Budapest, Hungary, 09–11 Jun 2011. PMLR.

Gianfranco Doretto, Alessandro Chiuso, Ying Nian Wu, and Stefano Soatto. Dynamic textures. *International journal of computer vision*, 51:91–109, 2003.

Jeffrey L Elman. Language as a dynamical system. *Mind as motion: Explorations in the dynamics of cognition*, pages 195–223, 1995.

Dylan Foster, Tuhin Sarkar, and Alexander Rakhlin. Learning nonlinear dynamical systems from a single trajectory. In *Learning for Dynamics and Control*, pages 851–861. PMLR, 2020.

Amin Ghadami and Bogdan I Epureanu. Data-driven prediction in dynamical systems: recent developments. *Philosophical Transactions of the Royal Society A*, 380(2229):20210213, 2022.

Udaya Ghai, Holden Lee, Karan Singh, Cyril Zhang, and Yi Zhang. No-regret prediction in marginally stable systems. In *Conference on Learning Theory*, pages 1714–1757. PMLR, 2020.

Dimitrios Giannakis, Amelia Henriksen, Joel A Tropp, and Rachel Ward. Learning to forecast dynamical systems from streaming data. *SIAM Journal on Applied Dynamical Systems*, 22(2):527–558, 2023.

Daniele Grattarola, Lorenzo Livi, and Cesare Alippi. Learning graph cellular automata. *Advances in Neural Information Processing Systems*, 34:20983–20994, 2021.

Steve Hanneke, Shay Moran, Vinod Raman, Unique Subedi, and Ambuj Tewari. Multiclass online learning and uniform convergence. *Proceedings of the 36th Annual Conference on Learning Theory (COLT)*, 2023.

Elad Hazan, Karan Singh, and Cyril Zhang. Learning linear dynamical systems via spectral filtering. *Advances in Neural Information Processing Systems*, 30, 2017.

Elad Hazan, Holden Lee, Karan Singh, Cyril Zhang, and Yi Zhang. Spectral filtering for general linear dynamical systems. *Advances in Neural Information Processing Systems*, 31, 2018.

Alfons G Hoekstra, Jiri Kroc, and Peter MA Sloot. *Simulating complex systems by cellular automata*. Springer, 2010.

Prateek Jain, Suhas S Kowshik, Dheeraj Nagaraj, and Praneeth Netrapalli. Streaming linear system identification with reverse experience replay. *arXiv preprint arXiv:2103.05896*, 2021.

Sham Kakade, Akshay Krishnamurthy, Kendall Lowrey, Motoya Ohnishi, and Wen Sun. Information theoretic regret bounds for online nonlinear control. *Advances in Neural Information Processing Systems*, 33:15312–15325, 2020.

Stuart Kauffman. Homeostasis and differentiation in random genetic control networks. *Nature*, 224 (5215):177–178, 1969.

David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

Milan Korda and Igor Mezić. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*, 93:149–160, 2018.

Suhas Kowshik, Dheeraj Nagaraj, Prateek Jain, and Praneeth Netrapalli. Near-optimal offline and streaming algorithms for learning non-linear dynamical systems. *Advances in Neural Information Processing Systems*, 34:8518–8531, 2021a.

Suhas Kowshik, Dheeraj Nagaraj, Prateek Jain, and Praneeth Netrapalli. Streaming linear system identification with reverse experience replay. *Advances in Neural Information Processing Systems*, 34:30140–30152, 2021b.

Mark Kozdoba, Jakub Marecek, Tigran Tchrakian, and Shie Mannor. On-line learning of linear dynamical systems: Exponential forgetting in kalman filters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4098–4105, 2019.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Logarithmic regret bound in partially observable linear dynamical systems. *Advances in Neural Information Processing Systems*, 33:20876–20888, 2020.

Holden Lee. Improved rates for prediction and identification of partially observed linear dynamical systems. In *International Conference on Algorithmic Learning Theory*, pages 668–698. PMLR, 2022.

Yingying Li, Xin Chen, and Na Li. Online optimal control with linear dynamics and predictions: Algorithms and regret analysis. *Advances in Neural Information Processing Systems*, 32, 2019.

Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2:285–318, 1987.

Chenghao Liu, Steven CH Hoi, Peilin Zhao, and Jianling Sun. Online arima algorithms for time series prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Lennart Ljung. System identification: theory for the user. *PTR Prentice Hall, Upper Saddle River, NJ*, 28:540, 1999.

Luis Mendoza and Elena R Alvarez-Buylla. Dynamics of the genetic regulatory network for arabiopsis thaliana flower morphogenesis. *Journal of theoretical biology*, 193(2):307–319, 1998.

Deepan Muthirayan and Pramod P Khargonekar. Online learning robust control of nonlinear dynamical systems. *arXiv preprint arXiv:2106.04092*, 2021.

Zirou Qiu, Abhijin Adiga, Madhav V Marathe, SS Ravi, Daniel J Rosenkrantz, Richard E Stearns, and Anil Vullikanti. Learning the topology and behavior of discrete dynamical systems.

Paria Rashidinejad, Jiantao Jiao, and Stuart Russell. Slip: Learning to predict in unknown dynamical systems with long-term memory. *Advances in Neural Information Processing Systems*, 33:5716–5728, 2020.

Daniel J Rosenkrantz, Abhijin Adiga, Madhav Marathe, Zirou Qiu, SS Ravi, Richard Stearns, and Anil Vullikanti. Efficiently learning the topology and behavior of a networked dynamical system via active queries. In *International Conference on Machine Learning*, pages 18796–18808. PMLR, 2022.

Yahya Sattar and Samet Oymak. Non-asymptotic and accurate learning of nonlinear dynamical systems. *The Journal of Machine Learning Research*, 23(1):6248–6296, 2022.

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

Ilya Shmulevich, Edward R Dougherty, and Wei Zhang. From boolean to probabilistic boolean networks as models of genetic regulatory networks. *Proceedings of the IEEE*, 90(11):1778–1792, 2002.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

Anastasios Tsiamis and George J Pappas. Online learning of the kalman filter with logarithmic regret. *IEEE Transactions on Automatic Control*, 68(5):2774–2789, 2022.

Vladimir Naumovich Vapnik and Aleksei Yakovlevich Chervonenkis. On uniform convergence of the frequencies of events to their probabilities. *Teoriya Veroyatnostei i ee Primeneniya*, 16(2):264–279, 1971.

Mathukumalli Vidyasagar and Rajeva L Karandikar. A learning theory approach to system identification and stochastic adaptive control. *Probabilistic and randomized methods for design under uncertainty*, pages 265–302, 2006.

Wen-Xu Wang, Ying-Cheng Lai, and Celso Grebogi. Data-based identification and prediction of nonlinear and complex dynamical systems. *Physics Reports*, 644:1–76, 2016.

Stephen Wolfram. Theory and applications of cellular automata. *World Scientific*, 1986.

N Wulff and J A Hertz. Learning cellular automaton dynamics with neural networks. *Advances in Neural Information Processing Systems*, 5, 1992.

Haimin Yang, Zhisong Pan, Qing Tao, and Junyang Qiu. Online learning for vector autoregressive moving-average time series prediction. *Neurocomputing*, 315:9–17, 2018.

Appendix A. Proofs for Realizable learnability

Proof (of lower bound of Theorem 9). Let $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ be any evolution class and $T \in \mathbb{N}$ be the time horizon. Let \mathcal{A} be any randomized learner. Our goal will be to construct a hard realizable trajectory $\{x_t\}_{t=0}^T$ such that \mathcal{A} 's expected number of mistakes is at least $\frac{C_T(\mathcal{F})}{2}$. Without loss of generality, suppose $d := C_T(\mathcal{F}) > 0$ as otherwise the lower bound holds trivially. Then, by definition of the Evolution complexity, there exists a trajectory tree \mathcal{T} of depth T shattered by \mathcal{F} with branching factor at least d . This means that for every path $\sigma \in \{-1, 1\}^T$ down \mathcal{T} , we have $\sum_{t=1}^T \mathbb{1}\{\mathcal{T}_t((\sigma_{<t}, -1)) \neq \mathcal{T}_t((\sigma_{<t}, +1))\} \geq d$.

Let $\sigma \sim \{-1, 1\}^T$ denote a random path down \mathcal{T} and consider the trajectory $\mathcal{T}_0 \cup \{\mathcal{T}_t(\sigma_{\leq t})\}_{t=1}^T$. Define $\mathcal{T}_{<t}(\sigma_{<t}) := (\mathcal{T}_0, \mathcal{T}_1(\sigma_1), \dots, \mathcal{T}_{t-1}(\sigma_{<t}))$. Then, observe that

$$\begin{aligned} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(\mathcal{T}_{<t}(\sigma_{<t})) \neq \mathcal{T}_t(\sigma_{\leq t})\} \right] &= \sum_{t=1}^T \mathbb{E} \left[\mathbb{E} \left[\mathbb{1}\{\mathcal{A}(\mathcal{T}_{<t}(\sigma_{<t})) \neq \mathcal{T}_t((\sigma_{<t}, \sigma_t))\} \mid \sigma_{<t} \right] \right] \\ &\geq \frac{1}{2} \sum_{t=1}^T \mathbb{E} \left[\mathbb{E} \left[\mathbb{1}\{\mathcal{T}_t((\sigma_{<t}, -1)) \neq \mathcal{T}_t((\sigma_{<t}, +1))\} \mid \sigma_{<t} \right] \right] \\ &= \frac{1}{2} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{T}_t((\sigma_{<t}, -1)) \neq \mathcal{T}_t((\sigma_{<t}, +1))\} \right] \geq \frac{d}{2} \end{aligned}$$

where the first inequality follows from the fact that $\sigma_t \sim \{-1, 1\}$. ■

Proof (of upper bound of Theorem 9). Let $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ be any evolution function class and let $T \in \mathbb{N}$ be the time horizon. Our goal will be to construct a deterministic learner \mathcal{A} such that for any realizable trajectory $\{x_t\}_{t=0}^T$, \mathcal{A} makes at most $C_T(\mathcal{F})$ mistakes. To that end, it will be useful to define an instance-dependent version of the evolution complexity.

Given a state $x \in \mathcal{X}$, we say \mathcal{T} is a trajectory tree rooted at x if the root node is labeled by x . For an initial state $x_0 \in \mathcal{X}$ and evolution class $V \subseteq \mathcal{F}$, let

$$C_T(V, x_0) := \sup \{B(\mathcal{T}) \mid \mathcal{T} \in \mathcal{S}(V, T) \text{ and } \mathcal{T} \text{ rooted at } x_0\}$$

denote an instance-dependent Evolution complexity of V , where $\mathcal{S}(V, T)$ is set of all trajectory trees of depth T shattered by V . Note that $C_T(V) = \sup_{x_0 \in \mathcal{X}} C_T(V, x_0)$. \mathcal{A} is a version-space algorithm that uses the instance-dependent Evolution complexity to make its prediction such that whenever \mathcal{A} errs, the instance-dependent Evolution complexity decreases. In this way, $C_T(V, x_0)$ acts as a potential function. Algorithm 1 formalizes this idea.

Algorithm 1 Deterministic Realizable Algorithm

Input: Initial state $x_0 \in \mathcal{X}$

Let $V_1 = \mathcal{F}$

for $t = 1, \dots, T$ **do**

For $x \in \mathcal{X}$, define $V_t^x = \{f \in V_t : f(x_{t-1}) = x\}$.
if $\{f(x_{t-1}) : f \in V_t\} = \{x\}$ **then**
 | Predict $\hat{x}_t = x$.
else
 | Predict $\hat{x}_t \in \arg \max_{x \in \mathcal{X}} C_{T-t}(V_t^x, x)$.
 Receive x_t and update $V_{t+1} \leftarrow V_t^{x_t}$.

end

We now show that Algorithm 1 makes at most $C_T(\mathcal{F})$ mistakes on any realizable trajectory. Let $\{x_t\}_{t=0}^T$ denote the realizable trajectory to be observed. It suffices to show that

$$C_{T-t}(V_{t+1}, x_t) \leq C_{T-t+1}(V_t, x_{t-1}) - \mathbb{1}\{x_t \neq \hat{x}_t\} \quad (3)$$

for all $t \in [T]$. To see why this is true, note that (3) implies

$$\sum_{t=1}^T C_{T-t}(V_{t+1}, x_t) \leq \sum_{t=1}^T C_{T-t+1}(V_t, x_{t-1}) - \sum_{t=1}^T \mathbb{1}\{x_t \neq \hat{x}_t\}.$$

Rearranging, we get that

$$\begin{aligned} \sum_{t=1}^T \mathbb{1}\{x_t \neq \hat{x}_t\} &\leq \sum_{t=1}^T C_{T-t+1}(V_t, x_{t-1}) - \sum_{t=1}^T C_{T-t}(V_{t+1}, x_t) \\ &= \sum_{t=1}^T (C_{T-t+1}(V_t, x_{t-1}) - C_{T-t}(V_{t+1}, x_t)) \\ &= C_T(V_1, x_0) - C_0(V_{T+1}, x_T) \leq C_T(\mathcal{F}). \end{aligned}$$

as needed. To prove (3), we need to consider three cases: (a) $C_{T-t+1}(V_t, x_{t-1}) > 0$ and $\hat{x}_t \neq x_t$, (b) $C_{T-t+1}(V_t, x_{t-1}) > 0$ and $\hat{x}_t = x_t$, and (c) $C_{T-t+1}(V_t, x_{t-1}) = 0$.

Starting with (a), let $t \in [T]$ such that $C_{T-t+1}(V_t, x_{t-1}) > 0$ and $\hat{x}_t \neq x_t$. We need to show that $C_{T-t}(V_{t+1}, x_t) < C_{T-t+1}(V_t, x_{t-1})$. Suppose for the sake of contradiction that $C_{T-t}(V_{t+1}, x_t) \geq C_{T-t+1}(V_t, x_{t-1}) := d$. Then, by the prediction rule, we have that $C_{T-t}(V_t^{\hat{x}_t}, \hat{x}_t) \geq C_{T-t}(V_t^{x_t}, x_t) \geq d$. By definition of the instance-dependent Evolution complexity, we are guaranteed the existence of a trajectory tree \mathcal{T}_{x_t} of depth $T-t$, rooted at x_t , shattered by $V_t^{x_t}$ with branching factor at least d and a tree $\mathcal{T}_{\hat{x}_t}$ of depth $T-t$, rooted at \hat{x}_t , shattered by $V_t^{\hat{x}_t}$ with branching factor at least d . Consider a binary tree \mathcal{T} whose root node v is labeled by x_{t-1} and left and right subtrees are \mathcal{T}_{x_t} and $\mathcal{T}_{\hat{x}_t}$ respectively. Then, observe that \mathcal{T} is a trajectory tree of depth $T-t+1$, rooted at x_{t-1} , shattered by V_t with branching factor at least $d+1$ (because $x_t \neq \hat{x}_t$). This contradicts our assumption that $C_{T-t+1}(V_t, x_{t-1}) = d$. Thus, we must have $C_{T-t}(V_{t+1}, x_t) < C_{T-t+1}(V_t, x_{t-1})$.

Moving to (b), let $t \in [T]$ be such that $\hat{x}_t = x_t$. Then, we need to show that $d := C_{T-t}(V_{t+1}, x_t) \leq C_{T-t+1}(V_t, x_{t-1})$. If $d = 0$, the inequality holds trivially. Thus, assume $d > 0$. Then, by definition, there exists a trajectory tree \mathcal{T} of depth $T-t$, rooted at x_t , shattered by V_{t+1} with branching

factor at least d . Consider a binary tree $\tilde{\mathcal{T}}$ whose root node v is labeled by x_{t-1} and left and right subtrees are both \mathcal{T} . Then since $V_{t+1} = V_t^{x_t} \subseteq V_t$, we have that $\tilde{\mathcal{T}}$ is a trajectory tree of depth $T - t + 1$, rooted at x_{t-1} , shattered by V_t with branching factor at least d . Thus, we must have $C_{T-t+1}(V_t, x_{t-1}) \geq d$.

Finally for (c), let $t \in [T]$ be such that $C_{T-t+1}(V_t, x_{t-1}) = 0$. Using the same logic as (b), $C_{T-t}(V_{t+1}, x_t) = 0$. Thus, to prove that (3) holds, it suffices to show that $x_t = \hat{x}_t$. To do so, we will show that the projection size $|\{f(x_{t-1}) : f \in V_t\}| = 1$. Thus, Algorithm 1 does not make a mistake. Suppose for the sake of contradiction that $|\{f(x_{t-1}) : f \in V_t\}| \geq 2$. Then, there exists two functions $f_1, f_2 \in V_t$ such that $f_1(x_{t-1}) \neq f_2(x_{t-1})$. Consider a binary tree \mathcal{T} of depth $T - t + 1$ where the root node is labeled by x_{t-1} , and its left and right child by $f_1(x_{t-1})$ and $f_2(x_{t-1})$ respectively. Label all remaining internal nodes in the left subtree of the root node such that every path is shattered by f_1 . Likewise, for the right subtree of the root node, label every internal node such that every path is shattered by f_2 . Following this procedure, \mathcal{T} is a trajectory tree of depth $T - t + 1$, rooted at x_{t-1} , shattered by V_t with branching factor at least 1 (because $f_1(x_{t-1}) \neq f_2(x_{t-1})$). Thus, $C_{T-t+1}(V_t, x_{t-1}) \geq 1$, which is a contradiction. \blacksquare

Appendix B. Proof of Theorem 10

Let $S \subset \mathbb{N} \cup \{0\}$ be an arbitrary subset of the extended natural numbers. For every $\sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}}$, define the evolution function

$$f_\sigma(x) = \left(\sigma_{|x|} \mathbb{1}\{|x| \in S\} + \mathbb{1}\{|x| \notin S\} \right) (|x| + 1)$$

and consider the class $\mathcal{F}_S = \left\{ f_\sigma : \sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}} \right\}$. By construction, functions in \mathcal{F}_S only disagree on states in S and their negation. Moreover, given any initial state $x_0 \in \mathbb{Z}$ and a time horizon $T \in \mathbb{N}$, the trajectory of any evolution in \mathcal{F} is $\varepsilon_1(|x_0| + 1), \dots, \varepsilon_T(|x_0| + T)$ for some $\varepsilon \in \{-1, 1\}^T$.

We now show that $C_T(\mathcal{F}) \leq \sup_{x_0 \in \mathbb{Z}} |S \cap \{|x_0|, \dots, |x_0| + T - 1\}|$. Let \mathcal{T} be any trajectory tree of depth T shattered by \mathcal{F} . It suffices to show that $B(\mathcal{T}) \leq |S \cap \{|\mathcal{T}_0|, \dots, |\mathcal{T}_0| + T - 1\}|$. Fix any path $\varepsilon \in \{-1, 1\}^T$ down \mathcal{T} and consider the sequence of states $\mathcal{T}_0, \dots, \mathcal{T}_T(\varepsilon_{\leq T})$. Note that $\mathcal{T}_t((\varepsilon_{<t}, -1)) \neq \mathcal{T}_t((\varepsilon_{<t}, +1))$ only if $|\mathcal{T}_{t-1}(\varepsilon_{<t})| \in S$. Thus,

$$\begin{aligned} \sum_{t=1}^T \mathbb{1}\{\mathcal{T}_t((\varepsilon_{<t}, -1)) \neq \mathcal{T}_t((\varepsilon_{<t}, +1))\} &\leq \sum_{t=1}^T \mathbb{1}\{|\mathcal{T}_{t-1}(\varepsilon_{<t})| \in S\} \\ &= \sum_{t=0}^{T-1} \mathbb{1}\{|\mathcal{T}_t(\varepsilon_{\leq t})| \in S\} \\ &= \mathbb{1}\{|\mathcal{T}_0| \in S\} + \sum_{t=1}^{T-1} \mathbb{1}\{|\mathcal{T}_t(\varepsilon_{\leq t})| \in S\} \end{aligned}$$

Moreover, since the path ε is shattered by \mathcal{F}_S , it must be the case that for every $t \in [T]$, we have that $|\mathcal{T}_t(\varepsilon_{\leq t})| = |\mathcal{T}_0| + t$. Thus,

$$\begin{aligned}
 \sum_{t=1}^T \mathbb{1}\{\mathcal{T}_t((\varepsilon_{<t}, -1)) \neq \mathcal{T}_t((\varepsilon_{<t}, +1))\} &\leq \mathbb{1}\{|\mathcal{T}_0| \in S\} + \sum_{t=1}^{T-1} \mathbb{1}\{|\mathcal{T}_t(\varepsilon_{\leq t})| \in S\} \\
 &= \sum_{t=0}^{T-1} \mathbb{1}\{|\mathcal{T}_0| + t \in S\} \\
 &= |S \cap \{|\mathcal{T}_0|, \dots, |\mathcal{T}_0| + T - 1\}|,
 \end{aligned}$$

Taking the supremum of both sides with respect to \mathcal{T}_0 completes the proof of the upper bound.

To prove the lower bound, fix $x \in \mathbb{Z}$ and consider the following trajectory tree \mathcal{T}^x of depth T . Let $\mathcal{T}_0^x = x$. For every path $\varepsilon \in \{-1, 1\}^T$ and $t \in [T]$, let

$$\mathcal{T}_t^x(\varepsilon_{\leq t}) = \begin{cases} \varepsilon_t(|\mathcal{T}_0^x| + t), & \text{if } |\mathcal{T}_0^x| + t - 1 \in S. \\ |\mathcal{T}_0^x| + t & \text{if } |\mathcal{T}_0^x| + t - 1 \notin S. \end{cases}$$

Note $|\mathcal{T}_t^x(\varepsilon_{\leq t})| = |\mathcal{T}_0^x| + t$ for all $t \in [T]$. Moreover, for every path $\varepsilon \in \{-1, 1\}^T$, there exists a function $\sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}}$ such that $\sigma|_{\mathcal{T}_0^x} = \varepsilon_1$ and $\sigma|_{\mathcal{T}_t^x(\varepsilon_{\leq t})} = \varepsilon_{t+1}$ for all $t \in [T-1]$. Thus, the function $f_\sigma \in \mathcal{F}_S$ shatters the path ε and the tree \mathcal{T}^x is shattered by \mathcal{F}_S . We claim that $B(\mathcal{T}^x) = |S \cap \{|\mathcal{T}_0^x|, \dots, |\mathcal{T}_0^x| + T - 1\}|$. To see this, observe that for every path $\varepsilon \in \{-1, 1\}^T$, we have

$$\begin{aligned}
 \sum_{t=1}^T \mathbb{1}\{\mathcal{T}_t^x((\varepsilon_{<t}, -1)) \neq \mathcal{T}_t^x((\varepsilon_{<t}, +1))\} &= \sum_{t=1}^T \mathbb{1}\{|\mathcal{T}_0^x| + t - 1 \in S\} \\
 &= |S \cap \{|x_0|, \dots, |x_0| + T - 1\}|.
 \end{aligned}$$

Thus, $B(\mathcal{T}^x) = |S \cap \{|x_0|, \dots, |x_0| + T - 1\}|$ and $C_T(\mathcal{F}) \geq \sup_{x \in \mathbb{Z}} B(\mathcal{T}^x) = \sup_{x \in \mathbb{Z}} |S \cap \{|x|, \dots, |x| + T - 1\}|$. This completes the proof.

Appendix C. Proof of Theorem 12

The proof of (i) follows from Theorem 9 and the fact that $C_T(\mathcal{F}) \leq Bd(\mathcal{F})$. To prove (ii), observe that by Theorem 9, it suffices to show that when $Bd(\mathcal{F}) = \infty$, we have that $C_T(\mathcal{F}) = \omega(1)$. If $Bd(\mathcal{F}) = \infty$, then for every $d \in \mathbb{N}$, there exists a shattered trajectory tree \mathcal{T} such that $B(\mathcal{T}) > d$. Thus, for every $d \in \mathbb{N}$, there exists a depth $d' \in \mathbb{N}$, such that for every $T \geq d'$, there exists a shattered tree \mathcal{T} of depth T with $B(\mathcal{T}) > d$. In other words, for every $d \in \mathbb{N}$, there exists a $d' \in \mathbb{N}$ such that for all $T \geq d'$ we have that $C_T(\mathcal{F}) > d$. By definition of $\omega(\cdot)$, this means that $C_T(\mathcal{F}) = \omega(1)$ as needed.

Appendix D. Proof of Theorem 13

Proof (of (i) in Theorem 13) To prove (i), pick $S = \{2^t : t \in \mathbb{N} \cup \{0\}\}$ and consider the function class \mathcal{F}_S from Theorem 10. It is not too hard to see that $C_T(\mathcal{F}) = \Theta(\log(T))$. In addition, one can verify that every finite subset of S is a shattered set according to Definition 7. Indeed, consider any finite subset $A \subset S$. For every $i \in [|A|]$, let A_i denote the i 'th element of A after sorting A in

increasing order. Then, observe that by the construction of \mathcal{F}_S , for every sequence $\varepsilon \in \{-1, 1\}^{|A|}$, there exists a function $f_\varepsilon \in \mathcal{F}_S$ such that $f_\varepsilon(A_i) = \varepsilon_i(A_i + 1)$ for every $i \in [|A|]$. By letting $\mathcal{H} = \{f_\varepsilon : \varepsilon \in \{-1, 1\}^{|A|}\}$ in Definition 7, one can verify that $|\mathcal{H}| = 2^{|A|} < \infty$, and for every $x \in A$ and $h \in \mathcal{H}$, there exists a $g \in \mathcal{H}$ such that $h(x) = -g(x)$ and $h(z) = g(z)$ for all $z \in A \setminus \{x\}$. Thus, \mathcal{F}_S shatters $A \subset S$. Since finite subsets of S can be arbitrary large, we have that $\text{DS}(\mathcal{F}) = \infty$. \blacksquare

Proof (of (ii) in Theorem 13) To prove (ii), let $\mathcal{X} = \mathbb{N} \cup \{\star\}$ and consider the following evolution function class. For every $\sigma \in \{-1, 1\}^{\mathbb{N}}$, define a sequence $a_\sigma : \mathbb{N} \rightarrow \mathbb{N}$ recursively such that $a_\sigma(1) = 1$ and $a_\sigma(n) = 2a_\sigma(n-1) + \frac{1+\sigma_{n-1}}{2}$ for $n \geq 2$. Equivalently, we can define the sequence a_σ explicitly by $a_\sigma(1) = 1$ and $a_\sigma(n) = 2^{n-1} + \sum_{i=1}^{n-1} \left(\frac{1+\sigma_i}{2}\right) 2^{n-(i+1)}$ for $n \geq 2$. Let $S_n := \bigcup_\sigma \{a_\sigma(n)\}$ and note that $S_n = \{2^{n-1}, \dots, 2^n - 1\}$ for every $n \geq 1$ and $S_n \cap S_r = \emptyset$ for all $n \neq r$. We establish some important properties about these sequences.

- (1) The sequence a_σ is strictly monotonically increasing in its input, and hence invertible. Accordingly, given a sequence a_σ and an element $x \in \text{im}(a_\sigma)$, let $a_\sigma^{-1}(x)$ denote the index $n \in \mathbb{N}$ such that $a_\sigma(n) = x$.
- (2) For every σ_1, σ_2 , if $a_{\sigma_1}(n) = a_{\sigma_2}(r)$, then $n = r$ since $S_n \cap S_r = \emptyset$ for all $n \neq r$.
- (3) The value of $a_\sigma(n)$ depends only on the prefix $(\sigma_1, \dots, \sigma_{n-1})$. Hence, two strings σ_1, σ_2 that share the same prefix up to and including index d will have the property that $a_{\sigma_1}(n) = a_{\sigma_2}(n)$ for all $n \leq d + 1$.
- (4) If $a_{\sigma_1}(d) = a_{\sigma_2}(d)$, then $a_{\sigma_1}(i) = a_{\sigma_2}(i)$ for all $i \leq d$. This follows by induction. For the base case, consider $i = d - 1$. If $a_{\sigma_1}(i) \neq a_{\sigma_2}(i)$, then $a_{\sigma_1}(d) = 2a_{\sigma_1}(i) + \frac{1+\sigma_{1,i}}{2} \neq 2a_{\sigma_2}(i) + \frac{1+\sigma_{2,i}}{2} = a_{\sigma_2}(d)$ for any value of $\sigma_{1,i}$ and $\sigma_{2,i}$. For the induction step, suppose $a_{\sigma_1}(i) = a_{\sigma_2}(i)$ for some $2 \leq i < d$. Then, if $a_{\sigma_1}(i-1) \neq a_{\sigma_2}(i-1)$, we have that $a_{\sigma_1}(i) = 2a_{\sigma_1}(i-1) + \frac{1+\sigma_{1,i-1}}{2} \neq 2a_{\sigma_2}(i-1) + \frac{1+\sigma_{2,i-1}}{2} = a_{\sigma_2}(i)$ for any value of $\sigma_{1,i-1}$ and $\sigma_{2,i-1}$.

We now construct a function class. For every $\sigma \in \{-1, 1\}^{\mathbb{N}}$, define the evolution function

$$f_\sigma(x) = \star \mathbb{1}\{x \notin \text{im}(a_\sigma)\} + a_\sigma\left(a_\sigma^{-1}(x) + 1\right) \mathbb{1}\{x \in \text{im}(a_\sigma)\}.$$

At a high-level, the evolution function f_σ maps every state in $\mathcal{X} \setminus \text{im}(a_\sigma)$ to \star and every state in $\text{im}(a_\sigma)$ to the next element in the sequence corresponding to a_σ .

Consider the function class $\mathcal{F} = \{f_\sigma : \sigma \in \{-1, 1\}^{\mathbb{N}}\}$. We now claim that $\text{C}_T(\mathcal{F}) = T$. To see this, fix a depth $d \in \mathbb{N}$, and consider the following trajectory tree \mathcal{T} of depth d . Let the root node be labeled by 1, that is $\mathcal{T}_0 = 1$. For all $t \in [d]$ and $\varepsilon \in \{-1, 1\}^t$, let $\mathcal{T}_t(\varepsilon) = a_{\tilde{\varepsilon}}(t+1)$ where $\tilde{\varepsilon}$ denotes an arbitrary extension of ε over \mathbb{N} . Note that the completion can be arbitrary because the value of $a_\sigma(t+1)$ for any $\sigma \in \{-1, 1\}^{\mathbb{N}}$ depends only on the prefix $(\sigma_1, \dots, \sigma_t)$. One can verify that such a tree \mathcal{T} is a complete binary tree of depth d where the root node is labeled with 1 and the internal nodes on depth $i \geq 1$ are labeled from left to right by $2^i, 2^i + 1, \dots, 2^{i+1} - 1$. Thus, it is clear that $\text{B}(\mathcal{T}) = d$ since for every internal node including the root, its two children are labeled by differing states. In addition, observe that for every path $\varepsilon \in \{-1, 1\}^d$ down \mathcal{T} , the function $f_{\tilde{\varepsilon}} \in \mathcal{F}$ shatters the associated sequence of states, where $\tilde{\varepsilon}$ again is an arbitrary completion of ε over \mathbb{N} . Indeed,

fix a $\varepsilon \in \{-1, 1\}^d$, a completion $\tilde{\varepsilon} \in \{-1, 1\}^{\mathbb{N}}$, and consider any $t \in [d]$. Then, by definition of \mathcal{T} , we have that $\mathcal{T}_{t-1}(\varepsilon_{<t}) = a_{\tilde{\varepsilon}}(t)$ and $\mathcal{T}_t(\varepsilon_{\leq t}) = a_{\tilde{\varepsilon}}(t+1)$. Consider the function $f_{\tilde{\varepsilon}} \in \mathcal{F}$. By definition of $f_{\tilde{\varepsilon}}$, we have that $f_{\tilde{\varepsilon}}(a_{\tilde{\varepsilon}}(t)) = a_{\tilde{\varepsilon}}(t+1)$, which implies that $f_{\tilde{\varepsilon}}(\mathcal{T}_{t-1}(\varepsilon_{<t})) = \mathcal{T}_t(\varepsilon_{\leq t})$. Since $t \in [d]$ was arbitrary, this is true for every $t \in [d]$, and thus $f_{\tilde{\varepsilon}}$ shatters the path ε down \mathcal{T} as claimed. Since $\varepsilon \in \{-1, 1\}^d$ was also arbitrary, we have that the entire tree \mathcal{T} is shattered by \mathcal{F} . Finally, since $d \in \mathbb{N}$ was arbitrary, this is true for arbitrarily large depths. Thus, $C_T(\mathcal{F}) = T$.

We now show that $DS(\mathcal{F}) = 1$ by proving that \mathcal{F} cannot DS-shatter any two instances $x_1, x_2 \in \mathcal{X}$. Our proof will be in cases. First, observe that if either $x_1 = \star$ or $x_2 = \star$, then (x_1, x_2) cannot be shattered since all functions in \mathcal{F} will output \star on either x_1 or x_2 . Thus, without loss of generality, suppose both $x_1, x_2 \in \mathbb{N}$ and $x_1 < x_2$. Consider any finite subset $\mathcal{H} \subset \mathcal{F}$ and suppose there exists a function $h \in \mathcal{H}$ such that $h(x_2) \neq \star$. Then, in order to shatter (x_1, x_2) , there must exist a function $g \in \mathcal{H}$ such that $h(x_1) \neq g(x_1)$ but $h(x_2) = g(x_2)$. However, if $h(x_2) = g(x_2) \neq \star$, then it must be the case that $h(x_1) = g(x_1)$. To see why, fix an instance $x \in \mathbb{N}$, and suppose $f_{\sigma_1}(x) = f_{\sigma_2}(x) \neq \star$. Then, by properties (2) and (4) above, it must be the case that $a_{\sigma_1}^{-1}(x) = a_{\sigma_2}^{-1}(x) = c$ and $a_{\sigma_1}(i) = a_{\sigma_2}(i)$ for all $i \leq c$. Thus, if $x_1 < x_2$ and $f_{\sigma_1}(x_2) = f_{\sigma_2}(x_2) \neq \star$, we must have that $f_{\sigma_1}(x_1) = f_{\sigma_2}(x_1)$ because either $x_1 \notin \text{im}(a_{\sigma_1}) \cup \text{im}(a_{\sigma_2})$ or $a_{\sigma_1}^{-1}(x_1) < a_{\sigma_1}^{-1}(x_2) = a_{\sigma_2}^{-1}(x_2)$. Thus, if \mathcal{H} were to satisfy the property in Definition 7, there cannot exist a hypothesis $h \in \mathcal{H}$ such that $h(x_2) \neq \star$. However, if for every $h \in \mathcal{H}$, we have that $h(x_2) = \star$, then for every $h \in \mathcal{H}$, there cannot exist a $g \in \mathcal{H}$ such that $h(x_1) = g(x_1)$ and $h(x_2) \neq g(x_2)$. Therefore, the two points (x_1, x_2) cannot be shattered. Since (x_1, x_2) and $\mathcal{H} \subset \mathcal{F}$ were arbitrary, this is true for all such points, implying that $DS(\mathcal{F}) \leq 1$. Since $|\mathcal{F}| \geq 2$, we have also that $DS(\mathcal{F}) \geq 1$, completing the proof that $DS(\mathcal{F}) = 1$. \blacksquare

Proof (of (iii) in Theorem 13) To prove part (iii) of Theorem 13, we reduce learning dynamical systems to online multiclass classification. Let $\mathcal{F} \subseteq \mathcal{X}^{\mathcal{X}}$ and \mathcal{B} be any (potentially randomized) online learner for \mathcal{F} for online multiclass classification with expected regret bound R . We will construct a learner \mathcal{A} that uses \mathcal{B} as a subroutine such that $M_{\mathcal{A}}(T, \mathcal{F}) \leq R$.

To that end, let (x_0, x_1, \dots, x_T) be the *realizable* stream to be observed by the learner and consider the following learning algorithm \mathcal{A} which makes the same predictions as \mathcal{B} while simulating the stream of labeled instance $(x_0, x_1), \dots, (x_{T-1}, x_T)$ to \mathcal{B} .

Algorithm 2 Learning algorithm \mathcal{A} .

Input: Online multiclass learner \mathcal{B} , initial state x_0 .

for $t = 1, \dots, T$ **do**

| Pass x_{t-1} to \mathcal{B} and receive prediction \hat{z}_t .
 | Predict $\hat{x}_t = \hat{z}_t$.
 | Receive next state x_t and update \mathcal{B} using labeled instance (x_{t-1}, x_t) .

end

Then, observe that

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\hat{x}_t \neq x_t\} \right] = \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\hat{z}_t \neq x_t\} \right] \stackrel{(i)}{\leq} \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} + R \stackrel{(ii)}{=} R \quad (4)$$

where (i) follows from the expected regret guarantee of \mathcal{B} and (ii) follows from the fact that the stream of states is realizable. Thus, $\inf_{\mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) \leq R$. Since \mathcal{B} is always guaranteed to observe a realizable sequence of labeled instances, picking the SOA (Littlestone, 1987; Daniely et al., 2011) gives that $R = L(\mathcal{F})$. Since the SOA is deterministic, this choice of \mathcal{B} implies that \mathcal{A} is deterministic. Part (i) then follows from that the fact that for any deterministic learner \mathcal{A} we have $M_{\mathcal{A}}(T, \mathcal{F}) \geq C_T(\mathcal{F})$. \blacksquare

Proof (of (iv) in Theorem 13) To prove part (iv), let $\mathcal{X} = [0, 1]$ and consider the class of thresholds $\mathcal{F} = \{x \mapsto \mathbb{1}\{x \geq a\} : a \in (0, 1)\}$. It is well known that $L(\mathcal{F}) = \infty$. On the other hand, note that $\dim(\mathcal{F}) = 1$ since $\mathcal{F}(1) = \{1\}$, $\mathcal{F}(0) = \{0\}$, and $\mathcal{F}(x) = \{0, 1\}$ for all $x \in (0, 1)$. \blacksquare

Appendix E. Proofs for Linear Systems

Proof (of (i) in Theorem 14) We first prove the lower bound. It suffices to show that

$$\inf_{\text{Deterministic } \mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) \geq r + 1.$$

Let \mathcal{A} be any deterministic algorithm and $\{e_0, e_1, \dots, e_r\}$ be arbitrary $r + 1$ standard basis on \mathbb{R}^n . This set exists because $n \geq r + 1$. Consider a stream such that $x_0 = e_0$, $x_t = \{-e_t, e_t\} \setminus \mathcal{A}(x_{<t})$ for all $t \in [r]$, and $x_{r+1} = \{-e_1, e_1\} \setminus \mathcal{A}(x_{<r+1})$. Here, $\forall t \in [r]$, we choose x_t to be an element of the set $\{-e_t, e_t\}$ other than \mathcal{A} 's prediction on round t . Similarly, x_{r+1} is chosen among $\{-e_1, e_1\}$. We can define a stream using \mathcal{A} because such a stream can be simulated before the game starts as \mathcal{A} is deterministic. Let $\{\sigma_t\}_{t=0}^{r+1} \in \{-1, 1\}^{r+2}$ such that $(x_0, \dots, x_{r+1}) = (\sigma_0 e_0, \sigma_1 e_1, \sigma_2 e_2, \dots, \sigma_r e_r, \sigma_{r+1} e_1)$. Consider the integer-valued matrix $W^* = \sum_{t=1}^r \sigma_{t-1} \sigma_t e_t \otimes e_{t-1} + \sigma_r \sigma_{r+1} e_1 \otimes e_r$. For $r + 2 \leq t \leq T$, one can define the stream to be $x_t = W^* x_{t-1}$. Note that the $\text{image}(W^*) \subseteq \text{span}(\{e_1, \dots, e_r\})$. Thus, $\text{rank}(W^*) \leq r$ and $W^* \in \mathcal{F}$. Finally, since W^* satisfies $W^*(\sigma_{t-1} e_{t-1}) = \sigma_t e_t$ for $t \in [r]$, $W^*(\sigma_r e_r) = \sigma_{r+1} e_1$, and $W^* x_{t-1} = x_t$ for $t \geq r + 2$, the stream is realizable. By the definition of the stream, \mathcal{A} makes mistakes on all rounds $t \in \{1, \dots, r + 1\}$. Thus, we have $M_{\mathcal{A}}(T, \mathcal{F}) \geq r + 1$.

To prove the upper bound, fix $T \in \mathbb{N}$ and suppose that $T \geq r + 1$ (otherwise $C_T(\mathcal{F}) \leq T \leq r + 1$). Consider a trajectory tree \mathcal{T} of depth T shattered by \mathcal{F} . We first show that, for any path $\sigma \in \{-1, 1\}^T$ down \mathcal{T} and $t \in [T - 1]$, if $\mathcal{T}_{t+1}((\sigma_{\leq t}, -1)) \neq \mathcal{T}_{t+1}((\sigma_{\leq t}, +1))$, then $\mathcal{T}_t(\sigma_{\leq t}) \notin \text{span}(\{\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t-1}(\sigma_{\leq t-1})\})$. To prove the contraposition of this statement, suppose $\mathcal{T}_t(\sigma_{\leq t}) \in \text{span}(\{\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t-1}(\sigma_{\leq t-1})\})$. Then, there exists constants $a_0, \dots, a_{t-1} \in \mathbb{Z}$ such that $a_0 \mathcal{T}_0 + \dots + a_{t-1} \mathcal{T}_{t-1}(\sigma_{\leq t-1}) = \mathcal{T}_t(\sigma_{\leq t})$. Thus, every matrix W consistent with the sequence $\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t-1}(\sigma_{\leq t-1})$ satisfies $W \mathcal{T}_t(\sigma_{\leq t}) = W(a_0 \mathcal{T}_0 + \dots + a_{t-1} \mathcal{T}_{t-1}(\sigma_{\leq t-1})) = a_0 \mathcal{T}_1(\sigma_{\leq 1}) + \dots + a_{t-1} \mathcal{T}_t(\sigma_{\leq t})$. This implies that the path $\sigma_{\leq t+1}$ is shattered only if $\mathcal{T}_{t+1}((\sigma_{\leq t}, -1)) = \mathcal{T}_{t+1}((\sigma_{\leq t}, +1))$.

Next, we show that the branching factor of every path in \mathcal{T} can be at most $r + 1$. Suppose, for the sake of contradiction, there exists a path $\sigma \in \{-1, 1\}^T$ whose branching factor is $k > r + 1$. We are guaranteed an increasing sequence of time points t_1, t_2, \dots, t_k in $\{0, \dots, T - 1\}$ such that $\mathcal{T}_{t_i+1}((\sigma_{\leq t_i}, -1)) \neq \mathcal{T}_{t_i+1}((\sigma_{\leq t_i}, +1))$ for all $i \in [k]$. Since $t_2 \geq 1$, by our argument above, we are guaranteed that $\mathcal{T}_{t_i}(\sigma_{\leq t_i}) \notin \text{span}(\{\mathcal{T}_0, \dots, \mathcal{T}_{t_i-1}(\sigma_{\leq t_i-1})\})$ for all $i \in \{2, \dots, k\}$. This further implies that the set $S := \{\mathcal{T}_{t_2}(\sigma_{\leq t_2}), \dots, \mathcal{T}_{t_k}(\sigma_{\leq t_k})\}$ is a linearly independent set. Note that $|S| = k - 1 > r$. Let W_{σ} be the matrix such that the associated function $f_{\sigma} \in \mathcal{F}$ shatters this path

σ . By the definition of the tree, we must have that $S \subseteq \text{image}(W_\sigma)$. Since $\text{rank}(W_\sigma) \leq r$, the set $\text{image}(W_\sigma)$ must be a subset of a vectorspace of dimension $\leq r$. However, this contradicts the fact that S contains at least $r+1$ linearly independent vectors in \mathbb{R}^n . Thus, the branching factor of every path in \mathcal{T} is at most $r+1$ and $B(\mathcal{T}) \leq r+1$. Since \mathcal{T} is arbitrary, we have $C_T(\mathcal{F}) \leq r+1$. ■

Proof (of (ii) in Theorem 14) We claim that it is without loss of generality to consider the function class $\mathcal{G} = \{x \mapsto Mx \pmod{2} : M \in \{0, 1\}^{n \times n}\} \subset \mathcal{F}$. Indeed, for every $f \in \mathcal{F}$, there exists a $g \in \mathcal{G}$ such that $f(x) = g(x)$ for all $x \in \mathcal{X}$. To see this, let $W \in \mathbb{Z}^{n \times n}$ be such that $f(x) = Wx \pmod{2}$ and fix some $x \in \{0, 1\}^n$. Then, observe that $f(x) = Wx \pmod{2} = (W \pmod{2}x) \pmod{2} = Mx \pmod{2}$ where $M = W \pmod{2} \in \{0, 1\}^{n \times n}$. Thus the function $g \in \mathcal{G}$ parameterized by M matches f everywhere on \mathcal{X} . We now prove that $C_T(\mathcal{G}) = n$. In the lower bound, we will use the fact that for a system of n linearly independent equations with n free variables defined over the integers modulo 2, there exists a unique solution.

Starting with the upper bound, fix $T \in \mathbb{N}$ and suppose that $T \geq n$ (as otherwise $C_T(\mathcal{G}) \leq T \leq n$). Consider a trajectory tree \mathcal{T} of depth T shattered by \mathcal{G} . Let $\sigma \in \{-1, 1\}^T$ denote an arbitrary path down \mathcal{T} . We first claim that for $t \geq 1$ if $\mathcal{T}_t(\sigma_{\leq t})$ is linearly dependent modulo 2 on the preceding sequence of states $\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t-1}(\sigma_{<t})$, then $\mathcal{T}_{t+1}((\sigma_{\leq t}, -1)) = \mathcal{T}_{t+1}((\sigma_{\leq t}, +1))$. To see this, suppose that $\mathcal{T}_t(\sigma_{\leq t})$ is linearly dependent modulo 2 on the preceding sequence of states $\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t-1}(\sigma_{<t})$. Then, there exists constants $a_0, \dots, a_{t-1} \in \{0, 1\}$ such that $a_0 \mathcal{T}_0 + \dots + a_{t-1} \mathcal{T}_{t-1}(\sigma_{<t}) \pmod{2} = \mathcal{T}_t(\sigma_{\leq t})$. Thus, every function $g \in \mathcal{G}$ consistent with the sequence $\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t-1}(\sigma_{<t})$ outputs $a_0 \mathcal{T}_1(\sigma_{\leq 1}) + \dots + a_{t-1} \mathcal{T}_t(\sigma_{\leq t}) \pmod{2}$ on $\mathcal{T}_t(\sigma_{\leq t})$, implying that the paths $(\sigma_{\leq t}, -1)$ and $(\sigma_{\leq t}, +1)$ are shattered only if $\mathcal{T}_{t+1}((\sigma_{\leq t}, -1)) = \mathcal{T}_{t+1}((\sigma_{\leq t}, +1)) = a_0 \mathcal{T}_1(\sigma_{\leq 1}) + \dots + a_{t-1} \mathcal{T}_t(\sigma_{\leq t}) \pmod{2}$. As a consequence, for $t \geq 1$, if $\mathcal{T}_{t+1}((\sigma_{\leq t}, -1)) \neq \mathcal{T}_{t+1}((\sigma_{\leq t}, +1))$, then $\mathcal{T}_t(\sigma_{\leq t})$ is linearly independent modulo 2 of its preceding states. Next, we claim that there can be at most $n-1$ timepoints $t_1, \dots, t_{n-1} \in [T-1]$ such that $\mathcal{T}_{t_i}(\sigma_{\leq t_i})$ is linearly independent of its preceding sequence of states $\mathcal{T}_0, \mathcal{T}_1(\sigma_{\leq 1}), \dots, \mathcal{T}_{t_i-1}(\sigma_{<t_i})$. Indeed, suppose for sake of contradiction there exists n timepoints $t_1, \dots, t_n \in [T-1]$ such that $\mathcal{T}_{t_i}(\sigma_{\leq t_i})$ is linearly independent of its preceding states. Then, note that the following set of $n+1$ states $\{\mathcal{T}_0, \mathcal{T}_{t_1}(\sigma_{\leq t_1}), \dots, \mathcal{T}_{t_n}(\sigma_{\leq t_n})\}$ are linearly independent modulo 2. This is a contradiction since \mathcal{X} is n -dimensional. Combining the two claims, we get that

$$\begin{aligned} \sum_{t=1}^T \mathbb{1}\{\mathcal{T}_t((\sigma_{<t}, -1)) \neq \mathcal{T}_t((\sigma_{<t}, +1))\} &\leq 1 + \sum_{t=2}^T \mathbb{1}\{\mathcal{T}_t((\sigma_{<t}, -1)) \neq \mathcal{T}_t((\sigma_{<t}, +1))\} \\ &\leq 1 + \sum_{t=1}^{T-1} \mathbb{1}\{\mathcal{T}_t(\sigma_{\leq t}) \text{ linearly indep. of preceding states}\} \\ &\leq 1 + n - 1 = n. \end{aligned}$$

Since $\sigma \in \{-1, 1\}^T$ is arbitrary, we get that $B(\mathcal{T}) \leq n$. Finally, since \mathcal{T} is arbitrary, we have that $C_T(\mathcal{F}) \leq n$ which completes the proof of the upper bound.

To prove the lower bound, let \mathcal{A} be any deterministic algorithm. Let $\mathcal{E} = \{e_0, \dots, e_{n-1}\} \subset \mathcal{X}$ be the standard basis over \mathbb{R}^n . Consider the stream where we pick $x_0 = e_0$, for all $t \in [n-2]$, we pick $x_t \in \mathcal{E} \setminus \{x_0, \dots, x_{t-1}, \mathcal{A}(x_{<t})\}$. Let $e = \mathcal{E} \setminus \{x_0, \dots, x_{n-2}\}$ be the remaining basis. Pick $x_{n-1} \in \{e, e + e_0\} \setminus \{\mathcal{A}(x_{<n-1})\}$. Finally, pick $x_n \neq \mathcal{A}(x_{<n})$. We can define a stream using \mathcal{A}

because it is deterministic and thus can be simulated before the game begins. Next, we show that x_0, \dots, x_n is a realizable stream. This follows from the fact that x_0, \dots, x_{n-1} are linearly independent by definition, and thus there exists a function $g \in \mathcal{G}$ such that $g(x_{t-1}) = x_t$ for all $t \in [n]$. Moreover, by definition of the stream, \mathcal{A} makes a mistake in every round. Thus, $M_{\mathcal{A}}(T, \mathcal{F}) \geq n$. Using the fact that $M_{\mathcal{A}}(T, \mathcal{F}) \leq C_T(\mathcal{F})$ completes the proof. \blacksquare

Proof (of (iii) in Theorem 14) Starting with the lower bound, it suffices to show that

$$\inf_{\text{Deterministic } \mathcal{A}} M_{\mathcal{A}}(T, \mathcal{F}) \geq n.$$

Let \mathcal{A} be any deterministic algorithm and $\{e_1, \dots, e_n\}$ be the standard basis on \mathbb{R}^n . Consider a stream such that $x_0 = e_1 + e_2 + \dots + e_n$, then $x_1 = x_0 - e_{i_1}$ for some $i_1 \in [n]$ such that $x_1 \neq \mathcal{A}(x_0)$, and for all $t \in \{2, \dots, n-1\}$,

$$x_t = x_0 - (e_{i_1} + \dots + e_{i_t}) \text{ such that } e_{i_t} \notin \{e_{i_1}, \dots, e_{i_{t-1}}\} \text{ and } x_t \neq \mathcal{A}(x_{<t}).$$

Define $e_{i_n} = x_0 - (e_{i_1} + \dots + e_{i_{n-1}})$. Note that $x_{n-1} = e_{i_n}$. Finally, we choose $x_n \in \{e_{i_n}, \mathbf{0}\}$ such that $x_n \notin \mathcal{A}(x_{<n-1})$. Here, $\forall t \in [n-2]$, we choose x_t by subtracting a basis e_{i_t} , which has not been subtracted before, from x_{t-1} such that x_t is other than what \mathcal{A} would have predicted on round t . Finally, for x_n , we either choose it to be equal to $x_{n-1} = e_{i_n}$ again or $\mathbf{0}$ while ensuring that $x_n \neq \mathcal{A}(x_{<n-1})$. We can define a stream using \mathcal{A} because such a stream can be simulated before the game starts as \mathcal{A} is deterministic.

Next, we show that (x_0, \dots, x_n) is a realizable stream. Indeed, the boolean matrix

$$W^* = \sum_{k=2}^n e_{i_k} \otimes (e_{i_1} + \dots + e_{i_{k-1}}) + x_n \otimes e_{i_n}.$$

satisfies $\mathbb{1}\{W^* x_{t-1} > 0\} = x_t$ for all $t \in [n]$. For $t = 1$, we have

$$\begin{aligned} W^* x_0 &= \left(\sum_{k=2}^n e_{i_k} \langle e_{i_1} + \dots + e_{i_{k-1}}, x_0 \rangle \right) + x_n \\ &= \sum_{k=2}^n (k-1) e_{i_k} + x_n \\ &= (k-1)(x_0 - e_{i_1}) + x_n \end{aligned}$$

Thus, $\mathbb{1}\{W^* x_0 > 0\} = x_0 - e_{i_1} = x_1$. For $t \in \{2, \dots, n-1\}$, we have

$$\begin{aligned} W^* x_{t-1} &= \left(\sum_{k=2}^n e_{i_k} \langle e_{i_1} + \dots + e_{i_{k-1}}, x_0 - (e_{i_1} + \dots + e_{i_{t-1}}) \rangle \right) + x_n \\ &= \sum_{k=2}^n e_{i_k} \langle e_{i_1} + \dots + e_{i_{k-1}}, e_{i_t} + \dots + e_{i_n} \rangle + x_n \\ &= \sum_{k=t+1}^n (k-t) e_{i_k} + x_n. \end{aligned}$$

So, we obtain $\mathbb{1}\{W^*x_{t-1} > 0\} = x_0 - (e_{i_1} + \dots + e_{i_t}) > 0 = x_t$. Finally, since $x_{n-1} = e_{i_n}$ and $W^*e_{i_n} = x_n$, we have $\mathbb{1}\{W^*x_{n-1} > 0\} = x_n$. By the definition of the stream, \mathcal{A} makes mistakes in every round on this stream. Thus, $M_{\mathcal{A}}(T, \mathcal{F}) \geq n$.

As for the upper bound, since $|\{0, 1\}^{n \times n}| = 2^{n^2}$, we have $L(\mathcal{F}) \leq \log(2^{n^2}) = n^2$. Thus, by Theorem 13 (iii), we must have $C_T(\mathcal{F}) \leq n^2$. \blacksquare

Appendix F. Proof of Theorem 15

Proof (of upper bound in Theorem 15) Let (x_0, x_1, \dots, x_T) be the stream to be observed by the learner. Given a multiclass classification learner \mathcal{B} for \mathcal{F} , consider the algorithm \mathcal{A} as defined in Algorithm D. Here, \mathcal{A} makes the same predictions as \mathcal{B} while simulating the stream of labeled instance $(x_0, x_1), \dots, (x_{T-1}, x_T)$ to \mathcal{B} . Using the bound (i) in Equation (4), we obtain

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\hat{x}_t \neq x_t\} \right] \leq \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} + R,$$

where expectation is taken with respect to the randomness of \mathcal{B} and R is the expected regret of the \mathcal{B} . This shows that $MR_{\mathcal{A}}(T, \mathcal{F}) \leq R$. Taking \mathcal{B} to be \mathbb{A}_{AG} defined in (Hanneke et al., 2023, Section 3), Theorem 4 in (Hanneke et al., 2023) implies that $R \leq L(\mathcal{F}) + \sqrt{T L(\mathcal{F}) \log T}$, matching the claimed upper bound in Theorem 15. \blacksquare

Proof (of lower bound of $\frac{L(\mathcal{F})}{18}$ in Theorem 15) Define $d := L(\mathcal{F})$ and let \mathcal{T} be the Littlestone tree of depth d shattered by \mathcal{F} . Let $(\mathcal{T}_0, \dots, \mathcal{T}_{d-1})$ be the sequence of node-labeling functions and (Y_1, \dots, Y_d) be the sequence of edge labeling functions of the shattered tree \mathcal{T} . To construct a stream, take $r = \lfloor \frac{d+2}{3} \rfloor$ and define a random sequence $\{n_i\}_{i=1}^r$ such that

$$n_1 = 1, \quad n_{i+1} \sim \text{Unif}(\{3i-1, 3i, 3i+1\}) \text{ for all } i \in [r-1].$$

Note that $n_r \leq 3(r-1) + 1 = 3r - 2 \leq d$. Next, pick a random path $(\sigma_1, \dots, \sigma_d)$ down the tree such that $\sigma_i \sim \text{Unif}\{-1, 1\}$. Let $x_0 = \mathcal{T}_0$ be the initial state and consider the stream

$$Y_{n_1}(\sigma_{\leq n_1}), \mathcal{T}_{n_2-1}(\sigma_{< n_2}), Y_{n_2}(\sigma_{\leq n_2}), \dots, \mathcal{T}_{n_r-1}(\sigma_{< n_r}), Y_{n_r}(\sigma_{\leq n_r}).$$

For any randomized algorithm \mathcal{A} , let \mathcal{A}_t denote its prediction on round t . Then, its expected cumulative loss on this stream is

$$\mathbb{E} \left[\sum_{t=1}^r \mathbb{1}\{\mathcal{A}_t \neq Y_{n_t}(\sigma_{\leq n_t})\} + \sum_{t=2}^r \mathbb{1}\{\mathcal{A}_t \neq \mathcal{T}_{n_{t-1}}(\sigma_{< n_t})\} \right],$$

where the expectation is taken with respect to \mathcal{A} , σ , and n_t 's. Note that

$$\mathbb{E}[\mathbb{1}\{\mathcal{A}_t \neq Y_{n_t}(\sigma_{\leq n_t})\}] \geq \frac{1}{2}.$$

where the inequality holds because conditioned on the history up to time point $t-1$, the state $Y_{n_t}(\sigma_{\leq n_t})$ is chosen uniformly at random between $Y_{n_t}((\sigma_{< n_t}, -1))$ and $Y_{n_t}((\sigma_{< n_t}, +1))$. So, the

algorithm cannot do better than random guessing. Similarly, given a path σ , the state $\mathcal{T}_{n_t-1}(\sigma_{<n_t})$ is selected uniformly from the set

$$\{\mathcal{T}_{m-1}(\sigma_{<m}) : m = 3(t-1) - 1, 3(t-1), 3(t-1) + 1\}.$$

Since each node along a path must have distinct elements, the aforementioned set must contain 3 elements. Thus, we have

$$\mathbb{E}[\mathbb{1}\{\mathcal{A}_t \neq \mathcal{T}_{n_t-1}(\sigma_{<n_t})\}] \geq \frac{2}{3}.$$

Overall, the total expected cumulative loss of \mathcal{A} is

$$\geq \sum_{t=1}^r \frac{1}{2} + \sum_{t=2}^r \frac{2}{3} \geq \frac{r}{2} + \frac{2(r-1)}{3} = \frac{7r-4}{6}.$$

To upper bound the loss of the best-fixed competitor, let $f_\sigma \in \mathcal{F}$ denote the function that shatters the path σ in the tree \mathcal{T} . Then, by definition of shattering, we have $f_\sigma(\mathcal{T}_{n_t-1}(\sigma_{<n_t})) = Y_{n_t}(\sigma_{\leq n_t})$. Thus, the cumulative loss of f_σ is

$$\begin{aligned} \mathbb{E}\left[\sum_{t=1}^r \mathbb{1}\{f_\sigma(\mathcal{T}_{n_t-1}(\sigma_{<n_t})) \neq Y_{n_t}(\sigma_{\leq n_t})\} + \sum_{t=2}^r \mathbb{1}\{f_\sigma(Y_{n_{t-1}}(\sigma_{\leq n_{t-1}})) \neq \mathcal{T}_{n_t-1}(\sigma_{<n_t})\}\right] \\ \leq 0 + \sum_{t=2}^r 1 = r-1. \end{aligned}$$

Hence, combining everything and plugging the value of r , the regret of \mathcal{A} is

$$\text{MR}_{\mathcal{A}}(T, \mathcal{F}) \geq \frac{7r-4}{6} - (r-1) = \frac{7r-4-6r+6}{6} = \frac{r+2}{6} \geq \frac{(d+2)/3-1+2}{6} \geq \frac{d}{18}.$$

This completes our proof. ■

Proof (of lower bound of $\frac{\sqrt{T}}{16\sqrt{3}}$ in Theorem 15) Let f_{-1} and f_{+1} be any two distinct functions in \mathcal{F} and $\bar{x}_0 \in \mathcal{X}$ be the point where they differ. That is, $f_{-1}(\bar{x}_0) \neq f_{+1}(\bar{x}_0)$. Define $S \subset \mathcal{X}$ such that $f_{-1}(x) = f_{+1}(x)$ for all $x \in S$. Moreover, define $S_0 = \{x \in S \mid f_{-1}(x) = f_{+1}(x) = \bar{x}_0\}$. Let $(\sigma_1, \dots, \sigma_T)$ be a sequence such that $\sigma_t \sim \text{Uniform}\{-1, 1\}$. Consider a random stream with initial state $x_0 = \bar{x}_0$ and for all $t \in [T]$,

$$x_t = \bar{x}_0 \mathbb{1}\{x_{t-1} \in S_0\} + \text{Uniform}(\{\bar{x}_0, f_{\sigma_t}(x_{t-1})\}) \mathbb{1}\{x_{t-1} \in S \setminus S_0\} + f_{\sigma_t}(x_{t-1}) \mathbb{1}\{x_{t-1} \notin S\}.$$

For any algorithm \mathcal{A} , its expected cumulative loss is

$$\begin{aligned} \mathbb{E}\left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\}\right] &= \mathbb{E}\left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \mathbb{1}\{x_{t-1} \in S_0\}\right] + \mathbb{E}\left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \mathbb{1}\{x_{t-1} \notin S_0\}\right] \\ &\geq 0 + \mathbb{E}\left[\sum_{t=1}^T \mathbb{E}[\mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \mid x_{<t}, \mathcal{A}] \mathbb{1}\{x_{t-1} \notin S_0\}\right] \\ &\geq \frac{1}{2} \mathbb{E}\left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S_0\}\right], \end{aligned}$$

where the final step follows because when $x_{t-1} \notin S_0$, the state x_t is sampled uniformly at random between two states. So, the expected loss of algorithm in this round is $\geq \frac{1}{2}$.

To upper bound the cumulative loss of the best-fixed function in hindsight, define $\sigma = \text{sign} \left(\sum_{t=1}^T \sigma_t \mathbb{1}\{x_{t-1} \notin S\} \right)$. Note that

$$\begin{aligned} & \mathbb{E} \left[\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} \right] \\ & \leq \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{f_\sigma(x_{t-1}) \neq x_t\} \right] \\ & = 0 + \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{f_\sigma(x_{t-1}) \neq x_t\} \mathbb{1}\{x_{t-1} \in S \setminus S_0\} \right] + \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{f_\sigma(x_{t-1}) \neq f_{\sigma_t}(x_{t-1})\} \mathbb{1}\{x_{t-1} \notin S\} \right] \\ & \leq \frac{1}{2} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \in S \setminus S_0\} \right] + \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{f_\sigma(x_{t-1}) \neq f_{\sigma_t}(x_{t-1})\} \mathbb{1}\{x_{t-1} \notin S\} \right], \end{aligned}$$

where the final step follows because conditioned on the event that $x_{t-1} \in S \setminus S_0$, the state x_t is $\sim \text{Uniform}(\{\bar{x}_0, f_{\sigma_t}(x_{t-1})\})$. Since $f_\sigma(x_{t-1}) = f_{\sigma_t}(x_{t-1})$ whenever $x_{t-1} \in S$, the expected loss of f_σ on round t is equal to the conditional probability that $x_t = x_0$, which is $\leq 1/2$. Moreover, using the fact that $\mathbb{1}\{f_\sigma(x_{t-1}) \neq f_{\sigma_t}(x_{t-1})\} \leq \mathbb{1}\{\sigma \neq \sigma_t\}$, we have

$$\begin{aligned} \mathbb{E} \left[\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} \right] & \leq \frac{1}{2} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \in S \setminus S_0\} \right] + \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\sigma \neq \sigma_t\} \mathbb{1}\{x_{t-1} \notin S\} \right] \\ & = \frac{1}{2} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \in S \setminus S_0\} \right] + \mathbb{E} \left[\sum_{t=1}^T \frac{1 - \sigma \sigma_t}{2} \mathbb{1}\{x_{t-1} \notin S\} \right] \\ & = \frac{1}{2} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S_0\} \right] - \frac{1}{2} \mathbb{E} \left[\sigma \sum_{t=1}^T \sigma_t \mathbb{1}\{x_{t-1} \notin S\} \right] \\ & = \frac{1}{2} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S_0\} \right] - \frac{1}{2} \mathbb{E} \left[\left| \sum_{t=1}^T \sigma_t \mathbb{1}\{x_{t-1} \notin S\} \right| \right], \end{aligned}$$

where the final equality follows from the definition of σ . Thus, the regret of \mathcal{A} is

$$\text{MR}_{\mathcal{A}}(T, \mathcal{F}) = \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] - \mathbb{E} \left[\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} \right] \geq \frac{1}{2} \mathbb{E} \left[\left| \sum_{t=1}^T \sigma_t \mathbb{1}\{x_{t-1} \notin S\} \right| \right].$$

We now lower bound the Rademacher sum above by closely following the proof of Khinchine's inequality ([Cesa-Bianchi and Lugosi, 2006](#), Lemma 8.1).

For any random variable Y with the finite-fourth moment, a simple application of Holder's inequality implies that

$$\mathbb{E}[|Y|] \geq \frac{(\mathbb{E}[Y^2])^{3/2}}{(\mathbb{E}[Y^4])^{1/2}}.$$

We apply this inequality to $Y := \sum_{t=1}^T \sigma_t \mathbb{1}\{x_{t-1} \notin S\}$. Since $Y^2 = \sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S\} + 2 \sum_{i < j} \sigma_i \sigma_j \mathbb{1}\{x_{i-1}, x_{j-1} \notin S\}$, we have

$$\mathbb{E}[Y^2] = \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S\} \right] + 2 \mathbb{E} \left[\sum_{i < j} \mathbb{E}[\sigma_j \mid \sigma_{<j}, x_{<j}] \sigma_i \mathbb{1}\{x_{i-1}, x_{j-1} \notin S\} \right] = \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S\} \right],$$

where the final equality follows because σ_j is still a Rademacher random variable conditioned on the past and $\mathbb{E}[\sigma_j \mid \sigma_{<j}, x_{<j}] = 0$. Furthermore, note that

$$\begin{aligned} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S\} \right] &\geq \mathbb{E}[\mathbb{1}\{x_0 \notin S\}] + \sum_{r=1}^{\lfloor \frac{T}{2} \rfloor - 1} \mathbb{E}[\mathbb{1}\{x_{2r-1} \notin S\} + \mathbb{1}\{x_{2r} \notin S\}] \\ &\geq 1 + \sum_{r=1}^{\lfloor \frac{T}{2} \rfloor - 1} \frac{1}{2} \\ &= \frac{1}{2} \left(\left\lfloor \frac{T}{2} \right\rfloor + 1 \right) \\ &\geq \frac{T}{4}. \end{aligned}$$

Here, the second inequality above uses (i) $x_0 = \bar{x}_0 \notin S$ and (ii) $\mathbb{E}[\mathbb{1}\{x_{2r-1} \notin S\} + \mathbb{1}\{x_{2r} \notin S\}] \geq \frac{1}{2}$ for all $r \geq 1$. To see (ii), one can consider two cases. First, if $x_{2r-1} \notin S$, then the inequality is trivially true. On the other hand, $x_{2r-1} \in S$, then $x_{2r} = \bar{x}_0 \notin S$ with probability 1/2. Thus, we have $\mathbb{E}[Y^2] \geq \frac{T}{4}$.

Similarly, for the fourth moment of Y , all the cross-term vanishes and we are left with the terms with fourth power and symmetric second powers.

$$\begin{aligned} \mathbb{E}[Y^4] &= \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{x_{t-1} \notin S\} \right] + \binom{4}{2} \mathbb{E} \left[\sum_{i < j} \mathbb{1}\{x_{i-1} \notin S\} \mathbb{1}\{x_{j-1} \notin S\} \right] \\ &\leq T + \binom{4}{2} \frac{T(T-1)}{2} \\ &\leq T + 3T(T-1) \\ &\leq 3T^2. \end{aligned}$$

The lower bound on the second moment of Y and the upper bound on the fourth moment of Y collectively implies that

$$\mathbb{E}[|Y|] \geq \frac{(\mathbb{E}[Y^2])^{3/2}}{(\mathbb{E}[Y^4])^{1/2}} \geq \frac{(T/4)^{3/2}}{(3T^2)^{1/2}} = \frac{\sqrt{T}}{8\sqrt{3}}.$$

Finally, combining everything, we obtain

$$\text{MR}_{\mathcal{A}}(T, \mathcal{F}) \geq \frac{1}{2} \mathbb{E} \left[\left| \sum_{t=1}^T \sigma_t \mathbb{1}\{x_{t-1} \notin S\} \right| \right] = \frac{1}{2} \mathbb{E}[|Y|] = \frac{\sqrt{T}}{16\sqrt{3}}.$$

This completes our proof. ■

Appendix G. Tightness of upper bound in Theorem 15

Fix $d \in \mathbb{N}$ and let $\mathcal{X} = \{1, 2, \dots, d\} \cup \{\pm(d+1), \pm(d+2), \dots, \pm 2d\}$. For every $\sigma \in \{-1, 1\}^d$, define a function $f_\sigma : \mathcal{X} \rightarrow \mathcal{X}$ such that

$$f_\sigma(x) = \sigma_x(d+x)\mathbb{1}\{x \in [d]\} + (|x| - d)\mathbb{1}\{x \notin [d]\}.$$

Consider $\mathcal{F} = \{f_\sigma : \sigma \in \{-1, 1\}^d\}$. It is not too hard to see that $L(\mathcal{F}) = d$. The fact that $L(\mathcal{F}) \leq d$ is trivial because any Littlestone tree can only have $\{1, 2, \dots, d\}$ in the internal nodes. This is because all functions output the same state in the domain $\mathcal{X} \setminus [d]$. Since the internal nodes in any given path of the Littlestone tree need to be different, the depth of any shattered tree is $\leq d$. To see why $L(\mathcal{F}) \geq d$, consider a complete binary tree \mathcal{T} of depth d with all the internal nodes in level $i \in [d]$ containing the element i . Let $-(d+i)$ and $(d+i)$ label the left and right outgoing edges respectively of every node in level i . Note that \mathcal{T} is shattered by \mathcal{F} as for any path $\varepsilon \in \{-1, 1\}^d$ down the tree \mathcal{T} , there exists a $f_\varepsilon \in \mathcal{F}$ such that $f_\varepsilon(i) = \varepsilon_i(d+i)$ for all $i \in [d]$. Thus, we have shown that $L(\mathcal{F}) = d$.

We now show that $\inf_{\mathcal{A}} \text{MR}_{\mathcal{A}}(T, \mathcal{F}) = \Omega(\sqrt{T L(\mathcal{F})})$, proving that the upper bound in Theorem 15 is tight up to $\sqrt{\log T}$. For a $k \in \mathbb{N}$, pick $T = 2kd - 1$. Draw $\varepsilon \in \{-1, +1\}^T$ where $\varepsilon_i \sim \text{Uniform}(\{-1, 1\})$ and consider a stream

$$x_{2k(i-1)} = i, x_t = \varepsilon_t(d+i)\mathbb{1}\{x_{t-1} = i\} + i\mathbb{1}\{x_{t-1} \neq i\} \quad \forall i \in [d] \text{ and } t \in \{2k(i-1) + 1, \dots, 2ki - 1\}.$$

Note that $x_0 = 1$ is the initial state and there are $T = 2kd - 1$ more states in this stream. For any algorithm \mathcal{A} , its expected cumulative loss is

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] \geq \mathbb{E} \left[\sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] \geq \frac{1}{2} \sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \mathbb{1}\{x_{t-1} = i\}.$$

Here, the first inequality is true because we just rewrite the sum without including the rounds $t = 2k, 4k, 6k, \dots, 2(d-1)k$. The second inequality holds because x_t is sampled uniformly at random between $-(d+i)$ and $d+i$ whenever $x_{t-1} = i$, and the algorithm cannot do better than guessing on these rounds. Note that there is no expectation following the second inequality because the event $x_{t-1} = i$ is deterministic.

To upper bound the loss of the best function in hindsight, define $\bar{\varepsilon}_i = \text{sign}(\sum_{t=2k(i-1)+1}^{2ki-1} \varepsilon_t \mathbb{1}\{x_{t-1} = i\})$ and consider $f_{\bar{\varepsilon}}$ where $\bar{\varepsilon} = (\bar{\varepsilon}_1, \dots, \bar{\varepsilon}_d)$. Then,

$$\begin{aligned} \mathbb{E} \left[\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} \right] &\leq \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{f_{\bar{\varepsilon}}(x_{t-1}) \neq x_t\} \right] \\ &\leq (d-1) + \mathbb{E} \left[\sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \mathbb{1}\{f_{\bar{\varepsilon}}(x_{t-1}) \neq x_t\} \right] \\ &= (d-1) + \mathbb{E} \left[\sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \mathbb{1}\{f_{\bar{\varepsilon}}(x_{t-1}) \neq x_t\} \mathbb{1}\{x_{t-1} = i\} \right]. \end{aligned}$$

Here, the second inequality holds because we trivially upper bound the losses on rounds $t = 2k, 4k, \dots, 2(d-1)k$ by 1. And the final equality follows because $f_{\bar{\varepsilon}}(x_{t-1}) = i = x_t$ whenever $x_{t-1} \neq$

i for $2k(i-1) + 1 \leq t \leq 2ki - 1$. Moreover, when $x_{t-1} = i$, we have $f_{\bar{\varepsilon}}(x_{t-1}) = \bar{\varepsilon}_i(d+i)$ and $x_t = \varepsilon_t(d+i)$. This implies that $\mathbb{1}\{f_{\bar{\varepsilon}}(x_{t-1}) \neq x_t\}\mathbb{1}\{x_{t-1} = i\} = \mathbb{1}\{\bar{\varepsilon}_i \neq \varepsilon_t\}\mathbb{1}\{x_{t-1} = i\}$. Thus, we can write

$$\begin{aligned}
& \mathbb{E} \left[\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} \right] \\
& \leq (d-1) + \mathbb{E} \left[\sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \mathbb{1}\{\bar{\varepsilon}_i \neq \varepsilon_t\}\mathbb{1}\{x_{t-1} = i\} \right] \\
& = (d-1) + \mathbb{E} \left[\sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \frac{1 - \bar{\varepsilon}_i \varepsilon_t}{2} \mathbb{1}\{x_{t-1} = i\} \right] \\
& = (d-1) + \sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \frac{1}{2} \mathbb{1}\{x_{t-1} = i\} - \frac{1}{2} \mathbb{E} \left[\sum_{i=1}^d \bar{\varepsilon}_i \sum_{t=2k(i-1)+1}^{2ki-1} \varepsilon_t \mathbb{1}\{x_{t-1} = i\} \right] \\
& = (d-1) + \sum_{i=1}^d \sum_{t=2k(i-1)+1}^{2ki-1} \frac{1}{2} \mathbb{1}\{x_{t-1} = i\} - \frac{1}{2} \mathbb{E} \left[\sum_{i=1}^d \left| \sum_{t=2k(i-1)+1}^{2ki-1} \varepsilon_t \mathbb{1}\{x_{t-1} = i\} \right| \right]
\end{aligned}$$

Thus, the expected regret of \mathcal{A} is

$$\begin{aligned}
\text{MR}_{\mathcal{A}}(T, \mathcal{F}) &= \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{t-1}) \neq x_t\} \right] - \mathbb{E} \left[\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f(x_{t-1}) \neq x_t\} \right] \\
&\geq \frac{1}{2} \mathbb{E} \left[\sum_{i=1}^d \left| \sum_{t=2k(i-1)+1}^{2ki-1} \varepsilon_t \mathbb{1}\{x_{t-1} = i\} \right| \right] - (d-1) \\
&\geq \frac{1}{2} \sum_{i=1}^d \sqrt{\frac{|\{2k(i-1)+1 \leq t \leq 2ki-1 : x_{t-1} = i\}|}{2}} - (d-1) \\
&= \frac{1}{2} \sum_{i=1}^d \sqrt{\frac{k}{2}} - (d-1).
\end{aligned}$$

where the second inequality follows due to Khinchine's inequality (Cesa-Bianchi and Lugosi, 2006, Lemma 8.2). The final equality holds because in each block of $2k$ rounds from $t = 2k(i-1) + 1$ to $t = 2ki - 1$, we have $x_{t-1} = i$ in exactly k of those rounds. Using $T = 2kd - 1$, we further obtain $\text{MR}_{\mathcal{A}}(T, \mathcal{F}) \geq \frac{1}{4} \sqrt{(T+1)d} - (d-1) = \Omega(\sqrt{Td})$ for $T \gg d$. This establishes our claim that the upper bound in Theorem 15 is tight up to $\sqrt{\log T}$.

Appendix H. Proof of Theorem 17

The lower bound in Theorem 17 follows directly from the lower bound in the realizable setting. Accordingly, we only prove the upper bound. Let $K := \sup_{x \in \mathcal{X}} |\mathcal{F}(x)|$.

Proof (of the upper bound in Theorem 17) Let $\{x_t\}_{t=0}^T$ be the trajectory to be observed by the learner and $f^* \in \arg \min_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{x_t \neq f^t(x_0)\}$ the optimal evolution function in hindsight.

Given the time horizon T , let $L_T = \{L \subset [T] : |L| \leq C_T(\mathcal{F})\}$ denote the set of all possible subsets of $[T]$ with size at most $C_T(\mathcal{F})$. For every $L \in L_T$, let $\phi : L \rightarrow [K]$ denote a function mapping time points in L to an integer in $[K]$. On time point $t \in L$, we should think of $\phi(t)$ as an index into the set $\mathcal{F}(x_{t-1})$. To that end, for an index $i \in [|\mathcal{F}(x)|]$, let $\mathcal{F}_i(x)$ denote the i -th element of the list obtained after sorting $\mathcal{F}(x)$ in its natural order. Let $\Phi_L = [K]^L$ denote all such functions ϕ . For each $L \in L_T$ and $\phi \in \Phi_L$, define an expert $\mathcal{E}_{L,\phi}$. On time point $t \in [T]$, the prediction of expert $\mathcal{E}_{L,\phi}$ is defined recursively by

$$\mathcal{E}_{L,\phi}(x_0, t) = \begin{cases} \mathcal{A}(x_0, t | \{\mathcal{E}_{L,\phi}(x_0, i)\}_{i=1}^{t-1}), & \text{if } t \notin L \\ \mathcal{F}_{\phi(t)}(\mathcal{E}_{L,\phi}(x_0, t-1)), & \text{otherwise} \end{cases}$$

where $\mathcal{E}_{L,\phi}(x_0, 0) = x_0$ and $\mathcal{A}(x_0, t | \{\mathcal{E}_{L,\phi}(x_0, i)\}_{i=1}^{t-1})$ denotes the prediction of Algorithm 1 on timepoint t after running and updating on the trajectory $\{\mathcal{E}_{L,\phi}(x_0, i)\}_{i=1}^{t-1}$. Let $E = \bigcup_{L \in L_T} \bigcup_{\phi \in \Phi_L} \mathcal{E}_{L,\phi}$ denote the set of all experts parameterized by $L \in L_T$ and $\phi \in \Phi_L$.

We claim that there exists an expert \mathcal{E}_{L^*,ϕ^*} such that $\mathcal{E}_{L^*,\phi^*}(x_0, t) = f^{*,t}(x_0)$ for all $t \in [T]$. To see this, consider the hypothetical trajectory $S^* = \{f^{*,t}(x_0)\}_{t=1}^T$ generated by f^* . Let $L^* = \{t_1, t_2, \dots\}$ be the indices on which Algorithm 1 would have made a mistake had it run and updated on S^* . By the guarantee of Algorithm 1, we know that $|L^*| \leq C_T(\mathcal{F})$. Moreover, there exists a $\phi^* \in \Phi_{L^*}$ such that $\mathcal{F}_{\phi^*(i)}(f^{*,i-1}(x_0)) = f^{*,i}(x_0)$ for all $i \in L^*$. By construction of \mathcal{E} , there exists an expert \mathcal{E}_{L^*,ϕ^*} parameterized by L^* and ϕ^* . We claim that $\mathcal{E}_{L^*,\phi^*}(x_0, t) = f^{*,t}(x_0)$ for all $t \in [T]$. This follows by strong induction on $t \in [T]$. For the base case $t = 1$, there are two subcases to consider. If $1 \in L^*$, then we have that $\mathcal{E}_{L^*,\phi^*}(x_0, 1) = \mathcal{F}_{\phi^*(1)}(\mathcal{E}_{L^*,\phi^*}(x_0, 0)) = \mathcal{F}_{\phi^*(1)}(x_0) = f^{*,1}(x_0)$. If $1 \notin L^*$, then $\mathcal{E}_{L^*,\phi^*}(x_0, 1) = \mathcal{A}(x_0, 1 | \{\}) = f^{*,1}(x_0)$, where the last equality follows by definition of L^* . Now for the induction step, suppose that $\mathcal{E}_{L^*,\phi^*}(x_0, i) = f^{*,i}(x_0)$ for all $i \leq t$. Then, if $t+1 \in L^*$, we have that $\mathcal{E}_{L^*,\phi^*}(x_0, t+1) = \mathcal{F}_{\phi^*(t+1)}(\mathcal{E}_{L^*,\phi^*}(x_0, t)) = \mathcal{F}_{\phi^*(t+1)}(f^{*,t}(x_0)) = f^{*,t+1}(x_0)$. If $t+1 \notin L^*$, then $\mathcal{E}_{L^*,\phi^*}(x_0, t+1) = \mathcal{A}(x_0, t+1 | \{\mathcal{E}_{L^*,\phi^*}(x_0, i)\}_{i=1}^t) = \mathcal{A}(x_0, t+1 | \{f^{*,i}(x_0)\}_{i=1}^t) = f^{*,t+1}(x_0)$, where the last equality again is due to the definition of L^* .

Now, consider the learner that runs the celebrated Randomized Exponential Weights Algorithm, denoted hereinafter by \mathcal{P} , using the set of experts E with learning rate $\eta = \sqrt{\frac{2 \ln(|E|)}{T}}$. By Theorem 21.11 of Shalev-Shwartz and Ben-David (2014), we have that

$$\begin{aligned} \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{P}(x_0, t) \neq x_t\} \right] &\leq \inf_{\mathcal{E} \in E} \sum_{t=1}^T \mathbb{1}\{\mathcal{E}(x_0, t) \neq x_t\} + \sqrt{2T \ln(|E|)} \\ &\leq \inf_{\mathcal{E} \in E} \sum_{t=1}^T \mathbb{1}\{\mathcal{E}(x_0, t) \neq x_t\} + \sqrt{2 C_T(\mathcal{F}) T \ln\left(\frac{KT}{C_T(\mathcal{F})}\right)} \\ &\leq \sum_{t=1}^T \mathbb{1}\{\mathcal{E}_{L^*,\phi^*}(x_0, t) \neq x_t\} + \sqrt{2 C_T(\mathcal{F}) T \ln\left(\frac{KT}{C_T(\mathcal{F})}\right)} \\ &= \sum_{t=1}^T \mathbb{1}\{f^{*,t}(x_0) \neq x_t\} + \sqrt{2 C_T(\mathcal{F}) T \ln\left(\frac{KT}{C_T(\mathcal{F})}\right)} \end{aligned}$$

where the second inequality follows because $|E| = \sum_{i=0}^{C_T(\mathcal{F})} K^i \binom{T}{i} \leq \left(\frac{KT}{C_T(\mathcal{F})}\right)^{C_T(\mathcal{F})}$. This completes the proof as \mathcal{P} achieves the stated upper bound on expected regret. \blacksquare

The next two examples show that both the lower- and upper bounds in Theorem 17 can be tight.

H.1. Tightness of lower bound in Theorem 17

Let $\mathcal{X} = \mathbb{Z}$ and fix $p \in \mathbb{N}$. For every $\sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}}$, define the evolution function

$$f_\sigma(x) = (\sigma_{|x|} \mathbb{1}\{|x| \leq p-1\} + \mathbb{1}\{|x| \geq p\})(|x| + 1)$$

and consider the class $\mathcal{F} = \{f_\sigma : \sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}}\}$. By construction of \mathcal{F} , branching only occurs on states in $\{0, 1, \dots, p-1\}$ and their negation. Moreover, given any initial state $x_0 \in \mathcal{X}$ and a time horizon $T \in \mathbb{N}$, the trajectory of any evolution in \mathcal{F} is some signed version of the sequence $|x_0| + 1, \dots, |x_0| + T$. From Theorem 10, it's not too hard to see that $C_T(\mathcal{F}) = \min\{p, T\}$. We now show that $\inf_{\mathcal{A}} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) \leq \frac{C_T(\mathcal{F})}{2}$. Consider the learner \mathcal{A} which, given initial state x_0 , checks whether $|x_0| \leq p-1$. If $|x_0| \leq p-1$, the learner \mathcal{A} samples a random sequence $\varepsilon \sim \{-1, 1\}^{p-|x_0|}$ and plays $\varepsilon_t(|x_0| + t)$ for $t \leq p-|x_0|$ and $|x_0| + t$ in all future rounds $t > p-|x_0|$. If $|x_0| > p-1$, the learner \mathcal{A} plays $|x_0| + t$ for all $t \geq 1$.

Let $\{x_t\}_{t=0}^T$ be the trajectory to be observed by the learner. If $|x_0| > p-1$, then observe that

$$\begin{aligned} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) &= \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} - \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f^t(x_0) \neq x_t\} \right] \\ &= \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{|x_0| + t \neq x_t\} - \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{|x_0| + t \neq x_t\} \right] = 0 \end{aligned}$$

On the other hand, if $|x_0| \leq p-1$, we can write the learner's expected loss as

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] = \mathbb{E} \left[\sum_{t=1}^{p-|x_0|} \mathbb{1}\{\varepsilon_t(|x_0| + t) \neq x_t\} \right] + \sum_{t=p-|x_0|+1}^T \mathbb{1}\{|x_0| + t \neq x_t\}.$$

Similarly, we can write the cumulative loss of the competitor term as:

$$\inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f^t(x_0) \neq x_t\} = \inf_{f \in \mathcal{F}} \sum_{t=1}^{p-|x_0|} \mathbb{1}\{f^t(x_0) \neq x_t\} + \sum_{t=p-|x_0|+1}^T \mathbb{1}\{|x_0| + t \neq x_t\}.$$

Let $f_\sigma = \arg \min_{f \in \mathcal{F}} \sum_{t=1}^{p-|x_0|} \mathbb{1}\{f^t(x_0) \neq x_t\}$. Combining both bounds, we get that:

$$\begin{aligned}
 \text{FR}_{\mathcal{A}}(T, \mathcal{F}) &= \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} - \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f^t(x_0) \neq x_t\} \right] \\
 &\leq \mathbb{E} \left[\sum_{t=1}^{p-|x_0|} \mathbb{1}\{\varepsilon_t(|x_0|+t) \neq x_t\} - \mathbb{1}\{f_\sigma^t(x_0) \neq x_t\} \right] \\
 &\leq \mathbb{E} \left[\sum_{t=1}^{p-|x_0|} \mathbb{1}\{\varepsilon_t(|x_0|+t) \neq f_\sigma^t(x_0)\} \right] \\
 &= \mathbb{E} \left[\sum_{t=1}^{p-|x_0|} \mathbb{1}\{\varepsilon_t(|x_0|+t) \neq \sigma_{|x_0|+t-1}(|x_0|+t)\} \right] = \frac{p-|x_0|}{2},
 \end{aligned}$$

where in the last equality we used the fact that $\varepsilon_t \sim \text{Unif}(-1, 1)$. Thus, the expected regret of such an learner \mathcal{A} is at most $\frac{\min\{\max\{p-|x_0|, 0\}, T\}}{2}$. Taking $x_0 = 0$, gives that $\inf_{\mathcal{A}} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) \leq \frac{C_T(\mathcal{F})}{2}$.

H.2. Tightness of upper bound in Theorem 17

Let $\mathcal{X} = \mathbb{Z}$ and fix $p \in \mathbb{N}$. Let $\mathcal{G} = \{(0, s_1, \dots, s_p) : s_1 < s_2 < \dots < s_p \in \mathbb{N}\}$ be the set of all ordered tuples of extended natural numbers with size $p+1$. For any $S \in \mathcal{G}$ and $x \in \mathbb{N} \cup \{0\}$, let $S_x = \max\{s \in S : s \leq x\}$ be the largest element in S smaller than x . Note that $S_0 = 0$ for all $S \in \mathcal{G}$. For every $\sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}}$ and $S \in \mathcal{G}$, consider the evolution function:

$$f_{\sigma, S}(x) = \sigma_{S_{|x|}}(|x|+1).$$

Given any initial state $x_0 \in \mathcal{X}$ and time horizon $T \in \mathbb{N}$, the trajectory $\{f_{\sigma, S}^t(x_0)\}_{t=1}^T$ is some signed version of the sequence $(|x_0|+1, \dots, |x_0|+T)$, where the signs depend on S and σ . More importantly, the trajectory $\{f_{\sigma, S}^t(x_0)\}_{t=1}^T$ switches signs at most $p+1$ times. Finally, consider the evolution class $\mathcal{F} = \{f_{\sigma, S} : \sigma \in \{-1, 1\}^{\mathbb{N} \cup \{0\}}, S \in \mathcal{G}\}$. Since the trajectory of any evolution $f \in \mathcal{F}$ can switch signs at most $p+1$ times, it follows that $C_T(\mathcal{F}) \leq p+1$. Moreover, by considering a trajectory tree with the root node labeled by 0 and the set of evolutions parameterized by the tuple $(0, 1, \dots, p) \in \mathcal{G}$, its not hard to see that $C_T(\mathcal{F}) \geq p+1$. Thus, $C_T(\mathcal{F}) = p+1$.

We now claim that $\inf_{\mathcal{A}} \text{FR}_{\mathcal{A}}(T, \mathcal{F}) \geq \sqrt{\frac{C_T(\mathcal{F})T}{8}}$, which shows that the upper bound in Theorem 17 is tight up to a logarithmic factor in T since $\sup_{x \in \mathcal{X}} |\mathcal{F}(x)| = 2$. Consider $T = k(p+1)$ for some odd $k \in \mathbb{N}$. For $\varepsilon \in \{-1, 1\}^T$, define $\tilde{\varepsilon}_i = \text{sign}\left(\sum_{t=(i-1)k+1}^{ik} \varepsilon_t\right)$ for all $i \in \{1, 2, \dots, p+1\}$. The game proceeds as follows. The adversary samples a string $\varepsilon \in \{-1, 1\}^T$ uniformly at random and constructs the random trajectory $\{x_t\}_{t=0}^T$ where $x_0 = 0$ is the initial state and $x_t = \varepsilon_t t$ for all $t \geq 1$. The adversary then passes $\{x_t\}_{t=0}^T$ to the learner.

Let \mathcal{A} be any randomized learner. Then, for each block $i \in [p+1]$, we have that

$$\mathbb{E} \left[\sum_{t=(i-1)k+1}^{ik} \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} \right] \geq \sum_{t=(i-1)k+1}^{ik} \frac{1}{2} = \frac{k}{2},$$

where the inequality follows from the fact that x_t is chosen uniformly at random between t and $-t$. Let $f_{\sigma,S} \in \mathcal{F}$ be the function in \mathcal{F} such that $S = (0, k, 2k, \dots, (p+1)k)$ and $\sigma_{S_{|x|}} = \tilde{\varepsilon}_i$ for all $(i-1)k \leq |x| \leq ik - 1$ and $i \in \{1, \dots, p+1\}$. For each block $i \in [p+1]$, we have

$$\begin{aligned}
\mathbb{E} \left[\sum_{t=(i-1)k+1}^{ik} \mathbb{1}\{f_{\sigma,S}^t(0) \neq x_t\} \right] &= \mathbb{E} \left[\sum_{t=(i-1)k+1}^{ik} \mathbb{1}\{\sigma_{S_{|t-1|}} t \neq \varepsilon_t t\} \right] \\
&= \mathbb{E} \left[\sum_{t=(i-1)k+1}^{ik} \mathbb{1}\{\tilde{\varepsilon}_i \neq \varepsilon_t\} \right] \\
&= \frac{k}{2} - \frac{1}{2} \mathbb{E} \left[\sum_{t=(i-1)k+1}^{ik} \tilde{\varepsilon}_i \varepsilon_t \right] \\
&= \frac{k}{2} - \frac{1}{2} \mathbb{E} \left[\left| \sum_{t=(i-1)k+1}^{ik} \varepsilon_t \right| \right] \\
&\leq \frac{k}{2} - \sqrt{\frac{k}{8}},
\end{aligned}$$

where the final step follows upon using Khinchine's inequality (Cesa-Bianchi and Lugosi, 2006). Combining these two bounds above, we obtain,

$$\mathbb{E} \left[\sum_{t=(i-1)k+1}^{ik} \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} - \sum_{t=(i-1)k+1}^{ik} \mathbb{1}\{f_{\sigma,S}^t(0) \neq x_t\} \right] \geq \sqrt{\frac{k}{8}}.$$

Summing this inequality over $p+1$ blocks, we obtain

$$\begin{aligned}
\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} - \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f^t(x_0) \neq x_t\} \right] &\geq \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq x_t\} - \sum_{t=1}^T \mathbb{1}\{f_{\sigma,S}^t(x_0) \neq x_t\} \right] \\
&\geq (p+1) \sqrt{\frac{k}{8}} = \sqrt{\frac{C_T(\mathcal{F})T}{8}},
\end{aligned}$$

which completes our proof.

Appendix I. Proof of Theorem 18

Our proof of Theorem 18 is constructive. That is, we provide an evolution function class and show that there exists a deterministic algorithm with a mistake bound of 3 in the realizable setting. As for the agnostic setting, we use the probabilistic method to argue the existence of a hard stream such that every algorithm incurs $T/6$ regret.

Let $\mathcal{X} = \{-1, 1\}^{\mathbb{N}} \times \mathbb{Z}$. For every $\sigma \in \{-1, 1\}^{\mathbb{N}}$, define an evolution function

$$f_{\sigma}((\theta, z)) = (\sigma, \sigma_{|z|+1}(|z|+1)) \mathbb{1}\{|z|+1 = 0 \pmod{3}\} + (\mathbf{1}, \sigma_{|z|+1}(|z|+1)) \mathbb{1}\{|z|+1 \neq 0 \pmod{3}\},$$

where $\theta \in \{-1, 1\}^{\mathbb{N}}, z \in \mathbb{Z}$, and $\mathbf{1} := (1, 1, \dots, 1, 1)$ is the string of all ones. Consider the evolution function class $\mathcal{F} = \{f_\sigma : \sigma \in \{-1, 1\}^{\mathbb{N}}\}$.

To prove (i), it suffices to note that for any initial state (θ_0, z_0) and $f_\sigma \in \mathcal{F}$, the realizable stream $\{f_\sigma^t((\theta_0, z_0))\}_{t=1}^T$ must reveal σ within the first three rounds $t \in \{1, 2, 3\}$. A deterministic algorithm \mathcal{A} that plays arbitrarily in the beginning and $f_\sigma^t((\theta_0, z_0))$ for all $t \geq 4$ makes no more than 3 mistakes.

To prove (ii), consider a random trajectory such that $x_0 = (\mathbf{1}, 0)$ is the initial state and $\{x_t\}_{t=1}^T = \{(\mathbf{1}, \varepsilon_t t)\}_{t=1}^T$, where $\varepsilon_t \sim \text{Uniform}(\{-1, 1\})$. Since the stream is generated uniformly at random, for any algorithm \mathcal{A} , we must have

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}\{\mathcal{A}(x_{<t}) \neq (\mathbf{1}, \varepsilon_t t)\} \right] \geq \frac{T}{2}.$$

Next, let $\varepsilon \in \{-1, 1\}^{\mathbb{N}}$ be any completion of $(\varepsilon_1, \dots, \varepsilon_T)$ and consider the function $f_\varepsilon \in \mathcal{F}$. Note that, for every $t \in [T]$, we have $f_\varepsilon(\mathbf{1}, \varepsilon_{t-1}(t-1)) = (\mathbf{1}, \varepsilon_t t)$ if $t \neq 0 \pmod{3}$ and $f_\varepsilon(\mathbf{1}, \varepsilon_{t-1}(t-1)) = (\varepsilon, \varepsilon_t t) \neq (\mathbf{1}, \varepsilon_t t)$ if $t = 0 \pmod{3}$. Moreover, it is not too hard to see that

$$f_\varepsilon^t((\mathbf{1}, 0)) = (\varepsilon, \varepsilon_t t) \mathbb{1}\{t = 0 \pmod{3}\} + (\mathbf{1}, \varepsilon_t t) \mathbb{1}\{t \neq 0 \pmod{3}\}.$$

Since the functions $f \in \mathcal{F}$ ignore the first argument of the tuple, we have $f_\varepsilon^t((\mathbf{1}, 0)) = f_\varepsilon(f_\varepsilon^{t-1}(\mathbf{1}, 0)) = f_\varepsilon((\mathbf{1}, \varepsilon_{t-1}(t-1)))$. So, using the equality established above, we obtain

$$\begin{aligned} \inf_{f \in \mathcal{F}} \sum_{t=1}^T \mathbb{1}\{f^t(x_0) \neq x_t\} &\leq \sum_{t=1}^T \mathbb{1}\{f_\varepsilon^t(x_0) \neq x_t\} \\ &\leq \sum_{t=1}^T \mathbb{1}\{f_\varepsilon^t((\mathbf{1}, 0)) \neq (\mathbf{1}, \varepsilon_t t)\} \\ &= \sum_{t=1}^T \mathbb{1}\{f_\varepsilon((\mathbf{1}, \varepsilon_{t-1}(t-1))) \neq (\mathbf{1}, \varepsilon_t t)\} \\ &= \sum_{t=1}^T \mathbb{1}\{t = 0 \pmod{3}\} \leq \frac{T}{3}. \end{aligned}$$

Therefore, we have $\text{FR}_{\mathcal{A}}(T, \mathcal{F}) \geq \frac{T}{2} - \frac{T}{3} \geq \frac{T}{6}$.