
ORACLE-EFFICIENT HYBRID ONLINE LEARNING WITH CONSTRAINED ADVERSARIES

Anonymous authors

Paper under double-blind review

ABSTRACT

The Hybrid Online Learning Problem, where features are drawn i.i.d. from an unknown distribution but labels are generated adversarially, is a well-motivated setting positioned between statistical and fully-adversarial online learning. Prior work has presented a dichotomy: algorithms that are statistically-optimal, but computationally intractable (Wu et al., 2023), and algorithms that are computationally-efficient (given an ERM oracle), but statistically-suboptimal (Wu et al., 2024).

This paper takes a significant step towards achieving statistical optimality and computational efficiency *simultaneously* in the Hybrid Learning setting. To do so, we consider a structured setting, where the Adversary is constrained to pick labels from an expressive, but fixed, class of functions \mathcal{R} . Our main result is a new learning algorithm, which runs efficiently given an ERM oracle and obtains regret scaling with the Rademacher complexity of a class derived from the Learner’s hypothesis class \mathcal{H} and the Adversary’s label class \mathcal{R} . As a key corollary, we give an oracle-efficient algorithm for computing equilibria in stochastic zero-sum games when action sets may be high-dimensional but the payoff function exhibits a type of low-dimensional structure. Technically, we develop a number of novel tools for the design and analysis of our learning algorithm, including a novel Frank-Wolfe reduction with “truncated entropy regularizer” and a new tail bound for sums of “hybrid” martingale difference sequences.

1 INTRODUCTION

Online learning is a fundamental paradigm in machine learning, where an algorithm learns sequentially from a stream of data, making predictions and updating its model in real-time. Within the broad landscape of online learning, different assumptions can be made about how the data is generated. Two prominent extremes are the statistical setting, where data is drawn independently and identically distributed (i.i.d.) from a fixed, unknown distribution, and the fully-adversarial setting, where data is chosen by an adaptive adversary aiming to maximize the learner’s error. While these are well-studied, the guarantees a learner can obtain can vary starkly between the two extremes. For example, the problem of learning thresholds from a small number of samples is straightforward in the statistical setting, but impossible in the fully-adversarial setting (Littlestone, 1988).

The Hybrid Online Learning Problem (Lazaric & Munos, 2009) has emerged as a compelling middle ground, capturing aspects of both statistical and adversarial scenarios. In this model, features are assumed to be drawn i.i.d. from an *unknown* distribution, much like in the statistical setting. The corresponding labels, however, are determined by a potentially malicious adversary. On a practical level, this hybrid model captures real-world situations where typical instances follow statistical patterns, but the labels associated with these instances are influenced by strategic actors, system dynamics, or other worst-case forces. Theoretically, the model serves as an important frontier for exploring the limits of efficient online learning with provable guarantees.

The current state of research in Hybrid Online Learning hints at a computational-statistical divide. Algorithms that achieve statistically optimal performance (Lazaric & Munos, 2009; Wu et al., 2023) are typically computationally intractable with time and space complexity both scaling linearly in the size of the learner’s hypothesis class. On the other hand, algorithms that are computationally efficient typically assume the learner has full knowledge of or unlimited sample access to the underlying feature distribution (Rakhlin et al., 2011; Haghtalab et al., 2024; Block et al., 2022), or they achieve suboptimal regret (Wu et al., 2024).

This chapter takes a crucial step towards bridging this gap, aiming to develop learning algorithms that are both statistically optimal and computationally efficient in the Hybrid Learning setting. To make progress on this challenging goal, we focus on a structured version of the problem. Specifically, we introduce a constraint on the adversary, assuming that the adversarial labels must be chosen from an expressive, but fixed, class of functions \mathcal{R} . This structural assumption allows for a more fine-grained analysis and algorithm design. Our main contribution is the development of a novel oracle-efficient learning algorithm for this structured setting.

1.1 PROBLEM FORMULATION

We consider the following Hybrid Online Learning Problem: Let \mathcal{X} be the feature space and $\mathcal{H} \subseteq [0, 1]^{\mathcal{X}}$ and $\mathcal{R} \subseteq [0, 1]^{\mathcal{X}}$ be the learner's hypothesis class and the adversary's constrained label function class, respectively, which are known to the learner. We assume the learner's loss function $\ell : [0, 1] \times [0, 1] \rightarrow \mathbb{R}$ is convex and L -Lipschitz with respect to its first argument for some constant $L > 0$ and measurable in the second argument. The learning process proceeds over T rounds. Nature commits to a fixed, unknown distribution \mathcal{D} over \mathcal{X} . In each round $t = 1, \dots, T$:

1. The learner selects a hypothesis h_t .
2. The adversary, with knowledge of the learner's strategy but not the future feature x_t , selects a function r_t from the adversary's label function class \mathcal{R} .
3. Nature samples a feature x_t i.i.d. from \mathcal{D} . The learner incurs loss $\ell(h_t(x_t), r_t(x_t))$. The pair (x_t, r_t) is revealed to the learner.

The learner's goal is to minimize its cumulative loss. The learner's strategy at time t is a function of the history $(x_1, r_1), \dots, (x_{t-1}, r_{t-1})$. We evaluate the performance of a learner by its regret with respect to the best fixed hypothesis in \mathcal{H} in hindsight. The regret over T rounds is defined as:

$$\text{Reg}(T) = \mathbb{E}_{x_1, \dots, x_T \sim \mathcal{D}} \left[\sum_{t=1}^T \ell(h_t(x_t), r_t(x_t)) - \min_{h \in \mathcal{H}} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) \right]$$

The expectation is taken over the random draws of x_1, \dots, x_T from the distribution \mathcal{D} . Our goal is to design an oracle-efficient learner that minimizes this regret. In achieving this goal, we will also consider an in-expectation regret guarantee, i.e., $\sum_{t=1}^T \mathbb{E}_{x \sim \mathcal{D}} [\ell(h_t(x), r_t(x))] - \min_{h \in \mathcal{H}} \sum_{t=1}^T \mathbb{E}_{x \sim \mathcal{D}} [\ell(h(x), r_t(x))]$

1.2 OVERVIEW OF RESULTS

Our main contribution is the development of an oracle-efficient learning algorithm for the Hybrid Online Learning Problem in a structured setting where the adversary's labeling function is constrained to a class \mathcal{R} . Our algorithm achieves a statistically near-optimal (up to the dependence on the adversary's constraint set \mathcal{R}) regret bound while being computationally efficient given access to a linear optimization oracle over the hypothesis class \mathcal{H} .

A key quantity characterizing the statistical complexity of function classes is the Rademacher complexity (see [Section 1.4](#) for definition). In statistical learning theory, the Rademacher complexity of a hypothesis class \mathcal{H} provides a tight characterization of the generalization error and hence the statistical error rate ([Mohri et al., 2012](#)). Our main result provides a high-probability regret bound for our hybrid learner in terms of Rademacher complexity of the function classes.

Theorem 1.1. Let $\mathcal{H} \subseteq [0, 1]^{\mathcal{X}}$ be a class of hypothesis functions and let $\mathcal{R} \subseteq [0, 1]^{\mathcal{X}}$ be a class of labeling functions. Let $\ell : [0, 1] \times [0, 1] \rightarrow \mathbb{R}$ be a convex, L -Lipschitz loss function in its first argument. There exists an online algorithm that outputs a sequence of hypothesis functions h_1, \dots, h_T such that with probability at least $1 - \delta$ over the draw of $x_1, \dots, x_T \sim \mathcal{D}$, the following bound on the cumulative loss holds:

$$\sum_{t=1}^T \ell(h_t(x_t), r_t(x_t)) - \min_{h \in \mathcal{H}} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) \leq O\left(T \text{rad}_T(\ell \circ \mathcal{H} \times \mathcal{R})^1 + L T \text{rad}_T(\mathcal{H}) + L \sqrt{T \log(T/\delta)}\right)$$

where $\ell \circ \mathcal{H} \times \mathcal{R}$ denotes the class of functions $\{x \mapsto \ell(h(x), r(x)) \mid h \in \mathcal{H}, r \in \mathcal{R}\}$. The algorithm runs in $O(T^2)$ time per round and makes $O(T^2)$ calls to a *linear optimization oracle* for \mathcal{H} throughout T rounds.

¹As defined in [Section 1.4](#), $\text{rad}_T(\mathcal{F})$ is at most 1 for any \mathcal{F} and $O(\sqrt{d/T})$ for binary classes of VC dimension d

Theorem 1.1 provides a strong statistical guarantee, showing that the regret scales with the Rademacher complexity of the composite class $\ell \circ \mathcal{H} \times \mathcal{R}$. This term captures the complexity introduced by both the learner’s hypothesis class \mathcal{H} and the adversary’s class \mathcal{R} . The bound is near-optimal up to the dependence on the Rademacher complexity of the adversary’s label class \mathcal{R} and a logarithmic factor in T . This follows from the fact that Hybrid Learning is at least as hard as statistically learning and this implies a lower bound of $L\text{rad}_T(\mathcal{H}) + L\sqrt{T \log(1/\delta)}$ on the regret rate of hybrid learning (Mohri et al., 2012). Note that if the hypothesis class \mathcal{H} is a binary valued class of VC dimension d and the composite class $\ell \circ \mathcal{H} \times \mathcal{R}$ is also a binary valued class of dimension d^* , then the regret guarantee of Theorem 1.1 can be upper bounded by $O(\sqrt{Td^*} + L\sqrt{Td} + L\sqrt{T \log(T/\delta)})$ (see Section 1.4).

Our Hybrid Online Learning framework and our hybrid learner can be applied to the area of game theory and optimization, specifically for finding approximate solutions to stochastic saddle-point problems, or equivalently, finding approximate equilibria of stochastic zero-sum games. While it is known that oracle-efficient algorithms for finding equilibria of arbitrary zero-sum games do not exist in general (see Theorem 4 of Hazan & Koren (2016)), our results enable designing oracle-efficient algorithms whenever the game’s payoff function factorizes as the composition of a bivariate convex-concave Lipschitz-continuous function with (stochastic) scalar-valued functions of each player’s action. Intuitively, any such factorization of the payoff function gives the game a low-dimensional structure that is useful for efficient equilibrium computation. However, since the players’ action sets themselves remain (potentially) high-dimensional, to take advantage of this low-dimensional structure in an oracle-efficient way one must design algorithms for a player to learn an approximate best-response to their opponent’s adaptively-chosen action sequence in the stochastic zero-sum game, leading naturally to a Hybrid Online Learning problem.

Corollary 1.2. Let \mathcal{X} be a domain space and \mathcal{D} be a distribution over \mathcal{X} . Let $\mathcal{H}, \mathcal{R} \subseteq [0, 1]^{\mathcal{X}}$ be classes of functions (assumed to be closed under convex combinations) and $u : [0, 1] \times [0, 1] \rightarrow \mathbb{R}$ be a convex-concave payoff function that is L -Lipschitz in its first parameter. Consider the saddle-point optimization problem

$$\min_{h \in \mathcal{H}} \max_{r \in \mathcal{R}} \mathbb{E}_{x \sim \mathcal{D}}[u(h(x), r(x))]$$

Given m samples from \mathcal{D} and access to best-response oracles for \mathcal{H} and \mathcal{R} , our online learning algorithm can be used to find an $\epsilon(m)$ -approximate saddle point solution (h^*, r^*) in polynomial time in m and the complexities of \mathcal{H} and \mathcal{R} . The approximation guarantee is $\epsilon(m) = \text{rad}_m(\mathcal{F}) + O(L\sqrt{\log m/m})$, where $\mathcal{F} = \{f : f(x) = u(h(x), r(x)) \mid h \in \mathcal{H}, r \in \mathcal{R}\}$. Note that $\text{rad}_m(\mathcal{F}) \rightarrow 0$ is necessary for uniform convergence of the payoff matrix.

Finally, along the way to establishing our main result, we prove a general uniform convergence bound that may be of independent interest. This bound addresses the challenge of concentration for function classes evaluated on i.i.d. data where the functions themselves are chosen adaptively based on the previous data samples:

Proposition 1.3. Let \mathcal{H} be a class of hypothesis functions and ℓ be a loss function that is L -Lipschitz in the first parameter. Let x_1, x_2, \dots, x_T be a sequence of i.i.d samples from a fixed distribution \mathcal{D} . Let $r_1, r_2, \dots, r_T \in [0, 1]^{\mathcal{X}}$ be a sequence of functions where r_t depends only on x_1, \dots, x_{t-1} (and potentially prior adversarial choices). The following holds with probability at least $1 - \delta$ over the draw of x_1, \dots, x_T , for all $h \in \mathcal{H}$:

$$\left| \frac{1}{T} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) - \frac{1}{T} \sum_{t=1}^T \mathbb{E}_{x \sim \mathcal{D}}[\ell(h(x), r_t(x))] \right| \leq O\left(L \cdot \text{rad}_T(\mathcal{H}) + L\sqrt{\frac{\log(T/\delta)}{T}}\right)$$

This result provides a uniform convergence bound that effectively handles the data-dependent nature of the sequence r_1, \dots, r_T . The sequence $\ell(h(x_t), r_t(x_t)) - \mathbb{E}_{\mathcal{D}}[\ell(h(x), r_t(x))]$ is a martingale difference sequence since x_t is sampled after the choice of r_t is made. Applying Azuma-Hoeffding together with a union bound over the class \mathcal{H} would only work for finite classes and would lead to a suboptimal bound of $\log|\mathcal{H}|$. We instead prove this lemma by employing a symmetrization technique and the application of a bound based on the distribution-dependent sequential Rademacher complexity, a measure introduced by Rakhlin et al. (2011). The L -Lipschitzness of the loss function with respect to its first parameter is key and ensures the bound depends only on the complexity of the hypothesis class \mathcal{H} and the Lipschitz constant L , rather than the complexity of the r_t sequence itself. We defer the full proof to Appendix A.2. We use Proposition 1.3 to obtain the high probability guarantee in Theorem 1.1 on the sampled sequence.

1.3 TECHNICAL OVERVIEW

Our technical approach begins by considering the in-expectation regret objective: to guarantee a bound on $\sum_{t=1}^T \mathbb{E}_{\mathcal{D}}[\ell(h_t(x), r_t(x))] - \min_{h \in \mathcal{H}} \sum_{t=1}^T \mathbb{E}_{\mathcal{D}}[\ell(h(x), r_t(x))]$. We note that achieving a bound on this quantity is a weaker benchmark compared to the standard regret definition (which is measured against the sum of losses on observed samples).

A key limitation in this setting is that we do not have direct access to the distribution \mathcal{D} . To build intuition, suppose for a moment that we had access to m i.i.d. samples, $S = \{s_1, \dots, s_m\}$, from the distribution \mathcal{D} *a priori*. We make the crucial observation that m samples are sufficient to guarantee uniform convergence for the combined function class $\mathcal{F} = \{f : f(x) = \ell(h(x), r(x)) \mid h \in \mathcal{H}, r \in \mathcal{R}\}$ at a rate characterized by $\text{rad}_m(\mathcal{F})$. Therefore, if we had these samples upfront, the problem could be formulated as an online learning over \mathcal{H} . In each round t , given r_t , the loss for a hypothesis h would be the empirical average loss over the sample set S : $\mathbb{E}_S[\ell(h(x), r_t(x))] = \frac{1}{m} \sum_{i=1}^m \ell(h(s_i), r_t(s_i))$. Due to the uniform convergence property, for any $h \in \mathcal{H}$ and adaptive $r_t \in \mathcal{R}$, the empirical average $\mathbb{E}_S[\ell(h(x), r_t(x))]$ would be a good approximation of the true expectation $\mathbb{E}_{\mathcal{D}}[\ell(h(x), r_t(x))]$. Since the loss function only depends on the m samples, this online learning problem is essentially an Online Convex Optimization problem with action set $(h(s_1)/m, \dots, h(s_m)/m) \in [0, 1/m]^m$ for each $h \in \mathcal{H}$ and where the loss vector in each round corresponds to the empirical losses $(\ell(h(s_1), r_t(s_1)), \dots, \ell(h(s_m), r_t(s_m))) \in [0, 1]^m$. Since the action set — the projection of \mathcal{H} on the m samples — is a subset of $[0, 1/m]^m$ which is a subset of the m -dimensional simplex, then applying Follow the Regularized Leader (FTRL) achieves regret of $\sqrt{T \log m}$. Unfortunately, a naive application of FTRL will return actions on the m dimensional simplex which may not correspond to any hypothesis in the class \mathcal{H} . To solve this problem, we introduce a Frank-Wolfe reduction to the linear optimization oracle in [Section 3](#).

However, in the Hybrid Online Learning problem, we do not have the samples upfront. Instead, we observe samples sequentially as part of the online process itself. We thus use the dataset accumulated up to round $t - 1$, $S_t = \{x_1, \dots, x_{t-1}\}$, to define an empirical loss at round t : $\mathbb{E}_{S_t}[\ell(h_t(x), r_t(x))] = \frac{1}{t-1} \sum_{i=1}^{t-1} \ell(h_t(x_i), r_t(x_i))$. Unfortunately, due to the dynamically changing structure of this empirical loss function (as the dataset D_t grows with t), this problem cannot be directly modeled as an Online Convex Optimization problem with a fixed vector space and a sequence of linear loss functions.

Despite this challenge posed by the adaptive structure of the empirical loss, we are still able to make progress by constructing an adaptive sequence of entropy regularizers. In a departure from standard FTRL analysis, the regularizers we employ are not strongly convex over the entire ambient vector space (which is of dimension T). This is because we never observe the “full vector” of losses or learner’s actions on all T samples at any given time $t < T$. Nevertheless, we bypass this difficulty by demonstrating that our adaptive entropy regularizers are strongly convex on the *relevant coordinates* (the first $t - 1$ dimensions) at step t . This careful construction allows us to achieve a favorable bound of $O(\sqrt{T \log T})$ with respect to our in-expectation regret benchmark (the sum of expected losses).

Finally, the remaining step is to transition from the weaker benchmark (regret against the sum of expected losses over \mathcal{D}) to the stronger benchmark (regret against the sum of actual losses incurred on the observed samples x_1, \dots, x_T). This is where uniform convergence arguments shown in [Proposition 1.3](#) come into play, allowing us to convert the bound on the weaker benchmark into the desired bound on the standard regret definition.

1.4 TECHNICAL PRELIMINARIES

Complexity Measures For a function class $\mathcal{F} \subseteq \mathbb{R}^X$ and samples $x_1, \dots, x_T \in X$, the empirical Rademacher complexity is $\widehat{\text{rad}}_T(\{f|_{x_1, \dots, x_T} : f \in \mathcal{F}\}) = \mathbb{E}_{\sigma} \left[\sup_{f \in \mathcal{F}} \frac{1}{T} \sum_{t=1}^T \sigma_t f(x_t) \right]$, where $\sigma_1, \dots, \sigma_T$ are independent random variables uniformly drawn from $\{\pm 1\}$. The Rademacher complexity at horizon T with respect to distribution \mathcal{D} is $\text{rad}_T(\mathcal{F}) = \mathbb{E}_{x_1, \dots, x_T \sim \mathcal{D}} [\widehat{\text{rad}}_T(\{f|_{x_1, \dots, x_T} : f \in \mathcal{F}\})]$. It is well known that for binary classes, the Rademacher complexity is tightly controlled by the VC dimension: it is both upper and lower bounded (up to logarithmic factors) by $\sqrt{\text{VCdim}(\mathcal{F})/T}$ ([Bartlett & Mendelson, 2003](#); [Mohri et al., 2012](#)). A similar result holds for real-valued classes and the fat-shattering dimension ([Mohri et al., 2012](#)).

We additionally define the composite function class: $\ell \circ \mathcal{H} \times \mathcal{R} = \{x \mapsto \ell(h(x), r(x)) \mid h \in \mathcal{H}, r \in \mathcal{R}\}$.

Linear Optimization Oracle Our algorithm’s computational efficiency is measured in terms of calls to a Linear Optimization Oracle for the hypothesis class \mathcal{H} . A Linear Optimization Oracle for \mathcal{H} is an algorithm that, given a set of points $S = \{s_1, \dots, s_m\} \subset \mathcal{X}$ and a set of weights for those points $V = \{v_1, \dots, v_m\} \subset \mathbb{R}$, returns a hypothesis $h^* \in \mathcal{H}$ that minimizes $\sum_{i=1}^m v_i h(s_i)$ over \mathcal{H} . In our context, the set S will typically be the set of observed samples x_1, \dots, x_{t-1} at round t .

1.5 COMPARISON TO PRIOR WORK

The study of Hybrid Online Learning with an unknown i.i.d source was initiated by [Lazaric & Munos \(2009\)](#), who showed $O(\sqrt{dT \log T})$ regret for hypothesis classes with finite VC dimension d and absolute loss. [Wu et al. \(2023\)](#) extended this to real-valued functions and general convex losses, achieving statistically optimal expected regret for VC classes. Their algorithms rely on constructing a stochastic cover of the hypothesis class, which is computationally intractable for some classes.

The first oracle-efficient algorithm for this setting was presented by [Wu et al. \(2024\)](#). They achieve $\tilde{O}(d^{1/2} T^{3/4})$ regret for finite VC classes oracle-efficiently. However, this rate is statistically suboptimal. Their approach uses a relaxation-based Follow the Perturbed Leader method, distinct from ours. This work aims to close this gap by providing a learning algorithm that is both statistically optimal and computationally efficient in a structured setting with a constrained adversary class \mathcal{R} .

Hybrid online learning can be viewed as a form of Smoothed Online Learning. However, most existing work in smoothed online learning ([Haghtalab et al., 2020; 2024; Block et al., 2022](#)) assumes knowledge of or sampling access to the underlying stochastic source. This key difference highlights the specific challenge addressed in our work: handling adversarial labels when the i.i.d. feature distribution is entirely unknown. Relatedly, [Rakhlin et al. \(2011\)](#) studied a distribution-dependent online learning problem where Nature adaptively selects sampling distributions. Again, their results primarily apply to the “distribution non-blind” case where the distribution is known (see [Rakhlin et al. \(2011\)](#), Section 7), unlike our “distribution blind” setting with an unknown i.i.d. source.

Our work conceptually relates to the comparative learning setting introduced by ([Hu & Peale, 2023](#)), where labeling functions are also restricted to a known class. However, a critical distinction is that their models, both offline and online learning settings, do not involve adaptive labeling functions. Consequently, their results have no direct bearing on our model. Nevertheless, exploring connections between the two works is an enticing direction for future work. It would be particularly interesting if the sample complexity in our setting can be characterized by the “mutual VC dimension” defined by [Hu & Peale \(2023\)](#), or if our algorithms can be adapted to yield oracle-efficient and statistically-optimal comparative learning algorithms in their setting.

2 ORACLE-EFFICIENT HYBRID LEARNING

In [Section 2.1](#), we show a hybrid learner that provides the in-expectation guarantee in [Section 1.4](#). In [Section 2.2](#), we prove our main result ([Theorem 1.1](#)).

2.1 IN-EXPECTATION REGRET GUARANTEE USING TRUNCATED ENTROPY REGULARIZATION

This subsection presents and analyzes an algorithm for hybrid learning that makes use of a subroutine called an *entropy-regularized ℓ -ERM oracle* over \mathcal{H} , defined as follows.

Definition 2.1. An entropy-regularized ℓ -ERM oracle is initialized with a class of functions $\mathcal{H} : \mathcal{X} \rightarrow [0, 1]$. The oracle takes, as input, a subset $S \subset \mathcal{X}$ of features, a set of triples $(x_1, y_1, w_1), \dots, (x_m, y_m, w_m) \in S \times \mathbb{R} \times \mathbb{R}$ and parameters η, ϵ . It outputs an element h in the convex hull of \mathcal{H} , such that h minimizes (within ϵ) the function $\sum_{i=1}^m w_i \ell(h(x_i), y_i) + \frac{1}{\eta} \sum_{s \in S} h(s) \log(h(s) + 1)$.

We use $\log(h(x_i) + 1)$ rather than $\log h(x_i)$ in the regularizer to ensure the argument to the log is well-defined on $[0, 1]$ but more importantly, $\log(a + 1)$ is strongly convex on the entire interval $[0, 1]$ with a uniform positive parameter. In [Section 3](#) below, we show how to use the Frank-Wolfe method to implement an ϵ -approximate regularized ℓ -ERM oracle using polynomial number of calls to a linear optimization oracle.

Theorem 2.1. Let $\mathcal{H} \subseteq [0, 1]^{\mathcal{X}}$ be a class of hypothesis functions and let $\mathcal{R} \subseteq [0, 1]^{\mathcal{X}}$ be a class of labeling functions. Let ℓ be a loss function that is convex, L -Lipschitz in the first parameter. Given an

entropy-regularized ℓ -ERM oracle for \mathcal{H} , Algorithm 1 outputs a sequence of hypothesis functions h_1, \dots, h_T such that with probability at least $1 - \delta$,

$$\sum_{t=1}^T \mathbb{E}[\ell(h_t(x), r_t(x))] \leq \min_{h \in \mathcal{H}} \sum_{t=1}^T \mathbb{E}[\ell(h(x), r_t(x))] + T \cdot \text{rad}_T(\ell \circ \mathcal{H} \times \mathcal{R}) + O(L \sqrt{T \log T})$$

The algorithm runs in time $O(T^2)$ per timestep and makes T calls the entropy-regularized ℓ -ERM oracle for \mathcal{H} .

Overview of Algorithm 1 The algorithm implements a hybrid learner using the Follow The Regularized Leader (FTRL) approach over the class \mathcal{H} . We define a surrogate loss for each timestep based on the empirical average of the actual loss with respect to the adversary's choice r_t on the samples x_1, \dots, x_{t-1} seen so far. Then we choose the approximate minimizer of the cumulative surrogate loss and an entropy regularizer that only depends on x_1, \dots, x_{t-1} .

Concretely, at each timestep t , the algorithm outputs a predictor $h_t \in \text{conv}(\mathcal{H})$. After observing the sample x_t and receiving the adversary's labeling function r_t , the algorithm prepares the input dataset for the entropy-regularized ERM oracle to compute the next predictor h_{t+1} . The dataset provided to the oracle at step t consists of triples (x_i, y_i, w_i) derived from the samples $\{x_1, \dots, x_{t-1}\}$ and the past adversarial functions $\{r_2, \dots, r_t\}$. Specifically, for each pair of $s \in \{2, \dots, t\}$ and $i \in \{1, \dots, s-1\}$, the oracle receives a triple $(x_i, r_s(x_i), \frac{1}{s-1})$. The oracle finds an ε -approximate minimizer of this cumulative regularized empirical loss, and this minimizer becomes the predictor h_{t+1} for the next round.

We introduce the following notation: let $v(h) = (h(x_1), \dots, h(x_T))$ and $\mathcal{V} = \text{conv}(\{v(h) \mid h \in \mathcal{H}\})$. We define the surrogate loss function at time $t \geq 2$ as $\tilde{\ell}_t(v) = \frac{1}{t-1} \sum_{s=1}^{t-1} \ell(v^{(s)}, r_t(x_s))$, and $\tilde{\ell}_1(v) = 0$ (where $v^{(s)}$ refers to the s -th coordinate of the vector v). Define the regularizer at time $t > 1$ as $\psi_t(v) = \frac{1}{\eta} \sum_{s=1}^{t-1} v^{(s)} \log(v^{(s)} + 1)$, and $\psi_1(v) = 0$. The algorithm at step t outputs h_t (corresponding to \bar{v}_t) where \bar{v}_t is an ε -approximate minimizer of $F_t(v) = \sum_{s=1}^{t-1} \tilde{\ell}_s(v) + \psi_t(v)$ for $t > 1$.

Algorithm 1 Hybrid Learner using Exponentiated Gradient

Require: Sequence of i.i.d. samples $\{x_t\}_{t=1}^T \sim \mathcal{D}^T$, time horizon T , failure probability δ , approximation parameter ε for the oracle

Ensure: Sequence of predictors $\{h_t\}_{t=1}^T$ where each $h_t \in \text{conv}(\mathcal{H})$

- 1: Set $\eta \leftarrow \sqrt{T/L^2 \log T}$, $\varepsilon = L \log^{3/2} T / \sqrt{T}$
- 2: Initialize h_1 to some arbitrary hypothesis in \mathcal{H}
- 3: **for** $t = 1$ to T **do**
- 4: Output h_t , Observe x_t .
- 5: Receive adversary function $r_t \in \mathcal{R}$.
- 6: Construct the set of triples $\mathcal{S}_t = \bigcup_{s=2}^t \{(x_i, r_s(x_i), \frac{1}{s-1}) \mid i \in \{1, \dots, s-1\}\}$. (For $t = 1$, $\mathcal{S}_1 = \emptyset$).
- 7: Obtain next predictor $h_{t+1} \in \text{conv}(\mathcal{H})$ by calling the entropy-regularized ERM oracle (in Algorithm 2) with input dataset \mathcal{S}_t and feature set $\{x_1, \dots, x_t\}$:

$$h_{t+1} \leftarrow \arg \min_{h \in \text{conv}(\mathcal{H})}^{\varepsilon} \left\{ \sum_{(x,y,w) \in \mathcal{S}_t} w \ell(h(x), y) + \frac{1}{\eta} \sum_{s=1}^t h(x_s) \log(h(x_s) + 1) \right\}.$$

- 8: **return:** Sequence of predictors $\{h_t\}_{t=1}^T$
-

Lemma 2.2 (Approximate FTRL for Hybrid Learning). For $\eta, \varepsilon > 0$, the empirical regret of Algorithm 1 is bounded by

$$\sum_{t=1}^T \tilde{\ell}_t(\bar{v}_t) - \min_{u \in \mathcal{V}} \tilde{\ell}_t(u) \leq \frac{T \log 2}{\eta} + \frac{4\eta L^2 \log T}{3} + 5L \sqrt{\eta \varepsilon T}.$$

To prove this lemma, we first bound the regret of playing the exact minimizers of F_t . Then we appeal to the strong convexity of F_t and Lipschitzness of ℓ to bound the loss of playing the approximate minimizers. We view this as an OCO problem with ambient vector space $\mathcal{V} \subset [0, 1]^T$ and convex

loss vectors $\tilde{\ell}_t$ with an adaptive sequence of regularizers ψ_t . We adapt the analysis of FTRL to deal with the fact that the algorithm never observes the full vector $v \in \mathcal{V}$ and the regularizers are not strongly convex with respect to the ℓ_1 -norm of the full ambient space. However, the loss functions $\tilde{\ell}_t$ and the regularizer ψ_t only depend on the first t coordinates, which means its gradients are zero for coordinates $s > t$. As a result, the ψ_t is strongly convex w.r.t the ℓ_1 norm of the first t coordinates and this suffices for the proof. The full proof can be found in [Appendix A.1](#).

Now we present a uniform convergence result necessary for relating the average loss on the samples seen so far to the expected loss under the true distribution. The full proof can be found in [Appendix A.1](#).

Lemma 2.3. Let $\mathcal{F} \subset [0, 1]^X$ be a class of functions. Let x_1, \dots, x_T be a sequence of samples drawn i.i.d from a fixed distribution \mathcal{D} . With probability at least $1 - \delta$, for all $t \in [T]$, $f \in \mathcal{F}$,

$$\frac{1}{t} \sum_{s=1}^t f(x_s) - \mathbb{E}_{x \sim \mathcal{D}}[f(x)] \leq 2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\log(2T/\delta)}{t}}$$

Proof of Theorem 2.1. By [Lemma 2.2](#), the empirical regret of the sequence $\bar{v}_2, \dots, \bar{v}_T$ (using $\tilde{\ell}_2, \dots, \tilde{\ell}_T$) with respect to any $u \in \mathcal{V}$ is bounded by: $\sum_{t=2}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(u)) \leq O(L\sqrt{T \log T}) + O(L\sqrt{\eta \varepsilon T})$. Let $h \in \mathcal{H}$ be arbitrary, and $u = v(h)$. Since $\bar{v}_t = v(h_t)$,

$$\sum_{t=2}^T \left(\frac{1}{t-1} \sum_{s=1}^{t-1} \ell(h_t(x_s), r_t(x_s)) - \frac{1}{t-1} \sum_{s=1}^{t-1} \ell(h(x_s), r_t(x_s)) \right) \leq O(L\sqrt{T \log T}) + O(L\sqrt{\eta \varepsilon T}).$$

Applying [Lemma 2.3](#) to the function class $F = \{x \rightarrow \ell(h(x), r(x)) \mid \forall h \in \mathcal{H}, r \in \mathcal{R}\}$, we have that, with probability at least $1 - \delta$, for all $t \geq 2$ and $h \in \mathcal{H}$,

$$\left| \mathbb{E}_{x \sim \mathcal{D}}[\ell(h(x), r_t(x))] - \frac{1}{t-1} \sum_{s=1}^{t-1} \ell(h(x_s), r_t(x_s)) \right| \leq 2\text{rad}_{t-1}(\ell \circ \mathcal{H} \times \mathcal{R}) + \sqrt{\frac{\log(2T/\delta)}{t-1}}$$

Plugging back in to the regret guarantee, we obtain that with probability at least $1 - \delta$, for all $h \in \mathcal{H}$,

$$\sum_{t=2}^T \mathbb{E}_{x \sim \mathcal{D}}[\ell(h_t(x), r_t(x))] - \mathbb{E}_{x \sim \mathcal{D}}[\ell(h(x), r_t(x))] \tag{1}$$

$$\leq \sum_{t=2}^T 2\text{rad}_{t-1}(\ell \circ \mathcal{H} \times \mathcal{R}) + \sum_{t=2}^T \sqrt{\frac{\log(2T/\delta)}{t-1}} + O(L\sqrt{T \log T}) + O(L\sqrt{\eta \varepsilon T}) \tag{2}$$

$$\leq O(T \cdot \text{rad}_T(\ell \circ \mathcal{H} \times \mathcal{R})) + O(L\sqrt{T \log T}) + O(L\sqrt{\eta \varepsilon T}) \tag{3}$$

Including the $t = 1$ term, minimizing over $h \in \mathcal{H}$ and setting $\eta = \sqrt{T/L^2 \log T}$, $\varepsilon = L \log^{3/2} T / \sqrt{T}$:

$$\sum_{t=1}^T \mathbb{E}[\ell(h_t(x), r_t(x))] \leq \min_{h \in \mathcal{H}} \sum_{t=1}^T \mathbb{E}[\ell(h(x), r_t(x))] + O(T \cdot \text{rad}_T(\ell \circ \mathcal{H} \times \mathcal{R})) + O(L\sqrt{T \log T}).$$

The runtime analysis is dominated by constructing the set of samples S_t to be sent to the regularized ERM oracle. But since $|S_t| < t^2$, the runtime of the algorithm is $O(T^2)$ per timestep. \square

2.2 PROOF OF [THEOREM 1.1](#)

Proof. We decompose the quantity to bound: $\sum_{t=1}^T \ell(h_t(x_t), r_t(x_t)) - \min_{h \in \mathcal{H}} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) = A + B + C$ where

$$A = \sum_{t=1}^T (\ell(h_t(x_t), r_t(x_t)) - \mathbb{E}_{\mathcal{D}}[\ell(h_t(x), r_t(x))])$$

and

$$B = \sum_{t=1}^T \mathbb{E}_{\mathcal{D}}[\ell(h(x), r_t(x))] - \min_{h \in \mathcal{H}} \sum_{t=1}^T \mathbb{E}_{\mathcal{D}}[\ell(h(x), r_t(x))]$$

and

$$C = \min_{h \in \mathcal{H}} \sum_{t=1}^T \mathbb{E}_{\mathcal{D}}[\ell(h(x_t), r_t(x_t))] - \min_{h \in \mathcal{H}} \sum_{t=1}^T \ell(h(x_t), r_t(x_t))$$

. We bound each term with high probability and allocate a $\delta/3$ failure probability to each.

Term A is a sum of martingale differences, as $Z_t = \ell(h_t(x_t), r_t(x_t)) - \mathbb{E}_{\mathcal{D}}[\ell(h_t(x), r_t(x))]$ satisfies $\mathbb{E}[Z_t | \mathcal{F}_{t-1}] = 0$. Since ℓ is L -Lipschitz over the interval $[0, 1]$, $|Z_t| \leq 2L$. By the Azuma-Hoeffding inequality, with probability at least $1 - \delta/3$: $A \leq \sqrt{2 \sum_{t=1}^T L^2 \log(1/(\delta/3))} = L \sqrt{2T \log(3/\delta)} = O(L \sqrt{T \log(1/\delta)})$

Term B is the in-expectation regret guarantee. By [Theorem 2.1](#), the algorithm guarantees: $B \leq T \cdot \text{rad}_T(\ell \circ \mathcal{H} \times \mathcal{R}) + O(L \sqrt{T \log T})$ with probability at least $1 - \delta/3$.

Term C is the generalization gap for the best hypothesis. By [Proposition 1.3](#), for all $h \in \mathcal{H}$, the difference between empirical and expected sums is bounded uniformly: $|\sum \ell(h, x_t, r_t) - \sum \mathbb{E}_{\mathcal{D}}[\ell(h, x, r_t)]| \leq L \cdot T \cdot \text{rad}_T(\mathcal{H}) + O(\sqrt{T \log(T/\delta)})$ with probability at least $1 - \delta/3$. Using this uniform bound, we get $C \leq L \cdot T \cdot \text{rad}_T(\mathcal{H}) + O(\sqrt{T \log(T/\delta)})$.

Summing the bounds for A , B , and C , using a union bound, we obtain that with probability at least $1 - \delta$:

$$\begin{aligned} & \sum_{t=1}^T \ell(h_t(x_t), r_t(x_t)) - \min_{h \in \mathcal{H}} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) \\ & \leq T \cdot \text{rad}_T(\ell \circ \mathcal{H} \times \mathcal{R}) + L \cdot T \cdot \text{rad}_T(\mathcal{H}) + O(L \sqrt{T \log(T/\delta)}) \end{aligned}$$

This proves the regret bound. The computational efficiency follows from [Theorem 2.1](#)'s use of an entropy-regularized ERM oracle, which is shown in [Lemma 3.1](#) to be implementable efficiently using a linear optimization oracle for \mathcal{H} . \square

3 FRANK-WOLFE REDUCTION TO LINEAR OPTIMIZATION ORACLE

In this section, we explain how to implement a regularized ERM oracle over \mathcal{H} ([Definition 2.1](#)) through a sequence of calls to a linear optimization oracle over \mathcal{H} . This algorithm for the regularized ERM oracle is presented below as [Algorithm 2](#). We assume the loss function is both convex and β -smooth in the first parameter. However, this should not be seen as a limiting assumption due to the fact that if the losses are convex, L -Lipschitz but not smooth then we can use a linearized surrogate of the loss (similar to the OCO to OLO reduction). That is, we can define a new loss $\ell'(a, b) = a \nabla \ell(a, b)$ and by convexity, low regret learning with ℓ' implies low regret learning for ℓ .

Lemma 3.1 (Frank-Wolfe for smooth loss functions). Given a finite set of features $S \subset \mathcal{X}$, dataset $\{(x_i, y_i, w_i)\}_{i=1}^m$ where $x_i \in S$ for all i , a loss function ℓ that is convex and β -smooth in the first parameter, a class of functions $\mathcal{H} \subseteq [0, 1]^{\mathcal{X}}$, a linear optimization oracle for \mathcal{H} over S , and parameters $\eta, \epsilon > 0$, [Algorithm 2](#) returns an ϵ -approximate solution h^* to the entropy-regularized ℓ -ERM problem

$$\arg \min_{h \in \mathcal{H}} \left\{ \eta \sum_{i=1}^m w_i \ell(h(x_i), y_i) + \sum_{s \in S} h(s) \log(h(s) + 1) \right\}$$

after $O\left(\frac{|S|(\eta W_{\max} \beta + 1)}{\epsilon}\right)$ iterations, where $W_{\max} = \max_{s \in S} \sum_{i: x_i = s} |w_i|$ is the maximum sum of absolute weights for any feature in S .

Overview of Algorithm 2: The objective requires (approximately) solving a constrained smooth minimization problem $\min\{G(z) \mid z \in \mathcal{K}_S\}$ where $z \in [0, 1]^{|S|}$ is a vector indexed by elements of S , and the function $G : [0, 1]^{|S|} \rightarrow \mathbb{R}$ is defined as

$$G(z) = \eta \sum_{i=1}^m w_i \ell(z_{x_i}, y_i) + \sum_{s \in S} z_s \log(z_s + 1),$$

where z_s denotes the component of z corresponding to $s \in S$. The set \mathcal{K}_S denotes the convex hull of the set of vectors $z(h) = (h(s))_{s \in S}$ as h ranges over \mathcal{H} . In this section, we assume we are given a *linear*

optimization oracle for \mathcal{H} over the set S , that is, an algorithm for selecting the $h \in \mathcal{H}$ that minimizes $\sum_{s \in S} c_s h(s)$ for given coefficients $\{c_s\}_{s \in S}$. [Algorithm 2](#) below uses such an oracle to implement the Frank-Wolfe method, also known as conditional gradient descent, for approximately minimizing the convex function $G(z)$ over \mathcal{K}_S . At each iteration, the algorithm computes the gradient of the objective function $G(z)$ with respect to z , maps these components to weights c_s for $s \in S$, and invokes the linear optimization oracle to find an extreme point in the original function class \mathcal{H} that minimizes the corresponding linear function over S . After $O(\frac{|S|(\eta W_{\max} \beta + 1)}{\epsilon})$ iterations, it returns an ϵ -approximate solution to the original problem.

Algorithm 2 Frank-Wolfe for Entropy Regularized ℓ -ERM

```

1: procedure FRANKWOLFE( $\{(x_i, y_i, w_i)\}_{i=1}^m, \mathcal{H}, \eta, \epsilon, S$ )
2:   Initialize  $h_1$  to an arbitrary function in  $\mathcal{H}$ 
3:   for  $t = 1, 2, \dots, T$  do
4:     Let  $z_t$  be the vector  $(h_t(s))_{s \in S}$ .
5:     Compute gradient components  $c_s = \frac{\partial G(z_t)}{\partial z_s}$  for each  $s \in S$ :

$$c_s = \eta \sum_{i: x_i = s} w_i \frac{\partial \ell(h_t(s), y_i)}{\partial h} + \log(h_t(s) + 1) + \frac{h_t(s)}{h_t(s) + 1}$$

6:     Call a linear optimization oracle for  $\mathcal{H}$  over  $S$  with weights  $\{c_s\}_{s \in S}$  to obtain  $h'_t \in \mathcal{H}$ 
       minimizing  $\sum_{s \in S} c_s h(s)$ .
7:     Set  $\gamma_t = \frac{2}{t+1}$ 
8:     Update  $h_{t+1} = (1 - \gamma_t)h_t + \gamma_t h'_t$ .
9:   return  $h_{T+1}$ 

```

Lemma 3.2 (Conditional Gradient Descent; ([Hazan, 2023](#))). Let $K \subset \mathbb{R}^n$ with bounded ℓ_2 diameter R . Let f be a β -smooth function on K , then the sequence of points $x_t \in K$ computed by the conditional gradient descent algorithm satisfies

$$f(x_t) - f(x^*) \leq \frac{2\beta R^2}{t+1}$$

for all $t \geq 2$ where $x^* \in \arg \min_{x \in K} f(x)$.

The proof of [Lemma 3.1](#) uses the formulation of the problem as minimizing a smooth convex function $G(z)$ over a bounded set $\mathcal{K}_S \subset \mathbb{R}^{|S|}$. We then compute and bound the smoothness constant of $G(z)$ and the diameter of \mathcal{K}_S . Finally, we apply the standard convergence guarantee for the Frank-Wolfe algorithm in [Lemma 3.2](#) to obtain the stated convergence guarantee.

4 APPLICATION TO GAMES

Corollary 4.1. Let \mathcal{X} be a domain space and \mathcal{D} be a distribution over \mathcal{X} . Let $\mathcal{H}, \mathcal{R} \subseteq [0, 1]^{\mathcal{X}}$ be classes of functions (assumed to be closed under convex combinations) and $u : [0, 1] \times [0, 1] \rightarrow \mathbb{R}$ be a convex-concave payoff function that is L -Lipschitz in its first parameter. Consider the saddle-point optimization problem

$$\min_{h \in \mathcal{H}} \max_{r \in \mathcal{R}} \mathbb{E}_{x \sim \mathcal{D}} [u(h(x), r(x))]$$

Given m samples from \mathcal{D} and access to best-response oracles for \mathcal{H} and \mathcal{R} , our online learning algorithm can be used to find an $\epsilon(m)$ -approximate saddle point solution (h^*, r^*) in polynomial time in m and the complexities of \mathcal{H} and \mathcal{R} . The approximation guarantee is $\epsilon(m) = \text{rad}_m(\mathcal{F}) + O(L \sqrt{\log m / m})$, where $\mathcal{F} = \{f : f(x) = u(h(x), r(x)) \mid h \in \mathcal{H}, r \in \mathcal{R}\}$. Note that $\text{rad}_m(\mathcal{F}) \rightarrow 0$ is necessary for uniform convergence of the payoff matrix.

To prove [Corollary 4.1](#), we feed the m samples from \mathcal{D} sequentially into our hybrid learner. For each timestep t , we choose r_t to be the best response function to the h_t the algorithm outputs. We use the same m samples to compute the best response function $r_t = \arg \max_{r \in \mathcal{R}} \sum_{i=1}^m u(h_t(x_i), r(x_i))$ and this will be close to the true best response due to uniform convergence. Finally, using standard minimax analysis, we argue in [Appendix C](#) that the process converges to an approximate equilibrium.

REFERENCES

- Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: risk bounds and structural results. *J. Mach. Learn. Res.*, 3(null):463–482, March 2003. ISSN 1532-4435. 4
- Adam Block, Yuval Dagan, Noah Golowich, and Alexander Rakhlin. Smoothed online learning is as easy as statistical learning. In *Conference on Learning Theory*, pp. 1716–1786. PMLR, 2022. 1, 5
- Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis of online and differentially private learning. *Advances in Neural Information Processing Systems*, 33:9203–9215, 2020. 5
- Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis with adaptive adversaries. *Journal of the ACM*, 71(3):1–34, 2024. 1, 5
- Elad Hazan. Introduction to online convex optimization, 2023. URL <https://arxiv.org/abs/1909.05207>. 9
- Elad Hazan and Tomer Koren. The computational power of optimization in online learning. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing, STOC '16*, pp. 128–141, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450341325. doi: 10.1145/2897518.2897536. URL <https://doi.org/10.1145/2897518.2897536>. 3
- Lunjia Hu and Charlotte Peale. Comparative Learning: A Sample Complexity Theory for Two Hypothesis Classes. In Yael Tauman Kalai (ed.), *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 72:1–72:30, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-263-1. doi: 10.4230/LIPIcs.ITCS.2023.72. URL <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2023.72>. 5
- Alessandro Lazaric and Rémi Munos. Hybrid stochastic-adversarial on-line learning. In *COLT 2009 - 22nd Conference on Learning Theory*, Montreal, Canada, jun 2009. (inria-00392524). 1, 5
- Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning*, 2:285–318, 1988. 1
- Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, 2012. ISBN 026201825X. 2, 3, 4, 11
- Francesco Orabona. A modern introduction to online learning, 2023. URL <https://arxiv.org/abs/1912.13213>. 11, 13, 15
- Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Stochastic, constrained, and smoothed adversaries. *Advances in neural information processing systems*, 24, 2011. 1, 3, 5, 16, 17
- Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Sequential complexities and uniform martingale laws of large numbers. *Probability Theory and Related Fields*, 161(1):111–153, February 2015. ISSN 1432-2064. doi: 10.1007/s00440-013-0545-5. URL <https://doi.org/10.1007/s00440-013-0545-5>. 18
- Changlong Wu, Mohsen Heidari, Ananth Grama, and Wojciech Szpankowski. Expected worst case regret via stochastic sequential covering. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL <https://openreview.net/forum?id=H1SekypXKA>. 1, 5
- Changlong Wu, Jin Sima, and Wojciech Szpankowski. Oracle-efficient hybrid online learning with unknown distribution. In Shipra Agrawal and Aaron Roth (eds.), *Proceedings of Thirty Seventh Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, pp. 4992–5018. PMLR, 30 Jun–03 Jul 2024. URL <https://proceedings.mlr.press/v247/wu24a.html>. 1, 5

A DEFERRED PROOF FROM SECTION 2

A.1 DEFERRED PROOFS FROM SECTION 2.1

A.1.1 REFERENCE LEMMAS

Lemma A.1 (Lemma 7.8 of Orabona (2023)). Assume V is convex. If F_t is closed, subdifferentiable, and strongly convex in V , then v_t exists and is unique. In addition, assume $\partial\tilde{\ell}_t(v_t)$ to be non-empty and $F_t + \tilde{\ell}_t$ to be closed, subdifferentiable, and λ_t -strongly convex w.r.t. $\|\cdot\|$ in V . Then, we have

$$F_t(v_t) - F_{t+1}(v_{t+1}) + \ell_t(v_t) \leq \frac{\|g_t\|_*^2}{2\lambda_t} + \psi_t(v_{t+1}) - \psi_{t+1}(v_{t+1}), \forall g_t \in \partial\tilde{\ell}_t(v_t).$$

Theorem A.2 (Theorem 2.18 of Orabona (2023)). Let f_1, \dots, f_m be proper functions on \mathbb{R}^d , and $f = f_1 + \dots + f_m$. Then, $\partial f(v) \supseteq \partial f_1(v) + \dots + \partial f_m(v)$, $\forall v$. Moreover, if f_1, \dots, f_m are also convex, closed, and $\text{dom } f_m \cap \bigcap_{i=1}^{m-1} \text{int dom } f_i \neq \emptyset$, then actually $\partial f(v) = \partial f_1(v) + \dots + \partial f_m(v)$, $\forall v$.

Theorem A.3 (Theorem 3.3 of Mohri et al. (2012)). Fix distribution $D|_{\mathcal{X}}$ and parameter $\delta \in (0, 1)$. If $\mathcal{F} \subseteq \{f : \mathcal{X} \rightarrow [-1, 1]\}$ and $S = \{x_1, \dots, x_m\}$ is drawn i.i.d. from $D|_{\mathcal{X}}$, then with probability $\geq 1 - \delta$ over the draw of S , for every function $f \in \mathcal{F}$,

$$\mathbb{E}_D[f(x)] \leq \mathbb{E}_S[f(x)] + 2\text{rad}_m(\mathcal{F}) + \sqrt{\frac{\ln(1/\delta)}{m}}. \quad (1)$$

In addition, with probability $\geq 1 - \delta$, for every function $f \in \mathcal{F}$,

$$\mathbb{E}_D[f(x)] \leq \mathbb{E}_S[f(x)] + 2\hat{\text{rad}}_m(\mathcal{F}) + 3\sqrt{\frac{\ln(2/\delta)}{m}}. \quad (2)$$

A.1.2 PROOF OF LEMMA 2.3

Proof. For any fixed $t \in [T]$, consider the set of the first t samples $S_t = \{x_1, \dots, x_t\}$, drawn i.i.d. from D . The class $\mathcal{F} \subset \{\mathcal{X} \rightarrow [0, 1]\} \subseteq \{\mathcal{X} \rightarrow [-1, 1]\}$. Also, consider the class $-\mathcal{F} = \{-f \mid f \in \mathcal{F}\}$, which is also a subset of $\{\mathcal{X} \rightarrow [-1, 1]\}$, and $\text{rad}_t(-\mathcal{F}) = \text{rad}_t(\mathcal{F})$.

For a fixed t , applying Theorem A.3 to \mathcal{F} with confidence $\delta_t/2$, we get that with probability at least $1 - \delta_t/2$, for all $f \in \mathcal{F}$:

$$\mathbb{E}_D[f(x)] \leq \mathbb{E}_{S_t}[f(x)] + 2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\ln(2/\delta_t)}{t}}$$

This provides an upper bound on $\mathbb{E}_{S_t}[f(x)] - \mathbb{E}_D[f(x)]$.

Applying Theorem A.3 to $-\mathcal{F}$ with confidence $\delta_t/2$, we get that with probability at least $1 - \delta_t/2$, for all $g \in -\mathcal{F}$:

$$\mathbb{E}_D[g(x)] \leq \mathbb{E}_{S_t}[g(x)] + 2\text{rad}_t(-\mathcal{F}) + \sqrt{\frac{\ln(2/\delta_t)}{t}}$$

Substituting $g = -f$ for $f \in \mathcal{F}$ and using $\text{rad}_t(-\mathcal{F}) = \text{rad}_t(\mathcal{F})$:

$$-\mathbb{E}_D[f(x)] \leq -\mathbb{E}_{S_t}[f(x)] + 2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\ln(2/\delta_t)}{t}}$$

Multiplying by -1, we get a lower bound on $\mathbb{E}_{S_t}[f(x)] - \mathbb{E}_D[f(x)]$:

$$\mathbb{E}_D[f(x)] \geq \mathbb{E}_{S_t}[f(x)] - \left(2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\ln(2/\delta_t)}{t}}\right)$$

Combining the upper and lower bounds, with probability at least $1 - \delta_t/2 - \delta_t/2 = 1 - \delta_t$, for all $f \in \mathcal{F}$:

$$\left| \frac{1}{t} \sum_{s=1}^t f(x_s) - \mathbb{E}_D[f(x)] \right| \leq 2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\ln(2/\delta_t)}{t}}$$

Finally, we apply a union bound over $t \in [T]$. We want the bound to hold for all $t \in [T]$ with overall probability at least $1 - \delta$. Let F_t be the event that the inequality above does not hold for a specific t .

We have $P(F_t) \leq \delta_t$. By the union bound, $P(\cup_{t=1}^T F_t) \leq \sum_{t=1}^T P(F_t) \leq \sum_{t=1}^T \delta_t$. Setting $\delta_t = \delta/T$, we get $\sum_{t=1}^T \delta/T = \delta$.

Thus, with probability at least $1 - \delta$, for all $t \in [T]$ and for all $f \in \mathcal{F}$:

$$\left| \frac{1}{t} \sum_{s=1}^t f(x_s) - \mathbb{E}_D[f(x)] \right| \leq 2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\ln(2/(\delta/T))}{t}} = 2\text{rad}_t(\mathcal{F}) + \sqrt{\frac{\ln(2T/\delta)}{t}}$$

□

A.1.3 PROOF OF [LEMMA 2.2](#)

Before presenting the proof, we first state the following key lemma that analyzes the regret of the hybrid learner if the algorithm used an exact ERM instead of an approximate one.

Lemma A.4 (Exact FTRL for Hybrid Learning). Consider the (exact minimizer) Follow-the-Regularized-Leader (FTRL) approach with regularizer $\psi_t(v) = \frac{1}{\eta} \sum_{s=1}^{t-1} v^{(s)} \log(v^{(s)} + 1)$ and loss functions $\tilde{\ell}_t(v) = \frac{1}{t-1} \sum_{s=1}^{t-1} \ell(v^{(s)}, r_t(x_s))$ (for $t > 1$), where at each time step t , the decision $v_t \in V \subseteq [0, 1]^d$ minimizes $F_t(v) = \psi_t(v) + \sum_{i=1}^{t-1} \tilde{\ell}_i(v)$. Then, for any $u \in V$, the empirical regret is bounded by:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) \leq \frac{T \log 2}{\eta} + \frac{2\eta L^2}{3} (1 + \log(T-1)).$$

If $\eta = \sqrt{T/L^2 \log T}$, the empirical regret is bounded by $O(L\sqrt{T \log T})$.

The proof of this lemma has been deferred to later in the section. We now present the proof of [Lemma 2.2](#).

Proof of Lemma 2.2. We want to bound the empirical regret of the approximate FTRL algorithm, $\sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(u))$ for any $u \in V$. We decompose the sum as:

$$\sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(u)) = \sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t)) + \sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)),$$

where $v_t \in V$ is the exact minimizer of $F_t(v)$ at time t . The second term on the right-hand side is the empirical regret of the exact FTRL algorithm. By [Lemma A.4](#), this term is bounded by:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) \leq O(L\sqrt{T \log T})$$

Now, we bound the first term, which is the accumulated difference in loss between the approximate and exact minimizers: $\sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t))$. Since \bar{v}_t is an ε -approximate minimizer of $F_t(v)$, we have $F_t(\bar{v}_t) \leq F_t(v_t) + \varepsilon$. Using the strong convexity of F_t with parameter $\lambda_t = \frac{3}{4\eta(t-1)}$ for $t > 1$ with respect to the ℓ_1 norm of the first $t-1$ coordinates, the difference in function values is related to the squared distance between the points:

$$F_t(\bar{v}_t) - F_t(v_t) \geq \frac{\lambda_t}{2} \|\bar{v}_t^{(1:t-1)} - v_t^{(1:t-1)}\|_1^2.$$

Here the superscript refers to the first $t-1$ coordinates of the vector v_t . Combining with the ε -optimality, we get a bound on the ℓ_1 distance between $\bar{v}_t^{(1:t-1)}$ and $v_t^{(1:t-1)}$:

$$\frac{\lambda_t}{2} \|\bar{v}_t^{(1:t-1)} - v_t^{(1:t-1)}\|_1^2 \leq \varepsilon \implies \|\bar{v}_t^{(1:t-1)} - v_t^{(1:t-1)}\|_1 \leq \sqrt{\frac{2\varepsilon}{\lambda_t}}.$$

The function $\tilde{\ell}_t(v)$ is convex and L -Lipschitz. For $t > 1$, the ℓ_∞ norm of its subgradient is bounded by $\|\partial \tilde{\ell}_t(v)\|_\infty \leq \frac{L}{t-1}$. The difference in loss can be bounded using the Lipschitz property:

$$\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t) \leq \|\partial \tilde{\ell}_t\|_\infty \|\bar{v}_t^{(1:t-1)} - v_t^{(1:t-1)}\|_1 \leq \frac{L}{t-1} \|\bar{v}_t^{(1:t-1)} - v_t^{(1:t-1)}\|_1, \quad \text{for } t > 1.$$

Substitute the bound on $\|\bar{v}_t^{(1:t-1)} - v_t^{(1:t-1)}\|_1$:

$$\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t) \leq \frac{L}{t-1} \sqrt{\frac{2\varepsilon}{\lambda_t}}, \quad \text{for } t > 1.$$

Using $\lambda_t = \frac{3}{4\eta(t-1)}$ for $t > 1$:

$$\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t) \leq \frac{L}{t-1} \sqrt{\frac{2\varepsilon}{\frac{3}{4\eta(t-1)}}} = \frac{L}{t-1} \sqrt{\frac{8\eta\varepsilon(t-1)}{3}} = L \sqrt{\frac{8\eta\varepsilon}{3(t-1)}}, \quad \text{for } t > 1.$$

Now, sum this bound from $t = 2$ to T

$$\sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t)) \leq \sum_{t=2}^T L \sqrt{\frac{8\eta\varepsilon}{3(t-1)}} = L \sqrt{\frac{8\eta\varepsilon}{3}} \sum_{t=2}^T \frac{1}{\sqrt{t-1}}.$$

We use the bound $\sum_{k=1}^{T-1} \frac{1}{\sqrt{k}} \leq 1 + \int_1^{T-1} x^{-1/2} dx = 1 + [2\sqrt{x}]_1^{T-1} = O(\sqrt{T})$.

$$\sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(v_t)) \leq L \sqrt{\frac{8\eta\varepsilon}{3}} O(\sqrt{T}) = O(L\sqrt{\eta\varepsilon T}).$$

Combining the bounds for the two terms in the regret decomposition:

$$\sum_{t=1}^T (\tilde{\ell}_t(\bar{v}_t) - \tilde{\ell}_t(u)) \leq \left(\frac{T \log 2}{\eta} + \frac{2\eta L^2}{3} (1 + \log(T-1)) \right) + O(L\sqrt{\eta\varepsilon T}).$$

□

To prove [Lemma A.4](#), we first present the following helper lemmas:

Lemma A.5 (Lemma 7.1 of [Orabona \(2023\)](#)). The Follow-the-Regularized-Leader (FTRL) algorithm, at each time step t from 1 to T , outputs a decision v_t that minimizes $F_t(v)$ over a closed and non-empty set $V \subseteq \mathbb{R}^d$, where $F_t(v) = \psi_t(v) + \sum_{i=1}^{t-1} \tilde{\ell}_i(v)$. That is,

$$v_t \in \operatorname{argmin}_{v \in V} F_t(v).$$

Assume that $\operatorname{argmin}_{v \in V} F_t(v)$ is non-empty, and let v_t be an element of this set. Then, for any $u \in \mathbb{R}^d$, we have:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) = \psi_{T+1}(u) - \min_{v \in V} \psi_1(v) + \sum_{t=1}^T [F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t)] + (F_{T+1}(v_{T+1}) - F_{T+1}(u)).$$

The proof of this lemma can be found in the referenced text.

Lemma A.6 (Variant of Lemma 7.2 of [Orabona \(2023\)](#)). Assume V is convex and $\partial \tilde{\ell}_t(v_t)$ is non-empty for the Follow-the-Regularized-Leader (FTRL) approach with regularizer $\psi_t(v) = \frac{1}{\eta} \sum_{s=1}^{t-1} v^{(s)} \log(v^{(s)} + 1)$ and loss functions $\tilde{\ell}_t(v) = \frac{1}{t-1} \sum_{s=1}^{t-1} \ell(v^{(s)}, r_t(x_s))$ (for $t > 1$). Then for $F_t(v) = \psi_t(v) + \sum_{i=1}^{t-1} \tilde{\ell}_i(v)$, it holds that for any $g_t \in \partial \tilde{\ell}_t(v_t)$, we have

$$F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t) \leq \frac{\|g_t\|_\infty^2}{2\lambda_t} + \psi_t(v_t) - \psi_{t+1}(v_{t+1}).$$

for $\lambda_t = \frac{3}{4\eta(t-1)}$

Note that a direct application of [Lemma A.1](#) (Lemma 7.2 of [Orabona \(2023\)](#)) can be challenging because the objective function F_t depends only on the first $t-1$ coordinates of v , while the domain V is in a higher-dimensional space. The following lemma provides a bound on the change in the objective function plus current loss, similar to [Lemma A.1](#), adapted for this structure. The proof is deferred to later in the section.

Lemma A.7. For $t > 1$, the regularizer $\psi_t(v) = \frac{1}{\eta} \sum_{s=1}^{t-1} v^{(s)} \log(v^{(s)} + 1)$ defined over $V^{(t-1)} \subseteq [0, 1]^{t-1}$ is $\frac{3}{4\eta(t-1)}$ -strongly convex with respect to the ℓ_1 norm.

The proof of the lemma has been deferred to later in this section.

Proof of Lemma A.4. From Lemma A.5, for any $u \in V$, we have:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) = \psi_{T+1}(u) - \min_{v \in V} \psi_1(v) + \sum_{t=1}^T [F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t)] + (F_{T+1}(v_{T+1}) - F_{T+1}(u)).$$

From the definition of ψ_1 , $\psi_1(v) = \frac{1}{\eta} \sum_{s=1}^0 v^{(s)} \log(v^{(s)} + 1) = 0$. Thus, $\min_{v \in V} \psi_1(v) = 0$. The equality becomes:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) = \psi_{T+1}(u) + \sum_{t=1}^T [F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t)] + (F_{T+1}(v_{T+1}) - F_{T+1}(u)).$$

By Lemma A.6, for $t > 1$, we have for $\lambda_t = \frac{3}{4\eta(t-1)}$:

$$F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t) \leq \frac{\|g_t\|_\infty^2}{2\lambda_t} + \psi_t(v_t) - \psi_{t+1}(v_{t+1}), \quad \forall g_t \in \partial \tilde{\ell}_t(v_t).$$

Summing this inequality from $t = 1$ to T :

$$\sum_{t=1}^T [F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t)] \leq \sum_{t=1}^T \frac{\|g_t\|_\infty^2}{2\lambda_t} + \sum_{t=1}^T (\psi_t(v_t) - \psi_{t+1}(v_{t+1})).$$

The second sum on the right-hand side is a telescoping sum:

$$\sum_{t=1}^T (\psi_t(v_t) - \psi_{t+1}(v_{t+1})) = (\psi_1(v_1) - \psi_2(v_2)) + \dots + (\psi_T(v_T) - \psi_{T+1}(v_{T+1})) = \psi_1(v_1) - \psi_{T+1}(v_{T+1}).$$

Since $\psi_1(v_1) = 0$, this sum equals $-\psi_{T+1}(v_{T+1})$. Substituting this back into the sum bound:

$$\sum_{t=1}^T [F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t)] \leq \sum_{t=1}^T \frac{\|g_t\|_\infty^2}{2\lambda_t} - \psi_{T+1}(v_{T+1}).$$

Now, substitute this bound into the FTRL guarantee:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) \leq \psi_{T+1}(u) + \sum_{t=1}^T \frac{\|g_t\|_\infty^2}{2\lambda_t} - \psi_{T+1}(v_{T+1}) + (F_{T+1}(v_{T+1}) - F_{T+1}(u)).$$

Since $v_{T+1} = \operatorname{argmin}_{v \in V} F_{T+1}(v)$, we have $F_{T+1}(v_{T+1}) \leq F_{T+1}(u)$, so $F_{T+1}(v_{T+1}) - F_{T+1}(u) \leq 0$.

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) \leq \psi_{T+1}(u) - \psi_{T+1}(v_{T+1}) + \sum_{t=1}^T \frac{\|g_t\|_\infty^2}{2\lambda_t}.$$

Now we bound the terms on the right-hand side. First, consider the difference of the regularizer terms $\psi_{T+1}(u) - \psi_{T+1}(v_{T+1})$. For any $v \in V \subseteq [0, 1]^d$, $\psi_{T+1}(v) = \frac{1}{\eta} \sum_{s=1}^T v^{(s)} \log(v^{(s)} + 1)$. Since $v^{(s)} \in [0, 1]$, $0 \leq v^{(s)} \log(v^{(s)} + 1) \leq \log 2$. Thus, $0 \leq \psi_{T+1}(v) \leq \frac{T \log 2}{\eta}$ for any $v \in V$. Therefore, $\psi_{T+1}(u) - \psi_{T+1}(v_{T+1}) \leq \psi_{T+1}(u) \leq \frac{T \log 2}{\eta}$.

Next, consider the sum of gradient terms $\sum_{t=1}^T \frac{\|g_t\|_\infty^2}{2\lambda_t}$. From Lemma A.7, $\lambda_t = \frac{3}{4\eta(t-1)}$ for $t > 1$. The subgradient $g_t \in \partial \tilde{\ell}_t(v_t)$. Since $\tilde{\ell}_t(v) = \frac{1}{t-1} \sum_{s=1}^{t-1} \ell(v^{(s)}, r_t(x_s))$ and ℓ is L-Lipschitz, the infinity norm of g_t is bounded by $\|g_t\|_\infty \leq \frac{L}{t-1}$ for $t > 1$. Substituting these bounds:

$$\begin{aligned} \sum_{t=2}^T \frac{\|g_t\|_\infty^2}{2\lambda_t} &\leq \sum_{t=2}^T \frac{(L/(t-1))^2}{2 \cdot \frac{3}{4\eta(t-1)}} = \sum_{t=2}^T \frac{L^2}{(t-1)^2} \cdot \frac{2\eta(t-1)}{3} = \sum_{t=2}^T \frac{2\eta L^2}{3(t-1)}. \\ \sum_{t=2}^T \frac{2\eta L^2}{3(t-1)} &= \frac{2\eta L^2}{3} \sum_{t=2}^T \frac{1}{t-1} = \frac{2\eta L^2}{3} \sum_{k=1}^{T-1} \frac{1}{k}. \end{aligned}$$

Using the bound $\sum_{k=1}^{T-1} \frac{1}{k} \leq 1 + \log(T-1)$ for $T > 1$:

$$\sum_{t=1}^T \frac{\|g_t\|_\star^2}{2\lambda_t} \leq \frac{2\eta L^2}{3}(1 + \log(T-1)).$$

Combining the bounds for the two terms:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) \leq \frac{T \log 2}{\eta} + \frac{2\eta L^2}{3}(1 + \log(T-1)).$$

If we choose $\eta = \sqrt{\frac{T}{L^2 \log T}}$, the empirical regret is bounded by:

$$\sum_{t=1}^T (\tilde{\ell}_t(v_t) - \tilde{\ell}_t(u)) \leq \frac{T \log 2}{\sqrt{\frac{T}{L^2 \log T}}} + \frac{2\sqrt{\frac{T}{L^2 \log T}} L^2}{3}(1 + \log(T-1)) = O(L\sqrt{T \log T}).$$

Thus, with this choice of η , the empirical regret is bounded by $O(L\sqrt{T \log T})$.

□

Corollary A.8 (Corollary 7.7 of [Orabona \(2023\)](#)). Let $f : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ be closed, proper, subdifferentiable, and μ -strongly convex with respect to a norm $\|\cdot\|$ over its domain. Let $v^\star = \arg \min_v f(v)$. Then, for all $v \in \text{dom } \partial f$, and $g \in \partial f(v)$, we have

$$f(v) - f(v^\star) \leq \frac{1}{2\mu} \|g\|_\star^2.$$

Theorem A.9 (Theorem 6.12 of [Orabona \(2023\)](#)). Let $f : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ be proper. Then $v^\star \in \arg \min_{v \in \mathbb{R}^d} f(v)$ iff $0 \in \partial f(v^\star)$.

Proof of Lemma A.6. Let $V^{(t-1)} = \{v^{(1:t-1)} : v \in V\} \subseteq [0, 1]^{t-1}$. Define $\bar{F}_t : V^{(t-1)} \rightarrow \mathbb{R}$ such that $\bar{F}_t(v^{(1:t-1)}) = F_t(v)$ for $v \in V$. Note that $F_t(v) = \psi_t(v) + \sum_{s=1}^{t-1} \tilde{\ell}_s(v)$. Since ψ_t depends only on $v^{(1:t-1)}$ and $\tilde{\ell}_s$ (for $s < t$) depends on $v^{(1:s-1)} \subseteq v^{(1:t-1)}$, F_t indeed depends only on $v^{(1:t-1)}$. Similarly, we define $\bar{\ell}_t : V^{(t-1)} \rightarrow \mathbb{R}$ such that $\bar{\ell}_t(v^{(1:t-1)}) = \tilde{\ell}_t(v)$ for $v \in V, t \in [T]$. By [Lemma A.7](#), ψ_t is closed, subdifferentiable, and strongly convex with parameter $\lambda_t = \frac{3}{4\eta(t-1)}$ with respect to the ℓ_1 norm on $V^{(t-1)}$ (for $t > 1$). As a sum of a strongly convex function (ψ_t) and convex functions ($\tilde{\ell}_s$), \bar{F}_t is also strongly convex with parameter $\lambda_t = \frac{3}{4\eta(t-1)}$ on $V^{(t-1)}$ (for $t > 1$). The function $(v^{(1:t-1)}) \mapsto F_t(v) + \tilde{\ell}_t(v)$ is closed, subdifferentiable, and strongly convex with parameter λ_t with respect to the ℓ_1 norm on $V^{(t-1)}$.

$$\begin{aligned} F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t) &= (F_t(v_t) + \tilde{\ell}_t(v_t)) - (F_t(v_{t+1}) + \tilde{\ell}_t(v_{t+1})) + \psi_t(v_{t+1}) - \psi_{t+1}(v_{t+1}) \quad (\text{Rearranging terms}) \\ &= (\bar{F}_t(v_t^{(1:t-1)}) + \bar{\ell}_t(v_t^{(1:t-1)})) - (\bar{F}_t(v_{t+1}^{(1:t-1)}) + \bar{\ell}_t(v_{t+1}^{(1:t-1)})) + \psi_t(v_{t+1}) - \psi_{t+1}(v_{t+1}) \end{aligned}$$

Recall that \bar{F}_t is also strongly convex with parameter $\lambda_t = \frac{3}{4\eta(t-1)}$ on $V^{(t-1)}$ (for $t > 1$), thus, if we define $v_{t+1}^{*,(1:t-1)} := \arg \min_{v \in V^{(1:t-1)}} \{\bar{F}_t(v) + \bar{\ell}_t(v)\}$. By [Corollary A.8](#), $(\bar{F}_t(v_t^{(1:t-1)}) + \bar{\ell}_t(v_t^{(1:t-1)})) - (\bar{F}_t(v_{t+1}^{*,(1:t-1)}) + \bar{\ell}_t(v_{t+1}^{*,(1:t-1)})) \leq \frac{\|g_t\|_\infty^2}{2\lambda_t}$ where $g_t \in \delta(\bar{F}_t + \bar{\ell}_t)(v_t^{(1:t-1)})$. Now, use the fact that $v_t^{(1:t-1)} \in \arg \min_{v \in V^{(t-1)}} F_t(v)$, which by [Theorem A.9](#) implies $0 \in \delta \bar{F}_t(v_t^{(1:t-1)})$, which implies $g_t \in \delta \bar{\ell}_t(v_t^{(1:t-1)})$. And because $\bar{\ell}_t$ only depends on the first $t-1$ coordinates, then $\delta \bar{\ell}_t = \delta \tilde{\ell}_t$. Thus, for any $g_t \in \delta \tilde{\ell}_t(v_t)$, we have

$$F_t(v_t) - F_{t+1}(v_{t+1}) + \tilde{\ell}_t(v_t) \leq \frac{\|g_t\|_\infty^2}{2\lambda_t} + \psi_t(v_t) - \psi_{t+1}(v_{t+1}).$$

for $\lambda_t = \frac{3}{4\eta(t-1)}$

□

Proof of Lemma A.7. Let $v \in V^{(t-1)} \subseteq [0, 1]^{t-1}$, so v is a vector $(v^{(1)}, \dots, v^{(t-1)})$ with $v^{(s)} \in [0, 1]$ for $s = 1, \dots, t-1$. Let $u = y - x$ where $x, y \in V^{(t-1)}$, so u is a vector $(u_1, \dots, u_{t-1}) \in \mathbb{R}^{t-1}$. The

function is $\psi_t(v) = \frac{1}{\eta} \sum_{s=1}^{t-1} v^{(s)} \log(v^{(s)} + 1)$. Let $f(w) = w \log(w + 1)$. The second derivative is $f''(w) = \frac{1}{w+1} + \frac{1}{(w+1)^2}$. For $w \in [0, 1]$, $w + 1 \in [1, 2]$, so $f''(w) \geq \frac{1}{2} + \frac{1}{4} = \frac{3}{4}$.

The Hessian matrix $\nabla^2 \psi_t(v)$ is a $(t-1) \times (t-1)$ diagonal matrix with entries $(\nabla^2 \psi_t(v))_{ss} = \frac{1}{\eta} f''(v^{(s)})$ for $s = 1, \dots, t-1$.

To show λ_t -strong convexity with respect to the ℓ_1 norm, we show $u^T \nabla^2 \psi_t(v) u \geq \lambda_t \|u\|_1^2$ for all $v \in V^{(t-1)}$ and $u \in \mathbb{R}^{t-1}$.

$$u^T \nabla^2 \psi_t(v) u = \sum_{s=1}^{t-1} (\nabla^2 \psi_t(v))_{ss} u_s^2 = \sum_{s=1}^{t-1} \frac{1}{\eta} f''(v^{(s)}) u_s^2$$

Since $v^{(s)} \in [0, 1]$ for $s = 1, \dots, t-1$, $f''(v^{(s)}) \geq \frac{3}{4}$.

$$u^T \nabla^2 \psi_t(v) u \geq \sum_{s=1}^{t-1} \frac{1}{\eta} \left(\frac{3}{4}\right) u_s^2 = \frac{3}{4\eta} \sum_{s=1}^{t-1} u_s^2 = \frac{3}{4\eta} \|u\|_2^2$$

We use the relationship between the ℓ_2 and ℓ_1 norms in \mathbb{R}^{t-1} : $\|u\|_2^2 \geq \frac{1}{t-1} \|u\|_1^2$. This holds for $t-1 > 0$, i.e., $t > 1$.

$$\frac{3}{4\eta} \|u\|_2^2 \geq \frac{3}{4\eta} \left(\frac{1}{t-1} \|u\|_1^2\right) = \frac{3}{4\eta(t-1)} \|u\|_1^2$$

Thus, $u^T \nabla^2 \psi_t(v) u \geq \frac{3}{4\eta(t-1)} \|u\|_1^2$. Comparing this with the strong convexity condition $u^T \nabla^2 \psi_t(v) u \geq \lambda_t \|u\|_1^2$, we can choose $\lambda_t = \frac{3}{4\eta(t-1)}$.

□

A.2 DEFERRED PROOFS FROM SECTION 1.2

To aid with their introduction and analysis of the distribution-dependent sequential Rademacher complexity, [Rakhlin et al. \(2011\)](#) introduce the following notation:

They define the *selector function* $\chi : \mathcal{X} \times \mathcal{X} \times \{\pm 1\} \rightarrow \mathcal{X}$, which selects one of two elements based on a binary sign ϵ :

$$\chi(x, x', \epsilon) = \begin{cases} x' & \text{if } \epsilon = 1 \\ x & \text{if } \epsilon = -1 \end{cases}$$

In the context of sequences where x_t and x'_t implicitly depend on previous ϵ values, the shorthand $\chi_t(\epsilon) := \chi(x_t(\epsilon_{1:t-1}), x'_t(\epsilon_{1:t-1}), \epsilon_t)$. This notation indicates that χ_t chooses either x_t or x'_t at time step t , depending on the value of ϵ_t within the path $\epsilon = (\epsilon_1, \dots, \epsilon_T)$. The terms $x_t(\epsilon_{1:t-1})$ and $x'_t(\epsilon_{1:t-1})$ represent elements at depth t along a specific path determined by the preceding ϵ values.

A *Z-valued tree of depth T* is a sequence of T mappings, $(\mathbf{z}_1, \dots, \mathbf{z}_T)$. Each mapping $\mathbf{z}_t : \{\pm 1\}^{t-1} \rightarrow Z$ assigns a value from set Z to a specific node at depth t . The node's position is uniquely determined by a sequence of prior choices, $(\epsilon_1, \dots, \epsilon_{t-1}) \in \{\pm 1\}^{t-1}$. A complete sequence $\epsilon = (\epsilon_1, \dots, \epsilon_T) \in \{\pm 1\}^T$ defines a unique path from the root to a leaf of the tree. For conciseness, $\mathbf{z}_t(\epsilon_{1:t-1})$ is shorthand for $\mathbf{z}_t(\epsilon_1, \dots, \epsilon_{t-1})$.

Given an underlying joint distribution \mathbf{p} (over T length sequences of observations from \mathcal{X}), we define a *probability tree* $\rho = (\rho_1, \dots, \rho_T)$. This tree generates sequences of pairs of elements $(\mathbf{x}, \mathbf{x}') = ((x_1, x'_1), \dots, (x_T, x'_T))$. Each $\rho_t(\epsilon_{1:t-1})$ is a conditional probability distribution that determines (x_t, x'_t) given the preceding pairs $(x_1, x'_1), \dots, (x_{t-1}, x'_{t-1})$. The crucial aspect is how this conditioning is performed:

$$\rho_t(\epsilon_{1:t-1})((x_t, x'_t) | (x_{1:t-1}, x'_{1:t-1})) = \mathbf{p}_t((\chi_s(\epsilon_s))_{s=1}^{t-1})((x_t, x'_t) | (x_{1:t-1}, x'_{1:t-1})) \quad (4)$$

Here, $\mathbf{p}_t((\chi_s(\epsilon_s))_{s=1}^{t-1})$ denotes the conditional distribution for (x_t, x'_t) derived from \mathbf{p} , given the history sequence formed by dynamically applying the selector function at each step: $(\chi_1(\epsilon_1), \dots, \chi_{t-1}(\epsilon_{t-1}))$. This means the generation of each pair (x_t, x'_t) depends on a history that dynamically selects between x_s and x'_s based on the Rademacher variables ϵ_s .

Definition A.1 (Definition 2 of [Rakhlin et al. \(2011\)](#)). The distribution-dependent sequential Rademacher complexity of a function class $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}}$ is defined as

$$\mathfrak{R}_T(\mathcal{F}, \mathbf{p}) \triangleq \mathbb{E}_{(\mathbf{x}, \mathbf{x}') \sim \rho} \mathbb{E}_{\epsilon} \left[\sup_{f \in \mathcal{F}} \sum_{t=1}^T \epsilon_t f(\chi_t(\epsilon)) \right]$$

where $\epsilon = (\epsilon_1, \dots, \epsilon_T)$ is a sequence of i.i.d. Rademacher random variables and ρ is the probability tree associated with \mathbf{p} as explained in [Equation \(4\)](#).

Lemma A.10 (Lemma 17 of [Rakhlin et al. \(2011\)](#)). Fix a class $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}}$ and a function $\phi : \mathbb{R} \times \mathcal{Y} \rightarrow \mathbb{R}$. Given a distribution p over \mathcal{X} , let \mathfrak{P} consist of all joint distributions \mathbf{p} such that the conditional distribution $p_t^{x,y}(x_t, y_t | x^{t-1}, y^{t-1})$ can be written as $p(x_t) \times p_t(y_t | x^{t-1}, y^{t-1}, x_t)$ for some conditional distribution p_t . Then,

$$\sup_{\mathbf{p} \in \mathfrak{P}} \mathfrak{R}_T(\phi(\mathcal{F}), \mathbf{p}) \leq \mathbb{E}_{\mathbf{x}_1, \dots, \mathbf{x}_T \sim p, y \sim \mathbf{p}} \left[\mathbb{E}_{\epsilon} \sup_{f \in \mathcal{F}} \sum_{t=1}^T \epsilon_t \phi(f(x_t), y_t(\epsilon)) \right].$$

Lemma A.11 (Lemma 18 of [Rakhlin et al. \(2011\)](#)). Fix a class $\mathcal{F} \subseteq [-1, 1]^{\mathcal{X}}$ and a function $\phi : [-1, 1] \times \mathcal{Y} \rightarrow \mathbb{R}$. Assume, for all $y \in \mathcal{Y}$, $\phi(\cdot, y)$ is a Lipschitz function with a constant L . Let \mathfrak{P} be as in [Lemma A.10](#). Then, for any $\mathbf{p} \in \mathfrak{P}$,

$$\mathfrak{R}_T(\phi(\mathcal{F}), \mathbf{p}) \leq L \mathfrak{R}_T(\mathcal{F}, p).$$

Proposition 1.3. Let \mathcal{H} be a class of hypothesis functions and ℓ be a loss function that is L -Lipschitz in the first parameter. Let x_1, x_2, \dots, x_T be a sequence of i.i.d samples from a fixed distribution \mathcal{D} . Let $r_1, r_2, \dots, r_T \in [0, 1]^{\mathcal{X}}$ be a sequence of functions where r_t depends only on x_1, \dots, x_{t-1} (and potentially prior adversarial choices). The following holds with probability at least $1 - \delta$ over the draw of x_1, \dots, x_T , for all $h \in \mathcal{H}$:

$$\left| \frac{1}{T} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) - \frac{1}{T} \sum_{t=1}^T \mathbb{E}_{x \sim \mathcal{D}} [\ell(h(x), r_t(x))] \right| \leq O \left(L \cdot \text{rad}_T(\mathcal{H}) + L \sqrt{\frac{\log(T/\delta)}{T}} \right)$$

Proof of Proposition 1.3. Note that $\ell(h(x_t), r_t(x_t)) - \mathbb{E}_{\mathcal{D}}[\ell(h(x), r_t(x))]$ is a martingale difference sequence since x_t is sampled after the choice of r_t is made. We will apply the classic symmetrization technique, borrowing ideas from the proof of Theorem 3 of [Rakhlin et al. \(2011\)](#). We will consider a tangent sequence $\{x'_t\}_{t=1}^T$ that is drawn i.i.d from the distribution \mathcal{D} . Note that this tangent sequence is independent of $\{x_t\}_{t=1}^T$. For any sequence of r_1, \dots, r_T , The LHS of the equation reduces to the following:

$$\mathbb{E} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) - \frac{1}{T} \sum_{t=1}^T \ell(h(x'_t), r_t(x'_t)) \right\} \right] \quad (5)$$

$$= \mathbb{E}_{(x_1, x'_1) \sim \mathcal{D}} \mathbb{E}_{(x_2, x'_2) \sim \mathcal{D}} \dots \mathbb{E}_{(x_T, x'_T) \sim \mathcal{D}} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) - \frac{1}{T} \sum_{t=1}^T \ell(h(x'_t), r_t(x'_t)) \right\} \right] \quad (6)$$

$$\leq \sup_{r_1 \in \mathcal{R}} \mathbb{E}_{(x_1, x'_1) \sim \mathcal{D}} \sup_{r_2 \in \mathcal{R}(\cdot | x_1)} \mathbb{E}_{(x_2, x'_2) \sim \mathcal{D}} \dots \quad (7)$$

$$\dots \sup_{r_T \in \mathcal{R}(\cdot | x_1, \dots, x_{T-1})} \mathbb{E}_{(x_T, x'_T) \sim \mathcal{D}} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) - \frac{1}{T} \sum_{t=1}^T \ell(h(x'_t), r_t(x'_t)) \right\} \right] \quad (8)$$

Now fix $\epsilon \in \{\pm 1\}^T$ and let $-\epsilon_t$ denote whether we switch x_t with x'_t . Since these are from the same distribution, this does not affect the expectation over \mathcal{D} . Thus, the last equation simplifies to

$$\sup_{r_1 \in \mathcal{R}} \mathbb{E}_{(x_1, x'_1) \sim \mathcal{D}} \sup_{r_2 \in \mathcal{R}(\cdot | x_1(\epsilon_1))} \mathbb{E}_{(x_2, x'_2) \sim \mathcal{D}} \dots \sup_{r_T \in \mathcal{R}(\cdot | x_1(\epsilon_1), \dots, x_{T-1}(\epsilon_{T-1}))} \mathbb{E}_{(x_T, x'_T) \sim \mathcal{D}} \quad (9)$$

$$\left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \epsilon_t (\ell(h(x_t), r_t(x_t)) - \ell(h(x'_t), r_t(x'_t))) \right\} \right] \quad (10)$$

Taking expectation over $\epsilon \in \{\pm 1\}^T$, we have that

$$\mathbb{E} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \ell(h(x_t), r_t(x_t)) - \frac{1}{T} \sum_{t=1}^T \ell(h(x'_t), r_t(x'_t)) \right\} \right] \quad (11)$$

$$\leq \sup_{r_1 \in \mathcal{R}} \mathbb{E}_{(x_1, x'_1)} \mathbb{E}_{\epsilon_1} \sup_{r_2 \in \mathcal{R}(\cdot | x_1(\epsilon_1))} \mathbb{E}_{(x_2, x'_2)} \mathbb{E}_{\epsilon_2} \dots \sup_{r_T \in \mathcal{R}(\cdot | x_1(\epsilon_1), \dots, x_{T-1}(\epsilon_{T-1}))} \mathbb{E}_{(x_T, x'_T)} \mathbb{E}_{\epsilon_T} \quad (12)$$

$$\left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \epsilon_t (\ell(h(x_t), r_t(x_t)) - \ell(h(x'_t), r_t(x'_t))) \right\} \right] \quad (13)$$

The process above can be thought of as taking a path in a binary tree whose nodes are represented by functions $r \in \mathcal{R}$. At each step t , r_t is chosen and then a coin is flipped and this determines whether x_t or x'_t is to be used in the following steps. We write the last expression concisely as

$$\sup_{\mathbf{r}} \mathbb{E}_{(x, x') \sim \mathcal{D}} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \epsilon_t (\ell(h(x_t), r_t(x_t)) - \ell(h(x'_t), r_t(x'_t))) \right\} \right]$$

And this can be upper bounded by two times the distribution-dependent Rademacher complexity notion defined in [Definition A.1](#)

$$\sup_{\mathbf{r}} \mathbb{E}_{(x, x') \sim \mathcal{D}} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \epsilon_t (\ell(h(x_t), r_t(x_t)) - \ell(h(x'_t), r_t(x'_t))) \right\} \right] \quad (14)$$

$$\leq 2 \sup_{\mathbf{r}} \mathbb{E}_{(x, x') \sim \mathcal{D}} \left[\sup_{h \in \mathcal{H}} \left\{ \frac{1}{T} \sum_{t=1}^T \epsilon_t \ell(h(x_t), r_t(x_t)) \right\} \right] \quad (15)$$

$$\leq 2 \sup_{\mathbf{p} \in \mathfrak{B}} \mathfrak{R}_T(\ell \circ \mathcal{H}, \mathbf{p}) \quad (16)$$

where \mathfrak{B} consists of all joint distributions \mathbf{p} such that the conditional distribution $p_t^{x,y}(x_t, y_t | x^{t-1}, y^{t-1})$ can be written as $p(x_t) \times p_t(y_t | x^{t-1}, y^{t-1}, x_t)$ for some conditional distribution p_t . Applying [Lemma A.10](#) together with [Lemma A.11](#) gives the desired result. To obtain the high probability version of the statement, we follow the same steps here replacing the expected Rademacher with high probability Rademacher as done in Lemma 4 of [Rakhlin et al. \(2015\)](#). \square

B DEFERRED PROOF FROM SECTION 4

Proof of Lemma 3.1. Let $p = |S|$. We order the elements of S as (s_1, \dots, s_p) . The feasible set is $\mathcal{K}_S = \{(h(s_1), \dots, h(s_p)) \mid h \in \text{conv}(\mathcal{H})\}$. Since $h : [0, 1]^X$, $\mathcal{K}_S \subseteq [0, 1]^p$. The objective function is $G : \mathcal{K}_S \rightarrow \mathbb{R}$ defined as

$$G(z) = \eta \sum_{i=1}^m w_i \ell(z_{x_i}, y_i) + \sum_{s \in S} z_s \log(z_s + 1),$$

where $z \in \mathcal{K}_S$ and z_s denotes the component of z corresponding to $s \in S$. The function G is well-defined and differentiable on \mathcal{K}_S . Its partial derivative with respect to z_s for $s \in S$ is:

$$\frac{\partial G(z)}{\partial z_s} = \eta \sum_{i: x_i = s} w_i \frac{\partial \ell(z_s, y_i)}{\partial z_s} + \log(z_s + 1) + \frac{z_s}{z_s + 1}.$$

The Hessian of $G(z)$ is a diagonal matrix. The diagonal entry corresponding to z_s is $\frac{\partial^2 G(z)}{\partial z_s^2}$.

$$\frac{\partial^2 G(z)}{\partial z_s^2} = \eta \sum_{i: x_i = s} w_i \frac{\partial^2 \ell(z_s, y_i)}{\partial z_s^2} + \frac{1}{z_s + 1} + \frac{1}{(z_s + 1)^2}.$$

Since ℓ is β -smooth, $|\frac{\partial^2 \ell}{\partial u^2}| \leq \beta$. For $z_s \in [0, 1]$, we have $\frac{1}{z_s + 1} + \frac{1}{(z_s + 1)^2} \leq 1 + 1 = 2$. Let $I(s) = \{i \in \{1, \dots, m\} \mid x_i = s\}$. Then

$$\left| \frac{\partial^2 G(z)}{\partial z_s^2} \right| \leq \eta \sum_{i \in I(s)} |w_i| \left| \frac{\partial^2 \ell(z_s, y_i)}{\partial z_s^2} \right| + \left| \frac{1}{z_s + 1} + \frac{1}{(z_s + 1)^2} \right| \leq \eta \left(\sum_{i \in I(s)} |w_i| \right) \beta + 2.$$

Let $W_{\max} = \max_{s \in S} \sum_{i: x_i = s} |w_i|$. The maximum absolute value of the diagonal entries of the Hessian of $G(z)$ is bounded by $\eta W_{\max} \beta + 2$. Therefore, G is β_G -smooth with $\beta_G = \eta W_{\max} \beta + 2$.

The set $\mathcal{K}_S \subseteq [0, 1]^p$. The ℓ_2 -diameter of \mathcal{K}_S is at most the ℓ_2 -diameter of the hypercube $[0, 1]^p$, which is $\sqrt{\sum_{j=1}^p (1-0)^2} = \sqrt{p} = \sqrt{|S|}$. Let $R = \sqrt{|S|}$.

Applying Lemma 3.2 to G on \mathcal{K}_S :

$$G(z_T) - G(z^*) \leq \frac{2\beta_G R^2}{T+1} \leq \frac{2(\eta W_{\max} \beta + 2)|S|}{T+1}.$$

To achieve $G(z_T) - G(z^*) < \epsilon$, we need

$$\frac{2(\eta W_{\max} \beta + 2)|S|}{T+1} \leq \epsilon,$$

which implies

$$T+1 \geq \frac{2|S|(\eta W_{\max} \beta + 2)}{\epsilon}.$$

Thus, $T = O\left(\frac{|S|(\eta W_{\max} \beta + 1)}{\epsilon}\right)$. \square

C DEFERRED PROOFS FROM SECTION 4

Proof of Corollary 4.1. Let $S = \{x_1, \dots, x_m\}$ be a set of m i.i.d. samples drawn from \mathcal{D} . We will use the hybrid learner algorithm (Algorithm 1) with $T = m$ steps. The samples for the hybrid learner are the drawn samples x_1, \dots, x_m . At each step $t \in \{1, \dots, m\}$, the hybrid learner outputs a hypothesis $h_t \in \text{conv}(\mathcal{H})$. We define the adversary function for step t of the hybrid learner as the empirical best response to h_t on the full sample set S :

$$r_t = \operatorname{argmax}_{r \in \mathcal{R}} \frac{1}{t-1} \sum_{i=1}^{t-1} u(h_t(x_i), r(x_i)).$$

This sequence of adversaries r_1, \dots, r_m is provided to the hybrid learner. r_t is chosen only using the first $t-1$ samples observed by the algorithm in order to preserve any martingale properties of the algorithm². At timestep t , after outputting h_t and observing x_t , the hybrid learner receives r_t (computed as the empirical best response to h_t on S_{t-1}) as the adversary function for the current step.

Let $h^* \in \text{conv}(\mathcal{H})$ be the optimal hypothesis in expectation against the sequence r_1, \dots, r_m : $h^* = \arg \min_{h \in \text{conv}(\mathcal{H})} \sum_{t=1}^m \mathbb{E}[u(h(x), r_t(x))]$. The hybrid learner theorem (Theorem 2.1) guarantees that with probability at least $1 - \delta'$,

$$\sum_{t=1}^m \mathbb{E}[u(h_t(x), r_t(x))] \leq \min_{h \in \text{conv}(\mathcal{H})} \sum_{t=1}^m \mathbb{E}[u(h(x), r_t(x))] + 2m \cdot \text{rad}_m(\mathcal{F}) + O\left(L \sqrt{m \log m}\right).$$

Consider the average policies $h_A = \bar{h} = \frac{1}{m} \sum_{t=1}^m h_t$ and $r_A = \bar{r} = \frac{1}{m} \sum_{t=1}^m r_t$. By convexity of u in the first argument, $\mathbb{E}[u(h_A, r)] \leq \frac{1}{m} \sum_{t=1}^m \mathbb{E}[u(h_t, r)]$. By concavity of u in the second argument, $\mathbb{E}[u(h, r_A)] \geq \frac{1}{m} \sum_{t=1}^m \mathbb{E}[u(h, r_t)]$.

Consider the saddle point gap for (h_A, r_A) : $\max_{r \in \text{conv}(\mathcal{R})} \mathbb{E}[u(h_A, r)] - \min_{h \in \text{conv}(\mathcal{H})} \mathbb{E}[u(h, r_A)]$.

$$\max_{r \in \text{conv}(\mathcal{R})} \mathbb{E}[u(h_A, r)] \leq \frac{1}{m} \sum_{t=1}^m \max_{r \in \text{conv}(\mathcal{R})} \mathbb{E}[u(h_t, r)].$$

$$\min_{h \in \text{conv}(\mathcal{H})} \mathbb{E}[u(h, r_A)] \geq \min_{h \in \text{conv}(\mathcal{H})} \frac{1}{m} \sum_{t=1}^m \mathbb{E}[u(h, r_t)].$$

So,

$$\max_{r \in \text{conv}(\mathcal{R})} \mathbb{E}[u(h_A, r)] - \min_{h \in \text{conv}(\mathcal{H})} \mathbb{E}[u(h, r_A)] \leq \frac{1}{m} \sum_{t=1}^m \left(\max_{r \in \text{conv}(\mathcal{R})} \mathbb{E}[u(h_t, r)] - \min_{h \in \text{conv}(\mathcal{H})} \mathbb{E}[u(h, r_t)] \right).$$

²although Theorem 1 doesn't rely on the martingale nature of the data.

Applying [Lemma 2.3](#), we have that for all $t > 1$, for all $r \in \mathcal{R}$

$$\left| \mathbb{E}_{x \sim \mathcal{D}}[u(h_t(x), r(x))] - \frac{1}{t-1} \sum_{s=1}^{t-1} u(h_t(x_s), r(x_s)) \right| \leq 2\text{rad}_{t-1}(\mathcal{F}) + \sqrt{\frac{\log(2T/\delta)}{t-1}}$$

Thus,

$$\frac{1}{m} \sum_{t=1}^m \max_{r \in \text{conv}(\mathcal{R})} \mathbb{E}[u(h_t, r)] \leq \frac{1}{m} \sum_{t=1}^m \mathbb{E}[u(h_t, r_t)] - \sum_{t=1}^m 2\text{rad}_{t-1}(\mathcal{F}) - \sum_{t=1}^m \sqrt{\frac{\log(2m/\delta)}{t-1}}$$

Applying the regret guarantee from [Theorem 2.1](#) to $\sum_{t=1}^m \mathbb{E}[u(h_t, r_t)] - \min_{h \in \text{conv}(\mathcal{H})} \mathbb{E}[u(h, r_t)]$ gives the desired result.

The total running time is $\text{poly}(m) \cdot (\text{cost of } \mathcal{H} \text{ ERM oracle}) + m \cdot (\text{cost of } \mathcal{R} \text{ best response oracle})$. This is oracle-efficient in $\text{poly}(m)$. \square