Proceedings of Machine Learning Research vol 75:1–42, 2018 31st Annual Conference on Learning Theory

Logistic Regression: The Importance of Being Improper

Dylan J. Foster DJFOSTER@CS.CORNELL.EDU
Cornell University

Satyen Kale SATYENKALE@GOOGLE.COM
Google Research

Haipeng Luo HAIPENGL@USC.EDU
University of Southern California

Mehryar Mohri MOHRI@CS.NYU.EDU
New York University and Google Research

Karthik Sridharan SRIDHARAN@CS.CORNELL.EDU
Cornell University

Editors: Sebastien Bubeck, Vianney Perchet and Philippe Rigollet

Abstract

Learning linear predictors with the logistic loss—both in stochastic and online settings—is a fundamental task in machine learning and statistics, with direct connections to classification and boosting. Existing "fast rates" for this setting exhibit exponential dependence on the predictor norm, and Hazan et al. (2014) showed that this is unfortunately unimprovable. Starting with the simple observation that the logistic loss is 1-mixable, we design a new efficient *improper* learning algorithm for online logistic regression that circumvents the aforementioned lower bound with a regret bound exhibiting a *doubly-exponential* improvement in dependence on the predictor norm. This provides a positive resolution to a variant of the COLT 2012 open problem of McMahan and Streeter (2012) when improper learning is allowed. This improvement is obtained both in the online setting and, with some extra work, in the batch statistical setting with high probability. We also show that the improved dependence on predictor norm is near-optimal.

Leveraging this improved dependency on the predictor norm yields the following applications: (a) we give algorithms for online bandit multiclass learning with the logistic loss with an $\tilde{O}(\sqrt{n})$ relative mistake bound across essentially all parameter ranges, thus providing a solution to the COLT 2009 open problem of Abernethy and Rakhlin (2009), and (b) we give an adaptive algorithm for online multiclass boosting with optimal sample complexity, thus partially resolving an open problem of Beygelzimer et al. (2015) and Jung et al. (2017). Finally, we give information-theoretic bounds on the optimal rates for improper logistic regression with general function classes, thereby characterizing the extent to which our improvement for linear classes extends to other parametric and even nonparametric settings.

1. Introduction

Logistic regression is a classical model in statistics used for estimating conditional probabilities (Berkson, 1944). The model, also known as *conditional maximum entropy model* (Berger et al., 1996), has been extensively studied in statistical and online learning and has been widely used in practice both for binary classification and multi-class classification in a variety of applications.

This paper presents a new study of logistic regression in online learning. The basic logistic regression problem consists of learning a linear predictor with performance measured by the *logistic loss*. In the online setting, when the hypothesis class is that of d-dimensional linear predictors with ℓ_2 norm bounded by B, there are two main algorithmic approaches to logistic regression: Online Gradient Descent (Zinkevich, 2003; Shalev-Shwartz and Singer, 2007; Nemirovski et al., 2009), which admits a regret guarantee of $O(B\sqrt{n})$ over n rounds, and Online Newton Step (Hazan et al., 2007), whose regret bound is in $O(de^B \log(n))$. While the latter bound is logarithmic in n, its poor dependence on B makes it weaker and guarantees an improvement only when $B \ll \frac{1}{2}\log(n)$. The question of whether this dependence on B could be improved was posed as an open problem in COLT 2012 by McMahan and Streeter (2012). Hazan et al. (2014) answered this in the negative, showing a lower bound of $\Omega(\sqrt{n})$ for $B \geq \Omega(\log(n))$.

The starting point for this work is a simple observation: the logistic loss, when viewed as a function of the prediction and the true outcome, is 1-mixable[1] (see Section 1.1 for definitions). This observation can be used in conjunction with Vovk's Aggregating Algorithm (Vovk, 1995), which leverages mixability in order to achieve regret bounds scaling logarithmically in an appropriate notion of complexity of the space of predictors, and can be implemented in *polynomial time* in relevant parameters using MCMC methods (Section 2). Mixability and efficient implementability open the door to fast rates for online logistic regression and related problems via *improper learning*: using predictions that may not be linear in the instances x_ts.

The power of improper learning manifests itself in solutions we present for three open problems. First, we give an *efficient* online learning algorithm that circumvents the lower bound of Hazan et al. (2014) via improper learning and attains a substantially more favorable regret guarantee of $O(d\log(Bn))$; this is a *doubly-exponential improvement* of the dependence on the scale parameter B. This algorithm provides a positive resolution to to a variant of the open problem of McMahan and Streeter (2012) where improper predictions are allowed. Second, the same technique provides an algorithm (Section 3) for the *online multiclass learning with bandit feedback problem* (Kakade et al., 2008) with an $\tilde{O}(\sqrt{n})$ relative mistake bound with respect to the multiclass logistic loss. This algorithm provides a solution to an open problem of Abernethy and Rakhlin (2009), improving upon the previous algorithm of Hazan and Kale (2011) by providing the $\tilde{O}(\sqrt{n})$ mistake bound guarantee for all possible ranges of parameter sets. Third, the technique provides a new *online multiclass boosting* algorithm (Section 4) with optimal sample complexity, thus partially resolving an open problem from (Beygelzimer et al., 2015; Jung et al., 2017) (the algorithm is sub-optimal in the number of weak learners it uses, though it is no worse in this regard than previous adaptive algorithms). For clarity of exposition, descriptions of all of these applications are given as concisely as possible without presenting the results in the most general form possible.

We further present a series of new results for batch statistical learning. We show how to convert our online improper logistic regression algorithm into a solution admitting a high-probability excess risk guarantee of $O(d\log(Bn)/n)$ (Section 5). While it is straightforward to achieve such a result in expectation using standard online-to-batch conversion techniques, the a high-probability bound is more technically challenging. We achieve this using a new technique based on a modified version of the "boosting the confidence" scheme proposed by Mehta (2017) for exp-concave losses. We also prove a lower bound showing that the logarithmic dependence on B of the guarantee of our new algorithm cannot be improved. Finally, we show how to (non-constructively) generalize the $\log(B)$

1. To the best of our knowledge, the mixability of the logistic loss has surprisingly not appeared in the literature, although similar observations were made in (Kakade and Ng, 2005).

dependence on predictor norm from linear to arbitrary function classes via sequential symmetrization and chaining arguments (Section 6). Our general bound indicates that the extent to which dependence on the predictor range B can be improved for general classes is completely determined by their (sequential) metric entropy. We also show how to extend this technique to the log loss, where we obtain a minimax rate for general function classes that uniformly improves on the minimax log loss rates in Rakhlin and Sridharan (2015a).

1.1. Preliminaries

Notation. Let $\mathbb{R}^d$ be the d-dimensional Euclidean space with $\langle \cdot, \cdot \rangle$ denoting the standard inner product in $\mathbb{R}^d$. Let $\|\cdot\|$ be a norm on $\mathbb{R}^d$ with dual norm denoted by $\|\cdot\|_\star$. In the multiclass learning problem, the input feature space is the set $\mathcal{X} = \{x \in \mathbb{R}^d | \ \|x\|_\star \le R\}$ for some unknown $R > 0$. The number of output classes is K and the set of output classes is denoted by $[K] := \{1, 2, \dots, K\}$. The set of distributions over $[K]$ is denoted Δ_K. Linear predictors are parameterized by weight matrices in $\mathbb{R}^{K \times d}$ so that for an input vector $x \in \mathcal{X}$, $Wx \in \mathbb{R}^K$ is the vector of scores assigned by W to the classes in $[K]$. For a weight matrix W and $k \in [K]$, we denote by W_k the k-th row of W. The space of parameter weight matrices is a convex set $\mathcal{W} \subseteq \{W \in \mathbb{R}^{K \times D} | \ \forall k \in [K], \|W_k\| \le B\}$ for some known parameter $B > 0$. Thus for all $x \in \mathcal{X}$ and $W \in \mathcal{W}$, we have $\|Wx\|_\infty \le BR$.

Define the softmax function $\boldsymbol{\sigma} : \mathbb{R}^K \to \Delta_K$ via $\boldsymbol{\sigma}(z)_k = \frac{e^{z_k}}{\sum_{j \in [K]} e^{z_j}}$ for $k \in [K]$. We also define a pseudoinverse for $\boldsymbol{\sigma}$ via $\boldsymbol{\sigma}^+(p)_k = \log(p_k)$ which has the property that for all $p \in \Delta_K$, we have $\boldsymbol{\sigma}(\boldsymbol{\sigma}^+(p)) = p$ and $\sum_{k \in [K]} e^{\boldsymbol{\sigma}^+(p)_k} = 1$. The multiclass logistic loss, also referred to as *softmax-cross-entropy* loss, is defined as $\ell : \mathbb{R}^K \times [K] \to \mathbb{R}$ as $\ell(z, y) := -\log(\boldsymbol{\sigma}(z)_y)$.

It will be convenient to overload notation and define a weighted version of the multiclass logistic loss function as follows: let $\mathcal{Y} := \left\{y \in \mathbb{R}^K_+ \mid \|y\|_1 \le L\right\}$ for some known parameter $L > 0$. Then the weighted multiclass logistic loss function $\ell : \mathbb{R}^K \times \mathcal{Y} \to \mathbb{R}$ is defined by $\ell(z, y) = -\sum_{k \in [K]} y_k \log(\boldsymbol{\sigma}(z)_k)$. It can also be seen by straightforward manipulation that the above definition is equivalent to $\ell(z, y) = \sum_{j \in [K]} y_j \log\left(1 + \sum_{k \ne j} e^{z_k - z_j}\right)$.

In the binary classification setting, the standard definition of the logistic loss function is (superficially) different: the label set is is $\{-1, 1\}$, and the logistic loss $\ell : \mathbb{R} \times \{-1, 1\} \to \mathbb{R}$ is defined as $\ell_{\text{bin}}(z, y) = \log(1 + \exp(-yz))$. Linear predictors are parameterized by weight vectors $w \in \mathbb{R}^d$ with $\|w\|_2 \le B$, and the loss for a predictor with parameter $w \in \mathbb{R}^d$ on an example $(x, y) \in \mathbb{R}^d \times \{-1, 1\}$ is $\ell_{\text{bin}}(\langle w, x \rangle, y)$. This loss can be equivalently viewed in the multiclass framework above setting $K = 2$, $\mathcal{W} = \{W \in \mathbb{R}^{2 \times d} | \ \|W_1\|_2 \le B, W_2 = 0\}$, and mapping the labels $1 \mapsto 1$ and $-1 \mapsto 2$.

Finally, we make frequent use of a smoothing operator $\text{smooth}_\mu : \Delta_K \to \Delta_K$ for a parameter $\mu \in [0, 1/2]$, defined via $\text{smooth}_\mu(p) = (1 - \mu)p + \mu \mathbf{1}/K$ where $\mathbf{1} \in \mathbb{R}^K$ is the all ones vector. We use the notation $\mathbf{1}[\cdot]$ to denote the indicator random variable for an event.

Online multiclass logistic regression. We use the following multiclass logistic regression protocol. Learning proceeds over a series of rounds indexed by $t = 1, \dots, n$. In each round t, nature provides $x_t \in \mathcal{X}$, and the learner selects prediction $\hat{z}_t \in \mathbb{R}^K$ in response. Then nature provides an outcome $y_t \in [K]$ or $y_t \in \mathcal{Y}$, depending on application, and the learner incurs multiclass logistic loss $\ell(\hat{z}_t, y_t)$. The regret of the learner is defined to be $\sum_{t=1}^n \ell(\hat{z}_t, y_t) - \inf_{W \in \mathcal{W}} \sum_{t=1}^n \ell(Wx_t, y_t)$.

The learner is said to be *proper* if it generates $\hat{z}_t$ by choosing a weight matrix $W_t \in \mathcal{W}$ *before* observing the pair (x_t, y_t) and setting $\hat{z}_t = W_t x_t$. This is the standard protocol when the problem is viewed as an instance of online convex optimization, and is the setting for previous investigations

into fast rates for logistic regression (Bach, 2010; McMahan and Streeter, 2012; Bach and Moulines, 2013; Bach, 2014), including the negative result of Hazan et al. (2014). The more general online learning setting that is described above allows *improper* learners which may generate $\hat{z}_t$ arbitrarily using knowledge of x_t.

Fast rates and mixability. Conditions under which *fast rates* for online/statistical learning (meaning that average regret or generalization error scales as $\tilde{O}(1/n)$ rather than $O(1/\sqrt{n})$) are achievable have been studied extensively (see (Van Erven et al., 2015) and the references therein). For the purpose of this paper, a rather general condition on the structure of the problem that leads to fast rates is Vovk's notion of *mixability* (Vovk, 1995), which we define in an abstract setting below. Consider a prediction problem where the set of outcomes is $\mathcal{Y}$ and the set of predictions is $\mathcal{Z}$, and the loss of a prediction on an outcome is given by a function $\ell : \mathcal{Z} \times \mathcal{Y} \to \mathbb{R}$. For a parameter $\eta > 0$, the loss function ℓ is said to be η-mixable if for any probability distribution π over $\mathcal{Z}$, there exists a *"mixed"* prediction $z_\pi \in \mathcal{Z}$ such that for all possible outcomes $y \in \mathcal{Y}$, we have $\mathbb{E}_{z\sim\pi}[\exp(-\eta\ell(z,y))] \le \exp(-\eta\ell(z_{\text{mix}},y))$.

Now suppose that we are given a finite reference class of predictors $\mathcal{F}$ consisting of functions $f : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{X}$ is the input space. The problem of online learning over $\mathcal{F}$ with an η-mixable loss function admits an *improper* algorithm, viz. Vovk's Aggregating Algorithm (Vovk, 1995), with regret bounded by $\frac{\log|\mathcal{F}|}{\eta}$, a *constant* independent of the number of prediction rounds n. The algorithm simply runs the standard exponential weights/Hedge algorithm (Cesa-Bianchi and Lugosi, 2006) with learning rate set to η. In each round t, given an input x_t, the distribution over $\mathcal{F}$ generated by the exponential weights algorithm induces a distribution over $\mathcal{Z}$ via the outputs of the predictors on x_t, and the Aggregating Algorithm plays the mixed prediction for this distribution over $\mathcal{Z}$. Finally, if $\mathcal{F}$ is infinite, under appropriate conditions on $\mathcal{F}$ fast rates can be obtained by running a continuous version of the same algorithm. This is the strategy we employ in this paper for the logistic loss.

2. Improved Rates for Online Logistic Regression

We start by providing a simple proof of the mixability of the multiclass logisitic loss function for the case when the outcomes y is a class in $[K]$ (i.e. the unweighted case).

Proposition 1 *The unweighted multiclass logistic loss $\ell : \mathbb{R}^K \times [K] \to \mathbb{R}$ defined as $\ell(z,y) = -\log(\boldsymbol{\sigma}(z)_y)$ is 1-mixable.*

Proof The proof is by construction. Given a distribution π on $\mathbb{R}^K$, define $z_\pi = \boldsymbol{\sigma}^+(\mathbb{E}_{z\sim\pi}[\boldsymbol{\sigma}(z)])$. Now, for any $y \in [K]$, we have $\mathbb{E}_{z\sim\pi}[\exp(-\ell(z,y))] = \mathbb{E}_{z\sim\pi}[\boldsymbol{\sigma}(z)_y] = \boldsymbol{\sigma}(z_\pi)_y = \exp(-\ell(z_\pi,y))$. The second equality above uses the fact that for any $p \in \Delta_K$, $\boldsymbol{\sigma}(\boldsymbol{\sigma}^+(p)) = p$. Thus, ℓ is 1-mixable. ■

With a little more work, we can prove that the weighted multiclass logistic loss function is also mixable with a constant that inversely depends on the total weight. The proof appears in Appendix A.

Proposition 2 *Let $\mathcal{Y} := \left\{y \in \mathbb{R}_+^K \mid \|y\|_1 \le L\right\}$ for some parameter $L > 0$. The weighted multiclass logistic loss $\ell : \mathbb{R}^K \times \mathcal{Y} \to \mathbb{R}$ defined as $\ell(z,y) = -\sum_{k\in[K]} y_k \log(\boldsymbol{\sigma}(z)_k)$ is $\frac{1}{L}$-mixable. For any distribution π on $\mathbb{R}^K$, the mixed prediction $z_\pi = \boldsymbol{\sigma}^+(\mathbb{E}_{z\sim\pi}[\boldsymbol{\sigma}(z)])$ certifies $\frac{1}{L}$-mixability of ℓ.*

We are now ready to state a variant of Vovk's Aggregating Algorithm, Algorithm 1 for the online multiclass logistic regression problem from Section 1.1, operating over a class of linear predictors

parameterized by weight matrices W in some convex set $\mathcal{W}$. The algorithm and its regret bound (proved in Appendix A) are given in some generality that is useful for applications.

Algorithm 1

procedure (decision set $\mathcal{W}$, smoothing parameter $\mu \in [0, 1/2]$.)
  Initialize P_1 to be the uniform distribution over $\mathcal{W}$.
  for $t = 1, \ldots, n$ **do**
    Obtain x_t and predict $\hat{z}_t = \boldsymbol{\sigma}^+(\mathrm{smooth}_\mu(\mathbb{E}_{W\sim P_t}[\boldsymbol{\sigma}(Wx_t)]))$.
    Obtain y_t and define P_{t+1} as the distribution over $\mathcal{W}$ with density $P_{t+1}(W) \propto \exp(-\frac{1}{L}\sum_{s=1}^{t}\ell(Wx_s, y_s))$.
  end for
end procedure

Theorem 3 *The regret of Algorithm 1 is bounded by*

$$\sum_{t=1}^{n}\ell(\hat{z}_t, y_t) - \inf_{W\in\mathcal{W}}\sum_{t=1}^{n}\ell(Wx_t, y_t) \le 5LD_{\mathcal{W}}\cdot\log\left(\frac{BRn}{D_{\mathcal{W}}}+e\right) + 2\mu\sum_{t=1}^{n}\|y_t\|_1, \tag{1}$$

where $D_{\mathcal{W}} := \dim(\mathcal{W}) \le dK$ is the linear-algebraic dimension of $\mathcal{W}$. The predictions $(\hat{z}_t)_{t\le n}$ generated by the algorithm satisfy $\|\hat{z}_t\|_\infty \le \log(K/\mu)$.

Increasing the smoothing parameter μ only degrades the performance of Algorithm 1. However, smoothing ensures that each prediction $\hat{z}_t$ is bounded, which is important for our applications.

For the special case of multiclass prediction when $y \in [K]$, this algorithm enjoys a regret bound of $O(dK\log(\frac{BRn}{dK}+e))$. It thus provides a positive resolution to the open problem of McMahan and Streeter (2012) (in fact, with an exponentially better dependence on B than what the open problem asked for), using improper predictions to circumvent the lower bound of Hazan et al. (2014).

Turning to efficient implementation, it has been noted (e.g. (Hazan et al., 2007)) that log-concave sampling or integration techniques (Lovász and Vempala, 2006, 2007) can be applied to compute the expectation in Algorithm 1 in polynomial time. The following proposition makes this idea rigorous[2] and is proven formally in Appendix B. We note that this is not a practical algorithm, however, and obtaining a truly practical algorithm with a modest polynomial dependence on the dimension is a significant open problem.

Proposition 4 *Algorithm 1 can be implemented approximately so that the regret bound (1) is obtained up to additive constants in time* $\mathrm{poly}(d, n, B, R, K, L)$.

Finally, to conclude this section we state a lower bound, which shows that the $\log(B)$ factor in the regret bound in Theorem 3 cannot be improved for most values of B. This lower bound is by reduction to learning halfspaces with a margin in a Perceptron-type setting: We first show that Algorithm 1 can be configured to give a mistake bound of $O(d\log(\log(n)/\gamma))$ for binary classification with halfspaces and margin γ,[3] then give a lower bound against this type of rate.

For simplicity, the lower bound is only stated in the binary outcome settting and we use the standard definition of the binary logistic loss, ℓ_{bin} from Section 1.1. The proof is in Appendix A.

2. A subtlety is that since $\hat{z}_t$ is evaluated inside the nonlinear logistic loss we cannot exploit linearity of expectation.
3. It is a folklore result that this type of margin bound can be obtained by running a variant of the ellipsoid method online.

Theorem 5 (Lower bound) *Consider the binary logistic regression problem over the class of linear predictors with parameter set* $\mathcal{W} = \{w \in \mathbb{R}^d | \ \|w\|_2 \le B\}$ *with* $B = \Omega(\sqrt{d}\log(n))$. *Then for any algorithm for prediction with the binary logistic loss, there is a sequence of examples* $(x_t, y_t) \in \mathbb{R}^d \times \{-1, 1\}$ *for* $t \in [n]$ *with* $\|x_t\|_2 \le 1$ *such that the regret of the algorithm is* $\Omega\left(d \log\left(\frac{B}{\sqrt{d}\log(n)}\right)\right)$.

3. Application: Bandit Multiclass Learning

The now apply our techniques to the *bandit multiclass* problem. This problem, first studied by Kakade et al. (2008), considers the protocol of online multiclass learning in Section 1.1 with nature choosing $y_t \in [K]$ in each round, but with the added twist of bandit feedback: in each round, the learner predicts a class $\hat{y}_t \sim p_t$ and receives feedback only on whether the prediction was correct or not, i.e. $\mathbb{1}[\hat{y}_t \neq y_t]$. The goal is to minimize regret with respect to a reference class of linear predictors, using some appropriate surrogate loss function for the 0-1 loss.

Kakade et al. (2009) used the multiclass hinge loss $\ell_{\text{hinge}}(W, (x_t, y_t)) = \max_{k \in [K] \setminus \{y_t\}} [1 + \langle W_k, x_t \rangle - \langle W_{y_t}, x_t \rangle]_+$ and gave an algorithm based on the multiclass Perceptron algorithm achieving $O(n^{2/3})$ regret. For a Lipschitz continuous surrogate loss function, running the EXP4 algorithm (Auer et al., 2002) on a suitable discretization of the space of all linear predictors obtains $\tilde{O}(\sqrt{n})$ regret, albeit very inefficiently, i.e. with exponential dependence on the dimension. In COLT 2009, Abernethy and Rakhlin (2009) posed the open problem of obtaining an *efficient* algorithm for the problem with $O(\sqrt{n})$ regret. Specifically, they suggested the multiclass logistic loss as an appropriate surrogate loss function for the problem. Hazan and Kale (2011) solved the open problem and obtained an algorithm, Newtron, based on the Online Newton Step algorithm (Hazan et al., 2007) with $\tilde{O}(\sqrt{n})$ regret for the case when norm of the linear predictors scales at most logarithmically in n. Beygelzimer et al. (2017) also solved the open problem presenting a different algorithm called SOBA. SOBA is analyzed using a different family of surrogate loss functions parameterized by a scalar $\eta \in [0, 1]$ with $\eta = 0$ corresponding to the hinge loss and $\eta = 1$ corresponding to the squared hinge loss. For all values of $\eta \in [0, 1]$, SOBA simultaneously obtains relative bound mistake bounds of $\tilde{O}(\frac{1}{\eta}\sqrt{n})$ with the comparator's loss measured with respect to the corresponding loss function.

Now we present an algorithm, OBAMA (for *Online Bandit Aggregation Multiclass Algorithm*), depicted in Algorithm 2 in Appendix A.2, that obtains an $\tilde{O}(\sqrt{n})$ relative mistake bound for the multiclass logistic loss, thus providing another solution to the open problem of Abernethy and Rakhlin (2009). The mistake bound of OBAMA trumps that of Newtron, since both algorithms rely on the same loss function, and OBAMA obtains an $\tilde{O}(\sqrt{n})$ relative mistake bound on a larger range of parameter values compared to Newtron. While SOBA also has an $\tilde{O}(\sqrt{n})$ relative mistake bound, the two bounds are incomparable since they are relative to the comparator's loss measured using different loss functions.

Theorem 6 *There is a setting of the smoothing parameter* μ *such that OBAMA enjoys the following mistake bound:*

$$\sum_{t=1}^{n} \mathbb{1}[\hat{y}_t \neq y_t] \le \inf_{W \in \mathcal{W}} \sum_{t=1}^{n} \ell(W x_t, y_t) + O\left(\min\left\{dK^2 e^{2BR} \log\left(\tfrac{BRn}{dK} + e\right),\ \sqrt{dK^2 \log\left(\tfrac{BRn}{dK} + e\right) n}\right\}\right).$$

This bound significantly improves upon that of Newtron (Hazan and Kale, 2011), which is of order $O(dK^3 \min\{\exp(BR)\log(n), BRn^{\frac{2}{3}}\})$ under the same setting and surrogate loss. The proof of Theorem 6 appears in Appendix A.2.

4. Application: Online Multiclass Boosting

Another application of our techniques is to derive adaptive online boosting algorithms with optimal sample complexity, which improves the AdaBoost.OL algorithm of Beygelzimer et al. (2015) for the binary classification setting as well as its multiclass extension AdaBoost.OLM of Jung et al. (2017). We state our improved online boosting algorithm in the multiclass setting for maximum generality, following the exposition and notation of Jung et al. (2017) fairly closely.

We consider the following online multiclass prediction setting with 0-1 loss. In each round t, $t = 1, \dots, n$, the learner receives an instance $x_t \in \mathcal{X}$, then selects a class $\hat{y}_t \in [K]$, and finally observes the true class $y_t \in [K]$. The goal is to minimize the total number of mistakes $\sum_{t=1}^{n} \mathbb{1}\{\hat{y}_t \neq y_t\}$.

In the boosting setup, we are interested in obtaining strong mistake bounds with the help of *weak learners*. Specifically, the learner is given access to N copies of a weak learning algorithm for a cost-sensitive classification task. Each weak learner $i \in [N]$ works in the following protocol: for time $t = 1, \dots, n$, 1) receive $x_t \in \mathcal{X}$ and cost matrix $C_t^i \in \mathcal{C}$; 2) predict class $l_t^i \in [K]$; 3) receive true class $y_t \in [K]$ and suffer loss $C_t^i(y_t, l_t^i)$. Here $\mathcal{C}$ is some fixed cost matrices class and we follow (Jung et al., 2017) to restrict to $\mathcal{C} = \left\{C \in \mathbb{R}_+^{K \times K} \mid \forall y \in [K], C(y,y) = 0 \text{ and } \|C(y,\cdot)\|_1 \leq 1\right\}$.

To state the weak learning condition, we define a randomized baseline $u_{\gamma,y} \in \Delta_K$ for some edge parameter $\gamma \in [0,1]$ and some class $y \in [K]$, so that $u_{\gamma,y}(k) = (1-\gamma)/K$ for $k \neq y$ and $u_{\gamma,y}(k) = (1-\gamma)/K + \gamma$ for $k = y$. In other words, $u_{\gamma,y}$ puts equal weight to all classes except for the class y which gets γ more weight. The assumption we impose on the weak learners is then that their performance is comparable to that of a baseline which always picks the true class with slightly higher probability than the others, formally stated below.

Definition 7 (Weak Learning Condition (Jung et al., 2017)) *An environment and a learner outputting $(l_t)_{t \leq n}$ satisfy the multiclass weak learning condition with edge γ and sample complexity S if for all outcomes $(y_t)_{t \leq n}$ and cost matrices $(C_t)_{t \leq n}$ from the set $\mathcal{C}$ adaptively chosen by the environment, we have*[4] $\sum_{t=1}^{n} C_t(y_t, l_t) \leq \sum_{t=1}^{n} \mathbb{E}_{k \sim u_{\gamma,y_t}}[C_t(y_t, k)] + S$.

4.1. AdaBoost.OLM++

The high level idea of our algorithm is similar to that of AdaBoost.OL and AdaBoost.OLM: find a weighted combination of weak learners to minimize some version of the logistic loss in an online manner. The key difference is that previous works use simple gradient descent to find the weight for each weak learner via proper learning, while we translate the problem into the framework discussed in Section 2 and deploy the proposed improper learning techniques to obtain an improvement on the regret for learning these weights, which then leads to better and in fact optimal sample complexity.

Another difference compared to (Jung et al., 2017) is that the logistic loss we use here is more suitable for the multiclass problem than the one they use.[5] This simple modification leads to exponential improvement in the number of classes K for the number of weak learners required.

We now describe our algorithm, called AdaBoost.OLM++, in more detail (see Algorithm 3 in Appendix A.3). We denote the i-th weak learner as WL^i, which is seen as a stateful object and supports two operations: $\mathrm{WL}^i.\mathrm{Predict}(x, C)$ predicts a class given an instance and a cost matrix but does not update its internal state; $\mathrm{WL}^i.\mathrm{Update}(x, C, y)$ updates the state given an instance, a cost

4. This is in fact a weaker weak learning condition than that of (Jung et al., 2017), which also allows weights.
5. The loss Jung et al. (2017) use moves the sum over the incorrect classes outside the log, that is, $\ell(z, y) = \sum_{k \neq y} \log(1 + e^{z_k - z_y})$.

matrix and the true class y. To keep track of the state we use the notation WL_t^i to imply that it has been updated for $t-1$ times.

For each weak learner, the algorithm also maintains an instance of Algorithm 1, denoted by Logistic^i, to improperly learn the aforementioned weight for this weak learner. Similarly, we use $\text{Logistic}^i.\text{Predict}(x)$ to denote the prediction step (step 4) in Algorithm 1 and $\text{Logistic}^i.\text{Update}(x,y)$ to denote the update step (i.e. step 5). The notation Logistic_t^i again implies that the state has been updated for $t-1$ times.

Our algorithm maintains a variable $s_t^i \in \mathbb{R}^K$ which stands for the weighted accumulated scores of the first i weak learners for instance x_t. When updating s_t^i from s_t^{i-1} given the prediction $l_t^i \in [K]$ of weak learner i, our goal is to have the total loss $\sum_{t=1}^n \ell(s_t^i, y_t)$ close to $\sum_{t=1}^n \ell(s_t^{i-1} + \alpha e_{l_t^i}, y_t)$ for the best α within some range ($[-2,2]$ suffices). Previous works therefore try to learn this weight α via standard online learning approaches. However, realizing $s_t^{i-1} + \alpha e_{l_t^i}$ can be written as $W\widetilde{x}_t^i$ for $W = (\alpha I_{K\times K}, I_{K\times K}) \in \mathbb{R}^{K\times 2K}$ and $\widetilde{x}_t^i = (e_{l_t^i}, s_t^{i-1}) \in \mathbb{R}^{2K}$, in light of Theorem 3 we can in fact apply Algorithm 1 to learn s_t^i if we let the decision set be $\mathcal{W} = \left\{(\alpha I_{K\times K}, I_{K\times K}) \in \mathbb{R}^{K\times 2K} \mid \alpha \in [-2,2]\right\}$. To make sure that $\widetilde{x}_t^i$ has bounded norm, we also set the smoothing parameter μ to be $1/n$.

With the weighted score s_t^i, the prediction coming from the first i weak learner is naturally define as $\hat{y}_t^i = \arg\max_k s_t^i(k)$, the class with the largest score. As in AdaBoost.OL and AdaBoost.OLM, these predictions $(\hat{y}_t^i)_{i\le N}$ are treated as N experts and the final prediction y_t is determined by the classic Hedge algorithm (Freund and Schapire, 1997) over these experts (Lines 13 and 18).

Finally, the cost matrices fed to the weak learners are closely related to the gradient of the loss function. Formally, define the auxiliary cost matrix $\widehat{C}_t^i$ such that $\widehat{C}_t^i(y,k) = \frac{\partial \ell(z,y)}{\partial z_k}|_{z=s_t^{i-1}}$, which is simply $\boldsymbol{\sigma}(s_t^{i-1})_k$ for $k \neq y$ and $\boldsymbol{\sigma}(s_t^{i-1})_y - 1$ otherwise. The actual cost matrix is then a translated and scaled version of $\widehat{C}_t^i(y,k)$ so that it belongs to the class $\mathcal{C}$:

$$C_t^i(y,k) = \frac{1}{K}\left(\widehat{C}_t^i(y,k) - \widehat{C}_t^i(y,y)\right) \in \mathcal{C}. \tag{2}$$

We now give a mistake bound for AdaBoost.OLM++, which holds even without the weak learning condition and is adaptive to the empirical edge of the weak learners.[6] All proofs in this section appear in Appendix A.3.

Theorem 8 *With probability at least $1-\delta$, the predictions $(\hat{y}_t)_{t\le n}$ generated by Algorithm 3 satisfy*

$$\sum_{t=1}^n \mathbb{1}\{\hat{y}_t \neq y_t\} = \tilde{O}\left(\frac{n}{\sum_{i=1}^N \gamma_i^2} + \frac{N}{\sum_{i=1}^N \gamma_i^2}\right), \tag{3}$$

where $\gamma_i = \frac{\sum_{t=1}^n \widehat{C}_t^i(y_t, l_t^i)}{\sum_{t=1}^n \widehat{C}_t^i(y_t, y_t)} \in [-1,1]$ is the empirical edge of weak learner i.

We can now relate the empirical edges to the edge defined in the weak learning condition.

Proposition 9 *Suppose all weak learners satisfy the weak learning condition with edge γ and sample complexity S (Definition 7). Then with probability at least $1-\delta$, the predictions $(\hat{y}_t)_{t\le n}$ generated by Algorithm 3 satisfy*

$$\sum_{t=1}^n \mathbb{1}\{\hat{y}_t \neq y_t\} = \tilde{O}\left(\frac{n}{N\gamma^2} + \frac{1}{\gamma^2} + \frac{KS}{\gamma}\right). \tag{4}$$

Thus, to achieve a target error rate ε, it suffices to take $N = \tilde{\Omega}\left(\frac{1}{\varepsilon\gamma^2}\right)$ and $n = \tilde{\Omega}\left(\frac{1}{\varepsilon\gamma^2} + \frac{KS}{\varepsilon\gamma}\right)$.

6. We use notation $\tilde{O}$ and $\tilde{\Omega}$ to hide dependence logarithmic in n, N, K and $1/\delta$.

Comparison with prior algorithms Compared to (Jung et al., 2017), our sample complexity on n improves the dependence on K (for OnlineMBBM) and also ε and γ (for AdaBoost.OLM), and is in fact optimal according to their lower bound (Theorem 4). Our bound on the number of weak learners, on the other hand, is weaker compared to the non-adaptive algorithm OnlineMBBM (which has a logarithmic dependence on $1/\varepsilon$), but is still much stronger than that of AdaBoost.OLM since it improves the dependence on K from linear to $\log(K)$. Although not stated explicitly, our results also apply to the binary setting considered in (Beygelzimer et al., 2015) and improve the sample complexity of their AdaBoost.OL algorithm to the optimal bound $\tilde{\Omega}(\frac{1}{\varepsilon\gamma^2} + \frac{S}{\varepsilon\gamma})$. Overall, our results significantly reduce the gap between optimal and adaptive online boosting algorithms.

As a final remark, the same technique used here also readily applies to the online boosting setting for the multi-label ranking problem recently studied by Jung and Tewari (2018). Details are omitted.

5. High-Probability Online-to-Batch Conversion

Before the present work, the issue of improving on the $O(e^B)$ fast rate for logistic regression was not addressed even in the batch statistical learning setting. This is perhaps not surprising since the proper lower bound proven by Hazan et al. (2014) applies in this setting as well.

Using our improved online algorithm as a starting point, we will show that it is possible to obtain a predictor with excess risk bounded in *high-probability* by $O(d\log(Bn)/n)$ for the batch logistic regression problem. While it is quite straightforward to show that the standard online-to-batch conversion technique applied to Algorithm 1 provides a predictor that obtains such an excess risk bound in expectation, obtaining a high-probability bound is far less trivial, as we must ensure that deviations scale at most as $O(\log(B))$. Indeed, a different algorithm is necessary, and our approach is to use a modified version of the "boosting the confidence" scheme proposed by Mehta (2017) for exp-concave losses. Our main result for linear classes is Theorem 10 below. For notational convenience will use the shorthand $\mathbb{E}_{(x,y)}[\cdot]$ to denote $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\cdot]$ where $\mathcal{D}$ is an unknown distribution over $\mathcal{X} \times [K]$.

Theorem 10 (High-probability excess risk bound) *Let $\mathcal{D}$ be an unknown distribution over $\mathcal{X} \times [K]$. For any $\delta > 0$ and n samples $\{(x_t, y_t)\}_{t=1}^n$ drawn from $\mathcal{D}$, we can construct $g : \mathcal{X} \to \mathbb{R}^K$ such that w.p. at least $1-\delta$, the excess risk $\mathbb{E}_{(x,y)}[\ell(g(x), y)] - \inf_{W\in\mathcal{W}} \mathbb{E}_{(x,y)}[\ell(Wx, y)]$ is bounded by*

$$O\left(\frac{dK\log\left(\frac{BRn}{\log(1/\delta)dK} + e\right)\log\left(\frac{1}{\delta}\right) + \log(Kn)\log\left(\frac{\log(n)}{\delta}\right)}{n}\right).$$

Theorem 10 is a consequence of the more general Theorem 26—stated and proved in Appendix A.4—concerning prediction with the log loss $\ell_{\log} : \Delta_K \times [K] \to \mathbb{R}$ defined as $\ell_{\log}(p, y) = -\log(p_y)$. The theorem asserts that we can convert any online algorithm for multiclass learning with log loss that predicts distributions in Δ_K for any given input into a predictor for the batch problem with an excess bound essentially equal to the average regret with high probability.

6. Beyond Linear Classes

We now turn to the question of extending our techniques to general, non-linear predictors. We characterize the minimax regret for learning with the unweighted multiclass logistic loss[7] for a

7. We only consider the unweighted case in this section to avoid excessive notation.

general class $\mathcal{F}$ of predictors $f : \mathcal{X} \to \mathbb{R}^K$ and abstract instance space $\mathcal{X}$. This is the same setting as in Section 1.1, but with the benchmark class $\{x \mapsto Wx \mid W \in \mathcal{W}\}$ replaced with an arbitrary class $\mathcal{F}$, where the loss of a predictor $f \in \mathcal{F}$ on an example $(x, y) \in \mathcal{X} \times [K]$ is given by $\ell(f(x), y) = -\log(\boldsymbol{\sigma}(f(x))_y)$. The bounds we present in this section—based on sequential covering numbers—substantially increase the scope of results from earlier sections. We note however that they are purely information-theoretic results in the vein of Rakhlin et al. (2015a); Rakhlin and Sridharan (2014, 2015a), not algorithmic.

Recall that the minimax regret—the best regret bound achievable against the worst-case adaptively chosen sequence of examples—is given by

$$\mathcal{V}_n(\mathcal{F}) = \left\langle\!\!\left\langle \sup_{x_t \in \mathcal{X}} \inf_{\hat{z}_t \in \mathbb{R}^K} \max_{y_t \in [K]} \right\rangle\!\!\right\rangle_{t=1}^{n} \left[\sum_{t=1}^{n} \ell(\hat{z}_t, y_t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t) \right], \tag{5}$$

where, following Rakhlin et al. (2015a), the $\langle\!\langle \star \rangle\!\rangle_{t=1}^{n}$ notation indicates sequential application of the operators contained within n times.

Our bounds on $\mathcal{V}_n(\mathcal{F})$ exploit that the logistic loss can be viewed in two complementary ways: since the loss is 1-mixable, one can attain a bound of $O(\log|\mathcal{F}|)$ for finite function classes $\mathcal{F}$ using the Aggregating Algorithm, and since the loss is 2-Lipschitz (in the ℓ_∞ norm), for more complex classes one can obtain bounds using sequential complexity measures such as sequential Rademacher complexity (Rakhlin et al., 2015a). Our analysis uses both properties simultaneously.

Here is a sketch of the idea for a special case in which we make the simplifying assumption that $\mathcal{F}$ admits a pointwise cover. Recall that a pointwise cover for $\mathcal{F}$ at scale γ is a set V of functions $g : \mathcal{X} \to \mathbb{R}^K$ such that for any $f \in \mathcal{F}$, there is a $g \in V$ such that for all $x \in \mathcal{X}$, $\|f(x) - g(x)\|_\infty \le \gamma$. Let $N(\gamma)$ be the size of a minimal such cover. For every $g \in V$, let $\mathcal{F}_g = \{f \in \mathcal{F} \mid \sup_{x \in \mathcal{X}} \|f(x) - g(x)\|_\infty \le \gamma\}$. Now consider the following two-level algorithm. Within each $\mathcal{F}_g$, run the minimax online learning algorithm for this set, then aggregate the predictions for these algorithms over all $g \in V$ using the Aggregating Algorithm to produce the final prediction $\hat{z}_t$.

For each $g \in V$, the regret of the minimax optimal online learning algorithm competing with $\mathcal{F}_g$ can be bounded by the sequential Rademacher complexity of $\mathcal{F}_g$, which can in turn be bounded by the Dudley integral complexity using that the loss is 2-Lipschitz and that the L_∞ "radius" of $\mathcal{F}_g$ is at most γ (Rakhlin et al., 2015a). The Aggregating Algorithm, via 1-mixability, ensures a regret bound of $\log N(\gamma)$ against any sub-algorithm. This algorithm has the following regret bound:

$$\sum_{t=1}^{n} \ell(\hat{z}_t, y_t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t) \le \inf_{\gamma > 0} \left\{ \log N(\gamma) + \inf_{\alpha > 0} \left\{ 8\alpha n + 24\sqrt{n} \int_{\alpha}^{\gamma} \sqrt{\log N(\delta)}\, d\delta \right\} \right\}. \tag{6}$$

This procedure already yields the same bound for the d-dimensional linear setting explored earlier: For a class $x \mapsto Wx$ with $\|W\|_2 \le B$ it holds that $N(\gamma) \le \left(\frac{B}{\gamma}\right)^{Kd}$, and we can use this bound in conjunction with (6) and the setting $\alpha = \gamma = 1/n$ to get the desired regret bound of $O(dK \log(Bn/dK))$ on the minimax regret.

Unfortunately, this simple approach fails on classes $\mathcal{F}$ for which the pointwise cover is infinite. This can happen for well-behaved function classes that have small *sequential covering number*, even though bounded sequential covering number is sufficient for learnability in the online setting (Rakhlin et al., 2015a). We now provide a bound that replaces the pointwise covering number in the argument above with the sequential covering number. The definition of the L_2 covering number $\mathcal{N}_2(\alpha, \ell \circ \mathcal{F})$ that appears in the statement of the theorem below is based on a multiclass generalization of a sequential cover and appears in Appendix A.5 due to space limitations.

Theorem 11 *Any function class $\mathcal{F}$ that is uniformly bounded*[8] *over $\mathcal{X}$ enjoys the minimax value bound:*

$$\mathcal{V}_n(\mathcal{F}) \le \inf_{\gamma>0}\left\{\log \mathcal{N}_2(\gamma, \ell\circ\mathcal{F}) + \inf_{\gamma\ge\alpha>0}\left\{8\alpha n + 24\sqrt{n}\int_\alpha^\gamma \sqrt{\log(\mathcal{N}_2(\delta,\ell\circ\mathcal{F})\cdot n)}d\delta\right\}\right\} + 4. \quad (7)$$

This rate overcomes several shortcomings faced when trying to apply previously developed minimax bounds for general function classes to the logistic loss. Specifically, Rakhlin et al. (2015a) applies to our logistic loss setup but ignores the curvature of the loss and so cannot obtain fast rates, while Rakhlin and Sridharan (2015a) obtain fast rates but scale with e^B, where B is a bound on the magnitude of the predictions, because they use exp-concavity.

Our general function class bound is especially interesting in light of rates obtained in Rakhlin and Sridharan (2014) for the square loss, which are also based on sequential covering numbers. In the binary case the bound (7) precisely matches the general class bound of (Rakhlin and Sridharan, 2014, Lemma 5) in terms of dependence on the sequential metric entropy. However, (7) does not depend on B explicitly, whereas their Lemma 5 bound for the square loss explicitly scales with B^2. In other words, compared to other common curved losses the logistic loss has a desirable property:

The minimax rate for logistic regression only depends on scale through capacity of the class $\mathcal{F}$.

Let us examine some rates obtained from this bound for concrete settings. These examples are based on sequential covering bounds that appeared in Rakhlin and Sridharan (2014, 2015a).

Example 1 (Sparse linear predictors) *Let $\mathcal{G} = \{g_1,\dots,g_M\}$ be a set of M functions $g_i : \mathcal{X} \mapsto [-B,B]$. Define $\mathcal{F}$ to be the set of all convex combinations of at most s out of these M functions. The sequential covering number can be easily upper bounded: We can choose s out of M functions in $\binom{M}{s}$ ways. For each choice, the sequential covering number for the set of all convex combinations of these s bounded functions at scale β is bounded as $\frac{B^s}{\beta^s}$. Hence, using that the logistic loss is Lipschitz, we conclude that $\mathcal{N}_2(\mathcal{F},\beta) = O\big(\big(\frac{eM}{s}\big)^s\cdot\beta^{-s}B^s\big)$. Using this bound with Theorem 11 we obtain $\mathcal{V}_n(\mathcal{F}) \le O\left(s\log(BMn/s)\right)$.*

The bounds from Rakhlin et al. (2015a); Rakhlin and Sridharan (2014, 2015a) either pay $O(B\sqrt{n})$ or $O(e^B)$ on this example, whereas the new bound from (7) correctly obtains $O(\log(B))$ scaling.

Example 2 (Besov classes) *Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^d$. Let $\mathcal{F}$ be the ball of radius B in Besov space $B^s_{p,q}(\mathcal{X})$. When $s > d/p$ it can be shown that the pointwise log covering number of the space at scale β is of order $(B/\beta)^{d/s}$. When $p \ge 2$ one can obtain a sequential covering number bound of order $(B/\beta)^p$ (Rakhlin and Sridharan, 2015b, Section 5.8). These bounds imply:*

1. *If $s \ge d/2$, then $\mathcal{V}_n(\mathcal{F}) \le \tilde{O}\left(B^{\frac{2d}{d+2s}} n^{\frac{d}{d+2s}}\right)$.*
2. *$s < d/2$, then: if $p > 1 + d/2s$ then $\mathcal{V}_n(\mathcal{F}) \le \tilde{O}\left(Bn^{1-\frac{s}{d}}\right)$; if not, $\mathcal{V}_n(\mathcal{F}) \le \tilde{O}(Bn^{1-1/p})$.*

Remark 12 *Using the machinery from the previous section, we can generically lift the general function class bounds given by Theorem 11 to high-probability bounds for the i.i.d. batch setting.*

8. Boundedness is required to apply the minimax theorem, but does not explicitly enter our quantitative bounds.

7. General Function Class Bounds for Log Loss

In this section we show that our analysis techniques can also be used to obtain improved rates for prediction with the *log loss* $\ell_{\log} : \Delta_K \times [K] \to \mathbb{R}$, defined via $\ell_{\log}(p, y) = -\log(p_y)$. Characterizing optimal rates for online prediction with the log loss is a fundamental problem (Merhav and Feder, 1998), but there have been very few successful attempts to provide rates for general classes of functions. Cesa-Bianchi and Lugosi (1999) studied the multiclass case,[9] but provide bounds only in terms of pointwise covering numbers; this can lead to vacuous bounds even for well-behaved classes such as Hilbert spaces. More recently, Rakhlin and Sridharan (2015a) provided a bound for general classes in terms of sequential covering numbers, but their bound is known to not be tight for certain classes (see the discussion in their Section 6). We improve on their rates uniformly.

Note that the problems of learning with the logistic loss and learning with the log loss can easily be mapped onto each other to provide coarse rates. One can trivially write $\ell_{\log}(p, y)$ as $\ell(\boldsymbol{\sigma}^+(p), y)$ for any distribution $p \in \Delta_K$, and likewise it holds that $\ell(z, y) = \ell_{\log}(\boldsymbol{\sigma}(z), y)$ for any $z \in \mathbb{R}^K$. To obtain rates for competing with a class $\mathcal{F} : \mathcal{X} \to \Delta_K$ under the log loss, we can use this relationship to get a bound by applying Theorem 11 with the class $\boldsymbol{\sigma}^+ \circ \mathcal{F}$. This bound improves over Rakhlin and Sridharan (2015a) in the low complexity regime, though it is worse for high complexity classes.

By combining the style of proof in Theorem 11 with key technical observations from Rakhlin and Sridharan (2015a), we provide a bound on minimax rate for log loss that both uniformly improves on the rate in Rakhlin and Sridharan (2015a) for binary outcome case and also extends in general to $K > 2$. For brevity we present results only for the binary case. In this case we can restrict to real-valued outputs: We let $\ell_{\log} : [0, 1] \times \{0, 1\} \to \mathbb{R}$ be defined by $\ell_{\log}(p, y) = -y \log(p) - (1 - y) \log(1 - p)$, and take both $\mathcal{F}$ and the learner's predictions to be $[0, 1]$-valued. The minimax regret for learning with the log loss is given by

$$\mathcal{V}_n^{\log}(\mathcal{F}) = \left\langle\!\!\left\langle \sup_{x_t \in \mathcal{X}} \inf_{\hat{p}_t \in [0,1]} \max_{y_t \in \{0,1\}} \right\rangle\!\!\right\rangle_{t=1}^{n} \left[\sum_{t=1}^{n} \ell(\hat{p}_t, y_t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t) \right]. \tag{8}$$

The following theorem provides an upper bound on the minimax regret in terms of L_∞ covering numbers $\mathcal{N}_\infty(\alpha, \mathcal{F})$ (definition deferred to Appendix A.6).

Theorem 13 *For any class $\mathcal{F} \subseteq [0, 1]^{\mathcal{X}}$ and any $\delta \in (0, 1/2]$, $\mathcal{V}_n^{\log}(\mathcal{F})$ is bounded by*

$$\tilde{O}\left(\inf_{\gamma \geq \alpha > 0} \left\{ \log \mathcal{N}_\infty(\gamma, \mathcal{F}) + \frac{\alpha n}{\delta} + \sqrt{\frac{n}{\delta}} \int_\alpha^\gamma \sqrt{\log \mathcal{N}_\infty(\rho, \mathcal{F})} d\rho + \frac{1}{\delta} \int_\alpha^\gamma \log \mathcal{N}_\infty(\rho, \mathcal{F}) d\rho \right\} + \delta n \right).$$

where $\tilde{O}$ supresses $\log(n)$ and $\log(1/\delta)$ factors.

Comparing to (Rakhlin and Sridharan, 2015a, Theorem 4), the only difference is that their bound has an extra $\frac{1}{\delta}$ factor in the leading $\log \mathcal{N}_\infty(\gamma, \mathcal{F})$ term above. Theorem 13 is strictly better for low-complexity classes, e.g. when $\log \mathcal{N}_\infty(\gamma, \mathcal{F}) \asymp \left(\frac{C}{\gamma}\right)^p$ for $p \leq 1$.

Acknowledgements

DF thanks Matus Telgarsky for sparking an interest in logistic regression through a series of talks at the Simons Institute. KS acknowledges support from the NSF under grants CDS&E-MSS 1521544 and NSF CAREER Award 1750575. MM acknowledges support under NSF grants CCF-1535987 and IIS-1618662. DF is supported in part by the NDSEG PhD fellowship.

9. In literature on log loss the class size K we use is typically referred to as the *alphabet size*.

References

Jacob D. Abernethy and Alexander Rakhlin. An Efficient Bandit Algorithm for $\sqrt{T}$ Regret in Online Multiclass Prediction? In *Conference on Learning Theory*, 2009.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

Francis Bach. Self-concordant analysis for logistic regression. *Electronic Journal of Statistics*, 4:384–414, 2010.

Francis Bach and Eric Moulines. Non-strongly-convex smooth stochastic approximation with convergence rate o (1/n). In *Advances in neural information processing systems*, pages 773–781, 2013.

Francis R Bach. Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. *Journal of Machine Learning Research*, 15(1):595–627, 2014.

Adam L. Berger, Stephen Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Comp. Linguistics*, 22(1), 1996.

Joseph Berkson. Application of the logistic function to bio-assay. *Journal of the American Statistical Association*, 39:357–365, 1944.

Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 19–26, 2011.

Alina Beygelzimer, Satyen Kale, and Haipeng Luo. Optimal and adaptive algorithms for online boosting. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2323–2331, 2015.

Alina Beygelzimer, Francesco Orabona, and Chicheng Zhang. Efficient Online Bandit Multiclass Learning with $\tilde{O}(\sqrt{T})$ Regret. In *Inernational Conference on Machine Learning*, pages 488–497, 2017.

Sébastien Bubeck, Ronen Eldan, and Joseph Lehec. Sampling from a log-concave distribution with projected langevin monte carlo. *Advances in Neural Information Processing Systems*, 2015.

Nicolò Cesa-Bianchi and Gábor Lugosi. Minimax regret under log loss for general classes of experts. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, COLT '99, pages 12–18, New York, NY, USA, 1999. ACM. ISBN 1-58113-167-4.

Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

Elad Hazan and Satyen Kale. Newtron: an efficient bandit algorithm for online multiclass prediction. In *Advances in Neural Information Processing Systems*, pages 891–899, 2011.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007.

Elad Hazan, Tomer Koren, and Kfir Y Levy. Logistic regression: Tight bounds for stochastic and online optimization. In *Proceedings of The 27th Conference on Learning Theory*, pages 197–209, 2014.

David P. Helmbold and Manfred K. Warmuth. On weak learning. *J. Comput. Syst. Sci.*, 50(3):551–573, 1995.

Young Hun Jung and Ambuj Tewari. Online boosting algorithms for multi-label ranking. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, 2018.

Young Hun Jung, Jack Goetz, and Ambuj Tewari. Online multiclass boosting. In *Advances in Neural Information Processing Systems*, pages 920–929, 2017.

Sham M Kakade and Andrew Y Ng. Online bounds for bayesian algorithms. In *Advances in neural information processing systems*, pages 641–648, 2005.

Sham M. Kakade, Shai Shalev-Shwartz, and Ambuj Tewari. Efficient bandit algorithms for online multiclass prediction. In *Proceedings of the 25th international conference on Machine learning*, pages 440–447. ACM, 2008.

Sham M Kakade, Karthik Sridharan, and Ambuj Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in neural information processing systems*, pages 793–800, 2009.

László Lovász and Santosh Vempala. Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization. In *47th Annual IEEE Symposium on Foundations of Computer Science*, pages 57–68. IEEE, 2006.

László Lovász and Santosh Vempala. The geometry of logconcave functions and sampling algorithms. *Random Structures & Algorithms*, 30(3):307–358, 2007.

H Brendan McMahan and Matthew Streeter. Open problem: Better bounds for online logistic regression. In *Conference on Learning Theory*, pages 44–1, 2012.

Nishant A Mehta. Fast rates with high probability in exp-concave statistical learning. *International Conference on Artificial Intelligence and Statistics*, 2017.

Neri Merhav and Meir Feder. Universal prediction. *IEEE Transactions on Information Theory*, 44:2124–2147, 1998.

Hariharan Narayanan and Alexander Rakhlin. Efficient sampling from time-varying log-concave distributions. *Journal of Machine Learning Research*, 18:112:1–112:29, 2017.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Alexander Rakhlin and Karthik Sridharan. Online nonparametric regression. In *Conference on Learning Theory*, 2014.

Alexander Rakhlin and Karthik Sridharan. Sequential probability assignment with binary alphabets and large classes of experts. *CoRR*, abs/1501.07340, 2015a.

Alexander Rakhlin and Karthik Sridharan. Online nonparametric regression with general loss functions. *CoRR*, abs/1501.06598, 2015b. URL http://arxiv.org/abs/1501.06598.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Random averages, combinatorial parameters, and learnability. *Advances in Neural Information Processing Systems 23*, pages 1984–1992, 2010.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning via sequential complexities. *Journal of Machine Learning Research*, 2015a.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Sequential complexities and uniform martingale laws of large numbers. *Probability Theory and Related Fields*, 161(1-2):111–153, 2015b.

Shai Shalev-Shwartz and Yoram Singer. Convex repeated games and fenchel duality. In *Advances in neural information processing systems*, pages 1265–1272, 2007.

Tim Van Erven, Peter D Grünwald, Nishant A Mehta, Mark D Reid, and Robert C Williamson. Fast rates in statistical and online learning. *Journal of Machine Learning Research*, 16:1793–1861, 2015.

Vladimir Vovk. A game of prediction with expert advice. In *Proceedings of the eighth annual conference on Computational learning theory*, pages 51–60. ACM, 1995.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, pages 928–936, 2003.

Appendix A. Proofs

A.1. Proofs from Section 2

Lemma 14 *The generalized multiclass logisitic loss is $2L$-Lipschitz with respect to ℓ_∞ norm.*

Proof It is straightforward to verify the identity

$$\nabla_z \ell(z, y) = \left(\sum_k y_k\right)\boldsymbol{\sigma}(z) - y.$$

It follows that $\|\nabla_z \ell(z,y)\|_1 \le \|y\|_1 \|\boldsymbol{\sigma}(z)\|_1 + \|y\|_1 \le 2L$. By duality, this implies $2L$-Lipschitzness with respect to ℓ_∞. ■

Lemma 15 *The function $f(x) = \prod_{k\in[d]} x_k^{\alpha_k}$ is concave over $\mathbb{R}^d_+$ whenever $\alpha_k \ge 0 \ \forall k$ and $\sum_{k\in[d]} \alpha_k \le 1$.*

Proof We will prove that the Hessian of f is negative semidefinite. The Hessian can be written as

$$\nabla^2 f(x) = f(x) \cdot G(x),$$

where the matrix $G(x) \in \mathbb{R}^{d\times d}$ is given by $G(x)_{ii} = \alpha_i(\alpha_i - 1)x_i^{-2}$ and $G(x)_{ij} = \alpha_i\alpha_j x_i^{-1}x_j^{-1}$. Since f is nonnegative, it suffices to show that G is negative semidefinite. Using the reparameterization $y_i = x_i^{-1}$ and the notation $\odot$ for the element-wise product, we can write

$$G(y) = (\alpha \odot y)^{\otimes 2} - \mathrm{diag}(\alpha \odot y^2).$$

For any fixed $y \in \mathbb{R}^d_+$ and any $v \in \mathbb{R}^d$, we have

$$\begin{aligned}
\langle v, G(y)v\rangle &= \left(\sum_{k=1}^d \alpha_k y_k v_k\right)^2 - \sum_{k=1}^d \alpha_k y_k^2 v_k^2 \\
&\le \left(\sum_{k=1}^d \alpha_k y_k^2 v_k^2\right)\left(\sum_{k=1}^d \alpha_k\right) - \sum_{k=1}^d \alpha_k y_k^2 v_k^2 \\
&\le 0.
\end{aligned}$$

The first inequality above uses Cauchy-Schwarz and the second uses that $\sum \alpha_k \le 1$. ■

Proof [Proof of Proposition 2] We first show that the generalized multiclass log loss $\ell_{\log}(p, y) := -\sum_{k\in[K]} y_k \log(p_k)$ is $1/L$-mixable over predictions $p \in \Delta_K$ and outcomes $y \in \mathcal{Y}$. Recall that to show η-mixability it is sufficient to demonstrate that ℓ is η-exp-concave with respect to p (e.g. (Cesa-Bianchi and Lugosi, 2006)) for any $y \in \mathcal{Y}$.

Observe that we have

$$e^{-\eta\ell(p,y)} = \prod_{k\in[K]} p_k^{\eta y_k}.$$

When $\eta \le 1/L$, we have $\sum_{k\in[K]} \eta y_k \le 1$. Since $p \in \Delta_K$ and by the definition of $\mathcal{Y}$, Lemma 15 implies the function $p \mapsto \prod_{k\in[K]} p_k^{\eta y_k}$ is concave, which proves the result.

Exp-concavity implies that for any distribution $\tilde{\pi}$ over Δ_K, the predicition $p_{\tilde{\pi}} = \mathbb{E}_{p\sim\tilde{\pi}}[p]$ certifies the inequality

$$\mathbb{E}_{p\sim\tilde{\pi}}[\exp(-\eta\ell_{\log}(p, y))] \le \exp(-\eta\ell_{\log}(p_{\tilde{\pi}}, y)) \quad y \in \mathcal{Y}.$$

Now, turning to the multiclass logistic loss $\ell : \mathbb{R}^K \times \mathcal{Y} \to \mathbb{R}$ defined as $\ell(z, y) = -\sum_{k\in[K]} y_k \log(\boldsymbol{\sigma}(z)_k)$, let π be any distribution on $\mathbb{R}^K$. Let $\tilde{\pi}$ be the induced distribution on Δ_K via the softmax function, i.e. a sample from $\tilde{\pi}$ is generated by sampling $z \sim \pi$ and computing $p = \boldsymbol{\sigma}(z)$. Then define $z_\pi = \boldsymbol{\sigma}^+(\mathbb{E}_{z\sim\pi}[\boldsymbol{\sigma}(z)])$. Since $\boldsymbol{\sigma}(z_\pi) = \mathbb{E}_{z\sim\pi}[\boldsymbol{\sigma}(z)] = p_{\tilde{\pi}}$ and $\ell(z, y) = \ell_{\log}(\boldsymbol{\sigma}(z), y)$, the above inequality implies that

$$\mathbb{E}_{z\sim\pi}[\exp(-\eta\ell(z, y))] \le \exp(-\eta\ell(z_\pi, y)) \quad y \in \mathcal{Y}.$$

■

Lemma 16 *Suppose a strategy $(\tilde{z}_t))_{t\le n}$ guarantees a regret inequality*

$$\sum_{t=1}^n \ell(\tilde{z}_t, y_t) - \inf_{f\in\mathcal{F}} \sum_{t=1}^n \ell(f(x_t), y_t) \le \mathbf{R}.$$

Then for $0 \le \mu \le 1/2$ the strategy $\hat{z}_t := \boldsymbol{\sigma}^+(\mathrm{smooth}_\mu(\boldsymbol{\sigma}(\hat{z}_t)))$ guarantees

$$\sum_{t=1}^n \ell(\tilde{z}_t, y_t) - \inf_{f\in\mathcal{F}} \sum_{t=1}^n \ell(f(x_t), y_t) \le \mathbf{R} + 2\mu \sum_{t=1}^n \|y_t\|_1.$$

and satisfies $\|\hat{z}_t\|_\infty \le \log(K/\mu)$.

Proof [Proof of Lemma 16]

We write regret as

$$\begin{aligned}
&\sum_{t=1}^n \ell(\hat{z}_t, y_t) - \inf_{f\in\mathcal{F}} \sum_{t=1}^n \ell(f(x_t), y_t) \\
&= \sum_{t=1}^n \ell(\tilde{z}_t, y_t) - \inf_{f\in\mathcal{F}} \sum_{t=1}^n \ell(f(x_t), y_t) + \sum_{t=1}^n \ell(\hat{z}_t, y_t) - \sum_{t=1}^n \ell(\tilde{z}_t, y_t) \\
&\le \mathbf{R} + \sum_{t=1}^n \ell(\hat{z}_t, y_t) - \sum_{t=1}^n \ell(\tilde{z}_t, y_t).
\end{aligned}$$

For the last two terms, fix any round t and define $\tilde{p} = \boldsymbol{\sigma}(\tilde{z}_t)$. Since $\boldsymbol{\sigma}(\hat{z}_t) = (1-\mu)\tilde{p} + \mu\mathbf{1}/K$, we have

$$\ell(\hat{z}_t, y_t) - \ell(\tilde{z}_t, y_t) = \sum_{k\in[K]} y_{t,k} \log\left(\frac{\tilde{p}_k}{(1-\mu)\tilde{p}_k + \mu/K}\right) \le \log\left(\frac{1}{1-\mu}\right) \sum_{k\in[K]} y_{t,k} \le 2\mu\|y_t\|_1.$$

The last inequality uses that $\log(1/(1-x)) \le 2x$ for $x \le 1/2$. Summing up over all rounds t gives us the desired regret bound.

To establish boundedness of the predictions, recall that $\boldsymbol{\sigma}_k^+(p) = \log(p_k)$. Letting $p = (1-\mu)\,\mathbb{E}_{W\sim P_t}[\boldsymbol{\sigma}(Wx_t)] + \mu\mathbf{1}/K$, it clearly holds that $p_k \ge \mu/K$, and so $|\boldsymbol{\sigma}_k^+(p)| \le \log(K/\mu)$. ■

Proof [Proof of Theorem 3] Let $\eta = 1/L$. Let $\tilde{z}_t = \boldsymbol{\sigma}^+(\mathbb{E}_{W\sim P_t}[\boldsymbol{\sigma}(Wx_t)])$ — that is, the prediction for the setting $\mu = 0$. We will first establish a regret bound for the case $\mu = 0$, then reduce the general case to it by approximation.

First observe that due to mixability for $\eta \le 1/L$ (from Proposition 2), we have

$$\sum_{t=1}^n \ell(\tilde{z}_t, y_t) \le -\frac{1}{\eta}\sum_{t=1}^n \log\left(\int_{\mathcal{W}} \exp(-\eta\ell(Wx_t, y_t))dP_t(W)\right).$$

Let $Z_t = \int_{\mathcal{W}} \exp(-\eta\sum_{s=1}^t \ell(Wx_s, y_s))dW$ with the convention $Z_0 = \int_{\mathcal{W}} dW$. Using the definition of P_t, the right-hand-side in the displayed equation above is then equal to

$$-\frac{1}{\eta}\sum_{t=1}^n \log(Z_t/Z_{t-1}) = -\frac{1}{\eta}\log(Z_n/Z_0) = -\frac{1}{\eta}\log\left(\int_{\mathcal{W}} \exp\left(-\eta\sum_{t=1}^n \ell(Wx_t, y_t)\right)dW\right) + \frac{1}{\eta}\log(\mathrm{Vol}(\mathcal{W}))$$

We will focus on coming up with an upper bound on the term $-\log\left(\int_{\mathcal{W}} \exp(-\eta\sum_{t=1}^n \ell(Wx_t, y_t))dW\right)$. Let $W^\star = \arg\min_{W\in\mathcal{W}} \sum_{t=1}^n \ell(Wx_t, y_t)$. Fix $\theta \in [0,1)$ and let $S = \{\theta W^\star + (1-\theta)W \mid W \in \mathcal{W}\} \subseteq \mathcal{W}$. To upper bound the negative-log-integral term, we will lower bound the integral appearing inside.

$$\int_{\mathcal{W}} \exp\left(-\eta\sum_{t=1}^n \ell(Wx_t, y_t)\right)dW \ge \int_S \exp\left(-\eta\sum_{t=1}^n \ell(Wx_t, y_t)\right)dW.$$

Using a change of variables and noting that since $W \in \mathbb{R}^{K\times d}$ the Jacobian of the mapping $W \mapsto (1-\theta)W + \theta W^\star$ has determinant $(1-\theta)^{D_{\mathcal{W}}}$, the right-hand-side above equals

$$= (1-\theta)^{D_{\mathcal{W}}} \int_{\mathcal{W}} \exp\left(-\eta\sum_{t=1}^n \ell((\theta W^\star + (1-\theta)W)x_t, y_t)\right)dW.$$

Observe that $\|(\theta W^\star + (1-\theta)W)x_t - W^\star x_t\|_\infty = (1-\theta)\max_{k\in[K]}\left|\left\langle W_k^\star - W_k, x_t\right\rangle\right| \le 2(1-\theta)B\|x_t\|_\star$. Using this observation with the $2L$-Lipschitzness of ℓ with respect to ℓ_∞ from Lemma 14 implies that the above displayed expression is at most

$$(1-\theta)^{D_{\mathcal{W}}} \int_{\mathcal{W}} \exp\left(-\eta\sum_{t=1}^n \ell(W^\star x_t, y_t) - 4(1-\theta)BL\eta\sum_{t=1}^n \|x_t\|_\star\right)dW.$$

$$= (1-\theta)^{D_{\mathcal{W}}} \cdot \mathrm{Vol}(\mathcal{W}) \cdot \exp\left(-\eta\sum_{t=1}^n \ell(W^\star x_t, y_t)\right) \cdot \exp\left(-4(1-\theta)BL\eta\sum_{t=1}^n \|x_t\|_\star\right).$$

Combining all of the observations so far, we have proven the following regret bound:

$$\begin{aligned}
&\sum_{t=1}^{n} \ell(\hat{y}_t, y_t) - \sum_{t=1}^{n} \ell(W^\star x_t, y_t) \\
&\leq \frac{1}{\eta} \log(\mathrm{Vol}(\mathcal{W})) - \sum_{t=1}^{n} \ell(W^\star x_t, y_t) \\
&\quad + \frac{1}{\eta} \underbrace{\left(D_{\mathcal{W}} \log\left(\frac{1}{1-\theta}\right) - \log(\mathrm{Vol}(\mathcal{W})) + \eta \sum_{t=1}^{n} \ell(W^\star x_t, y_t) + 4(1-\theta)BL\eta \sum_{t=1}^{n} \|x_t\|_\star \right)}_{\text{Bound on negative log-integral-exp.}} \\
&= \frac{D_{\mathcal{W}}}{\eta} \log\left(\frac{1}{1-\theta}\right) + 4(1-\theta)BL \sum_{t=1}^{n} \|x_t\|_\star.
\end{aligned}$$

To conclude, we choose θ to satisfy $1-\theta = \min\{D_{\mathcal{W}}/(B\sum_{t=1}^{n}\|x_t\|_\star), 1\}$. Note that regardless of which argument obtains the minimum, we have $4(1-\theta)BL\sum_{t=1}^{n}\|x_t\|_\star \leq 4D_{\mathcal{W}}L$. The choice of θ also means that $\log\left(\frac{1}{1-\theta}\right) = \log(1 \vee B\sum_{t=1}^{n}\|x_t\|_\star / D_{\mathcal{W}})$. This leads to a final bound of

$$D_{\mathcal{W}}L \cdot \log\left(1 \vee \frac{B\sum_{t=1}^{n}\|x_t\|_\star}{D_{\mathcal{W}}}\right) + 4D_{\mathcal{W}}L.$$

To simplify we upper bound this by

$$5D_{\mathcal{W}}L \cdot \log\left(\frac{B\sum_{t=1}^{n}\|x_t\|_\star}{D_{\mathcal{W}}} + e\right) = 5D_{\mathcal{W}}L \cdot \log\left(\frac{BRn}{D_{\mathcal{W}}} + e\right).$$

To handle the general case where $\mu > 0$ we simply appeal to Lemma 16 and use that $\boldsymbol{\sigma}(\boldsymbol{\sigma}^+(p)) = p \ \forall p \in \Delta_K$.

■

We now state the proof of Theorem 5. This proof is a simple corollary of Theorem 18, a lower bound on mistakes for online binary classification with a margin. Theorem 18 is proven in the remainder of this section of the appendix. To begin, we need the following definition:

Definition 17 *Let $\mathcal{F} : \mathcal{X} \to [-1,1]$ be some function class. A dataset $(x_1, y_1), \ldots, (x_n, y_n) \in \cup_{t=1}^{n} \mathcal{X} \times \{\pm 1\}$ is shattered with γ margin if there exists $f \in \mathcal{F}$ such that*

$$f(x_t)y_t \geq \gamma.$$

Proof [Proof of Theorem 5] Let $\hat{z}_t$ for $t \in [n]$ be the sequence of predictions made by the algorithm for a sequence of examples (x_t, y_t), for $t \in [n]$. It is easy to check that

$$\sum_{t=1}^{n} \ell_{\mathrm{bin}}(\hat{z}_t, y_t) \geq \log(2) \sum_{t=1}^{n} \mathbb{1}\{\mathrm{sgn}(\hat{z}_t) \neq y_t\}.$$

Let $1/\gamma = B/\log(n)$. From Theorem 18, it holds that whenever $\gamma \leq O(1/\sqrt{d})$, there exists an adversarial sequence (x_t, y_t), for $t \in [n]$, for which

$$\sum_{t=1}^{n} \mathbb{1}\{\mathrm{sgn}(\hat{y}_t) \neq y_t\} \geq \frac{d}{4}\left\lfloor \log_2\left(\frac{1}{5\gamma d^{1/2}}\right)\right\rfloor,$$

and for which the dataset is γ-shattered by some $w \in \mathbb{R}^d$ with $\|w\|_2 \le 1$. Since the dataset is γ-shattered we also have

$$\inf_{w:\|w\|_2 \le B} \sum_{t=1}^{n} \ell_{\text{bin}}(\langle w, x_t \rangle, y_t) \le \sum_{t=1}^{n} \log(1 + e^{-\gamma B}) = \sum_{t=1}^{n} \log\left(1 + \frac{1}{n}\right) \le 1.$$

This yields the desired lower bound on the regret. ■

Theorem 18 *Fix a margin $\gamma \in (0, \frac{1}{4\sqrt{5d}}]$. Then for any randomized strategy $(\hat{y}_t)_{t \le n}$ there exists an adversary $(x_t)_{t \le n}$, $(y_t)_{t \le n}$ with $\|x_t\|_2 \le 2$ for which*

$$\mathbb{E}\left[\sum_{t=1}^{n} \mathbb{1}\{\text{sgn}(\hat{y}_t) \neq y_t\}\right] \ge \frac{d}{4}\left\lfloor \log_2\left(\frac{1}{5\gamma d^{1/2}}\right)\right\rfloor, \tag{9}$$

and the data sequence is realizable by a unit vector $w \in \mathbb{R}^{d+1}$ with margin γ.

Remark 19 *This lower bound only applies in the regime where $\frac{1}{\gamma^2} \ge d$, meaning that it does not contradict the dimension-independent Perceptron bound.*

To prove Theorem 18, we first state a standard lower bound based on Littlestone's dimension.

Definition 20 *An $\mathcal{X}$-valued tree is a sequence of mappings $\boldsymbol{x}_t : \{\pm 1\}^{t-1} \to \mathcal{X}$ for $1 \le t \le n$.*

We use the abbreviation of $\boldsymbol{x}_t(\epsilon) = \boldsymbol{x}_t(\epsilon_1, \ldots, \epsilon_{t-1})$ for such a tree, where $\epsilon \in \{\pm 1\}^n$.

Lemma 21 *Let $\mathcal{F} : \mathcal{X} \to [-1, 1]$ be some function class. Suppose there exists a $\mathcal{X}$-valued tree $\boldsymbol{x}$ of depth D_γ such that*

$$\forall \epsilon \in \{\pm 1\}^{D_\gamma} \quad \exists f \in \mathcal{F} \quad \text{s.t.} \quad f(\boldsymbol{x}_t(\epsilon))\epsilon_t \ge \gamma. \tag{10}$$

Then

$$\inf_{q_1, \ldots, q_n} \sup_{\substack{(x_1, y_1), \ldots, (x_n, y_n) \\ \text{separable with } \gamma \text{ margin}}} \mathbb{E}_{\hat{y}_1 \sim q_t, \ldots, \hat{y}_n \sim q_n}\left[\sum_{t=1}^{n} \mathbb{1}\{\hat{y}_t \neq y_t\}\right] \ge \frac{1}{2}\min\{D_\gamma, n\},$$

where the infimum and supremum above are understood to range over policies.

Proof [Proof of Lemma 21] Suppose that $n \le D_\gamma$. We will sample Rademacher random variables $\epsilon \in \{\pm 1\}^n$ and play $y_t = \epsilon_t$ and $x_t = \boldsymbol{x}_t(\epsilon_{1:t-1})$. This immediately implies that the expected number of mistakes is equal to $\frac{n}{2}$. Moreover, since $n \le D_\gamma$, the assumption in the statement of the lemma implies that there exists $f \in \mathcal{F}$ such that $f(\boldsymbol{x}_t(\epsilon))y_t \ge \gamma$, so the data is indeed separable with γ margin.

If $n > D_\gamma$ we can follow the strategy above, then continue to play $(x_{D_\gamma}, y_{D_\gamma})$ for all $t > D_\gamma$. ■

Proof [Proof of Theorem 18] By Lemma 21 it suffices to exhibit a tree $\boldsymbol{x}$ for which (10) is satisfied with $D_\gamma = \Omega(d \log(1/(\sqrt{d}\gamma)))$.

We first restate a well-known tree instance for the one-dimensional case. Consider a class of thresholds $\mathcal{F}_{\text{thresh}} = \{f_\theta : [0, 1] \to \{\pm 1\}\}$ defined by $f_\theta(z) = 1 - 2\mathbb{1}\{x < \theta\}$. The claim is as follows: For any $\delta \in (0, 1]$, there exists a $[0, 1]$-valued tree $\boldsymbol{z}$ of depth $D_\delta := \lfloor \log_2(2/\delta) \rfloor$ such that

1. $\forall \epsilon \in \{\pm 1\}^{D_\delta} \;\; \exists \theta \;\; \text{s.t.} \;\; f_\theta(\boldsymbol{z}_t(\epsilon))\epsilon_t = 1.$
2. $|\boldsymbol{z}_t(\epsilon) - \boldsymbol{z}_s(\epsilon)| \geq \delta \;\; \forall s \neq t.$

The construction is as follows. Let $u_1 = 1$, $l_1 = 0$. Recursively for $t = 1, \ldots, n$:

- $\boldsymbol{z}_t(\epsilon_{1:t-1}) = \frac{l_t + u_t}{2}$.
- If $\epsilon_t = -1$ set $l_{t+1} = \boldsymbol{z}_t(\epsilon_{1:t-1})$ and $u_{t+1} = u_t$, else set $u_{t+1} = \boldsymbol{z}_t(\epsilon_{1:t-1})$ and $l_{t+1} = l_t$.

Under this construction the sequence $\boldsymbol{z}_1(\cdot), \ldots, \boldsymbol{z}_{D_\delta}(\epsilon_{1:D_\delta - 1})$ can always be shattered. Furthermore $\boldsymbol{z}^\star(\epsilon) := \boldsymbol{z}_{D_\delta + 1}(\epsilon_{1:D_\delta})$ satisfies the additional property that $\boldsymbol{z}_t > \boldsymbol{z}^\star(\epsilon) \implies \epsilon_t = 1$ and $\boldsymbol{z}_t < \boldsymbol{z}^\star(\epsilon) \implies \epsilon_t = -1$. Also, $|\boldsymbol{z}^\star - \boldsymbol{z}_t| \geq \frac{\delta}{2} \; \forall t \leq D_\delta$.

We now show how to extend this instance to $d + 1$ dimensions for any $d \geq 1$. The approach is to concatenate d instances of the $\boldsymbol{z}$ tree constructed above, one for each of the first d coordinates. The final coordinate is left as a constant so that a bias can be implemented.

Let $n = d \cdot D_\delta$ be the tree depth for our $d+1$-dimensional instance. For any time t, let $k \in [d]$ and $\tau \in [D_\delta]$ be such that $t = (k-1)D_\delta + \tau$. Let any sequence $\epsilon \in \{\pm 1\}^n$ be partitioned as $(\boldsymbol{\epsilon}^1, \ldots, \boldsymbol{\epsilon}^d)$ with each $\boldsymbol{\epsilon}^k \in \{\pm 1\}^{D_\delta}$. Letting e_k denote the kth standard basis vector, we define a shattered tree $\boldsymbol{x}$ as follows:

$$\boldsymbol{x}_t(\epsilon_{1:t-1}) = e_{d+1} + e_k \boldsymbol{z}_\tau(\boldsymbol{\epsilon}^k_{1:\tau-1}).$$

We construct a vector $w \in \mathbb{R}^{d+1}$ whose sign correctly classifies each $\boldsymbol{x}_t$ as follows:

- $w_{d+1} = -\delta$.
- $w_k = \delta / \boldsymbol{z}^\star(\boldsymbol{\epsilon}^k)$.

For any $t = (k-1)D_\delta + \tau$ this choice gives

$$\langle w, \boldsymbol{x}_t(\epsilon) \rangle \epsilon_t = \delta(\boldsymbol{z}_\tau(\boldsymbol{\epsilon}^k_{1:\tau-1}) / \boldsymbol{z}^\star(\boldsymbol{\epsilon}^k) - 1)\epsilon_t.$$

As described above, $\boldsymbol{z}_t > \boldsymbol{z}^\star(\epsilon) \implies \epsilon_t = 1$ and $\boldsymbol{z}_t < \boldsymbol{z}^\star(\epsilon) \implies \epsilon_t = -1$, which immediately implies that the inner product is always non-negative, and so the dataset is shattered. Using that $|\boldsymbol{z}^\star(\epsilon) - \boldsymbol{z}_t(\epsilon)| \geq \frac{\delta}{2}$ and that both numbers lie in $[0, 1]$, we can lower bound the magnitude with which the shattering takes place:

$$\left|\boldsymbol{z}_\tau(\epsilon^k_{1:\tau-1}) / \boldsymbol{z}^\star(\boldsymbol{\epsilon}^k) - 1\right| = \frac{1}{\boldsymbol{z}^\star(\boldsymbol{\epsilon}^k)} \left|\boldsymbol{z}_\tau(\epsilon^k_{1:\tau-1}) - \boldsymbol{z}^\star(\boldsymbol{\epsilon}^k)\right| \geq \frac{1}{\boldsymbol{z}^\star(\boldsymbol{\epsilon}^k)} \frac{\delta}{2} \geq \frac{\delta}{4},$$

and so the shattering takes place with margin at least $\delta^2/4$.

Lastly, the norm of w is given by

$$\|w\|_2 = \sqrt{\delta^2 + \sum_{k=1}^{d} \left(\frac{\delta}{\boldsymbol{z}^\star(\boldsymbol{\epsilon}^k)}\right)^2} \leq \sqrt{\delta^2 + 4d} \leq \sqrt{5d},$$

where the first inequality uses that $\boldsymbol{z}^\star(\epsilon) \geq \delta/2$ and the second uses that $d \geq 1$

Rescaling, we have that the vector $w / \|w\|_2$ shatters the tree with margin at least $\frac{\delta^2}{4\sqrt{5d}}$. To rephrase the result as a function of a desired margin: For any margin $\gamma \in (0, \frac{1}{4\sqrt{5d}}]$, setting $\delta = \sqrt{\gamma 4\sqrt{5d}} \leq 1$, we have constructed a tree of depth $\left\lfloor \log_2(2/\sqrt{\gamma 4\sqrt{5d}}) \right\rfloor$ that can be shattered with margin γ.

■

A.2. OBAMA Algorithm and Proof of Theorem 6

Algorithm 2

procedure OBAMA(decision set $\mathcal{W}$, smoothing parameter μ.)
  Let $\mathcal{A}$ be Algorithm 1 initialized with $\mathcal{W}$ and μ.
  for $t = 1, \dots, n$ **do**
    Obtain x_t, pass it to $\mathcal{A}$ and let $\hat{z}_t \in \mathbb{R}_K$ be the output of $\mathcal{A}$.
    Play $\hat{y}_t \sim p_t := \boldsymbol{\sigma}(\hat{z}_t)$ and obtain $\mathbb{1}[\hat{y}_t \neq y_t]$.
    Define $\tilde{y}_t \in \mathbb{R}^K$ as $\tilde{y}_t(k) := \frac{\mathbb{1}[k=\hat{y}_t]\mathbb{1}[\hat{y}_t=y_t]}{p_t(\hat{y}_t)}$ for $k \in [K]$ and pass it as feedback to $\mathcal{A}$.
  end for
end procedure

Proof [Proof of Theorem 6] First, note that an easy calculation on the softmax function $\boldsymbol{\sigma}$ implies that for all $k \in [K]$, $p_t(k) \geq \frac{(1-\mu)\exp(-2BR)+\mu}{K}$. So, defining $L = \frac{K}{(1-\mu)\exp(-2BR)+\mu}$, we have $\|\tilde{y}_t\|_1 \leq L$. Thus, Theorem 3 applied to $\mathcal{A}$ guarantees that for any $W \in \mathcal{W}$,

$$\sum_{t=1}^{n} \ell(\hat{z}_t, \tilde{y}_t) - \sum_{t=1}^{n} \ell(Wx_t, \tilde{y}_t) \leq 5LdK \cdot \log\Big(\tfrac{BRn}{dK} + e\Big) + 2\mu \sum_{t=1}^{n} \|\tilde{y}_t\|_1.$$

Fix a round t and let $\mathbb{E}_t[\cdot]$ denote expecation conditioned on $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{t-1}$. The construction of the feedback vectors $\tilde{y}_t$ via importance weighting guarantees $\mathbb{E}_t[\tilde{y}_t] = \mathbf{1}_{y_t}$, where $\mathbf{1}_k$ denotes the indicator vector supported on coordinate k. Hence, $\mathbb{E}_t[\ell(\hat{z}_t, \tilde{y}_t)] = \ell(\hat{z}_t, y_t) = -\log(p_t(y_t))$ and $\mathbb{E}_t[\ell(Wx_t, \tilde{y}_t)] = \ell(Wx_t, y_t)$. Furthermore, it is easy to check that $\mathbb{E}_t[\|\tilde{y}_t\|_1] = 1$. Thus, we conclude that

$$\sum_{t=1}^{n} \mathbb{E}[-\log(p_t(y_t))] - \sum_{t=1}^{n} \ell(Wx_t, y_t) \leq 5LdK \cdot \log\Big(\tfrac{BRn}{dK} + e\Big) + 2\mu n.$$

Now if we set $\mu = 0$, then the right-hand side is bounded by $O(dK^2 \exp(2BR) \log\big(\frac{BRn}{dK} + e\big))$. If we set $\mu = \sqrt{\frac{dK^2 \log\left(\frac{BRn}{dK}+e\right)}{n}}$, the right-hand side is bounded by $O\left(\sqrt{dK^2 \log(\frac{BRn}{dK} + e)n}\right)$. Choosing the setting of μ that gives the smaller upper bound, and the fact that the log loss upper bounds the probability of making a mistake (because $-\log(p_t(y_t)) \geq 1 - p_t(y_t)$), we get the stated bound on the expected number of mistakes. ■

A.3. Pseudocode and Proofs from Section 4

Proof [Proof of Theorem 8] Denote the number of mistakes of the i-th expert (which is the combination of the first i weak learners) by

$$M_i = \sum_{t=1}^{n} \mathbb{1}\{\hat{y}_t^i \neq y_t\} = \sum_{t=1}^{n} \mathbb{1}\left\{\arg\max_k s_t^i(k) \neq y_t\right\},$$

with the convention that $M_0 = n$. The weights v_t^i simply implement the multiplicative weights strategy, and so Lemma 23, which gives a concentration bound based on Freedman's inequality

Algorithm 3 AdaBoost.OLM++

procedure ADABOOST.OLM++(weak learners $\mathrm{WL}^1, \dots, \mathrm{WL}^N$)
  For all $i \in [N]$, set $v_1^i \leftarrow 1$, initialize weak learner WL_1^i, and initialize logistic learner $\mathrm{Logistic}_1^i$ with $\mathcal{W} = \left\{(\alpha I_{K\times K}, I_{K\times K}) \in \mathbb{R}^{K\times 2K} \mid \alpha \in [-2,2]\right\}$ and $\mu = 1/n$.
  for $t = 1, \dots, n$ **do**
    Receive instance x_t.
    $s_t^0 \leftarrow 0 \in \mathbb{R}^K$.
    for $i = 1, \dots, N$ **do**
      Compute cost matrix C_t^i from s_t^{i-1} using (2).
      $l_t^i \leftarrow \mathrm{WL}_t^i.\mathrm{Predict}(x_t, C_t^i)$.
      $\widetilde{x}_t^i \leftarrow (e_{l_t^i}, s_t^{i-1}) \in \mathbb{R}^{2K}$.
      $s_t^i \leftarrow \mathrm{Logistic}_t^i.\mathrm{Predict}(\widetilde{x}_t^i)$.
      $\hat{y}_t^i \leftarrow \arg\max_k s_t^i(k)$.
    end for
    Sample i_t with $\Pr(i_t = i) \propto v_t^i$.
    Predict $\hat{y}_t = \hat{y}_t^{i_t}$ and receive true class $y_t \in [K]$.
    for $i = 1, \dots, N$ **do**
      $\mathrm{WL}_{t+1}^i \leftarrow \mathrm{WL}_t^i.\mathrm{Update}(x_t, C_t^i, y_t)$.
      $\mathrm{Logistic}_{t+1}^i \leftarrow \mathrm{Logistic}_t^i.\mathrm{Update}(\widetilde{x}_t^i, \mathbf{1}_{y_t})$.
      $v_{t+1}^i \leftarrow v_t^i \cdot \exp\left(-\mathbb{1}\left\{\hat{y}_t^i \neq y_t\right\}\right)$.
    end for
  end for
end procedure

implies that with probability at least $1 - \delta$,[10]

$$\sum_{t=1}^{n} \mathbb{1}\{\hat{y}_t \neq y_t\} \leq 4 \min_i M_i + 2\log(N/\delta). \tag{11}$$

Note that if $k^\star := \arg\max_k s_t^{i-1}(k) \neq y_t$, then $\boldsymbol{\sigma}(s_t^{i-1})_{k^\star} \geq \boldsymbol{\sigma}(s_t^{i-1})_{y_t}$ and $\boldsymbol{\sigma}(s_t^{i-1}) \in \Delta_K$ imply $\boldsymbol{\sigma}(s_t^{i-1})_{y_t} \leq 1/2$, which then implies $\sum_{k\neq y_t} \boldsymbol{\sigma}(s_t^{i-1})_k \geq 1/2$ and finally

$$-\sum_{t=1}^{n} \widehat{C}_t^i(y_t, y_t) = \sum_{t=1}^{n} \sum_{k\neq y_t} \boldsymbol{\sigma}(s_t^{i-1})_k \geq \frac{M_{i-1}}{2}. \tag{12}$$

This also holds for $i = 1$ because $s_t^0 = 0$ and $-C_t^1(y_t, y_t) = (K-1)/K \geq 1/2$.

We now examine the regret guarantee provided by each logistic regression instance. For each $i \in [N]$ we have

$$\sum_{t=1}^{n} \ell(s_t^i, y_t) - \inf_{W\in\mathcal{W}} \sum_{t=1}^{n} \ell(W\widetilde{x}_t^i, y_t) \leq O(\log(n\log(nK)))$$

10. Note that previous online boosting works (Beygelzimer et al., 2015; Jung et al., 2017) use a simpler Hoeffding bound at this stage, which picks up an extra $\sqrt{n}$ term. For their results this is not a dominant term, but in our case it can spoil the improvement given by improper logistic regression, and so we use Freedman's inequality to remove it.

This follows from Theorem 3 using $L = 1$, $D_{\mathcal{W}} = 1$, $B = 3$ for ℓ_1 norm, $\|y_t\|_1 = 1$, $\mu = 1/n$, and $\left\|\widetilde{x}_t^i\right\|_\infty \le \log(nK)$, where the last fact is implied by the second statement of Theorem 3: $\left\|s_t^i\right\|_\infty \le \log(K/\mu) = \log(nK)$ and thus $\left\|\widetilde{x}_t^i\right\|_\infty = \left\|(e_{l_t^i}, s_t^{i-1})\right\|_\infty \le \log(nK)$. Now define the difference between the total loss of the i-th and $(i-1)$-th expert to be

$$\Delta_i = \sum_{t=1}^n \ell(s_t^i, y_t) - \ell(s_t^{i-1}, y_t).$$

Since $\inf_{W\in\mathcal{W}} \sum_{t=1}^n \ell(W\widetilde{x}_t^i, y_t) = \inf_{\alpha\in[-2,2]} \sum_{t=1}^n \ell(\alpha e_{l_t^i} + s_t^{i-1}, y_t)$, the regret bound above implies

$$\Delta_i \le \inf_{\alpha\in[-2,2]} \left[\sum_{t=1}^n \ell(\alpha e_{l_t^i} + s_t^{i-1}, y_t) - \ell(s_t^{i-1}, y_t)\right] + O(\log(n\log(nK))).$$

By Lemma 24 each term in the sum above satisfies

$$\ell(\alpha e_{l_t^i} + s_t^{i-1}, y_t) - \ell(s_t^{i-1}, y_t) \le \begin{cases} (e^\alpha - 1)\boldsymbol{\sigma}(s_t^{i-1})_{l_t^i} = (e^\alpha - 1)\widehat{C}_t^i(y_t, l_t^i), & l_t^i \ne y_t, \\ (e^{-\alpha} - 1)(1 - \boldsymbol{\sigma}(s_t^{i-1})_{y_t}) = -(e^{-\alpha} - 1)\widehat{C}_t^i(y_t, y_t), & l_t^i = y_t. \end{cases}$$

With notation $w^i = -\sum_{t=1}^n \widehat{C}_t^i(y_t, y_t)$, $c_+^i = -\frac{1}{w^i}\sum_{t:l_t^i=y_t} \widehat{C}_t^i(y_t, y_t)$, and $c_-^i = \frac{1}{w^i}\sum_{t:l_t^i\ne y_t} \widehat{C}_t^i(y_t, l_t^i)$, we rewrite

$$\inf_{\alpha\in[-2,2]} \left[\sum_{t=1}^n \ell(\alpha e_{l_t^i} + s_t^{i-1}, y_t) - \ell(s_t^{i-1}, y_t)\right] = w^i \cdot \inf_{\alpha\in[-2,2]} \left[(e^\alpha - 1)c_-^i + (e^{-\alpha} - 1)c_+^i\right].$$

One can verify that $w^i > 0$, $c_-^i, c_+^i \ge 0$, $c_+^i - c_-^i = \gamma_i \in [-1, 1]$ and $c_+^i + c_-^i \le 1$. By Lemma 25, it follows that

$$w^i \cdot \inf_{\alpha\in[-2,2]} \left[(e^{-\alpha} - 1)c_-^i + (e^\alpha - 1)c_+^i\right] \le -\frac{w^i\gamma_i^2}{2}.$$

Summing Δ_i over $i \in [N]$, we have

$$\sum_{t=1}^n \ell(s_t^N, y_t) - \sum_{t=1}^n \ell(s_t^0, y_t) = \sum_{i=1}^N \Delta_i \le -\frac{1}{2}\sum_{i=1}^N w^i\gamma_i^2 + O(N\log(n\log(nK))). \tag{13}$$

We lower bound the left hand side as

$$\sum_{t=1}^n \ell(s_t^N, y_t) - \sum_{t=1}^n \ell(s_t^0, y_t) \ge -\sum_{t=1}^n \ell(s_t^0, y_t) = -n\log(K),$$

where the inequality uses non-negativity of the logistic loss and the equality is a direct calculation from $s_t^0 = 0$. Next we upper bound the right-hand side of (13). Since $w^i = -\sum_{t=1}^n \widehat{C}_t^i(y_t, y_t)$, Eq. (12) implies

$$-\frac{1}{2}\sum_{i=1}^N w^i\gamma_i^2 \le -\frac{1}{4}\sum_{i=1}^N M_{i-1}\gamma_i^2 \le -\min_{i\in[N]} M_{i-1} \cdot \frac{1}{4}\sum_{i=1}^N \gamma_i^2 \le -\min_{i\in[N]} M_i \cdot \frac{1}{4}\sum_{i=1}^N \gamma_i^2.$$

Combining our upper and lower bounds on $\sum_{i=1}^N \Delta_i$ now gives

$$-n\log(K) \le -\frac{1}{2}\sum_{i=1}^N w^i\gamma_i^2 + O(N\log(n\log(K))) \le -\min_{i\in[N]} M_i \cdot \frac{1}{4}\sum_{i=1}^N \gamma_i^2 + O(N\log(n\log(nK))). \tag{14}$$

Rearranging, we have

$$\min_{i\in[N]} M_i \le O\left(\frac{n\log(K)}{\sum_{i=1}^N \gamma_i^2}\right) + O\left(\frac{N\log(n\log(nK))}{\sum_{i=1}^N \gamma_i^2}\right).$$

Returning to (11), this implies that with probability at least $1-\delta$,

$$\sum_{t=1}^n \mathbb{1}\{\hat{y}_t \neq y_t\} \le O\left(\frac{n\log(K)}{\sum_{i=1}^N \gamma_i^2}\right) + O\left(\frac{N\log(n\log(nK))}{\sum_{i=1}^N \gamma_i^2}\right) + 2\log(N/\delta),$$

which finishes the proof. ■

Proof [Proof of Proposition 9] By the definition of the cost matrices, the weak learning condition

$$\sum_{t=1}^n C_t^i(y_t, l_t^i) \le \sum_{t=1}^n \mathop{\mathbb{E}}_{k\sim u_{\gamma,y_t}}\left[C_t^i(y_t,k)\right] + S$$

implies

$$\sum_{t=1}^n \widehat{C}_t^i(y_t, l_t^i) \le \sum_{t=1}^n \mathop{\mathbb{E}}_{k\sim u_{\gamma,y_t}}\left[\widehat{C}_t^i(y_t,k)\right] + KS$$

Expanding the definitions of u_{γ,y_t} and $\widehat{C}_t^i$, we have

$$\mathop{\mathbb{E}}_{k\sim u_{\gamma,y_t}}\left[\widehat{C}_t^i(y_t,k)\right] = \left(\frac{1-\gamma}{K}\right)\left((\boldsymbol{\sigma}(s_t^{i-1})_{y_t} - 1) + \sum_{k\neq y_t} \boldsymbol{\sigma}(s_t^{i-1})_k\right) + \gamma(\boldsymbol{\sigma}(s_t^{i-1})_{y_t} - 1) = \gamma\widehat{C}_t^i(y_t,y_t).$$

So we have

$$\sum_{t=1}^n \widehat{C}_t^i(y_t, l_t^i) \le \gamma\sum_{t=1}^n \widehat{C}_t^i(y_t,y_t) + KS,$$

or, since $\widehat{C}_t^i(y_t,y_t) < 0$,

$$\gamma_i \ge \gamma - \frac{KS}{w^i},$$

where $w^i = -\sum_{t=1}^n C_t^i(y_t,y_t)$ as in the proof of Theorem 8. Since $a \ge b-c$ implies $a^2 \ge b^2 - 2bc$ for non-negative a, b and c, we further have $\gamma_i^2 \ge \gamma^2 - 2\frac{\gamma KS}{w^i}$.

Returning to the inequality (14), the bound we just proved implies

$$\begin{aligned}
-n\log(K) &\le -\frac{1}{2}\sum_{i=1}^N w^i\gamma^2 + \gamma KSN + O(N\log(n\log(nK))) \\
&\le -\frac{\gamma^2}{4}\sum_{i=1}^N M_{i-1} + \gamma KSN + O(N\log(n\log(nK))) && \text{(by (12))} \\
&\le -\min_{i\in[N]} M_i \cdot \frac{\gamma^2 N}{4} + \gamma KSN + O(N\log(n\log(nK))).
\end{aligned}$$

From here we proceed as in the proof of Theorem 8 to get the result. ■

Lemma 22 (Freedman's Inequality (Beygelzimer et al., 2011)) *Let $(Z_t)_{t\le n}$ be a real-valued martingale difference sequence adapted to a filtration $(\mathcal{J}_t)_{t\le n}$ with $|Z_t| \le R$ almost surely. For any $\eta \in [0, 1/R]$, with probability at least $1-\delta$,*

$$\sum_{t=1}^{n} Z_t \le \eta(e-2)\sum_{t=1}^{n}\mathbb{E}\left[Z_t^2 \mid \mathcal{J}_t\right] + \frac{\log(1/\delta)}{\eta} \tag{15}$$

for all $\eta \in [0, 1/R]$.

Lemma 23 *With probability at least $1-\delta$, the predictions $(\hat{y}_t)_{t\le n}$ generated by Algorithm 3 satisfy*

$$\sum_{t=1}^{n}\mathbb{1}\{\hat{y}_t \neq y_t\} \le 4\min_i \sum_{t=1}^{n}\mathbb{1}\left\{\hat{y}_t^i \neq y_t\right\} + 2\log(N/\delta).$$

Proof Define a filtration $(\mathcal{J}_t)_{t\le n}$ via

$$\mathcal{J}_t = \sigma((x_1, (l_1^i)_{i\le N}, y_1, i_1), \ldots, (x_{t-1}, (l_{t-1}^i)_{i\le N}, y_{t-1}, i_{t-1}), x_t, (l_t^i)_{i\le N}).$$

Since Line 18 of Algorithm 3 implements the multiplicative weights strategy with learning rate 1, the standard analysis (e.g. Cesa-Bianchi and Lugosi (2006)) implies that the conditional expectations under this strategy enjoy a regret bound of

$$\sum_{t=1}^{n}\mathbb{E}[\mathbb{1}\{\hat{y}_t \neq y_t\} \mid \mathcal{J}_t] \le 2\min_i \sum_{t=1}^{n}\mathbb{1}\left\{\hat{y}_t^i \neq y_t\right\} + \log(N).$$

Let $Z_t = \mathbb{1}\{\hat{y}_t \neq y_t\} - \mathbb{E}[\mathbb{1}\{\hat{y}_t \neq y_t\} \mid \mathcal{J}_t]$. Lemma 22 applied with $\eta = 1$ shows that with probability at least $1-\delta$,

$$\sum_{t=1}^{n} Z_t \le \sum_{t=1}^{n}\mathbb{E}\left[Z_t^2 \mid \mathcal{J}_t\right] + \log(1/\delta).$$

Since variance is bounded by second moment, we have

$$\sum_{t=1}^{n}\mathbb{E}\left[Z_t^2 \mid \mathcal{J}_t\right] \le \sum_{t=1}^{n}\mathbb{E}\left[(\mathbb{1}\{\hat{y}_t \neq y_t\})^2 \mid \mathcal{J}_t\right] = \sum_{t=1}^{n}\mathbb{E}[\mathbb{1}\{\hat{y}_t \neq y_t\} \mid \mathcal{J}_t].$$

Rearranging, we have proved that with probability $1-\delta$,

$$\sum_{t=1}^{n}\mathbb{1}\{\hat{y}_t \neq y_t\} \le 2\sum_{t=1}^{n}\mathbb{E}[\mathbb{1}\{\hat{y}_t \neq y_t\} \mid \mathcal{J}_t] + \log(1/\delta) \le 4\min_i \sum_{t=1}^{n}\mathbb{1}\left\{\hat{y}_t^i \neq y_t\right\} + 2\log(N/\delta).$$

■

Lemma 24 *The multiclass logistic loss satisfies for any $z \in \mathbb{R}^K$ and $y \in [K]$,*

$$\ell(z + \alpha e_l, y) - \ell(z, y) \le \begin{cases} (e^{\alpha}-1)\boldsymbol{\sigma}(z)_l, & l \neq y, \\ (e^{-\alpha}-1)(1-\boldsymbol{\sigma}(z)_y), & l = y. \end{cases}$$

Proof When $l \neq y$ we have

$$\begin{aligned}
\ell(z+\alpha e_l, y) - \ell(z,y) &= \log\left(\frac{1+\sum_{k\neq y,l} e^{z_k - z_y} + e^{z_l+\alpha-z_y}}{1+\sum_{k\neq y} e^{z_k-z_y}}\right) \\
&= \log\left(1 + (e^{\alpha}-1)\frac{e^{z_l-z_y}}{1+\sum_{k\neq y} e^{z_k-z_y}}\right) \\
&= \log(1+(e^{\alpha}-1)\boldsymbol{\sigma}(z)_l) \\
&\leq (e^{\alpha}-1)\boldsymbol{\sigma}(z)_l. && (\log(1+x)\leq x)
\end{aligned}$$

When $l = y$ we have

$$\begin{aligned}
\ell(z+\alpha e_l, y) - \ell(z,y) &= \log\left(\frac{1+e^{-\alpha}\sum_{k\neq y} e^{z_k - z_y}}{1+\sum_{k\neq y} e^{z_k-z_y}}\right) \\
&= \log\left(1 + (e^{-\alpha}-1)\frac{\sum_{k\neq y} e^{z_k-z_y}}{1+\sum_{k\neq y} e^{z_k-z_y}}\right) \\
&= \log\left(1 + (e^{-\alpha}-1)\sum_{k\neq y}\boldsymbol{\sigma}(z)_k\right) \\
&= \log(1+(e^{-\alpha}-1)(1-\boldsymbol{\sigma}(z)_y)) \\
&\leq (e^{-\alpha}-1)(1-\boldsymbol{\sigma}(z)_y). && (\log(1+x)\leq x)
\end{aligned}$$

■

Lemma 25 (Jung et al. (2017)) *For any $A, B \geq 0$ with $A - B \in [-1,+1]$ and $A + B \leq 1$,*

$$\inf_{\alpha\in[-2,2]}[A(e^{\alpha}-1) + B(e^{-\alpha}-1)] \leq -\frac{(A-B)^2}{2}.$$

A.4. Proof from Section 5

Theorem 26 *Let $\mathcal{F}$ be a class of functions $f : \mathcal{X} \to \Delta_K$. Suppose there is an online multiclass learning algorithm over $\mathcal{F}$ using the log loss that for any data sequence $(x_t, y_t) \in \mathcal{X} \times [K]$ for $t = 1, 2, \ldots, n$ produces distributions $p_t \in \Delta_K$ such that the following regret bound holds:*

$$\sum_{t=1}^{n} \ell_{\log}(p_t, y_t) - \inf_{f\in\mathcal{F}} \sum_{t=1}^{n} \ell_{\log}(f(x_t), y_t) \leq R(n).$$

Here $R(n)$ is some function of n and other relevant problem dependent parameters. Then for any given $\delta > 0$ and any (unknown) distribution $\mathcal{D}$ over $\mathcal{X} \times [K]$, it is possible to construct a predictor $g : \mathcal{X} \to \Delta_K$ using n samples $\{(x_t, y_t)\}_{t=1}^n$ drawn from $\mathcal{D}$ such that with probability at least $1 - \delta$, the excess risk of g is bounded as

$$\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(g(x), y)] \leq \inf_{f\in\mathcal{F}} \mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(f(x), y)] + O\left(\frac{\log\left(\frac{1}{\delta}\right)R\left(\frac{n}{\log(1/\delta)}\right) + \log(Kn)\log\left(\frac{\log(n)}{\delta}\right)}{n}\right).$$

Proof [Proof of Theorem 26] Recall that the standard online-to-batch conversion (Helmbold and Warmuth, 1995) produces an (improper) predictor using n data samples by running the online algorithm on those samples and stopping at a random time. Then predictor is online algorithm with its the internal state frozen. This predictor has excess risk bounded by the average regret over n rounds, in expectation over the n data samples.

The algorithm to generate the predictor g with the specified excess risk bound in the theorem statement is given below:

1. Let $M = \lceil \log(2/\delta) \rceil$. Produce M predictors $h_1, \dots, h_M : \mathcal{X} \to \Delta_K$ by using the online-to-batch conversion on the online multiclass learning algorithm run using M disjoint sets of $n/2M$ samples each. Call the ith such set of samples S_i
2. For $i \in [M]$, define $\tilde{h}_i : \mathcal{X} \to \Delta_K$ as $\tilde{h}_i(x) = \mathrm{smooth}_\mu(h_i(x))$ for $\mu = \frac{R(n/M)}{2n/M}$.
3. Construct an online convex optimization instance as follows. The learner's decision set is Δ_M, the set of all distributions on $[M]$. For every data point $(x, y) \in \mathcal{X} \times [K]$, associate the loss function $\ell_{(x,y)} : \Delta_M \to \mathbb{R}$ defined as $\ell_{(x,y)}(q) = -\log(\mathbb{E}_{i\sim q}[(\tilde{h}_i(x))_y])$. These loss functions are 1-exp-concave, so run the EWOO algorithm (Hazan et al., 2007) using the remaining $n/2$ examples sequentially to generate loss functions. Let $\bar{q}$ be the average of all the distributions in Δ_M generated by EWOO. Define $g := \mathbb{E}_{i\sim\bar{q}}[\tilde{h}_i]$.

We now proceed to analyse the excess risk of g. First, using the regret bound for the online multiclass learning algorithm, and in-expecation bound on the excess risk for online-to-batch conversion, for every $i \in [M]$, we have

$$\mathop{\mathbb{E}}_{S_i}\left[\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(h_i(x), y)]\right] \le \inf_{f\in\mathcal{F}} \mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(f(x), y)] + \frac{R(n/M)}{n/M}.$$

For any $p \in \Delta_K$, if $\tilde{p} = \mathrm{smooth}_\mu(p)$, then for any $y \in [K]$ we have $-\log(\tilde{p}_y) + \log(p_y) = \log(\frac{p_y}{(1-\mu)p_y + \mu/K}) \le 2\mu$. So for every $i \in [M]$, we have

$$\mathop{\mathbb{E}}_{S_i}\left[\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(\tilde{h}_i(x), y)]\right] \le \mathop{\mathbb{E}}_{S_i}\left[\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(h_i(x), y)]\right] + 2\mu.$$

Putting the above two bounds together, using the specified value of μ and an application of Markov's inequality, with probability at least $1 - e^{-M} = 1 - \frac{\delta}{2}$, there exists some $i^\star \in [M]$ such that

$$\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(\tilde{h}_{i^\star}(x), y)] \le \inf_{f\in\mathcal{F}} \mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(f(x), y)] + \frac{2eR(n/M)}{n/M}. \tag{16}$$

The EWOO algorithm in step 3 of the procedure enjoys a regret bound of $O(M\log(n))$ (the online convex optimization problem is an instance of online portfolio selection over M instruments, see (Hazan et al., 2007)). Furthermore, the application of smooth_μ makes the range for the log loss be bounded by $\log(K/\mu)$. Thus, by Corollary 2 of Mehta (2017), with probability at least $1 - \frac{\delta}{2}$,

$$\begin{aligned}\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(g(x), y)] &= \mathop{\mathbb{E}}_{(x,y)}[-\log(\mathop{\mathbb{E}}_{i\sim\bar{q}}[(\tilde{h}_i(x))_y])] \\ &\le \mathop{\mathbb{E}}_{(x,y)}[-\log((\tilde{h}_{i^\star}(x))_y)] + O\left(\frac{M\log(n) + \log(K/\mu)\log(\log(n)/\delta)}{n}\right) \end{aligned} \tag{17}$$

Note that $\ell_{\log}(\tilde{h}_{i^\star}(x), y) = -\log((\tilde{h}_{i^\star}(x))_y)$. Applying the union bound and combining inequalities (16) and (17) with some simplification of the bounds using the value of M, with probability at least $1-\delta$ we have

$$\mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(g(x),y)] \le \inf_{f\in\mathcal{F}} \mathop{\mathbb{E}}_{(x,y)}[\ell_{\log}(f(x),y)] + O\left(\frac{\log\left(\frac{1}{\delta}\right)R\left(\frac{n}{\log(1/\delta)}\right) + \log(Kn)\log\left(\frac{\log(n)}{\delta}\right)}{n}\right).$$

■

A.5. Details from Section 6

For this section we let ℓ denote the unweighted multiclass logistic loss: the multiclass logistic loss defined in Section 1.1 for the special case where $\mathcal{Y} = \{e_i\}_{i\in[K]}$. Before proving Theorem 11 we need a few preliminaries. First, we state a version of the Aggregating Algorithm with the logistic loss for finite classes.

Lemma 27 *Let $\mathcal{F}$ be any finite class of sequences of the form $f = (f_t)_{t\le n}$ with $f_t \in \mathbb{R}^K$, where each f_t is available at time t and may depend on $y_{1:t-1}$. Define a strategy*

1. $P_t(f) \propto \exp\left(-\sum_{s=1}^{t-1}\ell(f_s, y_s)\right)$ *(so $P_1 = \mathrm{Uniform}(\mathcal{F})$).*
2. $\hat{z}_t = \boldsymbol{\sigma}^+(\mathrm{smooth}_{\frac{1}{n}}(\mathbb{E}_{f\sim P_t}[\boldsymbol{\sigma}(f_t)]))$.

This strategy enjoys a regret bound of

$$\sum_{t=1}^{n}\ell(\hat{z}_t, y_t) - \min_{f\in\mathcal{F}}\sum_{t=1}^{n}\ell(f_t, y_t) \le \log|\mathcal{F}| + 2. \tag{18}$$

Furthermore, the predictions satisfy $\|\hat{z}_t\|_\infty \le \log(Kn)$.

Proof [Proof of Lemma 27] First consider the closely related strategy $\widetilde{z}_t := \boldsymbol{\sigma}^+(\mathbb{E}_{f\sim P_t}[\boldsymbol{\sigma}(f(x_t))])$. In light of the 1-mixability for the logistic loss proven in Proposition 1, $\widetilde{z}_t$ is precisely the finite class version of the Aggregating Algorithm, which guarantees (Cesa-Bianchi and Lugosi, 2006):

$$\sum_{t=1}^{n}\ell(\widetilde{z}_t, y_t) - \min_{f\in\mathcal{F}}\sum_{t=1}^{n}\ell(f_t, y_t) \le \log|\mathcal{F}|.$$

To establish the final result we simply appeal to Lemma 16, using that $\boldsymbol{\sigma}(\boldsymbol{\sigma}^+(p)) = p\ \forall p \in \Delta_K$. ■

We now formally define a multiclass generalization of a sequential cover.

Definition 28 *For any set $\mathcal{Z}$, a $\mathcal{Z}$-valued K-ary tree of depth n is a sequence $\mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)$ of n mappings with $\mathbf{z}_t : [K]^{t-1} \to \mathcal{Z}$.*

Definition 29 *A set V of $\mathbb{R}^K$-valued K-ary trees is an α-cover (w.r.t. the L_p norm) of $\mathcal{F}$ on an $\mathcal{X}$-valued K-ary tree $\mathbf{x}$ of depth n with loss ℓ if*

$$\forall f\in\mathcal{F},\ y\in[K]^n,\ \exists \mathbf{v}\in V \text{ s.t. } \left(\frac{1}{n}\sum_{t=1}^{n}\max_{y'_t\in[K]}\left|\ell(f(\mathbf{x}_t(y)), y'_t) - \ell(\mathbf{v}_t(y), y'_t)\right|^p\right)^{1/p} \le \alpha.$$

Definition 30 *The L_p covering number of $\mathcal{F}$ on tree $\mathbf{x}$ is defined as*

$$\mathcal{N}_p(\alpha, \ell \circ \mathcal{F}, \mathbf{x}) := \min\{|V| : V \text{ is an } \alpha\text{-cover of } \mathcal{F} \text{ on } \mathbf{x} \text{ w.r.t. the } L_p \text{ norm}\}.$$

Further, define $\mathcal{N}_p(\alpha, \ell \circ \mathcal{F}) = \sup_{\mathbf{x}} \mathcal{N}_p(\alpha, \ell \circ \mathcal{F}, \mathbf{x})$.

If $K = 2$ then the above definition coincides with the definition of sequential cover in Rakhlin et al. (2015a) which was defined for real valued function classes.

We also need a slight generalization of the notion of covering number defined in Definition 29 for intermediate results.

Definition 31 *Let U be a collection of $\mathbb{R}^K$-valued K-ary trees. A set V of $\mathbb{R}^K$-valued K-ary trees is an α-cover with respect to the L_p norm for U if*

$$\forall \mathbf{u} \in U,\ y \in [K]^n,\ \exists \mathbf{v} \in V \text{ s.t. } \left(\frac{1}{n}\sum_{t=1}^{n} \max_{y'_t \in [K]} \left|\ell(\mathbf{u}_t(y), y'_t) - \ell(\mathbf{v}_t(y), y'_t)\right|^p\right)^{1/p} \le \alpha.$$

Definition 32 *The L_p covering number for a collection of trees U with loss ℓ is*

$$\mathcal{N}_p(\alpha, \ell \circ U) := \min\{|V| : V \text{ is an } \alpha\text{-cover of } U \text{ w.r.t. the } L_p \text{ norm}\}.$$

Proof [Proof of Theorem 11] Define a subset of the output space:

$$\mathcal{Z} := \left\{z \in \mathbb{R}^K \mid \|z\|_\infty \le \log(Kn)\right\}.$$

We move to an upper bound on the minimax value by restricting predictions to $\mathcal{Z}$:

$$\begin{aligned}\mathcal{V}_n(\mathcal{F}) &= \left\langle\!\!\left\langle \sup_{x_t \in \mathcal{X}} \inf_{\hat{z}_t \in \mathbb{R}^K} \max_{y_t \in [K]} \right\rangle\!\!\right\rangle_{t=1}^{n} \left[\sum_{t=1}^{n} \ell(\hat{z}_t, y_t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t)\right] \\ &\le \left\langle\!\!\left\langle \sup_{x_t \in \mathcal{X}} \inf_{\hat{z}_t \in \mathcal{Z}} \max_{y_t \in [K]} \right\rangle\!\!\right\rangle_{t=1}^{n} \left[\sum_{t=1}^{n} \ell(\hat{z}_t, y_t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t)\right].\end{aligned}$$

Note that $\mathcal{Z}$ is a compact subset of a separable metric space and that ℓ is convex with respect to $\hat{z}$. Therefore, using repeated application of minimax theorem following Rakhlin et al. (2010)[11] the minimax value can be written as:

$$= \left\langle\!\!\left\langle \sup_{x_t \in \mathcal{X}} \sup_{p_t \in \Delta_K} \inf_{\hat{z}_t \in \mathcal{Z}} \mathbb{E}_{y_t \sim p_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \left[\sum_{t=1}^{n} \ell(\hat{z}_t, y_t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t)\right].$$

Now we perform a standard manipulation of the sup and loss terms as in Rakhlin et al. (2010):

$$= \left\langle\!\!\left\langle \sup_{x_t \in \mathcal{X}} \sup_{p_t \in \Delta_K} \mathbb{E}_{y_t \sim p_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \left[\sum_{t=1}^{n} \inf_{\hat{z}_t \in \mathcal{Z}} \mathbb{E}_{y_t \sim p_t}\left[\ell(\hat{z}_t, y_t)\right] - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(x_t), y_t)\right] \tag{19}$$

$$= \sup_{\mathbf{x}, \mathbf{p}} \mathbb{E}_{y \sim \mathbf{p}} \left[\sum_{t=1}^{n} \inf_{\hat{z}_t \in \mathcal{Z}} \mathbb{E}_{y_t \sim \mathbf{p}_t(y)}\left[\ell(\hat{z}_t, y_t)\right] - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell(f(\mathbf{x}_t(y)), y_t)\right]. \tag{20}$$

11. See Rakhlin et al. (2010) for an extensive discussion of the technicalities.

In the final line above we have introduced new notation. $\mathbf{x}$ and $\mathbf{p}$ are $\mathcal{X}$- and Δ_K-valued K-ary trees of depth n. That is, $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ where $\mathbf{x}_t : [K]^{t-1} \to \mathcal{X}$ and similarly for the tree $\mathbf{p} = (\mathbf{p}_1, \dots, \mathbf{p}_n)$, $\mathbf{p}_t : [K]^{t-1} \to \Delta_K$. The notation "$y \sim \mathbf{p}$" refers to the process in which we first draw $y_1 \sim \mathbf{p}_1$, then draw $y_t \sim \mathbf{p}_t(y_1, \dots, y_{t-1})$ for subsequent timesteps t. We also overload the notation as $\mathbf{p}_t(y) := \mathbf{p}_t(y_{1:t-1})$, and likewise for $\mathbf{x}$.

With this notation, (20) is seen to be (19) rewritten using that at time t, based on draw of previous ys, x_t and p_t are chosen to maximize the remaining game value; this process be represented via K-ary tree.

Note that the sequence $(\hat{z}_t)_{t \le n}$ being minimized over in (19) can depend on the full trees $\mathbf{x}$ and $\mathbf{p}$, but that it is adapted to the path $(y_t)_{t \le n}$, meaning that the value at time t ($\hat{z}_t$) can only depend on the $y_{1:t-1}$. This property is imporant because the choice we exhibit for $(\hat{z}_t)_{t \le n}$ will indeed depend on the full trees.

In light of the discussion in Section 6, the key advantage of having moved to the dual game above is that we can condition on the K-ary tree $\mathbf{x}$ and cover $\mathcal{F}$ only on this tree. Let V^γ be a minimal γ-sequential cover of $\ell \circ \mathcal{F}$ on the tree $\mathbf{x}$ with respect to the L_2 norm (in the sense of Definition 29).

Keeping the tree $\mathbf{x}$ fixed, for each tree $\mathbf{v} \in V^\gamma$, each $f \in \mathcal{F}$, we define a class of trees $\mathcal{F}_\mathbf{v}$ "centered" at $\mathbf{v}$—in a sense that will be made precise in a moment—via the following procedure.

- $\mathcal{F}_\mathbf{v} = \emptyset$.
- For each $f \in \mathcal{F}$ and $y \in [K]^n$ with $\sqrt{\frac{1}{n}\sum_{t=1}^n \max_{y_t'' \in [K]} (\ell(f(\mathbf{x}_t(y)), y_t'') - \ell(\mathbf{v}_t(y), y_t''))^2} \le \gamma$:
  - Define a $\mathbb{R}^K$-valued K-ary tree $\mathbf{u}_{f,y}$ via: For each $y' \in [K]^n$,

    $$(\mathbf{u}_{f,y})_t(y') := f(\mathbf{x}_t(y'))\mathbb{1}\{y_1' = y_1, \dots, y_{t-1}' = y_{t-1}\} + \mathbf{v}_t(y')\mathbb{1}\{\neg(y_1' = y_1, \dots, y_{t-1}' = y_{t-1})\}.$$

    In other words, $\mathbf{u}_{f,y}$ is equal to $f \circ \mathbf{x}$ on the path y, and equal to $\mathbf{v}$ everywhere else.
  - Add $\mathbf{u}_{f,y}$ to $\mathcal{F}_\mathbf{v}$.

The class $\mathcal{F}_\mathbf{v}$ has two important properties which are formally proven in an auxiliary lemma, Lemma 33: First, its L_2 covering number is (up to low order terms) bounded in terms of the L_2 covering number of the class $\mathcal{F} \circ \mathbf{x}$, so it has similar complexity to this class. Second, its L_2 radius is bounded by γ, in the sense that its covering number at scale γ is at most 1.

Note that on any path $y \in [K]^n$ and for each $f \in \mathcal{F}$, there exist $\mathbf{v} \in V^\gamma$ and $\mathbf{u} \in \mathcal{F}_\mathbf{v}$ such that $f(\mathbf{x}_t(y)) = \mathbf{u}_t(y)$. This is because a $\mathbf{v}$ that is γ-close to f on the path y through $\mathbf{x}$ is guaranteed by the cover property of V^γ, and so we can take $\mathbf{u}_{f,y}$ in $\mathcal{F}_\mathbf{v}$ as the desired $\mathbf{u}$. This implies that

$$\inf_{f \in \mathcal{F}} \sum_{t=1}^n \ell(f(\mathbf{x}_t(y)), y_t) \ge \min_{\mathbf{v} \in V^\gamma} \inf_{\mathbf{u} \in \mathcal{F}_\mathbf{v}} \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t).$$

With this we are ready to return to the minimax rate. We already established that

$$\mathcal{V}_n(\mathcal{F}) \le \sup_{\mathbf{x},\mathbf{p}} \mathbb{E}_{y \sim \mathbf{p}} \left[\sum_{t=1}^n \inf_{\hat{z}_t \in \mathcal{Z}} \mathbb{E}_{y_t \sim \mathbf{p}_t(y)} [\ell(\hat{z}_t, y_t)] - \inf_{f \in \mathcal{F}} \sum_{t=1}^n \ell(f(\mathbf{x}_t(y)), y_t) \right].$$

We now move to an upper bound based on the constructions for the tree collections V^γ and $\{\mathcal{F}_\mathbf{v}\}_{\mathbf{v}\in V^\gamma}$. These collections depend only on the tree $\mathbf{x}$ at the outer supremum above. Writing the choice of these collections as an infimum to make its dependence on the other quantities in the random process as explicit as possible, and using the containment just shown:

$$\leq \sup_{\mathbf{x}} \inf_{V^\gamma} \inf_{\{\mathcal{F}_\mathbf{v}\}_{\mathbf{v}\in V^\gamma}} \sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\left[\sum_{t=1}^n \inf_{\hat{z}_t\in\mathcal{Z}} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)}\left[\ell(\hat{z}_t,y_t)\right] - \min_{\mathbf{v}\in V^\gamma}\inf_{\mathbf{u}\in\mathcal{F}_\mathbf{v}}\sum_{t=1}^n \ell(\mathbf{u}_t(y),y_t)\right].$$

For the last time in the proof, we introduce a new collection of trees. For each $\mathbf{v}\in V^\gamma$ we introduce a $\mathcal{Z}$-valued K-ary tree $\hat{\mathbf{y}}^\mathbf{v}$, with $\hat{\mathbf{y}}^\mathbf{v}_t : [K]^{t-1}\to\mathcal{Z}$. We postpone explicitly constructing the trees for now, but the reader may think of each tree $\hat{\mathbf{y}}^\mathbf{v}$ as representing the optimal strategy for the set $\mathcal{F}_\mathbf{v}$ in a sense that will be made precise in a moment.

$$\begin{aligned}
&= \sup_{\mathbf{x}} \inf_{V^\gamma} \inf_{\{\mathcal{F}_\mathbf{v}\}_{\mathbf{v}\in V^\gamma}} \inf_{\{\hat{\mathbf{y}}^\mathbf{v}\}_{\mathbf{v}\in V^\gamma}} \sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\Bigg[\sum_{t=1}^n \inf_{\hat{z}_t\in\mathcal{Z}} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)}\left[\ell(\hat{z}_t,y_t)\right] \\
&\qquad\qquad - \min_{\mathbf{v}\in V^\gamma}\left\{\sum_{t=1}^n \ell(\hat{\mathbf{y}}^\mathbf{v}_t(y),y_t) - \sum_{t=1}^n \ell(\hat{\mathbf{y}}^\mathbf{v}_t(y),y_t) + \inf_{\mathbf{u}\in\mathcal{F}_\mathbf{v}}\sum_{t=1}^n \ell(\mathbf{u}_t(y),y_t)\right\}\Bigg] \\
&\leq \sup_{\mathbf{x}} \inf_{V^\gamma} \inf_{\{\mathcal{F}_\mathbf{v}\}_{\mathbf{v}\in V^\gamma}} \inf_{\{\hat{\mathbf{y}}^\mathbf{v}\}_{\mathbf{v}\in V^\gamma}} \Bigg\{ \underbrace{\sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\left[\sum_{t=1}^n \inf_{\hat{z}_t\in\mathcal{Z}} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)}\left[\ell(\hat{z}_t,y_t)\right] - \min_{\mathbf{v}\in V^\gamma}\sum_{t=1}^n \ell(\hat{\mathbf{y}}^\mathbf{v}_t(y),y_t)\right]}_{(\star)} \\
&\qquad + \underbrace{\sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\left[\max_{\mathbf{v}\in V^\gamma}\left\{\sum_{t=1}^n \ell(\hat{\mathbf{y}}^\mathbf{v}_t(y),y_t) - \inf_{\mathbf{u}\in\mathcal{F}_\mathbf{v}}\sum_{t=1}^n \ell(\mathbf{u}_t(y),y_t)\right\}\right]}_{(\star\star)} \Bigg\}.
\end{aligned} \tag{21}$$

We now bound terms $(\star)$ and $(\star\star)$ individually by instantiating specific choices for $(\hat{z}_t)_{t\leq n}$ and $\{\hat{\mathbf{y}}^\mathbf{v}\}$.

Term $(\star)$ We select $(\hat{z}_t)_{t\leq n}$ using the Aggregating Algorithm as configured in Lemma 27, taking $\mathcal{F}$ to be the finite collection of sequences $\{\hat{\mathbf{y}}^\mathbf{v}\}_{\mathbf{v}\in V^\gamma}$. Since each tree has the property that $\hat{\mathbf{y}}^\mathbf{v}_t$ only depends on $y_{1:t-1}$, Lemma 27 indeed applies, which means that for any sequence $y_{1:n}\in[K]^n$ of labels the algorithm deterministically satisfies the regret inequality

$$\sum_{t=1}^n \ell(\hat{z}_t,y_t) - \min_{\mathbf{v}\in V^\gamma}\sum_{t=1}^n \ell(\hat{\mathbf{y}}^\mathbf{v}_t(y),y_t) \leq \log|V^\gamma| + 2.$$

Since the algorithm guarantees $\|\hat{z}_t\|_\infty \leq \log(Kn)$, one can verify that $\hat{z}_t\in\mathcal{Z}$. Furthermore, $\hat{z}_t$ depends only on $y_{1:t-1}$, and so the predictions of the Aggregating Algorithm are a valid choice for the infimum in $(\star)$. This implies that

$$(\star) \leq \sup_{\mathbf{x}} \log|V^\gamma| + 2 \leq \log\mathcal{N}_2(\gamma,\ell\circ\mathcal{F}) + 2,$$

since the regret inequality holds for every possible draw of $y_{1:n}$ in the expression $(\star)$.

Term $(\star\star)$ First, observe that each tree class $\mathcal{F}_\mathbf{v}$ is uniformly bounded in the sense that

$$\sup_{\mathbf{u}\in\mathcal{F}_\mathbf{v}}\sup_{y\in[K]^n}\max_{t\in[n]}\|\mathbf{u}_t(y)\|_\infty < \infty.$$

This holds because $\mathbf{u}_t(y)$ is either equal to $\mathbf{v}_t(y)$, which is finite, or is equal to $f(\mathbf{x}_t(y))$ for some $f \in \mathcal{F}$, and the class $\mathcal{F}$ was already assumed to be uniformly bounded.

To bound this term we need a variant of the sequential Rademacher complexity regret bound of (Rakhlin et al., 2010), which shows that there exists a deterministic strategy for competing against any collection of trees. This is proven in the auxiliary Lemma 34 following this proof.

In particular, for each tree class $\mathcal{F}_{\mathbf{v}}$, there exists a deterministic strategy $\hat{y}_t^{\mathbf{v}}$ that guarantees the inequality

$$\sum_{t=1}^{n} \ell(\hat{y}_t^{\mathbf{v}}, y_t) - \inf_{\mathbf{u} \in \mathcal{F}_{\mathbf{v}}} \sum_{t=1}^{n} \ell(\mathbf{u}_t(y), y_t) \le 2 \cdot \max_{\mathbf{y}, \mathbf{y}'} \mathbb{E}_\epsilon \sup_{\mathbf{u} \in \mathcal{F}_{\mathbf{v}}} \left[\sum_{t=1}^{n} \epsilon_t \ell(\mathbf{u}_t(\mathbf{y}_{1:t-1}(\epsilon)), \mathbf{y}'_t(\epsilon)) \right] + 2,$$

holds for every sequence, where the supremum on the right-hand-side ranges over $[K]$-valued binary trees. Futhermore, $\hat{y}_t^{\mathbf{v}}$ is guaranteed by Lemma 34 to lie in the class $\mathcal{Z}$. We choose this strategy for the collection $\{\hat{\mathbf{y}}^{\mathbf{v}}\}$ being minimized over in (21). Since the regret inequality from Lemma 34 holds deterministically for all sequences y for each $\mathbf{v}$, we have that

$$(\star\star) \le 2 \cdot \max_{\mathbf{v} \in V^\gamma} \max_{\mathbf{y}, \mathbf{y}'} \mathbb{E}_\epsilon \sup_{\mathbf{u} \in \mathcal{F}_{\mathbf{v}}} \left[\sum_{t=1}^{n} \epsilon_t \ell(\mathbf{u}_t(\mathbf{y}_{1:t-1}(\epsilon)), \mathbf{y}'_t(\epsilon)) \right] + 2.$$

For each choice of $\mathbf{v}$, $\mathbf{y}$, $\mathbf{y}'$ at the outer supremum, we define a class of real-valued trees $W_{\mathbf{v},\mathbf{y},\mathbf{y}'}$ via $\{(\mathbf{w}_t)_{t \le n} \,:\, \mathbf{w}_t(\epsilon) := \ell(\mathbf{u}_t(\mathbf{y}(\epsilon_{1:t-1})), \mathbf{y}'_t(\epsilon)) \mid \mathbf{u} \in \mathcal{F}_{\mathbf{v}}\}$. Lemma 35 then implies

$$(\star\star) \le 2 \max_{\mathbf{v} \in V^\gamma} \max_{\mathbf{y}, \mathbf{y}'} \inf_{\alpha > 0} \left\{ 4\alpha n + 12 \int_{\alpha}^{\mathrm{rad}_2(W_{\mathbf{v},\mathbf{y},\mathbf{y}'})} \sqrt{n \log \mathcal{N}_2(\delta, W_{\mathbf{v},\mathbf{y},\mathbf{y}'})} d\delta \right\} + 2,$$

with the real-valued covering number $\mathcal{N}_2$ and radius rad_2 defined as in Lemma 35.

We now show how to bound this covering number in terms of the covering number for $\mathcal{F}_{\mathbf{v}}$. Suppose that Z is a collection of $\mathbb{R}^K$-valued K-ary trees that form a δ-cover for $\mathcal{F}_{\mathbf{v}}$ in the sense of Definition 31. Then we have

$$\begin{aligned}
&\sup_{\mathbf{u} \in \mathcal{F}_{\mathbf{v}}} \max_{\epsilon \in \{\pm 1\}^n} \inf_{\mathbf{z} \in Z} \sqrt{\frac{1}{n} \sum_{t=1}^{n} \left(\ell(\mathbf{u}_t(\mathbf{y}(\epsilon)), \mathbf{y}'_t(\epsilon)) - \ell(\mathbf{z}_t(\mathbf{y}(\epsilon)), \mathbf{y}'_t(\epsilon))\right)^2} \\
&\le \sup_{\mathbf{u} \in \mathcal{F}_{\mathbf{v}}} \max_{\epsilon \in \{\pm 1\}^n} \inf_{\mathbf{z} \in Z} \sqrt{\frac{1}{n} \sum_{t=1}^{n} \max_{y'_t \in [K]} \left(\ell(\mathbf{u}_t(\mathbf{y}(\epsilon)), y'_t) - \ell(\mathbf{z}_t(\mathbf{y}(\epsilon)), y'_t)\right)^2} \\
&\le \sup_{\mathbf{u} \in \mathcal{F}_{\mathbf{v}}} \max_{y \in [K]^n} \inf_{\mathbf{z} \in Z} \sqrt{\frac{1}{n} \sum_{t=1}^{n} \max_{y'_t \in [K]} \left(\ell(\mathbf{u}_t(y), y'_t) - \ell(\mathbf{z}_t(y), y'_t)\right)^2} \\
&\le \delta.
\end{aligned}$$

This implies that for any cover of $\mathcal{F}_{\mathbf{v}}$ in the sense of Definition 31 we can construct a cover for $W_{\mathbf{v},\mathbf{y},\mathbf{y}'}$ at the same scale using the construction $\{(\mathbf{w}_t)_{t \le n} \,:\, \mathbf{w}_t(\epsilon) := \ell(\mathbf{z}_t(\mathbf{y}(\epsilon_{1:t-1})), \mathbf{y}'_t(\epsilon)) \mid \mathbf{z} \in Z\}$. Consequently, we have

$$(\star\star) \le 2 \max_{\mathbf{v} \in V^\gamma} \inf_{\alpha > 0} \left\{ 4\alpha n + 12 \int_{\alpha}^{\mathrm{rad}_2(\mathcal{F}_{\mathbf{v}})} \sqrt{n \log \mathcal{N}_2(\delta, \ell \circ \mathcal{F}_{\mathbf{v}})} d\delta \right\} + 2.$$

In light of Lemma 33, this is further upper bounded by

$$(\star\star) \le 2 \inf_{\alpha>0}\left\{4\alpha n + 12\int_\alpha^\gamma \sqrt{n\log(\mathcal{N}_2(\delta, \ell\circ\mathcal{F}, \mathbf{x})n)}d\delta\right\} + 2$$
$$\le 2 \inf_{\alpha>0}\left\{4\alpha n + 12\int_\alpha^\gamma \sqrt{n\log(\mathcal{N}_2(\delta, \ell\circ\mathcal{F})n)}d\delta\right\} + 2.$$

Final bound Combining $(\star)$ and $(\star\star)$, we have

$$\mathcal{V}_n(\mathcal{F}) \le \log\mathcal{N}_2(\gamma, \ell\circ\mathcal{F}) + \inf_{\gamma\ge\alpha>0}\left\{8\alpha n + 24\int_\alpha^\gamma \sqrt{n\log(\mathcal{N}_2(\delta, \ell\circ\mathcal{F})n)}d\delta\right\} + 4.$$

for any fixed γ. Optimizing over γ yields the result. ■

Lemma 33 *Let $\mathcal{F}_{\mathbf{v}}$ be defined as in the proof of Theorem 11 for trees $\mathbf{v}$ and $\mathbf{x}$ and scale γ. Then it holds that*

1. $\mathcal{N}_2(\gamma, \ell\circ\mathcal{F}_{\mathbf{v}}) \le 1$.
2. $\mathcal{N}_2(\alpha, \ell\circ\mathcal{F}_{\mathbf{v}}) \le n\cdot\mathcal{N}_2(\alpha, \ell\circ\mathcal{F}, \mathbf{x})$ *for all* $\alpha > 0$.

Proof [Proof of Lemma 33]
First claim This is essentially by construction. Recall that each element of $\mathcal{F}_v$ is of the form

$$(\mathbf{u}_{f,y})_t(y') := f(\mathbf{x}_t(y'))\mathbb{1}\{y'_1 = y_1, \ldots, y'_{t-1} = y_{t-1}\} + \mathbf{v}_t(y')\mathbb{1}\{\neg(y'_1 = y_1, \ldots, y'_{t-1} = y_{t-1})\}.$$

for some path $y \in [K]^n$ and $f \in \mathcal{F}$ for which

$$\sqrt{\frac{1}{n}\sum_{t=1}^n \max_{y''_t\in[K]}(\ell(f(\mathbf{x}_t(y)), y''_t) - \ell(\mathbf{v}_t(y), y''_t))^2} \le \gamma. \tag{22}$$

These properties imply that $\{\mathbf{v}\}$ is a sequential γ-cover. Indeed, using the explicit form for $\boldsymbol{u}_{f,y}$ above, it can be seen that for each path $y' \in [K]^n$, there exists some time $1 < \tau \le n+1$ such that

$$(\mathbf{u}_{f,y})_t(y') = \begin{cases} f(\mathbf{x}_t(y')), & \text{if } t < \tau, \\ \mathbf{v}_t(y'), & \text{if } t \ge \tau. \end{cases}$$

it also holds that $y_t = y'_t$ for all $t < \tau - 1$.

Using this representation we have that for any path $y' \in [K]^n$:

$$\sqrt{\frac{1}{n}\sum_{t=1}^n \max_{y''_t\in[K]}(\ell((\mathbf{u}_{f,y})_t(y'), y''_t) - \ell(\mathbf{v}_t(y'), y''_t))^2}$$
$$= \sqrt{\frac{1}{n}\sum_{t=1}^{\tau-1} \max_{y''_t\in[K]}(\ell(f(\mathbf{x}_t(y'), y''_t) - \ell(\mathbf{v}_t(y'), y''_t))^2}.$$

Now use that $\mathbf{x}_1, \ldots, \mathbf{x}_{\tau-1}$ and $\mathbf{v}_1, \ldots, \mathbf{v}_{\tau-1}$ only depend on $y'_1, \ldots, y'_{\tau-2}$, and that $y'_1, \ldots, y'_{\tau-2} = y_1, \ldots, y_{\tau-2}$:

$$\begin{aligned}
&= \sqrt{\frac{1}{n}\sum_{t=1}^{\tau-1} \max_{y''_t \in [K]} (\ell(f(\mathbf{x}_t(y), y''_t) - \ell(\mathbf{v}_t(y), y''_t))^2} \\
&\leq \sqrt{\frac{1}{n}\sum_{t=1}^{n} \max_{y''_t \in [K]} (\ell(f(\mathbf{x}_t(y), y''_t) - \ell(\mathbf{v}_t(y), y''_t))^2} \\
&\leq \gamma.
\end{aligned}$$

Second claim Let V be a cover for $\ell \circ \mathcal{F}$ on $\mathbf{x}$ of size $\mathcal{N}_2(\alpha, \ell \circ \mathcal{F}, \mathbf{x})$. Assume $|V| < \infty$ as the claim holds trivially otherwise. We will construct from V a cover $\widetilde{V}$ for $\ell \circ \mathcal{F}_{\mathbf{v}}$ with the following procedure:

- $\widetilde{V} = \varnothing$.
- For each K-ary $\mathbb{R}^K$-valued tree $\mathbf{z} \in V$ and each time $\tau \in \{2, \ldots, n+1\}$:
  - Construct tree K-ary $\mathbb{R}^K$-valued tree $\mathbf{z}^{(\tau)}$ via

$$\mathbf{z}_t^{(\tau)}(y) = \mathbf{z}_t(y)\mathbb{1}\{t < \tau\} + \mathbf{v}_t(y)\mathbb{1}\{t \geq \tau\}.$$

  - Add $\mathbf{z}^{(\tau)}$ to $\widetilde{V}$.

Clearly $\left|\widetilde{V}\right| \leq n \cdot |V|$. We now show that $\widetilde{V}$ is an α-cover for $\ell \circ \mathcal{F}_{\mathbf{v}}$.

Let $\mathbf{u}_{f,y}$ be an element of $\mathcal{F}_{\mathbf{v}}$ of the form described in the proof of the first claim and let $y' \in [K]^n$ be a particular path. Let τ be such that $(\mathbf{u}_{f,y})_t(y') = f(\mathbf{x}_t(y'))\mathbb{1}\{t < \tau\} + \mathbf{v}_t(y')\mathbb{1}\{t \geq \tau\}$. Let $\mathbf{z} \in V$ be α-close to f on the path y' through $\mathbf{x}$, i.e.

$$\sqrt{\frac{1}{n}\sum_{t=1}^{n} \max_{y''_t \in [K]} (\ell(f(\mathbf{x}_t(y'), y''_t) - \ell(\mathbf{z}_t(y'), y''_t))^2} \leq \alpha.$$

Existence of such a $\mathbf{z}$ is guaranteed by the cover property of V. We will show that $\mathbf{z}^{(\tau)}$ is α-close to $\mathbf{u}_{f,y}$ on y'. Indeed, we have

$$\begin{aligned}
&\sqrt{\frac{1}{n}\sum_{t=1}^{n} \max_{y''_t \in [K]} (\ell((\mathbf{u}_{f,y})_t(y'), y''_t) - \ell(\mathbf{z}_t^{(\tau)}(y'), y''_t))^2} \\
&= \sqrt{\frac{1}{n}\sum_{t=1}^{\tau-1} \max_{y''_t \in [K]} (\ell(f(\mathbf{x}_t(y'), y''_t) - \ell(\mathbf{z}_t(y'), y''_t))^2 + \frac{1}{n}\sum_{t=\tau}^{n} \max_{y''_t \in [K]} (\ell(\mathbf{v}_t(y'), y''_t) - \ell(\mathbf{v}_t(y'), y''_t))^2} \\
&= \sqrt{\frac{1}{n}\sum_{t=1}^{\tau-1} \max_{y''_t \in [K]} (\ell(f(\mathbf{x}_t(y'), y''_t) - \ell(\mathbf{z}_t(y'), y''_t))^2} \\
&\leq \sqrt{\frac{1}{n}\sum_{t=1}^{n} \max_{y''_t \in [K]} (\ell(f(\mathbf{x}_t(y'), y''_t) - \ell(\mathbf{z}_t(y'), y''_t))^2} \\
&\leq \alpha.
\end{aligned}$$

Since this argument works for any $\mathbf{u}_{f,y} \in \mathcal{F}_{\mathbf{v}}$ this establishes that $\widetilde{V}$ is an α-cover of $\mathcal{F}_{\mathbf{v}}$. ■

The next lemma is almost the same as the sequential Rademacher complexity bound in Rakhlin et al. (2010), with the only technical difference being that the learner competes with a class of trees rather than a class of fixed functions. It is proven using the same argument as in that paper.

Lemma 34 *Let U be any collection of $\mathbb{R}^K$-valued K-ary trees of depth n. Suppose that $C := \sup_{\mathbf{u}\in U} \sup_{y\in[K]^n} \max_{t\in[n]} \|\mathbf{u}_t(y)\|_\infty < \infty$. Then there exists a strategy $\hat{z}_t$ that guarantees*

$$\sum_{t=1}^n \ell(\hat{z}_t, y_t) - \inf_{\mathbf{u}\in U} \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t) \le 2 \cdot \max_{\mathbf{y},\mathbf{y}'} \mathbb{E}_\epsilon \sup_{\mathbf{u}\in U} \left[\sum_{t=1}^n \epsilon_t \ell(\mathbf{u}_t(\mathbf{y}_{1:t-1}(\epsilon)), \mathbf{y}'_t(\epsilon)) \right] + 2,$$

where $\mathbf{y}$ and $\mathbf{y}'$ are $[K]$-valued binary trees of depth n and $\epsilon = (\epsilon_1, \dots, \epsilon_n)$ are Rademacher random variables.

Furthermore, the predictions $(\hat{z}_t)_{t\le n}$ satisfy $\|\hat{z}_t\|_\infty \le \log(Kn)$.

Proof [Proof of Lemma 34] Define $\mathcal{Z} := \left\{ z \in \mathbb{R}^K \mid \|z\|_\infty \le C \right\}$. The minimax optimal regret amongst deterministic strategies taking values in $\mathcal{Z}$ is given by

$$\mathcal{V}_n(U) := \left\langle\!\!\left\langle \inf_{\hat{z}_t\in\mathbb{R}^K} \max_{y_t\in[K]} \right\rangle\!\!\right\rangle_{t=1}^n \left[\sum_{t=1}^n \ell(\hat{z}_t, y_t) - \inf_{\mathbf{u}\in U} \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t) \right].$$

Once again, this proof closely follows the sequential Rademacher complexity bound from Rakhlin et al. (2010). We only sketch the first few steps for this proof as they are identical to the first few steps of the proof of Theorem 11, which is admissible due to compactness of $\mathcal{Z}$. Using the minimax swap as in that theorem, we can move to an upper bound of

$$\le \left\langle\!\!\left\langle \sup_{p_t\in\Delta_K} \mathbb{E}_{y_t\sim p_t} \right\rangle\!\!\right\rangle_{t=1}^n \left[\sum_{t=1}^n \inf_{\hat{z}_t\in\mathcal{Z}} \mathbb{E}_{y_t\sim p_t} \left[\ell(\hat{z}_t, y_t)\right] - \inf_{\mathbf{u}\in U} \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t) \right]$$
$$= \left\langle\!\!\left\langle \sup_{p_t\in\Delta_K} \mathbb{E}_{y_t\sim p_t} \right\rangle\!\!\right\rangle_{t=1}^n \sup_{\mathbf{u}\in U} \left[\sum_{t=1}^n \inf_{\hat{z}_t\in\mathcal{Z}} \mathbb{E}_{y_t\sim p_t} \left[\ell(\hat{z}_t, y_t)\right] - \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t) \right].$$

Now we choose $\hat{z}_t$ to match the value of $\mathbf{u}_t(y) = \mathbf{u}_t(y_{1:t-1})$, which is possible by definition of $\mathcal{Z}$:

$$\le \left\langle\!\!\left\langle \sup_{p_t\in\Delta_K} \mathbb{E}_{y_t\sim p_t} \right\rangle\!\!\right\rangle_{t=1}^n \sup_{\mathbf{u}\in U} \left[\sum_{t=1}^n \mathbb{E}_{y_t\sim p_t} \left[\ell(\mathbf{u}_t(y), y_t)\right] - \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t) \right].$$

Using Jensen's inequality, we pull the conditional expectaitons in the first term outside the supremum over $\mathbf{u}$ by introducing a tangent sequence $(y'_t)_{t\le n}$, where y'_t follows the distribution p_t conditioned on $y_{1:t-1}$.

$$\le \left\langle\!\!\left\langle \sup_{p_t\in\Delta_K} \mathbb{E}_{y_t, y'_t\sim p_t} \right\rangle\!\!\right\rangle_{t=1}^n \sup_{\mathbf{u}\in U} \left[\sum_{t=1}^n \ell(\mathbf{u}_t(y), y'_t) - \sum_{t=1}^n \ell(\mathbf{u}_t(y), y_t) \right].$$

Since y_t and y'_t are conditionally i.i.d., we can introduce a Rademacher random variable ϵ_t at each timestep t as follows:

$$= \left\langle\!\!\left\langle \sup_{p_t\in\Delta_K} \mathbb{E}_{y_t, y'_t\sim p_t} \mathbb{E}_{\epsilon_t} \right\rangle\!\!\right\rangle_{t=1}^n \sup_{\mathbf{u}\in U} \left[\sum_{t=1}^n \epsilon_t \big(\ell(\mathbf{u}_t(y), y'_t) - \ell(\mathbf{u}_t(y), y_t) \big) \right].$$

To decouple the arguments to the losses from the arugments to the tree $\mathbf{u}$, we move to a pessimistic upper bound:

$$\leq \left\langle\!\!\left\langle \sup_{p_t \in \Delta_K} \mathbb{E}_{y_t \sim p_t} \max_{y'_t, y''_t \in [K]} \mathbb{E}_{\epsilon_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \sup_{\mathbf{u} \in U} \left[\sum_{t=1}^{n} \epsilon_t \big(\ell(\mathbf{u}_t(y), y'_t) - \ell(\mathbf{u}_t(y), y''_t) \big) \right]$$
$$= \left\langle\!\!\left\langle \max_{y_t, y'_t, y''_t \in [K]} \mathbb{E}_{\epsilon_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \sup_{\mathbf{u} \in U} \left[\sum_{t=1}^{n} \epsilon_t \big(\ell(\mathbf{u}_t(y), y'_t) - \ell(\mathbf{u}_t(y), y''_t) \big) \right].$$

We now complete the symmetrization as follows:

$$\leq \left\langle\!\!\left\langle \max_{y_t, y'_t, y''_t \in [K]} \mathbb{E}_{\epsilon_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \sup_{\mathbf{u} \in U} \left[\sum_{t=1}^{n} \epsilon_t \ell(\mathbf{u}_t(y), y'_t) \right] + \left\langle\!\!\left\langle \max_{y_t, y'_t, y''_t \in [K]} \mathbb{E}_{\epsilon_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \sup_{\mathbf{u} \in U} \left[\sum_{t=1}^{n} \epsilon_t \ell(\mathbf{u}_t(y), y''_t) \right]$$
$$= 2 \cdot \left\langle\!\!\left\langle \max_{y_t, y'_t \in [K]} \mathbb{E}_{\epsilon_t} \right\rangle\!\!\right\rangle_{t=1}^{n} \sup_{\mathbf{u} \in U} \left[\sum_{t=1}^{n} \epsilon_t \ell(\mathbf{u}_t(y), y'_t) \right]$$
$$= 2 \cdot \max_{\mathbf{y}, \mathbf{y}'} \mathbb{E}_{\epsilon} \sup_{\mathbf{u} \in U} \left[\sum_{t=1}^{n} \epsilon_t \ell(\mathbf{u}_t(\mathbf{y}_{1:t-1}(\epsilon)), \mathbf{y}'_t(\epsilon)) \right].$$

In the last line $\mathbf{y}$ and $\mathbf{y}'$ are taken to be $[K]$-valued binary trees of depth n, so that $\mathbf{y}_t(\epsilon) = \mathbf{y}_t(\epsilon_1, \ldots \epsilon_{t-1})$ and likewise for $\mathbf{y}'$.

Finally, to guarantee the boundedness of predictions claimed in the lemma statement, we apply Lemma 16 to the minimax optimal strategy, for which we just showed regret is bounded by the sequential Rademacher complexity. ■

The last auxiliary lemma in this section is a slight variant of the Dudley entropy integral bound for sequential Rademacher complexity. This lemma can be extracted from the proof of Theorem 4 in Rakhlin et al. (2015b). We do not repeat the proof here.

Lemma 35 *Let W be a collection of $\mathbb{R}$-valued binary trees. Define $\mathcal{N}_p(\alpha, W)$ to be the size of the smallest class of trees V such that*

$$\forall \mathbf{w} \in W, \epsilon \in \{\pm 1\}^n, \ \exists \mathbf{v} \in V \text{ s.t. } \left(\frac{1}{n} \sum_{t=1}^{n} (\mathbf{w}_t(\epsilon) - \mathbf{v}_t(\epsilon))^p \right)^{1/p} \leq \alpha. \tag{23}$$

Let $\mathrm{rad}_p(W) := \min\{\alpha \mid \mathcal{N}_p(\alpha, W) = 1\}$. *Then it holds that*

$$\mathbb{E}_{\epsilon} \sup_{\mathbf{w} \in W} \sum_{t=1}^{n} \epsilon_t \mathbf{w}_t(\epsilon) \leq \inf_{\alpha > 0} \left\{ 4\alpha n + 12 \int_{\alpha}^{\mathrm{rad}_2(W)} \sqrt{n \log \mathcal{N}_2(\delta, W)} d\delta \right\}. \tag{24}$$

A.6. Details from Section 7

We first define a suitable notion of sequential cover for the log loss setting:

Definition 36 *For a fixed $\mathcal{X}$-valued binary tree $\mathbf{x}$, define $\mathcal{N}_\infty(\alpha, \mathcal{F}, \mathbf{x})$ to be the size of the smallest set of $[0,1]$-valued binary trees V such that*

$$\forall f \in \mathcal{F}, \ \epsilon \in \{\pm 1\}^n, \ \exists \mathbf{v} \in V \text{ s.t. } \max_{t \in [n]} |f(\mathbf{x}_t(\epsilon)) - \mathbf{v}_t(\epsilon)| \leq \alpha.$$

Further, define $\mathcal{N}_\infty(\alpha, \mathcal{F}) = \sup_{\mathbf{x}} \mathcal{N}_\infty(\alpha, \mathcal{F}, \mathbf{x})$.

We also require a generalization of Definition 36 for general tree classes.

Definition 37 *For a class of $[0,1]$-valued binary trees U, define $\mathcal{N}_\infty(\alpha, U)$ to be the size of the smallest set of $[0,1]$-valued binary trees V such that*

$$\forall \mathbf{u} \in U,\ \epsilon \in \{\pm 1\}^n,\ \exists \mathbf{v} \in V \text{ s.t. } \max_{t\in[n]} |\mathbf{u}_t(\epsilon) - \mathbf{v}_t(\epsilon)| \le \alpha.$$

We now turn to the proof of Theorem 13. It follows the same structure as the proof in Appendix A.5 with a few technical differences related the slightly different notion of cover used and the non-Lipschitzness of the log loss. We first give one more definition.

Definition 38 *For any $\delta \in (0, 1/2]$, we define the truncation to the range $[\delta, 1-\delta]$ via $\mathrm{clip}_\delta(p) = \max\{\delta, \min\{1-\delta, p\}\}$.*

The following proposition is a simple consequence of the fact that clip_δ is 1-Lipschitz.

Proposition 39 *For any class of trees U and any $\delta \in (0, 1/2]$, $\mathcal{N}_\infty(\alpha, \mathrm{clip}_\delta \circ U) \le \mathcal{N}_\infty(\alpha, U)$.*

Proof [Proof of Theorem 13] The proof is very similar to that of Theorem 11. When it would otherwise be repetitive we will only sketch details and instead refer back to the proof of that theorem.

To begin, fix $\delta \in (0, 1/2]$. We will work with the clipped class $\mathcal{F}^\delta = \mathrm{clip}_\delta \circ \mathcal{F}$ just as in Cesa-Bianchi and Lugosi (2006). It was shown there that

$$\mathcal{V}_n^{\log}(\mathcal{F}) \le \mathcal{V}_n^{\log}(\mathcal{F}^\delta) + \delta n.$$

With this restriction, we proceed exactly as in the proof of Theorem 11. First, restrict the learner's predictions to $[\delta, 1-\delta]$ to guarantee boundedness of the loss:

$$\begin{aligned}\mathcal{V}_n^{\log}(\mathcal{F}^\delta) &= \left\langle\!\!\left\langle \sup_{x_t\in\mathcal{X}} \inf_{\hat{p}_t\in[0,1]} \max_{y_t\in\{0,1\}} \right\rangle\!\!\right\rangle_{t=1}^n \left[\sum_{t=1}^n \ell_{\log}(\hat{p}_t, y_t) - \inf_{f\in\mathcal{F}^\delta} \sum_{t=1}^n \ell_{\log}(f(x_t), y_t)\right] \\ &\le \left\langle\!\!\left\langle \sup_{x_t\in\mathcal{X}} \inf_{\hat{p}_t\in[\delta,1-\delta]} \max_{y_t\in\{0,1\}} \right\rangle\!\!\right\rangle_{t=1}^n \left[\sum_{t=1}^n \ell_{\log}(\hat{p}_t, y_t) - \inf_{f\in\mathcal{F}^\delta} \sum_{t=1}^n \ell_{\log}(f(x_t), y_t)\right].\end{aligned}$$

Since compactness holds, we can apply the minimax theorem and manipulate terms in the same fashion as in the proof of Theorem 11 to arrive at the following expression

$$= \sup_{\mathbf{x},\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}} \left[\sum_{t=1}^n \inf_{\hat{p}_t\in[\delta,1-\delta]} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)} \left[\ell_{\log}(\hat{p}_t, y_t)\right] - \inf_{f\in\mathcal{F}} \sum_{t=1}^n \ell_{\log}(f(\mathbf{x}_t(y)), y_t)\right]. \tag{25}$$

In the final line above $\mathbf{x}$ and $\mathbf{p}$ are $\mathcal{X}$- and $\Delta_{\{0,1\}}$-valued binary trees (indexed by $\{0,1\}$) of depth n. The notation "$y \sim \mathbf{p}$" refers to the process in which we first draw $y_1 \sim \mathbf{p}_1$, then draw $y_t \sim \mathbf{p}_t(y_1, \ldots, y_{t-1})$ for subsequent timesteps t.

Let V^γ be a minimal γ-sequential cover of $\mathcal{F}$ on the tree $\mathbf{x}$ with respect to the L_∞ norm in the sense of Definition 36.

Following the proof of Theorem 11, we define a collection of $[\delta, 1-\delta]$-valued binary trees for each element of V^γ, with the tree $\mathbf{x}$ fixed. For each tree $\mathbf{v} \in V^\gamma$, each $f \in \mathcal{F}^\delta$, we define a class of trees $\mathcal{F}_{\mathbf{v}}^\delta$ as follows:

- Initially $\mathcal{F}_{\mathbf{v}}^{\delta} = \varnothing$.
- For each $f \in \mathcal{F}^{\delta}$ and $y \in \{0,1\}^n$ with $\max_{t\in[n]}|f(\mathbf{x}_t(y)) - \mathbf{v}_t(y)| \le \gamma$:
  - Define a $[\delta, 1-\delta]$-valued binary tree $\mathbf{u}_{f,y}$ via: For each $y' \in \{\pm 1\}^n$,

    $$(\mathbf{u}_{f,y})_t(y'_{1:t-1}) := f(\mathbf{x}_t(y'))\mathbb{1}\{y'_1 = y_1, \dots, y'_{t-1} = y_{t-1}\} + \mathbf{v}_t(y')\mathbb{1}\{\neg(y'_1 = y_1, \dots, y'_{t-1} = y_{t-1})\}.$$

    (So that $\mathbf{u}_{f,y}$ is equal to $f \circ \mathbf{x}$ on the path y, and equal to $\mathbf{v}$ everywhere else.)
  - Add $\mathbf{u}_{f,y}$ to $\mathcal{F}_{\mathbf{v}}^{\delta}$.

Just like the construction in Theorem 11, $\mathcal{F}_{\mathbf{v}}^{\delta}$ has two properties: Its L_∞ covering number is bounded in terms of the L_∞ covering number of the class $\mathcal{F}^{\delta} \circ \mathbf{x}$, and its L_∞ radius is bounded by γ. These properties are stated in Lemma 40.

On any path $y \in \{0,1\}^n$ and for each $f \in \mathcal{F}$, there exist $\mathbf{v} \in V^{\gamma}$ and $\mathbf{u} \in \mathcal{F}_{\mathbf{v}}^{\delta}$ such that $f(\mathbf{x}_t(y)) = \mathbf{u}_t(y)$. This is because a $\mathbf{v}$ that is γ-close to f on the path y through $\mathbf{x}$ is guaranteed by the cover property of V^{γ}, and so we can take $\mathbf{u}_{f,y}$ in $\mathcal{F}_{\mathbf{v}}$ as the desired $\mathbf{u}$. This implies that

$$\inf_{f\in\mathcal{F}} \sum_{t=1}^{n} \ell_{\log}(f(\mathbf{x}_t(y)), y_t) \ge \min_{\mathbf{v}\in V^{\gamma}} \inf_{\mathbf{u}\in\mathcal{F}_{\mathbf{v}}^{\delta}} \sum_{t=1}^{n} \ell_{\log}(\mathbf{u}_t(y), y_t).$$

Returning to the minimax rate, all the properties of the tree families we have established so far imply

$$\begin{aligned}
&\mathcal{V}_n^{\log}(\mathcal{F}^{\delta}) \\
&\le \sup_{\mathbf{x}} \inf_{V^{\gamma}} \inf_{\{\mathcal{F}_{\mathbf{v}}^{\delta}\}_{\mathbf{v}\in V^{\gamma}}} \sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\left[\sum_{t=1}^{n} \inf_{\hat{p}_t\in[\delta,1-\delta]} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)}\left[\ell_{\log}(\hat{p}_t, y_t)\right] - \min_{\mathbf{v}\in V^{\gamma}} \inf_{\mathbf{u}\in\mathcal{F}_{\mathbf{v}}^{\delta}} \sum_{t=1}^{n} \ell_{\log}(\mathbf{u}_t(y), y_t)\right].
\end{aligned}$$

As in the proof of Theorem 11, we introduce a family of trees representing the minimax optimal strategy competing with each tree class $\mathcal{F}_{\mathbf{v}}^{\delta}$. For each $\mathbf{v} \in V^{\gamma}$, we introduce a $[\delta, 1-\delta]$-valued binary tree $\hat{\mathbf{p}}^{\mathbf{v}}$, with $\hat{\mathbf{p}}_t^{\mathbf{v}} : \{0,1\}^{t-1} \to [\delta, 1-\delta]$.

$$\begin{aligned}
&= \sup_{\mathbf{x}} \inf_{V^{\gamma}} \inf_{\{\mathcal{F}_{\mathbf{v}}^{\delta}\}_{\mathbf{v}\in V^{\gamma}}} \inf_{\{\hat{\mathbf{p}}^{\mathbf{v}}\}_{\mathbf{v}\in V^{\gamma}}} \sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\Bigg[\sum_{t=1}^{n} \inf_{\hat{p}_t\in[\delta,1-\delta]} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)}\left[\ell_{\log}(\hat{p}_t, y_t)\right] \\
&\qquad\qquad - \min_{\mathbf{v}\in V^{\gamma}}\left\{\sum_{t=1}^{n} \ell_{\log}(\hat{\mathbf{p}}_t^{\mathbf{v}}(y), y_t) - \sum_{t=1}^{n} \ell_{\log}(\hat{\mathbf{p}}_t^{\mathbf{v}}(y), y_t) + \inf_{\mathbf{u}\in\mathcal{F}_{\mathbf{v}}^{\delta}} \sum_{t=1}^{n} \ell_{\log}(\mathbf{u}_t(y), y_t)\right\}\Bigg]. \\
&\le \sup_{\mathbf{x}} \inf_{V^{\gamma}} \inf_{\{\mathcal{F}_{\mathbf{v}}^{\delta}\}_{\mathbf{v}\in V^{\gamma}}} \inf_{\{\hat{\mathbf{p}}^{\mathbf{v}}\}_{\mathbf{v}\in V^{\gamma}}} \Bigg\{ \underbrace{\sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\left[\sum_{t=1}^{n} \inf_{\hat{p}_t\in[\delta,1-\delta]} \mathbb{E}_{y_t\sim\mathbf{p}_t(y)}\left[\ell_{\log}(\hat{p}_t, y_t)\right] - \min_{\mathbf{v}\in V^{\gamma}} \sum_{t=1}^{n} \ell_{\log}(\hat{\mathbf{p}}_t^{\mathbf{v}}(y), y_t)\right]}_{(\star)} \\
&\qquad + \underbrace{\sup_{\mathbf{p}} \mathbb{E}_{y\sim\mathbf{p}}\left[\max_{\mathbf{v}\in V^{\gamma}}\left\{\sum_{t=1}^{n} \ell_{\log}(\hat{\mathbf{p}}_t^{\mathbf{v}}(y), y_t) - \inf_{\mathbf{u}\in\mathcal{F}_{\mathbf{v}}^{\delta}} \sum_{t=1}^{n} \ell_{\log}(\mathbf{u}_t(y), y_t)\right\}\right]}_{(\star\star)} \Bigg\}.
\end{aligned} \tag{26}$$

We now bound the terms $(\star)$ and $(\star\star)$ individually as follows:

Term $(\star)$ We select $(\hat{p}_t)_{t\le n}$ using the Aggregating Algorithm as configured in Lemma 41, with W as the finite collection of sequences $\{\hat{\mathbf{p}}^{\mathbf{v}}\}_{\mathbf{v}\in V^\gamma}$. This is possible because $\hat{\mathbf{p}}^{\mathbf{v}}_t$ only depends on $y_{1:t-1}$.

$$\sum_{t=1}^{n}\ell_{\log}(\hat{p}_t,y_t)-\min_{\mathbf{v}\in V^\gamma}\sum_{t=1}^{n}\ell_{\log}(\hat{\mathbf{p}}^{\mathbf{v}}_t(y),y_t)\le \log|V^\gamma|+2.$$

Since the algorithm's predictions lie in $[\delta,1-\delta]$ they are a valid choice for the infimum in $(\star)$. This implies that

$$(\star)\le \sup_{\mathbf{x}}\log|V^\gamma|\le \log\mathcal{N}_\infty(\gamma,\mathcal{F}^\delta).$$

Term $(\star\star)$ First, note that we can take each tree class $\mathcal{F}^\delta_{\mathbf{v}}$ to be $[\delta,1-\delta]$-valued without loss of generality. We exhibit a deterministic strategy for each class by invoking the generic minimax regret bound Lemma 42. Since the collection is $[\delta,1-\delta]$-valued, the lemma guarantees existence of a deterministic strategy $(\hat{p}_t)_{t\le n}$ with a regret bound of

$$\begin{aligned}&\sum_{t=1}^{n}\ell_{\log}(\hat{p}_t,y_t)-\inf_{\mathbf{u}\in\mathcal{F}^\delta_{\mathbf{v}}}\sum_{t=1}^{n}\ell_{\log}(\mathbf{u}_t(y),y_t)\\&\le 2n\delta\log(1/\delta)\\&\quad+\frac{C}{\delta}\log\mathcal{N}_\infty(\gamma,\mathcal{F}^\delta_{\mathbf{v}})+\inf_{\alpha\in(0,\gamma]}\left\{\frac{4n\alpha}{\delta}+30\sqrt{\frac{2n}{\delta}}\int_\alpha^\gamma\sqrt{\log\mathcal{N}_\infty(\rho,\mathcal{F}^\delta_{\mathbf{v}})}d\rho+\frac{8}{\delta}\int_\alpha^\gamma\log\mathcal{N}_\infty(\rho,\mathcal{F}^\delta_{\mathbf{v}})d\rho\right\}.\end{aligned}$$

By Lemma 40, $\mathcal{N}_\infty(\gamma,\mathcal{F}^\delta_{\mathbf{v}})\le 1$, and so we can drop the leading covering number term in the bound:

$$\le 2n\delta\log(1/\delta)+\inf_{\alpha\in(0,\gamma]}\left\{\frac{4n\alpha}{\delta}+30\sqrt{\frac{2n}{\delta}}\int_\alpha^\gamma\sqrt{\log\mathcal{N}_\infty(\rho,\mathcal{F}^\delta_{\mathbf{v}})}d\rho+\frac{8}{\delta}\int_\alpha^\gamma\log\mathcal{N}_\infty(\rho,\mathcal{F}^\delta_{\mathbf{v}})d\rho\right\}.$$

Lemma 40 also implies that we can upper bound the covering number in terms of that of $\mathcal{F}^\delta$:

$$\le 2n\delta\log(1/\delta)+\inf_{\alpha\in(0,\gamma]}\left\{\frac{4n\alpha}{\delta}+30\sqrt{\frac{2n}{\delta}}\int_\alpha^\gamma\sqrt{\log(n\mathcal{N}_\infty(\rho,\mathcal{F}^\delta,\mathbf{x}))}d\rho+\frac{8}{\delta}\int_\alpha^\gamma\log(n\mathcal{N}_\infty(\rho,\mathcal{F}^\delta,\mathbf{x}))d\rho\right\}.$$

Since the regret inequality holds deterministically and uniformly for all sequences y for each $\mathbf{v}$, we have that

$$(\star\star)$$
$$\le 2n\delta\log(1/\delta)+\inf_{\alpha\in(0,\gamma]}\left\{\frac{4n\alpha}{\delta}+30\sqrt{\frac{2n}{\delta}}\int_\alpha^\gamma\sqrt{\log(n\mathcal{N}_\infty(\rho,\mathcal{F}^\delta,\mathbf{x}))}d\rho+\frac{8}{\delta}\int_\alpha^\gamma\log(n\mathcal{N}_\infty(\rho,\mathcal{F}^\delta,\mathbf{x}))d\rho\right\}.$$

Final bound We combine $(\star)$ and $(\star\star)$, take the supremum over $\mathbf{x}$, and apply Proposition 39 to conclude that $\mathcal{V}^{\log}_n(\mathcal{F})$ is bounded by

$$\begin{aligned}&3n\delta\log(1/\delta)+\log\mathcal{N}_\infty(\gamma,\mathcal{F})\\&\quad+\inf_{\alpha\in(0,\gamma]}\left\{\frac{4n\alpha}{\delta}+30\sqrt{\frac{2n}{\delta}}\int_\alpha^\gamma\sqrt{\log(n\mathcal{N}_\infty(\rho,\mathcal{F}))}d\rho+\frac{8}{\delta}\int_\alpha^\gamma\log(n\mathcal{N}_\infty(\rho,\mathcal{F}))d\rho\right\}.\end{aligned}$$

The theorem statement uses that we are free to choose any value for δ and γ. ■

The remaining lemmas in this section mirror those used in the proof of Theorem 11, with the most substantive difference being that we required a more refined chaining bound for general classes under the log loss from Rakhlin and Sridharan (2015a). We omit their proofs.

Lemma 40 *Let $\mathcal{F}^{\delta}_{\mathbf{v}}$ be defined as in the proof of Theorem 11 for trees $\mathbf{v}$ and $\mathbf{x}$ and scale γ. Then it holds that*

1. $\mathcal{N}_\infty(\gamma, \mathcal{F}^{\delta}_{\mathbf{v}}) \le 1$.
2. $\mathcal{N}_\infty(\alpha, \mathcal{F}^{\delta}_{\mathbf{v}}) \le n \cdot \mathcal{N}_\infty(\alpha, \mathcal{F}^{\delta}, \mathbf{x})$ *for all* $\alpha > 0$.

Note that the covering number (Definition 37) was defined for trees indexed by $\{\pm 1\}^n$, but trees in $\mathcal{F}^{\delta}_{\mathbf{v}}$ are indexed by $\{0,1\}^n$. We overload the covering number in the natural way in the lemma above and subsequent lemmas.

Lemma 41 (Cesa-Bianchi and Lugosi (2006)) *Let W be any class of $[\delta, 1-\delta]$-valued binary trees of depth n. Then Vovk's Aggregating Algorithm configured with W as a benchmark class of experts generates predictions $(\hat{p}_t)_{t\le n}$ that enjoy regret*

$$\sum_{t=1}^{n} \ell_{\log}(\hat{p}_t, y_t) - \min_{\mathbf{w}\in W} \sum_{t=1}^{n} \ell_{\log}(\mathbf{w}_t(y), y_t) \le \log|\mathcal{W}|. \tag{27}$$

Furthermore, the predictions $(\hat{p}_t)_{t\le n}$ lie in $[\delta, 1-\delta]$.

Lemma 42 (Extracted from Rakhlin and Sridharan (2015a)) *Let W be any class of $[\delta, 1-\delta]$-valued binary trees of depth n. Then there exists a deterministic prediction strategy $(\hat{p}_t)_{t\le n}$ that enjoys regret*

$$\begin{aligned}
&\sum_{t=1}^{n} \ell_{\log}(\hat{p}_t, y_t) - \inf_{\mathbf{w}\in W} \sum_{t=1}^{n} \ell_{\log}(\mathbf{w}_t(y), y_t) \\
&\le 2n\delta \log(1/\delta) + \frac{C}{\delta} \log \mathcal{N}_\infty(\gamma, W) \\
&\quad + \inf_{\alpha\in(0,\gamma]} \left\{ \frac{4n\alpha}{\delta} + 30\sqrt{\frac{2n}{\delta}} \int_\alpha^\gamma \sqrt{\log \mathcal{N}_\infty(\rho, W)}\, d\rho + \frac{8}{\delta} \int_\alpha^\gamma \log \mathcal{N}_\infty(\rho, W)\, d\rho \right\},
\end{aligned}$$

for all $\gamma > 0$ and for some absolute constant $C > 0$. The predictions $(\hat{p}_t)_{t\le n}$ lie in $[\delta, 1-\delta]$.

Appendix B. Efficient Implementation

In this section we discuss an efficient (i.e. polynomial time in the parameters of the problem) randomized implementation of Algorithm 1. The main idea is to exploit the log-concavity of the density of P_t in the algorithm and to use well-established Markov chain Monte Carlo samplers for such densities to collect enough matrices W sampled from the distribution to approximate the prediction $\hat{z}_t$ sufficiently well to ensure the increase in regret is small.

Fix a round t. Recall that the density on $\mathcal{W}$ we wish to sample from in round t of the algorithm is

$$P_t(W) \propto \exp\left(-\tfrac{1}{L}\textstyle\sum_{s=1}^{t-1} \ell(Wx_s, y_s)\right).$$

For notational convenience, define the function $F_t : \mathcal{W} \to \mathbb{R}$ as $F_t(W) := \exp\left(-\tfrac{1}{L}\sum_{s=1}^{t-1} \ell(Wx_s, y_s)\right)$. It is easy to check that F_t is log-concave.

Assumption 1 *We have access to a sampler that makes poly$(1/\varepsilon, n, d, B, R)$ queries to F_t and produces a sample W with distribution $\tilde{P}_t$ such that $d_{TV}(\tilde{P}_t, P_t) \le \varepsilon$.*

Such samplers are well-known in the literature: for example, the hit-and-run sampler (Lovász and Vempala, 2006), the projected Langevin Monte Carlo sampler (Bubeck et al., 2015), and the Dikin walk sampler (Narayanan and Rakhlin, 2017). It is easy to derive appropriate bounds on all the relevant parameters of F_t that are involved in the analysis of these samplers so that the samplers run in polynomial time. While this gives a theoretically efficient implementation, the running time bounds are too loose to be practical (for example, see the calculations below for projected Langevin Monte Carlo sampler). We have not attempted to improve these running time bounds; that is a direction for future work.

Example 3 (Bubeck et al. (2015)) *Let W have density $P \propto e^{-f}$ for some β-smooth, S-Lipschitz convex function f over a convex body $\mathcal{W}$ contained in a euclidian ball of radius D in dimension d. Projected Langevin Monte Carlo produces a sample from $\widetilde{P}$ with $d_{TV}(\widetilde{P}, P) \le \varepsilon$ after $O\Big(\frac{D^6 \max\{d, DS, D\beta\}^{12}}{\varepsilon^{12}}\Big)$ evaluations. For our setting, when $\|x_t\|_2 \le R$ and $\|y_t\|_1 \le L$, the loss $w \mapsto \ell(\langle w, x_t\rangle, y_t)$ is $O(RL)$-Lipschitz and smooth. We therefore have $S, \beta \le RLn$ and $D = B$, which yields the following bound on the number of queries to F_t:*

$$O\Big(\frac{B^6 \max\{dK, BRLn\}^{12}}{\varepsilon^{12}}\Big).$$

Given access to a sampler, we can now prove Proposition 4. In the following, we use the phrase "with high probability" to indicate that the statement referred to holds with probability at least $1 - \delta$ for any $\delta > 0$. We also use the notation $\tilde{O}(\cdot)$ and $\tilde{\Omega}(\cdot)$ to suppress logarithmic dependence on $1/\delta$, d, K, and n.

Proof [Proof of Proposition 4.] The idea is very straightforward: for some parameters $m \in \mathbb{N}$ and $\varepsilon > 0$ to be specified later, in each round t, simply use the sampler with error tolerance $\frac{\varepsilon}{2}$ repeatedly m times to collect samples $W^{(i)}$ for $i \in [m]$ and then approximate the prediction by $\tilde{z}_t = \boldsymbol{\sigma}^+\big(\mathrm{smooth}_\mu\big(\mathbb{E}_{i\sim[m]}\big[\boldsymbol{\sigma}(W^{(i)}x_t)\big]\big)\big)$. Here, "$i \sim [m]$" denotes sampling i uniformly from $[m]$, and $m = \mathrm{poly}(n, d, B, R, 1/\delta)$ will be chosen to be large enough to ensure that this approximation incurs only $1/n$ additional loss in each round, with high probability, and thus at most $O(1)$ additional loss over all n rounds.

It remains to provide appropriate bounds on m. In the following, we will fix the round t and drop the subscript t from $P_t, \tilde{P}_t, x_t, y_t$, etc. for notational clarity.

Define the distributions $p = \mathrm{smooth}_\mu(\mathbb{E}_{W\sim P}[\boldsymbol{\sigma}(Wx)])$, $\tilde{p} = \mathrm{smooth}_\mu\big(\mathbb{E}_{W\sim \tilde{P}}[\boldsymbol{\sigma}(Wx)]\big)$ and $\tilde{\tilde{p}} = \mathrm{smooth}_\mu\big(\mathbb{E}_{i\sim[m]}\big[\boldsymbol{\sigma}(W^{(i)}x)\big]\big)$. Then standard Chernoff-Hoeffding bounds and a union bound over all $k \in [K]$ imply that if $m = \tilde{\Omega}\big(1/\varepsilon^2\big)$, then with high probability, we have $\|\tilde{p} - \tilde{\tilde{p}}\|_\infty \le \frac{\varepsilon}{2}$. Furthermore, the sampler ensures $d_{\mathrm{TV}}(\tilde{P}, P) \le \frac{\varepsilon}{2}$, which implies that $\|p - \tilde{p}\|_\infty \le \frac{\varepsilon}{2}$ since each coordinate of p and $\tilde{p}$ are i n $[0, 1$. Thus, by the triangle inequality, we have $\|p - \tilde{\tilde{p}}\|_\infty \le \varepsilon$.

We now bound the excess loss for using $\tilde{\tilde{p}}$ instead of p in the algorithm, using the fact the weighted multiclass logistic loss can be equivalently viewed as a weighted multiclass log loss after passing the logits through the softmax function $\boldsymbol{\sigma}$. Thus, the additional loss equals

$$\sum_{k\in[K]} y_k \log\Big(\tfrac{p_k}{\tilde{\tilde{p}}_k}\Big) \le \sum_{k\in[K]} y_k \log\Big(\tfrac{\tilde{\tilde{p}}_k + \varepsilon}{\tilde{\tilde{p}}_k}\Big) \le \sum_{k\in[K]} y_k \log\Big(1 + \tfrac{\varepsilon K}{\mu}\Big) \le \tfrac{\varepsilon K L}{\mu}.$$

The first inequality above follows from the bound $\|p - \tilde{\tilde{p}}\|_\infty \le \varepsilon$, and the second from the fact that $\tilde{\tilde{p}}_k \ge \frac{\mu}{K}$ for all $k \in [K]$, and the third from $\log(1+a) \le a$ for all $a \in \mathbb{R}_+$ and $\|y\|_1 \le L$. Thus, setting $\varepsilon = \frac{\mu}{KLn}$ ensures that the additional loss is at most $1/n$ with high probability, as required. ■