Sequential Prediction of Individual Sequences Under General Loss Functions

David Haussler, Jyrki Kivinen, and Manfred K. Warmuth

***Abstract*— We consider adaptive sequential prediction of arbitrary binary sequences when the performance is evaluated using a general loss function. The goal is to predict on each individual sequence nearly as well as the best prediction strategy in a given comparison class of (possibly adaptive) prediction strategies, called *experts*. By using a general loss function, we generalize previous work on universal prediction, forecasting, and data compression. However, here we restrict ourselves to the case when the comparison class is finite. For a given sequence, we define the regret as the total loss on the entire sequence suffered by the adaptive sequential predictor, minus the total loss suffered by the predictor in the comparison class that performs best on that particular sequence. We show that for a large class of loss functions, the minimax regret is either $\Theta(\log N)$ or $\Omega(\sqrt{\ell \log N})$, depending on the loss function, where N is the number of predictors in the comparison class and ℓ is the length of the sequence to be predicted. The former case was shown previously by Vovk; we give a simplified analysis with an explicit closed form for the constant in the minimax regret formula, and give a probabilistic argument that shows this constant is the best possible. Some weak regularity conditions are imposed on the loss function in obtaining these results. We also extend our analysis to the case of predicting arbitrary sequences that take real values in the interval $[0, 1]$.**

***Index Terms*— On-line learning, universal prediction, worst case loss bounds, worst case regret.**

I. Introduction

ASSUME that your data consists of a sequence $y_1, \cdots, y_\ell$ of binary *outcomes* that is revealed to you one outcome at a time. At each time step or *trial* t, after seeing the outcomes $y_1, \cdots, y_{t-1}$, you must *predict* the next outcome y_t by producing a number $\hat{y}_t \in [0, 1]$. When the actual next outcome y_t is revealed, you then suffer a loss $L(y_t, \hat{y}_t)$, where L is a fixed *loss function*. One example of this scenario is in producing sequential probability assignments for individual binary sequences [29], [32], [33], [39]. Here $\hat{y}_t$ is an estimate of the probability that $y_t = 1$, given the previous outcomes $y_1, \cdots, y_{t-1}$. In this case, the loss function is the *logarithmic loss function*, $L = L_{\log}$, which is defined by letting

$$L_{\log}(y_t, \hat{y}_t) = -\ln \hat{y}_t, \qquad \text{if } y_t = 1$$

and

$$L_{\log}(y_t, \hat{y}_t) = -\ln(1 - \hat{y}_t), \qquad \text{if } y_t = 0.$$

That is, the loss is the negative logarithm of the probability that was predicted for y_t. This is closely related to the number of bits required to encode y_t given $y_1, \cdots, y_{t-1}$ in an optimal sequential adaptive coding method based on the sequential probability assignments. Other possible loss functions are the *square loss* $L_{\text{sq}}(y_t, \hat{y}_t) = (y_t - \hat{y}_t)^2$, often used in the literature on sequential forecasting [9], [13], [16], [36] and the *absolute loss* $L_{\text{abs}}(y_t, \hat{y}_t) = |y_t - \hat{y}_t|$, often used in pattern recognition and computational learning theory [6]–[8], [25], where the term *on-line* is used to describe a sequential procedure of this type. The absolute loss can be interpreted as the probability of error in predicting y_t when you use a randomized strategy of predicting 1 with probability $\hat{y}_t$ and 0 with probability $1 - \hat{y}_t$.

In this on-line prediction setup, you play the role of an adaptive algorithm or *learning algorithm*, which produces the predictions for each trial. Nature provides the sequence of outcomes through some unknown process. In *universal* or *worst case* prediction over individual outcome sequences, nothing at all is assumed about the process used by nature to produce the sequence of outcomes. The performance of the learning algorithm is judged in the worst case over all possible outcome sequences of length ℓ, for each ℓ. However, in order to make the problem nontrivial, one only considers the performance of the learning algorithm relative to the performance of the best prediction strategy in a specified class $\mathcal{E}$ of on-line prediction strategies which we call the *comparison class* or the *set of experts*. Specifically, for every possible sequence $y_1, \cdots, y_\ell$, the total loss incurred by the learning algorithm for all trials t, $1 \le t \le \ell$, is measured, and from this we subtract the infimum over all experts $\mathcal{E}_i \in \mathcal{E}$ of the total loss incurred by expert $\mathcal{E}_i$ on this sequence. This difference represents a *regret* suffered by the learning algorithm, measured as the total loss it suffers minus the total loss it would have suffered if it had used the advice of the expert in $\mathcal{E}$ that performed best on this particular sequence of outcomes. In particular, for $0 \le p \le 1$ let A_p be the constant predictor that always predicts with $\hat{y}_t = p$, and let $\mathcal{E} = \{A_p | 0 \le p \le 1\}$ be

Manuscript received November 14, 1994; revised January 16, 1997. The work of D. Haussler was supported by NSF under Grant IRI-9123692 and by DOE under Grant DE-FG03-95ER62112. The work of J. Kivinen was supported by Emil Aaltonen Foundation, the Academy of Finland, and by ESPRIT Project NeuroCOLT. The work of M. K. Warmuth was supported by NSF under Grants IRI-9123692 and CCR 9700201. Preliminary results presented in this paper have appeared in *Computational Learning Theory: EuroCOLT' 93*, Oxford, U.K.: Clarendon, 1994, pp. 109–120; in *Computational Learning Theory: Second European Conference, EuroCOLT' 95*, Berlin, Germany: Springer, 1995, pp. 69–83; and in Technical Report UCSC-CRL-94-36, University of California, Santa Cruz, 1994.

D. Haussler and M. K. Warmuth are with Computer Science, University of California at Santa Cruz, Santa Cruz, CA 95064 USA (e-mail: haussler@cse.ucsc.edu; manfred@cse.ucsc.edu).

J. Kivinen is with the Department of Computer Science, University of Helsinki, P.O. Box 26 (Teollisuuskatu 23), FIN-00014 University of Helsinki, Finland (e-mail: jkivinen@cs.helsinki.fi).

Publisher Item Identifier S 0018-9448(98)04786-5.

the class of all constant sequential probability assignments (memoryless encoding schemes). Then the minimax regret for the logarithmic loss is (essentially) the redundancy of the adaptive code for $y_1, \cdots, y_\ell$, i.e., the total number of bits needed to encode this sequence of outcomes adaptively, minus the number of bits that would have been required if $\hat{y}_t = \sum_{t'} y_{t'}/\ell$ for all t, which is the best constant prediction for this particular outcome sequence. Extensions of this include the case where $\mathcal{E}$ consists of all Markov predictors of a given order, or all finite-state predictors of a given number of states [15], [26], [38].

Of course, since the learning algorithm must make its predictions on-line, it cannot know ahead of time which expert in $\mathcal{E}$ will perform best on the sequence of outcomes produced by Nature. Remarkably, however, the work on universal prediction, forecasting, and data compression has shown that in many cases the learning algorithm can achieve surprisingly small regret, i.e., it can make predictions almost as well as if it knew ahead of time which expert's advice to take [2], [4]–[7], [11], [14], [15], [17], [19], [20], [27], [28], [30], [31], [33], [35], [37]. In particular, it has been shown that the Lempel–Ziv algorithm universally achieves quite a small regret when compared to the best finite-state predictor [15]. A more general analysis of universal prediction is given in [31] for a comparison class that is a smooth parametric family, and in more general cases in [28]. Closely related work has also been done in the area of mathematical finance, where one seeks a stock portfolio rebalancing strategy that performs almost as well as the best strategy in a given comparison class on any market [12], [22]. More general decision-theoretic scenarios are considered in [1], [10], and [37]. Also related is the work on on-line competitive algorithms in computer science (see e.g., [34]).

In this paper we focus on the case in which the comparison class $\mathcal{E}$ is finite. Our goal is to develop the most general results possible for this finite case. Whereas most previous papers (with the exception of [10], [35], and [37]) have each focused on a single loss function, which has often been different in different disciplines, we give a unified treatment of a very general class of loss functions, including the usual ones. Whereas in many previous papers, very specific forms for the comparison class are studied, e.g., kth-order Markov models or finite-state predictors of a certain size, here we obtain bounds that hold for an arbitrary (finite) set of sequential prediction mechanisms. They are even allowed to be dependent on each other, in the sense that the prediction of one expert in $\mathcal{E}$ at trial t can depend not only on the previous outcomes $y_1, \cdots, y_{t-1}$ but on the predictions of the other experts in $\mathcal{E}$ up to and including time t as well. We use the term "experts" instead of "statistical models" for the predictors in the comparison class $\mathcal{E}$ to distinguish this setting from a setting in which the comparison class consists of simpler statistical models.

The standard universal prediction setting can be viewed as a game between Nature and the learner. Nature selects the sequence of outcomes, and the learner makes predictions on-line and suffers some regret, as defined above, after all outcomes have been seen. One is interested in the minimax value of this game, that is, the minimum over all possible prediction strategies of the maximum regret over all possible outcome sequences. We call this the *minimax regret*. The exact minimax regret depends strongly on the comparison class, and even for simple comparison classes it does not usually have a nice closed-form formula for each loss function L and length of play ℓ [6], [10]. So here we focus instead on obtaining good upper and lower bounds for the minimax regret.

Our upper bounds on the minimax regret are quite general, in that they depend only on the number N of experts in the comparison class. So they in fact hold for a more challenging game in which Nature chooses the (predictions of the) experts in the comparison class in an adversarial manner, in addition to choosing the outcomes. A game of this type was defined and analyzed by Cesa-Bianchi *et al.* [6] for the absolute loss; here we extend this analysis to more general loss functions. The upper bounds we obtain on the minimax regret of this more challenging game provide upper bounds on the minimax regret of any standard universal prediction game. Our lower bounds show that the leading constants in these general upper bounds cannot be improved. However, the adversary construction in these lower bounds does not require the full power available to Nature in the more challenging game. In particular, the lower bounds show that for large N and ℓ there is always a set of nonadaptive experts whose (predetermined) predictions for times 1 to ℓ can be known in advance to the on-line prediction algorithm, and still this algorithm must suffer minimax regret close to that given in the general upper bound, with respect to the worst case outcome sequence.

Vovk [35] introduced an on-line prediction algorithm that is applicable to all loss functions when the outcomes are binary. This algorithm can be used to obtain good general upper bounds on the minimax regret. For a large class of loss functions, Vovk proved that for this algorithm, the minimax regret was bounded by $c_L \ln N$, independent of the number ℓ of trials, where c_L is a positive constant determined by the loss function L and N is the number of experts in the comparison class. For instance, for the square loss Vovk's algorithm achieves this bound with $c_L = 1/2$ [35], and for logarithmic loss with $c_L = 1$, when the natural logarithm is used to define the loss function L [14], [35]. On the other hand, for the absolute loss L_{abs}, Cesa-Bianchi *et al.* [6] have shown that the best general bounds on the minimax regret that can be obtained are $\Theta(\sqrt{\ell \log N})$, and that the best possible constant in this bound approaches $1/\sqrt{2}$ for a large number N of experts, when the natural logarithm is used. Here there is a strong dependence on the number ℓ of trials. Slightly weaker results for the absolute loss were obtained earlier by Littlestone and Warmuth [25].

It is instructive to compare these general results obtained for a finite comparison class of size N and logarithmic loss to the universal prediction results of Rissanen and others for smooth parametric families of models, which form infinite comparison classes. Indeed, Rissanen has shown that for the purposes of universal prediction under logarithmic loss, essentially without loss of performance, under suitable smoothness conditions one can replace a continuous k-dimensional comparison class of models with a finite approximation to this class in which the parameters are given to precision roughly $1/\sqrt{\ell}$, where

ℓ is the number of data points (trials) [31] (see (27) and the preceding equation). This gives a finite comparison class of size $N = O(\ell^{k/2})$. Hence in this case, Vovk's bound of $c_L \ln N$ on the minimax regret, where $c_L = 1$ for logarithmic loss, gives a bound of roughly $(k/2) \ln \ell$, as obtained by Rissanen, which is optimal apart from the additive constant, supplied only by the deeper argument of Rissanen.

In this paper we give a simplified analysis of Vovk's general algorithm which yields an explicit definition of the constant c_L in the formula above for a wide class of loss functions L, including most usual loss functions, with the exception of absolute loss. We also provide a probabilistic argument that shows this is the best constant that can be obtained. Then we define another class of loss functions that includes the absolute loss, and prove that there is no general upper bound on the minimax regret for any loss function in this class that is smaller than $\Theta(\sqrt{\ell \log N})$. Thus for loss functions in this class, the minimax regret will in general depend strongly on the number ℓ of trials.

We make some weak regularity assumptions on the loss function. It is possible to construct loss functions that are in neither of our classes, and for which we thus do not know any bounds. It is an open problem to provide nontrivial bounds on the minimax risk that would apply to *all* loss functions. Nevertheless, the classes we define cover such a broad range of functions that we must conclude that the two asymptotic forms of the minimax regret that are obtained, $\Theta(\log N)$ and $\Theta(\sqrt{\ell \log N})$, are in some sense generic for this problem.

Section II gives a formal description of our framework of analysis. Our bounds are given in Section III-A together with a discussion of the regularity conditions assumed for the loss function. Section III-B restates Vovk's algorithm and upper-bound proof, simplified for our purposes. The lower-bound proof, given in Section III-C, is based on generating the outcome sequence by a simple randomized adversary, using simple randomly defined experts, and showing that already the expected regret of the algorithm approaches the worst case upper bound. Thus in a sense we see that in our particular setting, the average case is almost as difficult as the worst case. The proof technique with a randomized adversary was used previously by Cesa-Bianchi *et al.* [6] in the special case of the absolute loss.

Finally, in Section IV-A we show that for certain loss functions, such as the square and logarithmic loss, Vovk's algorithm achieves the same worst case regret even if the outcomes are allowed to be arbitrary real numbers in the interval $[0, 1]$. In this case, the logarithmic loss is generalized to the *relative entropy loss,* defined by

$$L_{\text{ent}}(y_t, \hat{y}_t) = y_t \ln (y_t/\hat{y}_t) + (1 - y_t) \ln ((1 - y_t)/(1 - \hat{y}_t)).$$

Combined with our lower bounds, this shows that the minimax regret in this case is the same as for binary outcomes. For the absolute loss, the worst case regret bounds proven for binary outcomes [6], [35] can be achieved with continuous-valued outcomes by using a slightly more complicated algorithm, as we show in Section IV-B.

II. On-Line Prediction and Loss Bounds

We consider the predictive performance of an on-line learning algorithm A on a sequence of outcomes $y_1, \cdots, y_\ell$, where $y_t \in [0, 1]$ for each $1 \le t \le \ell$. The algorithm's performance is compared to that of the best expert in a given set $\mathcal{E} = \{\mathcal{E}_1, \cdots, \mathcal{E}_N\}$ of experts, each of which is an arbitrary on-line prediction strategy. The prediction of the expert $\mathcal{E}_i$ for the outcome y_t is denoted by $x_{t,i}$ and is a real number in the interval $[0, 1]$. This prediction can depend on the previous (and current) predictions of the other experts as well as the previous outcomes. The vector of predictions by all the experts for trial t, called the *prediction vector,* is defined by $\boldsymbol{x}_t = (x_{t,1}, \cdots, x_{t,N})$. When the algorithm makes its prediction $\hat{y}_t \in [0, 1]$ for the outcome y_t, we assume that it has access to all previous outcomes, as well as all previous predictions of the experts, including the predictions $\boldsymbol{x}_t$ for the current trial t. This is always true if the algorithm has access to the previous outcomes and can simulate the predictive mechanisms of the experts. However, our upper-bound results also hold in more general cases in which the algorithm cannot simulate the experts; see [6] for further discussion. Also, most algorithms considered in this paper make their predictions $\hat{y}_t$ independently of the length ℓ of the whole trial sequence, but in some situations we also consider how the algorithms can be fine-tuned if ℓ is known in advance.

We define the (N-expert) *trial sequence* as

$$S = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell))$$

and each pair $(\boldsymbol{x}_t, y_t)$ is a *trial*. We consider separately the cases of *binary outcomes,* with the outcomes y_t either 0 or 1, and *continuous-valued outcomes,* with y_t any real number from the interval $[0, 1]$. The performance of the learner at trial t is measured by $L(y_t, \hat{y}_t)$, where L is a *loss function* with the range $[0, \infty)$, or sometimes $[0, \infty]$. For binary outcomes $y_t \in \{0, 1\}$ it suffices to consider the functions L_0 and L_1 defined by $L_0(\hat{y}) = L(0, \hat{y})$ and $L_1(\hat{y}) = L(1, \hat{y})$.

Example 2.1: The *relative entropy loss* L_{ent} is defined by

$$L_{\text{ent}}(y, \hat{y}) = y \ln (y/\hat{y}) + (1 - y) \ln ((1 - y)/(1 - \hat{y})).$$

By the usual convention $0 \ln 0 = 0$, this gives $L_0(\hat{y}) = -\ln (1 - \hat{y})$ and $L_1(\hat{y}) = -\ln \hat{y}$ for $L = L_{\text{ent}}$. In the binary case $y \in \{0, 1\}$, the relative entropy loss is better known as the *logarithmic loss*.

The *square loss* L_{sq} is defined by

$$L_{\text{sq}}(y, \hat{y}) = (y - \hat{y})^2.$$

Hence, for $L = L_{\text{sq}}$, we have $L_0(\hat{y}) = \hat{y}^2$ and $L_1(\hat{y}) = (1 - \hat{y})^2$.

The Hellinger loss L_{H} is given by

$$L_{\text{H}}(y, \hat{y}) = \tfrac{1}{2}((\sqrt{1 - y} - \sqrt{1 - \hat{y}})^2 + (\sqrt{y} - \sqrt{\hat{y}})^2).$$

Hence, for $L = L_H$ we have $L_0(\hat{y}) = 1 - \sqrt{1 - \hat{y}}$ and $L_1(\hat{y}) = 1 - \sqrt{\hat{y}}$.

The absolute loss L_{abs} is given by $L_{\text{abs}}(y, \hat{y}) = |y - \hat{y}|$, and we have $L_0(\hat{y}) = \hat{y}$ and $L_1(\hat{y}) = 1 - \hat{y}$ for $L = L_{\text{abs}}$. □

It is worth noting some properties of the loss functions of Example 2.1, since these will be important later. In each case, the function L_0 is increasing and L_1 decreasing in $[0, 1]$, so the loss $L(y, \hat{y})$ increases as the prediction $\hat{y}$ moves away from the outcome y. The functions L_0 and L_1 are differentiable, and by the previous remark, $L_0'(z) \geq 0$ and $L_1'(z) \leq 0$ for all z. Except for the absolute loss, the second derivatives $L_0''(z)$ and $L_1''(z)$ are positive for all z, which means that errors become progressively more expensive as the difference between the prediction and outcome increases.

Consider now a loss function L and an on-line prediction algorithm A. Let $S = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell))$ be an N-expert trial sequence, and let the prediction of the algorithm A at trial t of the sequence S be $\hat{y}_t$. We then define

$$\text{Loss}_L(A, S) = \sum_{t=1}^{\ell} L(y_t, \hat{y}_t)$$

as the loss of the algorithm and

$$\text{Loss}_L(\mathcal{E}_i, S) = \sum_{t=1}^{\ell} L(y_t, x_{t,i})$$

as the loss of the ith expert on the sequence S. We define

$$V_{L,A}(S) = \text{Loss}_L(A, S) - \min_{1 \leq i \leq N} \text{Loss}_L(\mathcal{E}_i, S)$$

to be the *regret* or *additional loss* of the algorithm, i.e., the amount by which the loss of the algorithm exceeds the loss of the best expert. We let

$$V_{L,A}(N, \ell) = \sup\{V_{L,A}(((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell))) | \boldsymbol{x}_t \in [0, 1]^N, y_t \in \{0, 1\}\}$$

be the worst case regret for A, when the outcomes in an N-expert trial of length ℓ are restricted to be binary. Here we are formalizing the more challenging game in which Nature is allowed to select both the outcomes and the predictions of the N experts in an adversarial fashion. Finally, we let $V_L(N, \ell) = \inf_A V_{L,A}(N, \ell)$ be the smallest regret obtainable by an on-line prediction algorithm A. This is the minimax value for this more challenging game. The goal of this paper is to study $V_L(N, \ell)$ for general loss functions L, and to generalize the results for continuous-valued outcomes $y_t \in [0, 1]$.

Some general mathematical notation we will need is as follows. We use $\text{E}[X]$ and $\text{Var}[X]$ to denote the expected value and variance of a random variable X. If we want to emphasize the underlying probability measure P, we write $\text{E}_{x \in P}[X(x)]$ and $\text{Var}_{x \in P}[X(x)]$. The probability of an event φ according to a probability measure P is denoted by $\Pr_{x \in P}[\varphi(x)]$.

We use $\boldsymbol{N}_+$ to denote the set $\{1, 2, 3, \cdots\}$ of the positive integers and $\boldsymbol{R}$ to denote the set of real numbers.

III. Binary Outcomes

We now consider the case of binary outcomes $y_t \in \{0, 1\}$. Our results include both upper and lower bounds for the minimax regret. In Section IV we show how at least for the usual loss functions, the upper bounds can be generalized to allow for continuous-valued outcomes $y_t \in [0, 1]$.

The main results are summarized in Section III-A. Section III-B gives the algorithm that obtains the upper bounds, and the proof that it does so. Both the algorithm and analysis are originally by Vovk [35]; here we are able to simplify them by considering only continuous loss functions. Section III-C contains the main lower bound proofs. Finally, in Section III-D we consider some other possibilities for lower bound proofs.

A. Main Results

The proofs of our upper and lower bounds require that the loss function satisfies certain constraints. We first state the main result with all the necessary restrictions and then discuss the meaning of these restrictions. First, given loss functions L_0 and L_1 that are twice differentiable, we define a function S by

$$S(z) = L_0'(z)L_1''(z) - L_1'(z)L_0''(z) \tag{3.1}$$

and a function R by

$$R(z) = \frac{L_0'(z)L_1'(z)^2 - L_1'(z)L_0'(z)^2}{S(z)}. \tag{3.2}$$

We then define a constant c_L by

$$c_L = \sup_{0<z<1} R(z). \tag{3.3}$$

If $S(z) = 0$ for some $0 < z < 1$, we write $c_L = \infty$. Our main result concerns the case where c_L is finite. When c_L is finite and the loss function satisfies certain other conditions, we can prove an upper bound $V_{L,A}(N, \ell) \leq c_L \ln N$ and show that the bound is asymptotically tight.

Theorem 3.1: Let L be a loss function such that $L_0(0) = L_1(1) = 0$, L_0 and L_1 are three times differentiable in $(0, 1)$, and $L_0'(z) > 0$ and $L_1'(z) < 0$ for $0 < z < 1$. Assume that the constant c_L defined in (3.3) is finite and $S(z)$ defined in (3.1) is positive for $0 < z < 1$. Then there is an on-line prediction algorithm A for which

$$V_{L,A}(N, \ell) \leq c_L \ln N \tag{3.4}$$

holds for all $N \geq 1$ and $\ell \geq 1$. Further, we have

$$V_L(N, \ell) \geq (c_L - o(1)) \ln N \tag{3.5}$$

where $o(1)$ denotes a quantity that approaches 0 as ℓ and N approach ∞.

The algorithm A that obtains the bound (3.4), as well as the proof of the bound, are already given by Vovk [35]. The algorithm makes its predictions independently of the length ℓ of the trial sequence. We give the algorithm and a simplified proof in Section III-B. Note that the length ℓ of the sequence does not appear on the right-hand side of (3.4). The lower bound (3.5) is based on a probabilistic proof that is given in Section III-C. The lower bound holds also for algorithms that get knowledge of ℓ beforehand.

Example 3.2: Consider the loss functions of Example 2.1. For these loss functions, we clearly have $L_0(0) = L_1(1) = 0$, $L_0'(z) > 0$, and $L_1'(z) < 1$. For the logarithmic loss we have

$$S(z) = 1/(z^2(1-z)) + 1/(z(1-z)^2) > 0.$$

Further, $R(z)$ is identically 1, and therefore $c_L = 1$. For the square loss, we have $S(z) = 4 > 0$ for all z. Further, $R(z) = 2z - 2z^2$, and hence $c_L = 1/2$. For the Hellinger loss, we have

$$S(z) = (1/8)z^{-3/2}(1-z)^{-3/2}.$$

Further,

$$R(z) = z\sqrt{1-z} + (1-z)\sqrt{z}$$

and it is straightforward to show that $R(z)$ is maximized for $z = 1/2$. Hence, $c_L = 2^{-1/2}$. For the absolute loss $S(z)$ is identically 0, so $c_L = \infty$ and Theorem 3.1 is not applicable for the absolute loss. □

For all typical loss functions the conditions $L_0(0) = L_1(1) = 0$, $L_0'(z) > 0$, and $L_1'(z) < 1$ hold. Thus Theorem 3.1 can be nonapplicable because $c_L = \infty$, or because $S(z) \le 0$ for some z. Since $S(z) = 0$ implies $c_L = \infty$, these two reasons often occur together. In Section III-C we prove the following lower bounds, which show that if the denominator $S(z)$ is not always strictly positive, the regret $V_L(N, \ell)$ cannot have an upper bound that is independent of ℓ.

Theorem 3.3: Let L be a loss function such that L_0 and L_1 are three times differentiable in $(0, 1)$, and $L_0'(z) > 0$ and $L_1'(z) < 0$ for all z. Let S be as in (3.1).

1) If $S(z) = 0$ for some $0 < z < 1$, we have
$$V_L(N, \ell) = \Omega(\ell^{1/6}\sqrt{\log N}). \tag{3.6}$$
2) If $S(z) < 0$ for some $0 < z < 1$, or there are values $a < b$ such that $S(z) = 0$ for all $a \le z \le b$, we have
$$V_L(N, \ell) = \Omega(\sqrt{\ell \log N}). \tag{3.7}$$

The special case of absolute loss was considered by Cesa-Bianchi *et al.* [6]. They show that for the optimal algorithm A we have $V_{L,A}(N, \ell) = \Theta(\sqrt{\ell \ln N})$. For the absolute loss, the denominator $S(z)$ is 0 for all z. Thus our lower bound (3.7) generalizes their lower bound for more general loss functions. Unfortunately, in the case of general loss functions we know of no corresponding upper bound.

Finally, it is possible that the value c_L is infinite, but the denominator $S(z)$ is positive for all z. We can construct an example to show that such behavior is possible, although none of the usual loss functions found in the literature exhibit it. For such loss functions the results of this paper have no implications whatsoever.

Example 3.4: Define a loss function by $L_0(z) = (1-z)^{-\alpha} - 1$ and $L_1(z) = z^{-\alpha} - 1$ for some positive value α. We then have

$$R(z) = \frac{\alpha}{\alpha+1}(z^{-\alpha}(1-z) + (1-z)^{-\alpha}z).$$

Therefore, $R(z)$ approaches ∞ as z approaches 0 or 1, and c_L is infinite. Hence, our results give no upper bound for $V_L(N, \ell)$. However, the denominator $S(z)$ is given by

$$S(z) = \alpha^2(\alpha+1)(z(1-z))^{-\alpha-2}$$

and is hence strictly positive for $0 < z < 1$. Therefore, we have no lower bound, either. For this loss function it is an open problem to define any bounds for $V_L(N, \ell)$. □

Ignoring the artificially constructed special case of Example 3.4, our results for specific loss functions are divided based on the sign of the function S. For the logarithmic loss, the square loss, and the Hellinger loss, the value $S(z)$ is positive for all z, and Theorem 3.1 applies. For the absolute loss, $S(z)$ is zero everywhere, and Theorem 3.3 applies. To conclude this section, we clarify the intuitive meaning of the function S by connecting it to Bayes-optimal predictions in a simple probabilistic prediction game.

Let Q be a probability measure on $\{0, 1\}$, with $\Pr_{y\in Q}[y=1] = q$. For a prediction $z \in [0, 1]$, the *expected loss* for probability measure Q, or for *bias* q, is

$$\mathrm{E}_{y\in Q}[L(y, z)] = (1-q)L_0(z) + qL_1(z).$$

Here we define $0\cdot\infty = 0$. For example, for the logarithmic loss we have $L_0(1) = \infty$, but the expected loss for prediction 1 is defined to be 0 for bias 1. For other biases it would be infinite. A prediction z is *Bayes-optimal* for bias q if it minimizes the expected loss. Note that since we assume L_0 and L_1 to be continuous in a closed interval, the expected loss always has a minimum value at some z. This holds even if we allow infinite losses. If L_0 is increasing and L_1 decreasing, then the prediction 0 is Bayes-optimal for bias 0 and the prediction 1 for bias 1. If a value $0 < z < 1$ is a local extremum point for the expected loss, then

$$(1-q)L_0'(z) + qL_1'(z) = 0. \tag{3.8}$$

If $1 - q \ne 0$ and $L_1'(z) \ne 0$, this implies

$$\frac{q}{1-q} = -\frac{L_0'(z)}{L_1'(z)}. \tag{3.9}$$

More generally, if either $L_0'(z)$ or $L_1'(z)$ is nonzero for a given value $z \in (0, 1)$, then there is a unique value $q \in (0, 1)$ for which (3.8) holds, and hence z cannot be a Bayes-optimal prediction for more than one bias. If

$$(1-q)L_0''(z) + qL_1''(z) > 0 \tag{3.10}$$

holds in addition to (3.8), then z is a local minimum point. There may be one or more Bayes-optimal predictions for a given bias.

Lemma 3.5: Let L be a loss function such that L_0 and L_1 are three times differentiable in $(0, 1)$, and $L_0'(z) > 0$ and $L_1'(z) < 0$ for all z. Let S be as in (3.1). If $S(z) > 0$ for all z, then for all biases $0 \le q \le 1$ there is a unique Bayes-optimal prediction z. If for all biases q the Bayes-optimal prediction is unique, then $S(z) \ge 0$ for all z, and there is no interval $[a, b]$ with $a < b$ such that $S(z) = 0$ for all $z \in [a, b]$.

The proof of Lemma 3.5 is given in Section III-C. We close the section by applying Lemma 3.5 to specific loss functions.

Example 3.6: For the logarithmic, square, and Hellinger losses, as well as for the loss function of Example 3.4, we have $S(z) > 0$ for all z and hence a unique Bayes-optimal prediction z for every bias q. The actual Bayes-optimal predictions can be determined by straightforward calculations. For the logarithmic and square losses we have $z = q$. For the Hellinger loss, we have

$$z = \frac{1}{1 + \left(\frac{1-q}{q}\right)^2}.$$

For the loss function considered in Example 3.4, with $L_0(z) = (1-z)^{-\alpha} - 1$ and $L_1(z) = z^{-\alpha} - 1$ for some positive value α, we have

$$z = \frac{1}{1 + \left(\frac{1-q}{q}\right)^{1/(\alpha+1)}}.$$

For the absolute loss, $S(z)$ is identically zero, and there must hence be at least one bias q for which there are more than one Bayes-optimal predictions. Easy calculations show that $z = 0$ is the unique Bayes-optimal prediction for biases $q < 1/2$ and $z = 1$ for biases $q > 1/2$. However, for the bias $q = 1/2$ any prediction is Bayes-optimal. □

B. The Algorithm and the Upper Bound

We consider an algorithm first introduced by Vovk [35]. We give the general algorithm and its analysis, applied to our situation in which the loss function is continuous. We then work out as examples the details for several interesting loss functions.

The algorithm has two positive real-valued parameters c and η. We first introduce the algorithm in a somewhat open form, leaving the parameters c and η unspecified and defining the prediction $\hat{y}_t$ only by giving a condition it must satisfy. For the moment we also leave open the possibility that there is no prediction that satisfies the condition, in which case we say that the algorithm fails. The parameter c can vaguely be characterized as a measure for the error allowed for the algorithm. The smaller the value c, the tighter upper bound we get for the regret assuming that the algorithm does not fail. Hence, for applying the algorithm we need to find the least value c for which the algorithm is guaranteed to never fail when the *learning rate* η is chosen suitably.

It turns out that for a loss function L that satisfies the assumptions of Theorem 3.1, the suitable choice is $c = c_L$ and $\eta = 1/c$. This gives a bound $V_{L,A}(N, \ell) \leq c_L \ln N$. The main part of the proof is in showing that for any choice $c \geq c_L$ the algorithm is guaranteed not to fail for $\eta = 1/c$. We also give a more direct way of choosing a prediction $\hat{y}_t$ that satisfies the required conditions, provided that such a prediction exists. Examples show that the seemingly complicated conditions for $\hat{y}_t$ are actually quite simple for the usual loss functions.

The algorithm uses an N-dimensional weight vector $\boldsymbol{w}_t = (w_{t,1}, \cdots, w_{t,N})$ as its internal state. The weight $w_{t,i}$ is always nonnegative and summarizes the performance of the ith expert in previous trials. At the end of the tth trial we have $-\ln w_{t,i} = \eta \text{Loss}_L(\mathcal{E}_i, S_t)$, where S_t consists of the first t trials of S. Note that the weights $w_{t,i}$ are invariant under permutations of the trial sequence S_t. The predictions $\hat{y}_t$ of the algorithm are independent of the total length ℓ of the trial sequence.

Algorithm 3.7 (The Generic Algorithm): Let L be a loss function and c and η be any positive constants.

Initialization: Set the weights to some initial values $w_{1,i} > 0$.

Prediction: Let $v_{t,i} = w_{t,i}/W_t$, where $W_t = \sum_{i=1}^N w_{t,i}$. At the beginning of trial t, compute for $y = 0$ and $y = 1$ the value

$$\Delta(y) = -c \ln \sum_{i=1}^N v_{t,i} e^{-\eta L(y, x_{t,i})}. \tag{3.11}$$

On receiving the tth input $\boldsymbol{x}_t$, predict with any value $\hat{y}_t$ that satisfies for $y = 0$ and $y = 1$ the condition

$$L(y, \hat{y}_t) \leq \Delta(y). \tag{3.12}$$

If no such value $\hat{y}_t$ exists, the algorithm fails.

Update: After receiving the tth outcome y_t, let

$$w_{t+1,i} = w_{t,i} e^{-\eta L(y_t, x_{t,i})}. \tag{3.13}$$

To understand the algorithm, note that by (3.11) and (3.13) we can write $\Delta(y_t) = U_{t+1} - U_t$, where $U_t = -c \ln W_t$. Hence, we can consider $-c \ln W_t$ as a potential function, and the condition $L(y_t, \hat{y}_t) \leq \Delta(y_t)$ means that at each trial, the increase of the potential must be at least as large as the loss of the algorithm.

In the case of the logarithmic loss, the key quantities in the Generic Algorithm have a natural statistical interpretation. In particular, it turns out that it is optimal to set $\eta = 1$, and thus $e^{-\eta L(y_t, x_{t,i})} = {x_{t,i}}^{y_t}(1 - x_{t,i})^{1-y_t}$. This latter quantity can be interpreted as the likelihood of y_t under a probability model used by the ith expert. Hence the update (3.13) can be interpreted as a Bayesian update of posterior probabilities $v_{t,i}$ over the set of experts [6]. The additivity of the logarithmic loss, and its associated statistical interpretation and chain rule, makes the analysis of this special loss more convenient, as pointed out in, e.g., [21]. In that paper, bounds for the logarithm loss are obtained first, and then these are used, along with certain inequalities, to derive bounds for other losses. Here we obtain better results by using Vovk's generalization of the likelihood, $e^{-\eta L(y_t, x_{t,i})}$, to directly obtain an analogous chain rule for a general loss.

The basic idea of proving the upper bound for the loss of the Generic Algorithm is based on relating the total potential increase $U_{\ell+1} - U_1$ to the total loss of the best expert. The following upper bound was already given by Vovk [35].

Theorem 3.8: Let L be any loss function. Let

$$S = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell))$$

be an N-expert trial sequence in which the outcomes $y_t \in \{0, 1\}$ are binary. Assume that during this trial sequence, the

Generic Algorithm 3.7 with parameters c and η does not fail but produces at each trial t a prediction $\hat{y}_t$. Then for all i the total loss satisfies

$$\begin{aligned}\mathrm{Loss}_L(A, S) &\le -c \ln \frac{W_{\ell+1}}{W_1} \\ &\le -c \ln \frac{w_{1,i}}{W_1} + c\eta \,\mathrm{Loss}_L(\mathcal{E}_i, S). \quad (3.14)\end{aligned}$$

Proof: The condition (3.12) for $y = y_t$ together with (3.11) and (3.13) implies

$$L(y_t, \hat{y}_t) \le -c \ln \frac{W_{t+1}}{W_t}$$

and hence

$$\sum_{t=1}^{\ell} L(y_t, \hat{y}_t) \le -c \ln \frac{W_{\ell+1}}{W_1} \le -c \ln \frac{w_{\ell+1,i}}{W_1}$$

for all i. Finally, by (3.13) we get

$$\frac{w_{\ell+1,i}}{W_1} = \frac{w_{1,i}}{W_1} \prod_{t=1}^{\ell} e^{-\eta L(y_t, x_{t,i})}$$

and the theorem follows. □

For given values c and η, we say that the loss function L is (c, η)-*realizable* if the condition (3.12) for $y = 0$ and $y = 1$ can always be satisfied by a suitable choice of $\hat{y}_t$. To prove the upper bound of Theorem 3.1, it now suffices to show that a loss function L that satisfies the assumptions of Theorem 3.1 is $(c, 1/c)$-realizable for $c = c_L$. The result then follows from Theorem 3.8 by setting $w_{1,i} = 1$ for all i. The rest of this section gives our formulation of Vovk's [35] proof for these results.

We first develop an equivalent version of condition (3.12). Write $\Delta_0 = \Delta(0)$ and $\Delta_1 = \Delta(1)$, so the condition (3.12) for $y \in \{0, 1\}$ can be expressed as $L_0(\hat{y}_t) \le \Delta_0$ and $L_1(\hat{y}_t) \le \Delta_1$. To obtain explicit bounds for $\hat{y}_t$ from these conditions, we need to have some notion of an inverse for L_0 and L_1. Assume that L_0 is continuous and strictly increasing and L_1 is continuous and strictly decreasing in $[0, 1]$, which is implied by the assumptions of Theorem 3.1. Then L_0 has a continuous strictly increasing inverse L_0^{-1} that is defined in $[L_0(0), L_0(1)]$, and L_1 has a continuous strictly decreasing inverse L_1^{-1} that is defined in $[L_1(1), L_1(0)]$.

Consider first the case with $\Delta_0 \in [L_0(0), L_0(1)]$ and $\Delta_1 \in [L_1(1), L_1(0)]$. Then the values $L_0^{-1}(\Delta_0)$ and $L_1^{-1}(\Delta_1)$ are defined, and (3.12) for $y \in \{0, 1\}$ can be equivalently written as

$$L_1^{-1}(\Delta_1) \le \hat{y}_t \le L_0^{-1}(\Delta_0). \quad (3.15)$$

A prediction $\hat{y}_t$ that satisfies (3.15) can be found if and only if

$$L_1^{-1}(\Delta_1) \le L_0^{-1}(\Delta_0). \quad (3.16)$$

If (3.16) holds, the prediction $\hat{y}_t$ can be chosen to be an arbitrary value between the bounds $L_1^{-1}(\Delta_1)$ and $L_0^{-1}(\Delta_0)$. For instance, their mean $(L_1^{-1}(\Delta_1) + L_0^{-1}(\Delta_0))/2$ is a valid choice for $\hat{y}_t$.

Consider now the possibility that the value Δ_0 or Δ_1 is outside of the range of L_0 or L_1, respectively. If, for instance, Δ_0 is larger than $L_0(1)$, then the condition $L_0(\hat{y}_t) \le \Delta_0$ in (3.12) holds for all $\hat{y}_t$. Thus the equivalence between (3.12) and (3.15) will be maintained for all nonnegative Δ_0 if the inverse L_0^{-1} is extended in such a way that the condition $\hat{y}_t \le L_0^{-1}(\Delta_0)$ holds for all $\hat{y}_t \in [0, 1]$ when $\Delta_0 > L_0(1)$. Hence, we say that L_0^{-1} is a *generalized inverse* of L_0 if $L_0^{-1}(L_0(\hat{y})) = \hat{y}$ for all $\hat{y} \in [0, 1]$ and $L_0^{-1}(\Delta_0) \ge 1$ whenever $\Delta_0 \ge L_0(1)$. Similarly, L_1^{-1} is a generalized inverse of L_1 if $L_1^{-1}(L_1(\hat{y})) = \hat{y}$ for all $\hat{y} \in [0, 1]$ and $L_1^{-1}(\Delta_1) \le 0$ whenever $\Delta_1 \ge L_1(0)$.

For instance, if L is the square loss L_{sq}, we have the generalized inverses $L_0^{-1}(z) = \sqrt{z}$ and $L_1^{-1}(z) = 1 - \sqrt{z}$ for $z \ge 0$, so (3.16) becomes

$$\sqrt{\Delta_0} + \sqrt{\Delta_1} \ge 1.$$

For the relative entropy loss L_{ent} we have $L_0^{-1}(z) = 1 - e^{-z}$ and $L_1^{-1}(z) = e^{-z}$, so we get

$$e^{-\Delta_0} + e^{-\Delta_1} \le 1.$$

For the absolute loss L_{abs} we have $L_0^{-1}(z) = z$ and $L_1^{-1}(z) = 1 - z$, so we need to have

$$\Delta_0 + \Delta_1 \ge 1.$$

Our definitions of generalized inverses let us show the equivalence between (3.15) and (3.12) for all values of Δ_0 and Δ_1.

Lemma 3.9: Assume that L is a loss function such that $L_0(0) = L_1(1) = 0$, L_0 is continuous and strictly increasing in $[0, 1]$, and L_1 is continuous and strictly decreasing in $[0, 1]$. For any generalized inverses L_0^{-1} and L_1^{-1}, the condition (3.15) is equivalent to (3.12) for $y \in \{0, 1\}$.

Proof: If $\Delta_0 \notin [0, L_0(1)]$, then both $L_0(\hat{y}_t) \le \Delta_0$ and $\hat{y}_t \le L_0^{-1}(\Delta_0)$ hold for all $\hat{y}_t \in [0, 1]$. If $\Delta_1 \notin [0, L_1(0)]$, then both $L_1(\hat{y}_t) \le \Delta_1$ and $L_1^{-1}(\Delta_1) \le \hat{y}_t$ hold for all $\hat{y}_t \in [0, 1]$. Hence, we may assume that Δ_0 is in the range of L_0 and Δ_1 is in the range of L_1. In this case (3.12) and (3.15) are equivalent because L_0 is strictly increasing and L_1 strictly decreasing. □

We are now ready to show that if in Algorithm 3.7 we use a value c such that $c \ge c_L$, where c_L is as defined in (3.3), and set $\eta = 1/c$, then the algorithm never fails.

Lemma 3.10: Let L be any loss function such that L_0 and L_1 are three times continuously differentiable, $L_0(0) = L_1(1) = 0$, and $L_0'(z) > 0$ as well as $L_1'(z) < 0$ hold for $0 < z < 1$. Assume that the value c_L defined in (3.3) is finite, and $S(z)$ defined in (3.1) is positive for all z. Then for all $\boldsymbol{w}_t$ and $\boldsymbol{x}_t$ such that $0 \le x_{t,i} \le 1$ and $w_{t,i} \ge 0$ for $1 \le i \le N$, condition (3.16) holds whenever $c \ge c_L$ and $\eta = 1/c$.

Proof: For $0 \le z \le 1$, define $p(z) = \exp(-L_0(z)/c)$ and $q(z) = \exp(-L_1(z)/c)$, and for r in the range of p define

$$f(r) = \exp(-L_1(L_0^{-1}(-c \ln r))/c). \quad (3.17)$$

Note that $f(p(z)) = q(z)$.

First, assume that $f''(p(z)) \le 0$ holds for $0 \le z \le 1$. We are later going to show that this is in fact true if $c \ge c_L$. Let

$r_i = p(x_{t,i})$ and $s_i = q(x_{t,i}) = f(r_i)$ for $i = 1, \cdots, N$. Then for $\eta = 1/c$ we have

$$\Delta_0 = -c \ln \left(\sum_{i=1}^{N} v_{t,i} r_i \right)$$

and

$$\Delta_1 = -c \ln \left(\sum_{i=1}^{N} v_{t,i} s_i \right).$$

The assumption $f''(r) \leq 0$ implies

$$\sum_{i=1}^{N} v_{t,i} s_i = \sum_{i=1}^{N} v_{t,i} f(r_i) \leq f\left(\sum_{i=1}^{N} v_{t,i} r_i \right).$$

We get

$$\begin{aligned} \Delta_1 &= -c \ln \left(\sum_{i=1}^{N} v_{t,i} s_i \right) \\ &\geq -c \ln \left(f\left(\sum_{i=1}^{N} v_{t,i} r_i \right) \right) \\ &= L_1 \left(L_0^{-1} \left(-c \ln \left(\sum_{i=1}^{N} v_{t,i} r_i \right) \right) \right) \\ &= L_1(L_0^{-1}(\Delta_0)) \end{aligned}$$

from which condition (3.16) follows since L_1^{-1} is decreasing.

We now show that our assumptions on L_0 and L_1 imply that for $c \geq c_L$ the function f has a nonpositive second derivative in the range of p. We have $f(p(z)) = q(z)$ and thus $f'(p(z)) = q'(z)/p'(z)$. Differentiating further, we obtain

$$f''(p(z))p'(z) = (q''(z)p'(z) - q'(z)p''(z))/p'(z)^2.$$

Since $p'(z) = -L_0'(z)p(z)/c < 0$, we have $f''(p(z)) \leq 0$ if and only if $q''(z)p'(z) - q'(z)p''(z) \geq 0$. By substituting

$$p'(z) = -L_0'(z)p(z)/c$$

and

$$p''(z) = (-L_0''(z)/c + (L_0'(z))^2/c^2)p(z)$$

and using similar expressions for $q'(z)$ and $q''(z)$, we see that $f''(p(z)) \leq 0$ if and only if

$$\Big(-L_0'(z)L_1'(z)^2 + L_1'(z)L_0'(z)^2 + c(L_0'(z)L_1''(z) - L_1'(z)L_0''(z)) \Big) \frac{p(z)q(z)}{c^3} \geq 0.$$

Finally, since our assumptions imply

$$L_0'(z)L_1''(z) - L_1'(z)L_0''(z) > 0$$

we conclude that $f''(p(z)) \leq 0$ holds if and only if $c \geq R(z)$. Hence, $c \geq c_L$ is a necessary and sufficient condition for having $f''(p(z)) \leq 0$ for all z. □

Note that above argument shows that the nonpositivity of $f''(r)$ is also a necessary condition. If $f''(r)$ is positive on some interval, by placing all the values $x_{t,i}$ in this interval but not making them equal we get

$$\sum_{i=1}^{N} v_{t,i} f(r_i) > f\left(\sum_{i=1}^{N} v_{t,i} r_i \right)$$

and, hence, $L_1^{-1}(\Delta_1) > L_0^{-1}(\Delta_0)$.

In particular, we see that since the Generic Algorithm 3.7 does not fail with the parameters $c = c_L$ and $\eta = 1/c_L$, we get the upper bound claimed in Theorem 3.1 by applying Theorem 3.8 with the initial weights $w_{1,i} = 1$ for all i.

Theorem 3.11: Let L be a loss function for which the constant c_L is finite. Let A be the Generic Algorithm 3.7 with the parameters $c = c_L$, $\eta = 1/c_L$, and the initial weights $w_{1,i} = 1$ for all i. Then for all N and ℓ, the regret of the algorithm satisfies

$$V_{L,A}(N, \ell) \leq c_L \ln N.$$

We are now ready to write the Generic Algorithm 3.7 in a more explicit form for particular loss functions.

Example 3.12: If L is the logarithmic loss, we have $c_L = 1$ and can therefore take $c = \eta = 1$ in the Generic Algorithm 3.7. After simple manipulations we get $\Delta_0 = -\ln(1 - p_t)$ and $\Delta_1 = -\ln p_t$, where $p_t = \sum_i v_{t,i} x_{t,i}$ is the weighted average of the experts' predictions. Hence,

$$L_0^{-1}(\Delta_0) = L_1^{-1}(\Delta_1) = p_t$$

and $\hat{y}_t = p_t$ is the only prediction for which (3.12) holds for $y \in \{0, 1\}$ with this choice of c and η. The loss bound we obtain was previously shown by De Santis *et al.* [14] and Vovk [35]. □

Example 3.13: Let L be the square loss. Vovk [35] has shown that the square loss is $(1/2, 2)$-realizable. Here the result follows from Lemma 3.10 and Example 3.2. The note after the proof of Lemma 3.10 further implies that the square loss is not $(c, 1/c)$-realizable for any $c < 1/2$. Hence, we take $c = 1/2$ and $\eta = 2$ in the Generic Algorithm 3.7 for the square loss. The condition (3.12) for $y \in \{0, 1\}$ now becomes

$$1 - \left(-\frac{\ln \sum_{i=1}^{N} v_{t,i} e^{-2(1-x_{t,i})^2}}{2} \right)^{1/2} \leq \hat{y}_t \leq \left(-\frac{\ln \sum_{i=1}^{N} v_{t,i} e^{-2x_{t,i}^2}}{2} \right)^{1/2}. \quad (3.18)$$

By numerically substituting random values for $\boldsymbol{v}_t$ and $\boldsymbol{x}_t$ we see that the seemingly natural choice $\hat{y}_t = \sum_i v_{t,i} x_{t,i}$ usually does not satisfy (3.18). More generally, there is no function f such that choosing $\hat{y}_t = f(\sum_i v_{t,i} x_{t,i})$ would guarantee (3.18) to hold. To see this, consider $N = 2$ and set first $\boldsymbol{x}_t = (0, 7/10)$ and $\boldsymbol{v}_t = (2/7, 5/7)$. Then $\sum_i v_{t,i} x_{t,i} = 1/2$, and evaluating the left-hand side of (3.18) with these values of $\boldsymbol{x}_t$ and $\boldsymbol{v}_t$ yields a bound $0.52 < f(1/2)$. On the other hand, we also have $\sum_i v_{t,i} x_{t,i} = 1/2$ when $\boldsymbol{x}_t = (3/10, 1)$ and $\boldsymbol{v}_t = (5/7, 2/7)$, and evaluating the right-hand side of (3.18) with these values gives the contradictory condition $f(1/2) < 0.48$.

Hence, the algorithm needs more information than is provided by merely the weighted average of the experts' predictions.

It can be proved that in the more restricted case that all the experts' predictions $x_{t,i}$ are in $\{0, 1\}$, we can guarantee (3.15) for the square loss with $c = 1/\eta \approx 0.41$ instead of $c = 0.5$. This gives a slightly improved bound. However, restricting the experts to predict with binary values while allowing the algorithm to predict with continuous values does not seem a natural setting. □

Example 3.14: Take L to be the absolute loss. As now $c_L = \infty$, we know that the absolute loss is not $(c, 1/c)$-realizable for any c. We therefore let $\eta > 0$ be arbitrary, and see for which values c the absolute loss is (c, η)-realizable.

By using the bound $e^{-\eta x} \le 1 - (1 - e^{-\eta})x$ that holds for all $x \in [0, 1]$, we obtain

$$\begin{aligned}
&L_0^{-1}(\Delta_0) - L_1^{-1}(\Delta_1) \\
&\quad = -c \ln \sum_{i=1}^{N} v_{t,i} e^{-\eta x_{t,i}} - \left(1 + c \ln \sum_{i=1}^{N} v_{t,i} e^{-\eta(1-x_{t,i})}\right) \\
&\quad \ge c\left(-\ln \sum_{i=1}^{N} v_{t,i}(1 - (1 - e^{-\eta})x_{t,i}) \right. \\
&\qquad \left. -\ln \sum_{i=1}^{N} v_{t,i}(1 - (1 - e^{-\eta})(1 - x_{t,i}))\right) - 1 \\
&\quad = c(-\ln(1 - p_t + p_t e^{-\eta}) - \ln(p_t + (1 - p_t)e^{-\eta})) - 1
\end{aligned}$$

where $p_t = \sum_i v_{t,i} x_{t,i}$. By Jensen's inequality, this is positive for $c \ge (2 \ln(2/(1 + e^{-\eta})))^{-1}$, and the prediction condition (3.12) for $y \in \{0, 1\}$ becomes

$$1 + \frac{\ln \sum_{i=1}^{N} v_{t,i} e^{-\eta(1-x_{t,i})}}{2 \ln \frac{2}{1+e^{-\eta}}} \le \hat{y}_t \le -\frac{\ln \sum_{i=1}^{N} v_{t,i} e^{-\eta x_{t,i}}}{2 \ln \frac{2}{1+e^{-\eta}}}. \tag{3.19}$$

Cesa-Bianchi *et al.* [6] have noted that (3.19) always holds if we choose

$$\hat{y}_t = \frac{\ln(1 - p_t + p_t e^{-\eta})}{\ln(1 - p_t + p_t e^{-\eta}) + \ln((1 - p_t)e^{-\eta} + p_t)}$$

but does not in general hold for $\hat{y}_t = p_t$. Hence, the weighted average of the experts' prediction provides sufficient information for the prediction, but cannot be used directly.

The bound obtained by applying Theorem 3.8 for the absolute loss with the choice $c = (2 \ln(2/(1 + e^{-\eta})))^{-1}$, namely

$$\mathrm{Loss}_L(A, S) \le \frac{-\ln \frac{w_{1,i}}{W_1} + \eta \,\mathrm{Loss}_L(\mathcal{E}_i, S)}{2 \ln \frac{2}{1+e^{-\eta}}} \tag{3.20}$$

was first proven by Vovk [35]. We would like to choose the learning rate η in such a way that the loss bound on the right-hand side of (3.20) is minimized. This tuning of the learning rate is discussed in detail by Cesa-Bianchi *et al.* [6], [7]. Here we just cite some of the basic results. If all the initial weights $w_{1,i}$ are 1 and η is chosen to be $\ln h(\sqrt{2(\ln N)/\ell})$ where $h(z) = 1 + 2z + z^2/\ln 2$, the Generic Algorithm 3.7 for absolute loss satisfies

$$V_{L,A}(N, \ell) \le \sqrt{\frac{\ell \ln(N+1)}{2}} + \frac{\log_2(N+1)}{2}.$$

Note that here it is necessary to know ℓ before the first trial in order to choose the learning rate η appropriately. Similar results can be obtained by basing the choice of η on an upper bound for the loss $\min_i \mathrm{Loss}_L(\mathcal{E}_i, S)$ of the best expert instead of on ℓ.

Finally, we consider the variations of the Generic Algorithm given by Cesa-Bianchi *et al.* [6] for the special case of the absolute loss. Instead of the update (3.13), we write more generally $w_{t+1,i} = \alpha_{t,i} w_{t,i}$ and

$$\Delta(y) = -c \ln \sum_{i=1}^{N} v_{t,i} \alpha_{t,i}$$

and consider choices for the factors $\alpha_{t,i}$ in addition to the choice $\alpha_{t,i} = \exp(-\eta|y_t - x_{t,i}|)$ of the Generic Algorithm. First, note that if $-\ln \alpha_{t,i} \le \eta|y_t - x_{t,i}|$, the proof of Theorem 3.8 can easily be generalized to yield the same loss bound. Second, note that the proof given for the inequality $L_1^{-1}(\Delta_1) \le L_0^{-1}(\Delta_0)$ is valid assuming

$$\alpha_{t,i} \le 1 - (1 - e^{-\eta})|y_t - x_{t,i}|.$$

Hence, the algorithm works and gives the same worst case loss bound for any choice

$$e^{-\eta|y_t - x_{t,i}|} \le \alpha_{t,i} \le 1 - (1 - e^{-\eta})|y_t - x_{t,i}|. \tag{3.21}$$

Interestingly enough, the weights obtained using

$$\alpha_{t,i} = 1 - (1 - e^{-\eta})|y_t - x_{t,i}|$$

have a Bayesian interpretation [6]. □

C. Lower Bounds

This subsection contains proofs of the lower bounds for $V_L(N, \ell)$ stated in Theorems 3.1 and 3.3 in Section III-A. The lower bounds hold even for algorithms that receive ℓ as input before the first trial.

The lower bound proofs are based on a probabilistic method. We consider trial sequences in which the outcomes y_t, $1 \le t \le \ell$, are independent, identically distributed random variables with some distribution Q (over $\{0, 1\}$) and the experts' predictions $x_{t,i}$, $t = 1, \cdots, \ell$, $i = 1, \cdots, N$, are independent, identically distributed random variables with some distribution P (over $[0, 1]$). We then derive for an arbitrary algorithm A a lower bound for the expected regret $\mathrm{E}_S V_{L,A}(S)$ when the trial sequence S is drawn from this distribution. As clearly $V_{L,A}(N, \ell) \ge \mathrm{E}_S V_{L,A}(S)$ holds for all A, this yields a lower bound for the minimax regret. Surprisingly, it turns out that the lower bound derived from this simple probabilistic setting is tight, i.e., it matches asymptotically the upper bounds

derived assuming an arbitrary adversarial choice of experts and outcomes.

We now outline the proof. First, consider arbitrary fixed distributions P and Q. Let $q = \Pr_{y\in Q}[y = 1]$ be the probability of drawing the outcome 1, or the bias of the distribution Q. Define

$$H(z) = (1-q)L_0(z) + qL_1(z)$$

to denote the expected loss of a prediction z with this bias. Recall from Section III-A that any $z \in [0, 1]$ for which $H(z)$ is minimized is called a Bayes-optimal prediction for the bias q. Assume that b is a Bayes-optimal prediction. Hence, the expected loss

$$\mathrm{E}_S[\mathrm{Loss}_L(A, S)] = \sum_{t=1}^{\ell} \mathrm{E}_{y_t\in Q}[L(y_t, \hat{y}_t)]$$

obtains its minimum when the algorithm A is such that $\hat{y}_t = b$ for all t. Therefore, also the expected regret $\mathrm{E}_S[V_{L,A}(S)]$ is minimized for this A, and for the purposes of bounding this expected regret from below we can without loss of generality assume $\hat{y}_t = b$.

Note that this is true regardless of the experts' predictions, so we could even allow the algorithm to know all the experts' predictions beforehand. Thus we are actually proving the stronger result that there is a fixed choice of experts' predictions such that for this choice the lower bound is always achieved for some set of outcomes, no matter what prediction algorithm is used.

Consider now the experts, choosing their predictions independently according to a distribution P. The two parameters we need for our lower bound calculation are given by

$$\tau = \mathrm{E}_{y\in Q,\, x\in P}[L(y, x)]$$

and

$$\sigma^2 = \mathrm{E}_{y\in Q}[\mathrm{Var}_{x\in P}[L(y, x)]].$$

We begin the lower bound proof by giving in Theorem 3.15 for large N and ℓ a lower bound of the form

$$V_L(N, \ell) \geq \ell H(b) - \ell\tau + \alpha\sigma\sqrt{\ell \ln N} \tag{3.22}$$

where α is positive and independent of N, ℓ, and P and Q. The first term on the right-hand side of (3.22) is simply the expected loss for the optimal prediction algorithm, and the second term is the expected loss for any fixed expert using distribution P. The final term shows how much better the best expert of N is expected to perform compared to a fixed single expert. Obviously, this final term is large if there is much variance in the experts' predictions; this variance is here measured by the parameter σ.

The bound of form (3.22) given in Theorem 3.15 holds for any P and Q, but is not likely to be useful unless P and Q are carefully chosen. Consider first choosing P when Q is already fixed. The simple case is the one in which there are two distinct Bayes-optimal predictions for bias q, say b_1 and b_2. This is covered in Lemma 3.17. Setting

$$\Pr_{x\in P}[x = b_1] = \Pr_{x\in P}[x = b_2] = 1/2$$

yields a distribution P with $\tau \leq H(z)$ for all z and $\sigma > 0$, which substituted into (3.22) gives the desired result

$$V_L(N, \ell) = \Omega(\sqrt{\ell \log N}).$$

The more interesting case, considered in Theorem 3.16, is that b is the unique Bayes-optimal prediction for bias q, i.e., b is the unique minimum point of H. In this case, we choose P such that

$$\Pr_{x\in P}[x = b - h] = \Pr_{x\in P}[x = b + h] = 1/2$$

where h is close to zero. By estimating the loss functions L_0 and L_1 by their second-order Taylor expansions around b we obtain values of τ and σ that substituted into (3.22) yield $V_L(N, \ell) \geq (R(b) - o(1)) \ln N$, where $R(b)$ is as in (3.2). Notice that the bound still has an implicit dependence on Q, as we assume that b is Bayes-optimal for bias q.

Now remains the choice of Q, or the bias q. If even for one bias there is more than one Bayes-optimal prediction, we directly get the lower bound $\Omega(\sqrt{\ell \log N})$ using this bias. Otherwise, Lemma 3.18 shows that by varying q from 0 to 1 we can also make the Bayes-optimal prediction b vary over the whole range from 0 to 1. Thus a suitable choice of q allows us to replace $R(b)$ in the bound by its supremum c_L.

As a minor technical complication, there is a third case: if for some bias q there is a unique Bayes-optimal prediction b, but $H''(b) = 0$, we get a bound that is slightly weaker than $\Omega(\sqrt{\ell \log N})$.

We now begin the actual proof. First we provide a bound that holds for arbitrary distributions P and Q.

Theorem 3.15: Let P be a probability measure on $[0, 1]$ and Q a probability measure on $\{0, 1\}$. Assume that for $y = 0$ and $y = 1$, the condition $\Pr_{x\in P}[L(y, x) > K] = 0$ holds for some constant K. Let b be a Bayes-optimal prediction for Q. Let

$$\tau = \mathrm{E}_{y\in Q,\, x\in P}[L(y, x)]$$

and

$$\sigma^2 = \mathrm{E}_{y\in Q}[\mathrm{Var}_{x\in P}[L(y, x)]].$$

Assume that for $y = 0$ and $y = 1$ the variance $\mathrm{Var}_{x\in P}[L(y, x)]$ is strictly positive. Then for all $\varepsilon > 0$ there is an ℓ_ε such that for all $\ell \geq \ell_\varepsilon$ we have

$$V_L(N, \ell) \geq \ell\,\mathrm{E}_{y\in Q}[L(y, b)] - \ell\tau + (a_N - \varepsilon)\sigma\sqrt{\ell \ln N} \tag{3.23}$$

where

$$\lim_{N\to\infty} a_N = \sqrt{2}.$$

Proof: Given $\boldsymbol{x} \in [0, 1]^{N\times\ell}$ and $\boldsymbol{y} \in \{0, 1\}^{\ell}$, we define an N-expert trial sequence of length ℓ by

$$\langle \boldsymbol{x}, \boldsymbol{y}\rangle = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell)).$$

For an on-line prediction algorithm A, consider $V_{L,A}(\langle \boldsymbol{x}, \boldsymbol{y}\rangle)$ as a random variable, with $\boldsymbol{x}$ and $\boldsymbol{y}$ drawn from the product measures $P^{N\times\ell}$ and Q^ℓ, respectively. The expected value of a random variable is clearly a lower bound for the supremum. Combining this with the linearity of expectation, we get

$$\begin{aligned} V_{L,A}(N,\ell) &\geq \mathrm{E}_{\boldsymbol{x}\in P^{N\times\ell}}\mathrm{E}_{\boldsymbol{y}\in Q^\ell} V_{L,A}(\langle \boldsymbol{x}, \boldsymbol{y}\rangle) \\ &= \sum_{j=1}^{\ell} \mathrm{E}_{y\in Q}[L(y,\hat{y}_t)] - \mathrm{E}_{\boldsymbol{x}\in P^{N\times\ell}}\mathrm{E}_{\boldsymbol{y}\in Q^\ell} \\ &\quad \cdot \left[\min_{1\leq i\leq N} \mathrm{Loss}_L(\mathcal{E}_i, \langle \boldsymbol{x}, \boldsymbol{y}\rangle)\right] \\ &\geq \ell \mathrm{E}_{y\in Q}[L(y,b)] - \mathrm{E}_{\boldsymbol{x}\in P^{N\times\ell}}\mathrm{E}_{\boldsymbol{y}\in Q^\ell} \\ &\quad \cdot \left[\min_{1\leq i\leq N} \mathrm{Loss}_L(\mathcal{E}_i, \langle \boldsymbol{x}, \boldsymbol{y}\rangle)\right]. \end{aligned}$$

Since this holds for any A, we obtain (3.23) if we can prove that

$$\begin{aligned} \mathrm{E}_{\boldsymbol{x}\in P^{N\times\ell}}\mathrm{E}_{\boldsymbol{y}\in Q^\ell}\left[\min_{1\leq i\leq N} \mathrm{Loss}_L(\mathcal{E}_i, \langle \boldsymbol{x}, \boldsymbol{y}\rangle)\right] \\ \leq \ell\tau - (a_N - \varepsilon)\sigma\sqrt{\ell \ln N}. \quad (3.24) \end{aligned}$$

Our basic method in estimating the expectation on the left-hand side of (3.24) consists of two steps. First, we apply the central limit theorem to each of the random variables $\mathrm{Loss}_L(\mathcal{E}_i, \langle \boldsymbol{x}, \boldsymbol{y}\rangle)$, $i = 1, \cdots, N$, and see that for large ℓ, they have an approximately normal distribution. Second, we apply known results that directly give the expectation of the minimum of a set of N identical independent normal random variables. Both of these steps are relatively simple in themselves. Unfortunately, the random variables $\mathrm{Loss}_L(\mathcal{E}_i, \langle \boldsymbol{x}, \boldsymbol{y}\rangle)$ that give the losses of the various experts are not independent, as the outcome sequence $\boldsymbol{y}$ affects them all. Therefore, to make the proof rigorous, we need to add some inelegant details by first considering only an arbitrary fixed outcome sequence y_t.

Let $q = \Pr_{y\in Q}[y = 1]$. Then

$$\tau = (1-q)\mathrm{E}_{x\in P}[L_0(x)] + q\mathrm{E}_{x\in P}[L_1(x)]$$

and

$$\sigma^2 = (1-q)\,\mathrm{Var}_{x\in P}[L_0(x)] + q\,\mathrm{Var}_{x\in P}[L_1(x)].$$

Given a sequence $\boldsymbol{y} \in \{0,1\}^\infty$ and $\ell \in \boldsymbol{N}_+$, define

$$\hat{q}_\ell(\boldsymbol{y}) = \frac{1}{\ell}\sum_{i=1}^{\ell} y_i.$$

We also let

$$\hat{\tau}_\ell = (1-\hat{q}_\ell(\boldsymbol{y}))\mathrm{E}_{x\in P}[L_0(x)] + \hat{q}_\ell(\boldsymbol{y})\mathrm{E}_{x\in P}[L_1(x)]$$

and

$$\hat{\sigma}_\ell(\boldsymbol{y})^2 = (1-\hat{q}_\ell(\boldsymbol{y}))\,\mathrm{Var}_{x\in P}[L_0(x)] + \hat{q}_\ell(\boldsymbol{y})\,\mathrm{Var}_{x\in P}[L_1(x)]$$

be the estimates obtained for τ and σ^2 by using $\hat{q}_\ell(\boldsymbol{y})$ instead of the true probability q.

For $\boldsymbol{x} \in [0,1]^{N\times\infty}$ and $\boldsymbol{y} \in \{0,1\}^\infty$, let $T_{ij}^{\boldsymbol{y}}(\boldsymbol{x}) = L(y_j, x_{j,i})$ be the loss of expert i at trial j, if $\boldsymbol{x}$ is the sequence of experts' predictions and $\boldsymbol{y}$ the sequence of outcomes. We consider $T_{ij}^{\boldsymbol{y}}$ as a random variable on the domain $[0,1]^{N\times\infty}$. We now define for $i = 1, \cdots, N$ and $\ell = 1, 2, \cdots$ the random variable $S_{i\ell}$ in the domain $[0,1]^{N\times\infty} \times \{0,1\}^\infty$ by

$$S_{i\ell}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j=1}^{\ell} L(y_j, x_{ij})$$

to denote the loss of expert i in the first ℓ trials. We also define for a given sequence $\boldsymbol{y} \in \{0,1\}^\infty$ the random variable $S_{i\ell}^{\boldsymbol{y}}$ by

$$S_{i\ell}^{\boldsymbol{y}}(\boldsymbol{x}) = S_{i\ell}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j=1}^{\ell} T_{ij}^{\boldsymbol{y}}(\boldsymbol{x}).$$

The underlying probability measures for these random variables are the product measures defined by P and Q, so for a fixed $\boldsymbol{y}$ the random variables $T_{ij}^{\boldsymbol{y}}$ and $T_{i'j'}^{\boldsymbol{y}}$ are independent for $(i,j) \neq (i',j')$. To study the distribution of $S_{i\ell}^{\boldsymbol{y}}$, we define a suitably normalized random variable $U_{i\ell}^{\boldsymbol{y}}$ by

$$U_{i\ell}^{\boldsymbol{y}} = \frac{S_{i\ell}^{\boldsymbol{y}} - \sum_{j=1}^{\ell}\mathrm{E}[T_{ij}^{\boldsymbol{y}}]}{\sqrt{\sum_{j=1}^{\ell}\mathrm{Var}\,[T_{ij}^{\boldsymbol{y}}]}}. \quad (3.25)$$

Then $\mathrm{E}[U_{i\ell}^{\boldsymbol{y}}] = 0$ and $\mathrm{Var}\,[U_{i\ell}^{\boldsymbol{y}}] = 1$. Further, since we have assumed that $\Pr[|T_{ij}^{\boldsymbol{y}}|] > K) = 0$, the Lindeberg form of the central limit theorem implies that each one of the N sequences $U_{i1}^{\boldsymbol{y}}, U_{i2}^{\boldsymbol{y}}, \cdots$, $i = 1, \cdots, N$, converges in distribution to a standard normal random variable.

Let $F_1, F_2, \cdots, F_N$ be N independent standard normal random variables. It is well known [18] that

$$\mathrm{E}\left[\min_{1\leq i\leq N} F_i\right] = -a_N\sqrt{\ln N}$$

where $\lim_{N\to\infty} a_N = \sqrt{2}$. Since for each i, the sequence $U_{i1}^{\boldsymbol{y}}, U_{i2}^{\boldsymbol{y}}, \cdots$ converges in distribution to F_i, we could now apply various convergence theorems to show

$$\lim_{\ell\to\infty} \mathrm{E}_{\boldsymbol{x}\in P^{N\times\ell}, \boldsymbol{y}\in Q^\ell}\left[\min_{1\leq i\leq N} U_{i\ell}^{\boldsymbol{y}}(\boldsymbol{x})\right] = -a_N\sqrt{\ln N}$$

where again $\lim_{N\to\infty} a_N = \sqrt{2}$. However, this is not quite what we need. We are really interested in the expected minimum of the variables $S_{i\ell}^{\boldsymbol{y}}$ that give the losses of the experts, not of the normalized variables $U_{i\ell}^{\boldsymbol{y}}$. As the denominator on the right-hand side of (3.25) has a complicated dependence on $\boldsymbol{y}$, the expectations of the normalized variables cannot readily be transformed back to expectations of the original ones. To get the desired result, we show that in considering expectations in the limit of large ℓ, we get the same results if we replace $\mathrm{Var}\,[T_{ij}^{\boldsymbol{y}}]$ in (3.25) by its expected value σ.

Thus define $r_\ell(\boldsymbol{y}) = \hat{\sigma}_\ell(\boldsymbol{y})/\sigma$. Then $|r_\ell(\boldsymbol{y})| \leq K/\sigma$, and by the strong law of large numbers we have $\lim_{\ell\to\infty} r_\ell(\boldsymbol{y}) = 1$ for almost all $\boldsymbol{y}$. We now apply the equation

$$\begin{aligned} \lim_{\ell\to\infty} \mathrm{E}_{\boldsymbol{y}\in Q^\infty}\left[r_\ell(\boldsymbol{y})\mathrm{E}_{\boldsymbol{x}\in P^{N\times\infty}}\left[\min_{1\leq i\leq N} U_{i\ell}^{\boldsymbol{y}}(\boldsymbol{x})\right]\right] \\ = \mathrm{E}\left[\min_{1\leq i\leq N} F_i\right] \end{aligned}$$

which we obtain directly by applying Lemma A.1, proved in the Appendix. Intuitively, we have here merely changed the order of taking the limit $\ell \to \infty$ and taking expectations, and taking a minimum of random variables. In other words, we now have

$$\lim_{\ell\to\infty} \mathrm{E}_{\boldsymbol{y}\in Q^\infty}\left[\frac{\hat{\sigma}_\ell(\boldsymbol{y})}{\sigma}\mathrm{E}_{\boldsymbol{x}\in P^{N\times\infty}}\left[\min_{1\le i\le N} U_{i\ell}^{\boldsymbol{y}}(\boldsymbol{x})\right]\right] = -a_N\sqrt{\ln N}. \quad (3.26)$$

We are now through the probability theoretic part of the proof, and the rest is straightforward.

By partitioning the summations in (3.25) into two parts according to whether $y_i = 0$ or $y_i = 1$, we can write

$$\begin{aligned} U_{i\ell}^{\boldsymbol{y}} &= \frac{S_{i\ell}^{\boldsymbol{y}} - \ell((1-\hat{q}_\ell(\boldsymbol{y}))\mathrm{E}_{x\in P}[L_0(x)] + \hat{q}_\ell(\boldsymbol{y})\mathrm{E}_{x\in P}[L_1(x)])}{\sqrt{\ell((1-\hat{q}_\ell(\boldsymbol{y}))\mathrm{Var}_{x\in P}[L_0(x)] + \hat{q}_\ell(\boldsymbol{y})\mathrm{Var}_{x\in P}[L_1(x)])}} \\ &= \frac{S_{i\ell}^{\boldsymbol{y}} - \ell\hat{\tau}_\ell(\boldsymbol{y})}{\hat{\sigma}_\ell(\boldsymbol{y})\sqrt{\ell}}. \end{aligned}$$

By substituting this into (3.26), we obtain

$$\lim_{\ell\to\infty} \frac{\mathrm{E}_{\boldsymbol{y}\in Q^\infty}[\mathrm{E}_{\boldsymbol{x}\in P^{N\times\infty}}[\min_{1\le i\le N} S_{i\ell}^{\boldsymbol{y}}(\boldsymbol{x}) - \ell\hat{\tau}_\ell(\boldsymbol{y})]]}{\sigma\sqrt{\ell}} = -a_N\sqrt{\ln N}.$$

Therefore, for all $\varepsilon > 0$ there is a value ℓ_ε such that for all $\ell \ge \ell_\varepsilon$ we have

$$\begin{aligned} &\mathrm{E}_{\boldsymbol{y}\in Q^\infty}\left[\mathrm{E}_{\boldsymbol{x}\in P^{N\times\infty}}\left[\min_{1\le i\le N} S_{i\ell}(\boldsymbol{x},\boldsymbol{y}) - \ell\hat{\tau}_\ell(\boldsymbol{y})\right]\right] \\ &\quad = \mathrm{E}_{\boldsymbol{y}\in Q^\ell}\left[\mathrm{E}_{\boldsymbol{x}\in P^{N\times\ell}}\left[\min_{1\le i\le N} \mathrm{Loss}_L(\mathcal{E}_i, \langle\boldsymbol{x},\boldsymbol{y}\rangle)\right]\right] - \ell\tau \\ &\quad \le -(a_N - \varepsilon)\sigma\sqrt{\ell\ln N}. \end{aligned}$$

This implies (3.24), as desired. □

We now see how Theorem 3.15 implies a lower bound for $V_L(N, \ell)$ when the probability measure P for the experts is chosen suitably. First we consider the case in which the Bayes-optimal prediction is unique. The Bayes-optimal prediction is the minimum point of the expected loss; the result has two cases depending on whether the second derivative of the expected loss is positive or zero at that minimum point.

Lemma 3.16: Let L be a loss function such that L_0 and L_1 are three times differentiable, and $L_0'(z) > 0$ and $L_1'(z) < 0$ hold for $0 < z < 1$. Assume that $b \in (0, 1)$ is a Bayes-optimal prediction for bias $q \in (0, 1)$.

1) If $(1-q)L_0''(b) + qL_1''(b) > 0$, then

$$V_L(N, \ell) \ge (R(b) - o(1))\ln N$$

where $R(b)$ is as in (3.2) and $o(1)$ denotes a quantity that approaches 0 as the values ℓ and N and the ratio $\ell/\ln N$ all approach ∞.

2) If $(1-q)L_0''(b) + qL_1''(b) = 0$, we have

$$V_L(N, \ell) = \Omega(\ell^{1/6}\sqrt{\log N}).$$

Proof: Let Q be the probability measure on $\{0, 1\}$ for which $\Pr_{y\in Q}[y = 1] = q$. Let A be an arbitrary on-line prediction algorithm. For any probability measure P on $[0, 1]$ and for any $\varepsilon > 0$, we have by Theorem 3.15 for sufficiently large ℓ the bound

$$V_{L,A}(N, \ell) \ge \ell(\mathrm{E}_{y\in Q}[L(y, b)] - \tau) + (a_N - \varepsilon)\sigma\sqrt{\ell\ln N} \quad (3.27)$$

where $\lim_{N\to\infty} a_N = \sqrt{2}$. For some positive parameter h, define P to give $x = b - h$ with probability $1/2$ and $x = b + h$ with probability $1/2$. We use some simple calculus to approximate the right-hand side of (3.27) as a function of h, within accuracy $o(h^3)$. We then choose the value h that maximizes the approximated value. We also see that the resulting value for h is such that the $o(h^3)$ terms can be safely ignored when ℓ and N approach infinity in the manner stated in the lemma.

We can expand

$$L_0(b \pm h) = L_0(b) \pm L_0'(b)h + \frac{L_0''(b)}{2}h^2 \pm \frac{L_0'''(b)}{6}h^3 + o(h^3)$$

where $o(h^3)$ denotes a quantity $f(h)$ such that

$$\lim_{h\to 0}(f(h)/h^3) = 0$$

and similarly for L_1. We now substitute these expansions into the various quantities in (3.27). First, note that

$$\mathrm{E}_{x\in P}[L_0(x)] = L_0(b) + h^2L_0''(b)/2 + o(h^3)$$

so

$$\begin{aligned} \mathrm{Var}_{x\in P}[L_0(x)] &= \mathrm{E}_{x\in P}[(L_0(x) - E_{x\in P}L_0(x))^2] \\ &= L_0'(b)^2h^2 + o(h^4). \end{aligned}$$

Similarly, $\mathrm{Var}_{x\in P}[L_1(x)] = L_1'(b)^2h^2 + o(h^4)$, and

$$\sigma^2 = h^2((1-q)L_0'(b)^2 + qL_1'(b)^2) + o(h^4).$$

We also have

$$\begin{aligned} \tau &= (1-q)(L_0(b) + h^2L_0''(b)/2) + q(L_1(b) \\ &\quad + h^2L_1''(b)/2) + o(h^3) \end{aligned}$$

so

$$\mathrm{E}_{y\in Q}[L(y, b)] - \tau = -\frac{h^2}{2}((1-q)L_0''(b) + qL_1''(b)) - o(h^3).$$

Hence, $V_{L,A}(N, \ell) \ge \ell(rh - sh^2) - O(\ell)o(h^3)$, where

$$r = (a_N - \varepsilon)\sqrt{\frac{\ln N}{\ell}}\sqrt{(1-q)L_0'(b)^2 + qL_1'(b)^2}$$

and

$$s = \frac{(1-q)L_0''(b) + qL_1''(b)}{2}.$$

We first consider the case $s > 0$, which gives the first part of the theorem. The main part $\ell(rh - sh^2)$ of the bound is

maximized by choosing $h = r/(2s)$. For this value of h, we get

$$\begin{aligned} V_{L,A}(N, \ell) &\geq \ell \frac{r^2}{4s} - O(\ell)o(h^3) \\ &= \frac{(a_N - \varepsilon)^2}{2} \frac{(1-q)L_0'(b)^2 + qL_1'(b)^2}{(1-q)L_0''(b) + qL_1''(b)} \\ &\quad \cdot \ln N - o((\log N)^{1/2}\ell^{-1/2}) \ln N \end{aligned}$$

since $h = \Theta((\log N)^{1/2}\ell^{-1/2})$. By applying (3.9) to eliminate q we now get the claimed result, since $\lim_{N\to\infty} a_N^2/2 = 1$.

Consider now the case $s = 0$, which gives the second part of the theorem. We now have

$$V_{L,A}(N, \ell) \geq ah\sqrt{\ell \ln N} - O(\ell)o(h^3)$$

where

$$a = (a_N - \varepsilon)\sqrt{(1-q)L_0'(b)^2 + qL_1'(b)^2} > 0.$$

By choosing $h = \ell^{-1/3}$ we get

$$V_{L,A}(N, \ell) \geq a\ell^{1/6}\sqrt{\ln N} + o(1). \qquad \square$$

If the Bayes-optimal prediction for the bias is not unique, we get an asymptotically stronger bound that grows as ℓ and N grow.

Lemma 3.17: Let L be a loss function such that L_0 is strictly increasing and L_1 strictly decreasing. Assume that for bias q there are two distinct Bayes-optimal predictions b_1 and b_2. Then for all $\varepsilon > 0$ there is an ℓ_ε such that for all $\ell \geq \ell_\varepsilon$ we have

$$V_L(N, \ell) \geq (a_N - \varepsilon)\sigma\sqrt{\ell \ln N}$$

where $\lim_{N\to\infty} a_N = \sqrt{2}$ and

$$\sigma^2 = \frac{1-q}{4}(L_0(b_1) - L_0(b_2))^2 + \frac{q}{4}(L_1(b_1) - L_1(b_2))^2. \tag{3.28}$$

Proof: Let b_1 and b_2 be two distinct Bayes-optimal predictions for some probability measure Q on $\{0, 1\}$. As L_0 and L_1 are strictly monotone, the bias of Q cannot be 0 or 1. We define a probability measure P by

$$\Pr_{x\in P}[x = b_1] = \Pr_{x\in P}[x = b_2] = 1/2$$

and apply Theorem 3.15. Then

$$\tau = \mathrm{E}_{y\in Q}[L(y, b_1)] = \mathrm{E}_{y\in Q}[L(y, b_2)].$$

Further, we get

$$\begin{aligned} \mathrm{Var}_{x\in P}[L(0, x)] &= \mathrm{E}_{x\in P}[L(0, x)^2] - \mathrm{E}_{x\in P}[L(0, x)]^2 \\ &= \frac{1}{2}L_0(b_1)^2 + \frac{1}{2}L_0(b_2)^2 \\ &\quad - \left(\frac{1}{2}L_0(b_1) + \frac{1}{2}L_0(b_2)\right)^2 \\ &= \frac{1}{4}(L_0(b_1) - L_0(b_2))^2 \end{aligned}$$

and, similarly,

$$\mathrm{Var}_{x\in P}[L(1, x)] = \frac{1}{4}(L_1(b_1) - L_1(b_2))^2.$$

Hence, σ is as given in (3.28). The result now follows from Theorem 3.15 with either $b = b_1$ or $b = b_2$. $\square$

Note that for strictly monotone L_0 and L_1, the right-hand side of (3.28) is strictly positive. For the absolute loss, we can apply Lemma 3.17 with $q = 1/2$, $b_1 = 0$, and $b_1 = 1$. This gives $\sigma = 1/2$, and hence

$$V_L(N, \ell) \geq (1 - o(1))\sqrt{(\ell \ln N)/2}$$

which is the result obtained by Cesa-Bianchi *et al.* [6].

Recall that in Lemma 3.16 we had a lower bound in terms of $R(b)$ assuming that b is the unique Bayes-optimal prediction for some bias. We now show that either every value b is the Bayes-optimal prediction for some bias, which allows us to replace $R(b)$ by its supremum c_L, or else for some bias there are multiple Bayes-optimal predictions, which gives us the stronger lower bound of $\Omega(\sqrt{\ell \log N})$ by Lemma 3.17.

Lemma 3.18: If a prediction $z \in (0, 1)$ is not Bayes-optimal for any bias $q \in [0, 1]$, then there are two predictions b_1 and b_2 with $b_1 < z < b_2$ such that for some bias q both b_1 and b_2 are Bayes-optimal.

Proof: Consider a prediction $z \in (0, 1)$ that is not Bayes-optimal for any bias. Let R_1 be the set of biases q for which there is a Bayes-optimal prediction $b < z$, and let R_2 be the set of biases q for which there is a Bayes-optimal prediction $b > z$. If we can show $R_1 \cap R_2 \neq \emptyset$, we are done. Since z is never Bayes-optimal, we have $R_1 \cup R_2 = [0, 1]$. Hence, if both R_1 and R_2 are closed, their intersection cannot be empty.

Suppose that R_1 is not closed. Let $p_1, p_2, \cdots$ be a monotone sequence of points in R_1 that converges to a point $p \notin R_1$. Let $b_n < z$ be a Bayes-optimal prediction for bias p_n, $n = 0, 1, \cdots$. The sequence $b_1, b_2, \cdots$ is also monotone and converges to some limit $b \leq z$. Let b' be a Bayes-optimal prediction for bias p. As $p \notin R_1$, we have $b' > z$. Define

$$F(q, x) = (1-q)L_0(x) + qL_1(x).$$

Since b_n is Bayes-optimal for bias p_n, we have $F(p_n, b_n) \leq F(p_n, b')$ for all n. Since F is continuous, this implies $F(p, b) \leq F(p, b')$. As b' is Bayes-optimal for bias p, so is b. Thus $p \in R_1$, a contradiction. A similar argument works if we assume R_2 to be not closed. $\square$

We are now ready to combine our lower bounds into one theorem. First, however, we wish to replace the various assumptions concerning Bayes-optimal predictions with assumptions about the function S defined in (3.1). For this purpose, we apply Lemma 3.5.

Proof of Lemma 3.5: Since we assume L_0 to be strictly increasing and L_1 to be strictly decreasing, 0 is the unique Bayes-optimal prediction for the bias 0 and 1 is the unique Bayes-optimal prediction for the bias 1.

Assume first that b_1 and b_2 are two Bayes-optimal predictions for some bias $0 < q < 1$, with $b_1 < b_2$. Thus the expected loss $f(z) = (1-q)L_0(z) + qL_1(z)$ has local minima at $z = b_1$ and $z = b_2$, and, therefore, $f(z)$ has a local maximum at some value a with $b_1 < a < b_2$. We then have $f'(a) = 0$ and $f''(a) \leq 0$. The condition $f'(a) = 0$ implies

$q/(1-q) = -L_0'(a)/L_1'(a)$, which substituted into $f''(a) \leq 0$ gives $S(a) \leq 0$.

Assume now that for every bias q there is a unique Bayes-optimal prediction. Then Lemma 3.18 implies that for all z there is a bias q for which z is Bayes-optimal, and we know that this bias q must be unique. Let $B(z)$ denote the bias for which z is the Bayes-optimal prediction. We know that B is strictly increasing. Let $f(z) = -L_0'(z)/L_1'(z)$. We then have $f(z) = g(B(z))$ where $g(q) = q/(1-q)$. Since g and B are strictly increasing, so is f, and, therefore, the derivative $f'(z)$ cannot be negative, and cannot be zero on any continuous interval. As

$$f'(z) = \frac{L_0'(z)L_1''(z) - L_1'(z)L_0''(z)}{L_1'(z)^2} = \frac{S(z)}{L_1'(z)^2},$$

the claim follows. □

The lower bounds in Theorems 3.1 and 3.3 now follow directly from the following theorem.

Theorem 3.19: Let L be a loss function such that L_0 and L_1 are three times differentiable, and $L_1'(z) > 0$ and $L_1'(z) < 0$ hold for all $0 < z < 1$. Let $S(z)$ be as in (3.1).

1) If $S(z) > 0$ for $0 < z < 1$, then
$$V_L(N, \ell) \geq (c_L - o(1)) \ln N$$
where c_L is as in (3.3).
2) If $S(z) = 0$ for some $0 < z < 1$, then
$$V_L(N, \ell) = \Omega(\ell^{1/6}\sqrt{\log N})$$
for all $\alpha > 0$.
3) If $S(z) < 0$ for some $0 < z < 1$, or $S(z) = 0$ for all the values z in some continuous interval, then
$$V_L(N, \ell) = \Omega(\sqrt{\ell \log N}).$$

Proof: If for some bias there are two distinct Bayes-optimal predictions, we have by Lemma 3.17 the bound $V_L(N, \ell) = \Omega(\sqrt{\ell \log N})$, which is the strongest of the bounds claimed here. Thus we only need to consider the case in which for each bias there is at most one Bayes-optimal prediction. By Lemma 3.18, we then have for all predictions z a bias such that z is Bayes-optimal. By Lemma 3.5, the value $S(z)$ is always nonnegative and cannot be zero on any continuous interval.

Recall that when z is Bayes-optimal for q, the condition (3.8) implies that $(1-q)L_0''(z) + qL_1''(z)$ has the same sign as $S(z)$. If $S(z) = 0$, then applying Lemma 3.16 Part 2) with the bias q that makes z Bayes-optimal gives the bound $V_L(N, \ell) = \Omega(\ell^{1/6}\sqrt{\log N})$ for all $\alpha > 0$. If $S(z) > 0$ for all z, Lemma 3.16 Part 1) gives $V_L(N, \ell) \geq (R(z) - o(1)) \ln N$ for all z, from which $V_L(N, \ell) \geq (c_L - o(1)) \ln N$ follows. □

D. Alternative Lower Bound Methods

The lower bounds we have proved are sufficient to show that we cannot improve upon the constant c_L in the upper bound of Theorem 3.1. However, the lower bounds are based on having both ℓ and N approach infinity. It would be interesting to get other bounds for, say, constant N with ℓ approaching infinity. Except for some special cases, we do not really have results along these lines. However, we give here some ideas and arguments that could be useful for such work.

First notice that for the logarithmic loss, there is a simple argument that shows the lower bound $V_L(N, \ell) \geq \ln N$ for $N = 2^k$ and $\ell \geq k$.

Example 3.20: For arbitrary positive integer k, let $N = 2^k$ and $\ell = k$. Let A be an arbitrary on-line prediction algorithm. For the trials $t = 1, \cdots, \ell$ we choose binary prediction vectors $\boldsymbol{x}_t \in \{0, 1\}^N$ in such a way that the set of the experts' prediction sequences $\{(x_{1,i}, \cdots, x_{t,i}) | 1 \leq i \leq N\}$ contains all the $2^\ell = N$ possible binary sequences of length ℓ. The outcomes y_t are chosen by an adversary in such a way that $y_t = 0$ if the prediction $\hat{y}_t$ of the algorithm A satisfies $\hat{y}_t \geq 1/2$, and $y_t = 1$, otherwise. Then at each trial the algorithm incurs loss at least $\ln 2$, and the total loss of the algorithm will be at least $\ell \ln 2 = \ln N$. One expert will have total loss 0, so we obtain $V_{L,A}(N, \ell) \geq \ln N$. This matches exactly the upper bound for $V_{L,A}(N, \ell)$ given in Theorem 3.1 and Example 3.2 when A is the Generic Algorithm 3.7.

Another way of thinking of this lower bound argument is as follows. At the first trial, half of the experts predict 0 and half of the experts predict 1. After the trial, those that made a mistake are eliminated, and those that were correct remain. At subsequent trials, half of the remaining experts predict 0 and half predict 1. Thus at trial t there are $N/2^{t-1}$ experts remaining, each with cumulative loss 0, while the rest of the experts have cumulative loss ∞ and have been eliminated. □

Note that by considering a single trial this easily gives for the logarithmic loss the bound $V_L(2, 1) \geq \ln 2$. The general lower bound $V_L(N, \ell) \geq \ln N$ for the logarithmic loss, when $N = 2^k$ and $\ell \geq k$, can also be obtained by applying the following Theorem 3.22 to this lower bound for $V_L(2, 1)$. Theorem 3.22 is proven using the following lemma.

Lemma 3.21: Assume that for all on-line prediction algorithms A' there is an N-expert trial sequence S' of length ℓ' such that $V_{L,A'}(S') \geq a$, and that for all on-line prediction algorithms A'' there is a two-expert trial sequence S'' of length ℓ'' such that $V_{L,A''}(S'') \geq b$. Then for all on-line prediction algorithms A there is a $2N$-expert trial sequence S of length $\ell' + \ell''$ such that $V_{L,A}(S) \geq a + b$.

Proof: A $2N$-expert *coupled* trial sequence is a sequence in which each instance $\boldsymbol{x}_t$ has the property $x_{t,i} = x_{t,N+i}$ for $1 \leq i \leq N$. A $2N$-expert *simple* trial sequence is a sequence where each instance $\boldsymbol{x}_t$ has the property

$$x_{t,1} = x_{t,2} = \cdots = x_{t,N}$$

and

$$x_{t,N+1} = x_{t,N+2} = \cdots = x_{t,2N}.$$

Note that $2N$-expert coupled trial sequences are essentially N-expert trial sequences and $2N$-expert simple trial sequences are essentially two-expert trial sequences.

Since we assumed that for all prediction algorithms A' there is an N-expert trial sequence S' of length ℓ' such that $V_{L,A'}(S') \geq a$, it follows that for all on-line prediction

algorithms A there is a $2N$-expert coupled trial sequence S_1 of length ℓ' such that $V_{L,A}(S_1) \geq a$. Similarly, since we assumed that for all prediction algorithms A'' there is a two-expert trial sequence S'' of length ℓ'' such that $V_{L,A''}(S'') \geq b$, it follows that for all on-line prediction algorithms A there is a $2N$-expert simple trial sequence S_2 of length ℓ'' such that $V_{L,A}(S_2) \geq b$.

Now let A be an arbitrary on-line prediction algorithm for trial sequences of length $\ell' + \ell''$. Given a trial sequence S' of length ℓ', let $A(S')$ denote the algorithm for trial sequences of length ℓ'' that simulates the algorithm A but processes the trial sequence S' before the first actual trial. Our assumptions imply that there is a $2N$-expert coupled trial sequence S_1 of length ℓ' for which $V_{L,A}(S_1) \geq a$, and that there is a $2N$-expert simple trial sequence S_2 of length ℓ'' for which $V_{L,A(S_1)}(S_2) \geq b$. Let S be the $2N$-expert trial sequence of length $\ell' + \ell''$ that is obtained by concatenating S_1 and S_2.

To complete the proof, we show that

$$\text{Loss}_L(A, S) - \text{Loss}_L(\mathcal{E}_i, S) \geq a + b$$

holds for some $1 \leq i \leq 2N$. Note that

$$\text{Loss}_L(A, S) = \text{Loss}_L(A, S_1) + \text{Loss}_L(A(S_1), S_2).$$

We know that

$$\text{Loss}_L(A, S_1) \geq \text{Loss}_L(\mathcal{E}_i, S_1) + a$$

holds for some $1 \leq i \leq 2N$. Since S_1 is a coupled trial sequence, this implies that for some $1 \leq k \leq N$ we have

$$\text{Loss}_L(A, S_1) \geq \text{Loss}_L(\mathcal{E}_i, S_1) + a$$

both for $i = k$ and for $i = N + k$. We also know that

$$\text{Loss}_L(A(S_1), S_2) \geq \text{Loss}_L(\mathcal{E}_j, S_2) + b$$

holds for some $1 \leq j \leq 2N$. Since S_2 is a simple trial sequence, this implies that

$$\text{Loss}_L(A(S_1), S_2) \geq \text{Loss}_L(\mathcal{E}_j, S_2) + b$$

holds for all $1 \leq j \leq N$ or for all $N + 1 \leq j \leq 2N$. Hence, we have

$$\text{Loss}_L(A, S_1) \geq \text{Loss}_L(\mathcal{E}_j, S_1) + a$$

and

$$\text{Loss}_L(A(S_1), S_2) \geq \text{Loss}_L(\mathcal{E}_j, S_2) + b$$

for $j = k$ or for $j = N + k$, which proves the claim. □

Again, the proof of Lemma 3.21 remains valid if the algorithms are allowed to know the length of the trial sequence beforehand. An obvious induction based on Lemma 3.21 gives the following result.

Theorem 3.22: For any loss function L and positive integer k, we have $V_L(2^k, k\ell) \geq kV_L(2, \ell)$.

In particular, if $\lim_{\ell \to \infty} V_L(2, \ell) \geq c \ln 2$ for some constant c, then for $N = 2^k$, Theorem 3.22 implies

$$\lim_{\ell \to \infty} V_L(N, \ell) \geq c \log_2 N \ln 2 = c \ln N.$$

Hence, if we were able to prove

$$\lim_{\ell \to \infty} V_L(2, \ell) \geq c_L \ln 2$$

for the constant c_L defined in (3.3), we would again obtain the asymptotic lower bound $V_L(N, \ell) \geq (c_L - o(1)) \ln N$ stated in Theorem 3.1. However, this new bound would be stronger because the term $o(1)$ approaches 0 as ℓ approaches ∞ for all N of the form $N = 2^k$, whereas in the bound of Theorem 3.1 the term $o(1)$ is stated to approach 0 only when both N and ℓ approach ∞.

To obtain the lower bound $V_L(N, \ell) \geq (1/2 - o(1)) \ln N$ given in Theorem 3.1 and Example 3.2 for the square loss by applying Theorem 3.22, we would need to show

$$\lim_{\ell \to \infty} V_L(2, \ell) = \frac{\ln 2}{2}. \tag{3.29}$$

We conjecture that (3.29) indeed is true. We have numerically obtained lower bounds such as $V_L(2, 500) \geq 0.3456$, while $(\ln 2)/2 \approx 0.3466$. (Obviously, $V_L(2, \ell)$ is an increasing function of ℓ, and $V_L(2, \ell) \leq (\ln 2)/2$ by the upper bound of Theorem 3.1 and Example 3.2.) These numerical results are based on a recurrence we have not been able to solve in a closed form. Note that for the square loss, the simple construction used for the logarithmic loss does not yield an optimal lower bound. If we have $\ell = 1$ and $N = 2$, with $\boldsymbol{x}_1 = (0, 1)$, we have $V_{L,A}((\boldsymbol{x}_1, y_1)) \leq 1/4 = 0.25$ for the algorithm A that predicts $1/2$, and this bound falls short of the required $(\ln 2)/2 \approx 0.3466$.

The preceding remarks show that for the logarithmic loss we have

$$\lim_{\ell \to \infty} V_L(2^k, \ell) = k \lim_{\ell \to \infty} V_L(2, \ell).$$

It is an interesting open question to see which loss functions L have this property. Theorem 3.22 gives

$$\lim_{\ell \to \infty} V_L(2^k, \ell) \geq k \lim_{\ell \to \infty} V_L(2, \ell)$$

for all loss functions. To show equality it is sufficient to show

$$\lim_{\ell \to \infty} V_L(2, \ell) \geq c_L \ln 2$$

and our conjecture is that this is true for the square loss.

IV. Continuous-Valued Outcomes

A. Applying the Generic Algorithm

We now show that under certain assumptions, The Generic Algorithm 3.7 also works for continuous-valued outcomes $y_t \in [0, 1]$. These assumptions hold for the square and relative entropy loss, but not for the absolute loss, which will be considered in Section IV-B. We also consider the more general situation where the values $x_{t,i}$ and y_t are not in the range $[0, 1]$.

Lemma 4.1: Assume that for all $y, a, b \in [0, 1]$, the function g defined by $g(y, a, b) = L(y, a)/c - \eta L(y, b)$ satisfies

$$\frac{\partial^2 g(y, a, b)}{\partial y^2} + \left(\frac{\partial g(y, a, b)}{\partial y}\right)^2 \geq 0. \tag{4.1}$$

If (3.12) holds for binary values $y \in \{0, 1\}$, then it holds for all values $y \in [0, 1]$.

Proof: We write (3.12) as $(L(y, \hat{y}_t) - \Delta(y))/c \leq 0$. By exponentiating both sides and applying (3.11), we get

$$e^{L(y, \hat{y}_t)/c} \sum_{i=1}^{N} v_{t,i} e^{-\eta L(y, x_{t,i})} \leq 1. \tag{4.2}$$

Let us denote the left-hand side of (4.2) by $f(y)$. Then

$$f(y) = \sum_{i=1}^{N} v_{t,i} e^{g(y, \hat{y}_t, x_{t,i})}$$

so for the second derivative of F we get

$$\frac{\partial^2 f(y)}{\partial y^2} = \sum_{i=1}^{N} \cdot v_{t,i} \left(\frac{\partial^2 g(y, \hat{y}_t, x_{t,i})}{\partial y^2} + \left(\frac{\partial g(y, \hat{y}_t, x_{t,i})}{\partial y}\right)^2\right) \cdot e^{g(y, \hat{y}_t, x_{t,i})}.$$

As our assumption implies this to be nonnegative, the maximum value of f for y in the interval $[0, 1]$ occurs for $y = 0$ or $y = 1$. Since (3.12) is equivalent to $f(y) \leq 1$ for $y \in \{0, 1\}$, this proves our claim. □

Theorem 4.2: Let L be a loss function for which the constant c_L is finite and the condition (4.1) holds for $c = c_L$ and $\eta = 1/c_L$. Let A be the Generic Algorithm 3.7 with the parameters $c = c_L$, $\eta = 1/c_L$, and the initial weights $w_{1,i} = 1$ for all i. Let $S = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell))$ be a trial sequence for which $\boldsymbol{x}_t \in [0, 1]^N$ and $y_t \in [0, 1]$ hold for all t. Then the algorithm does not fail during the trial sequence, and its regret satisfies

$$V_{L,A}(S) \leq c_L \ln N.$$

Proof: First note that by Lemma 3.10, the algorithm A does not fail. By Lemma 4.1, the predictions $\hat{y}_t$ of the algorithm satisfy $L(y_t, \hat{y}_t) \leq \Delta(y_t)$. We then proceed as in the proof of Theorem 3.8, and obtain the claimed bound by choosing $w_{1,i} = 1$ for all i. □

Example 4.3: Let L be the relative entropy loss L_{ent}. We have

$$\frac{\partial L(y, z)}{\partial y} = \ln y - \ln(1 - y) - \ln z + \ln(1 - z)$$

so the second derivative $\partial^2 L(y, z)/\partial y^2 = 1/y + 1/(1 - y)$ does not depend on z. Hence, if $c = 1/\eta$, the second derivative of the function g of Lemma 4.1 is 0, and (4.1) holds. Recall that $c_L = 1$ for the relative entropy loss. Hence, by Theorem 4.2, if A is the Generic Algorithm 3.7 with $c = \eta = 1$, we have $V_{L,A}(S) \leq \ln N$ for any N-expert trial sequence S even if the outcomes $y_t \in [0, 1]$ are continuous-valued. □

Example 4.4: Let L be the square loss L_{sq}. As the second derivative $\partial^2 L(y, z)/\partial y^2$ is constant, the second derivative of the function g of Lemma 4.1 is 0 whenever $c = 1/\eta$, and hence (4.1) trivially holds. Since $c_L = 1/2$, we let A be the Generic Algorithm 3.7 with $c = 1/2$ and $\eta = 2$. Then by Theorem 4.2 we have $V_{L,A}(S) \leq \frac{1}{2} \ln N$ even if the trial sequence S contains continuous-valued outcomes.

Consider now the more general case that at trial t, the experts' predictions $x_{t,i}$ and the outcome y_t are in a known range $[s_t, s_t + r_t]$. Let

$$x'_{t,i} = (x_{t,i} - s_t)/r_t$$

and

$$y'_t = (y_t - s_t)/r_t$$

and let $\hat{y}'_t$ be the prediction of the Generic Algorithm when it is given these scaled inputs $x'_{t,i}$ and outcomes y'_t. Then Theorem 3.8 applies to this scaled sequence of trials. For an algorithm that predicts with $\hat{y}_t = s_t + r_t \hat{y}'_t$ we then have the following loss bound, if we choose $\eta = 2$ and the initial weights to be equal:

$$\sum_{i=1}^{\ell} \left(\frac{y_t - \hat{y}_t}{r_t}\right)^2 \leq \min_{1 \leq i \leq N} \sum_{i=1}^{\ell} \left(\frac{y_t - x_{t,i}}{r_t}\right)^2 + \frac{\ln N}{2}. \tag{4.3}$$

We can consider (4.3) as giving a loss bound similar to (3.14), but with a loss function that changes dynamically as the ranges of $x_{t,i}$ and y_t vary. Note that achieving this bound requires that s_t and r_t are known before the prediction $\hat{y}_t$ is to be made. This is the case, for instance, if the outcome y_t is assumed to be within the range defined by the smallest and largest expert prediction at trial t. Another special case is that before the first trial, we know that $x_{t,i}$ and y_t will always be in some range $[S, S + R]$. We can then take $r_t = R$ for all t, and (4.3) is equivalent with

$$\sum_{i=1}^{\ell} (y_t - \hat{y}_t)^2 \leq \min_{1 \leq i \leq N} \sum_{i=1}^{\ell} (y_t - x_{t,i})^2 + \frac{R^2 \ln N}{2}.$$

Note that if the range of y_t is not bounded, loss bounds of the above form cannot be attained. To see that, let $N = 2$, and consider a one-trial sequence in which the first prediction vector is $(-R/2, R/2)$. The outcome is chosen by an adversary to be either $y_1 = R/2 + \sqrt{K}$ or $y_1 = -R/2 - \sqrt{K}$, depending on whether the algorithm's prediction was negative or not. Then the loss of the best expert is K, and the loss of the algorithm is at least $(R/2 + \sqrt{K})^2 = K + R\sqrt{K} + R^2/4$. Thus if we let K grow, the regret of the algorithm grows as $\Omega(\sqrt{K})$. □

Since the absolute loss L_{abs} does not even have a first derivative everywhere, the technique of Lemma 4.1 does not give any results for this loss function. In the next subsection we devise a new algorithm particularly for this problem.

B. The Vee Algorithm

We now show how the loss bounds obtained for the absolute loss with binary outcomes can also be achieved when the outcomes are continuous-valued. The results of this section were obtained independently by Vovk (private communication).

We call our algorithm the Vee Algorithm. In choosing the prediction it is now necessary to explicitly also consider other outcomes than just $y = 0$ and $y = 1$. We will show that the prediction can still be computed in time $O(N \log N)$.

Algorithm 4.5 (The Vee Algorithm): As the Generic Algorithm 3.7, except that we have fixed the loss function to be the absolute loss, the parameter c to be $(2 \ln(2/(1+e^{-\eta})))^{-1}$, and predicting is done as follows.

Prediction: On receiving the tth input $\boldsymbol{x}_t$, let $Y = \{0, 1, x_{t,1}, \cdots, x_{t,N}\}$ and $v_{t,i} = w_{t,i}/W_t$.

Predict with any value $\hat{y}_t$ that satisfies the condition

$$\max_{y \in Y} \{y - \Delta(y)\} \leq \hat{y}_t \leq \min_{y \in Y} \{y + \Delta(y)\} \tag{4.4}$$

where

$$\Delta(y) = -\frac{\ln\left(\sum_{i=1}^{N} v_{t,i} e^{-\eta|y - x_{t,i}|}\right)}{2 \ln \frac{2}{1+e^{-\eta}}}.$$

It is easy to see how the prediction $\hat{y}_t$ can be obtained in time $O(N)$ once the values

$$s(y) = \sum_{i=1}^{N} v_{t,i} e^{-\eta|y - x_{t,i}|}$$

have been obtained for all the $N + 2$ choices of y. Let $\boldsymbol{x}'_t$ be a vector that contains the components of the prediction vector $\boldsymbol{x}_t$ sorted into an ascending order. Thus $x'_{t,i} \leq x'_{t,i+1}$ for $1 \leq i \leq N - 1$. The vector $\boldsymbol{x}'_t$ can be obtained in time $O(N \log N)$. Let $\boldsymbol{v}'_t$ be the vector obtained by applying to $\boldsymbol{v}_t$ the same permutation that applied to $\boldsymbol{x}_t$ gives $\boldsymbol{x}'_t$. Thus

$$\sum_{i=1}^{N} v_{t,i} \exp(|y - x_{t,i}|) = \sum_{i=1}^{N} v'_{t,i} \exp(|y - x'_{t,i}|).$$

We show how all the sums $s(y)$ for $y \in \{0, x_{t,1}, \cdots, x_{t,N}, 1\}$ can be obtained in time $O(N)$ given the sorted prediction vector $\boldsymbol{x}'_t$. To unify notation, write $x'_{t,0} = 0$ and $x'_{t,N+1} = 1$. Note that for $0 \leq j \leq N + 1$ we can write $s(x'_{t,j}) = a_j + b_j$ where

$$a_j = \sum_{i=1}^{j} v'_{t,i} e^{-\eta(x'_{t,j} - x'_{t,i})}$$

and

$$b_j = \sum_{i=j+1}^{N} v'_{t,i} e^{-\eta(x'_{t,i} - x'_{t,j})}.$$

We have $a_0 = 0$, and b_0 can be computed in time $O(N)$. Further, given a_j and b_j we obtain a_{j+1} and b_{j+1} in time $O(1)$ by

$$a_{j+1} = e^{-\eta(x'_{t,j+1} - x'_{t,j})} a_j + v'_{t,j+1}$$

and

$$b_{j+1} = e^{-\eta(x'_{t,j} - x'_{t,j+1})}(b_j - v'_{t,j+1} e^{-\eta(x'_{t,j+1} - x'_{t,j})}).$$

Hence, the prediction $\hat{y}_t$, if it exists, can be found in total time $O(N \log N)$.

We see in Lemma 4.6 that there always is a prediction $\hat{y}_t$ that satisfies (4.4) and that (4.4) implies $|y - \hat{y}_t| \leq \Delta(y)$ for *all* $y \in [0, 1]$ and not merely for $y \in \{0, 1\}$, which was the requirement in the Generic Algorithm. Hence, we now get for continuous-valued outcomes $y_t \in [0, 1]$ the bound (3.20) that was previously obtained for binary outcomes $y_t \in \{0, 1\}$. Note that if (3.20) holds for $y_t \in [0, 1]$, it actually holds for all y_t, provided we still have $x_{t,i} \in [0, 1]$. This is because moving y_t outside the range of the experts' predictions increases every $|y_t - x_{t,i}|$ as much as it increases $|y_t - \hat{y}_t|$, and the coefficient $\eta/(2 \ln(2/(1 + e^{-\eta})))$ that appears in front of $|y_t - x_{t,i}|$ in (3.20) is greater than 1. Again, the parameter η can be tuned as mentioned in Example 3.14, and the scaling method of Example 4.4 can be used if the values $x_{t,i}$ are not in the range $[0, 1]$.

For the absolute loss, (3.12) has a simple geometric interpretation. Fig. 1 gives an example of the graphs of the left-hand side $|y - \hat{y}|$ and the right-hand side $\Delta(y)$ as functions of y, fixing $\hat{y} = 0.58$ and $\boldsymbol{x} = (0.33, 0.83, 0.97, 0.52)$. The left-hand side of the inequality is represented by a vee-curve with its tip at $(\hat{y}, 0)$. The graph of Δ has a nondifferentiable tip at each value $y = x_i$. The condition (3.12) states that the vee-curve must be below the graph of Δ at y. For continuous-valued outcomes we wish (3.12) to hold for $y \in [0, 1]$ and hence the vee-curve to be below the graph of Δ everywhere. If we were to move the tip of the vee to the left of 0.51, the right arm of the vee would intersect the Δ-curve at the value $y = 0.97$. Hence, the value of the maximum on the left-hand side of (4.4) is roughly 0.51. Similarly, the minimum on the right-hand side is about 0.63, since moving the tip of the vee over this value would make its left arm intersect the Δ-curve at $y = 0.33$. For binary outcomes we only required (3.12) to hold for $y = 0$ and $y = 1$, which gives the weaker condition that the vee-curve must be below the graph of Δ at the endpoints.

For binary outcomes, the loss bound (3.20) was previously shown for a whole family of algorithms defined by a number of different prediction and update factors $\alpha_{t,i}$ [6], as was briefly explained in Example 3.14. In the continuous case we have less freedom. Suppose we were to use $\alpha_{t,i} = 1 - (1 - e^{-\eta})|y_t - x_{t,i}|$, and let $N = 1$, $\boldsymbol{x} = (0)$, and $\eta = 1$. Then $\Delta(0) = 0$, so to satisfy $|y - \hat{y}| \leq \Delta(y)$ for $y = 0$ we must choose $\hat{y} = 0$. However, as $\Delta(0.2) \approx 0.178$, we cannot then have $|y - \hat{y}| \leq \Delta(y)$ for $y = 0.2$. The Algorithm WMC [25] does work for the continuous case, and is allowed to use any update that satisfies (3.21). However, its worst case bound has $1 - e^{-\eta}$ in the denominator instead of $2 \ln(2/(1 + e^{-\eta}))$, and hence it is slightly worse than the bounds given here.

As we noticed in Example 3.14, for binary outcomes it was possible to choose the prediction $\hat{y}_t$ as a function of the weighted average of the experts' predictions. If the outcomes are allowed to be continuous-valued, this is not possible any more. To see that there is no function f such that $\hat{y}_t = f(\sum_i v_{t,i} x_{t,i})$ guarantees (4.4) to hold, we consider two cases. First, let $\boldsymbol{v}_t = (0.3, 0.7)$ and $\boldsymbol{x}_t = (0, 1)$, so $\sum_i v_{t,i} x_{t,i} = 0.7$. For the value $\eta = 1$, the left-hand side of (4.4) is approximately 0.72, and we obtain a constraint $0.72 \leq f(0.7)$ for f. On the other hand, considering $\boldsymbol{v}_t = (1, 0)$ and $\boldsymbol{x}_t = (0.7, 0)$ on the right-hand side of (4.4) gives a contradictory constraint $f(0.7) \leq 0.70$.

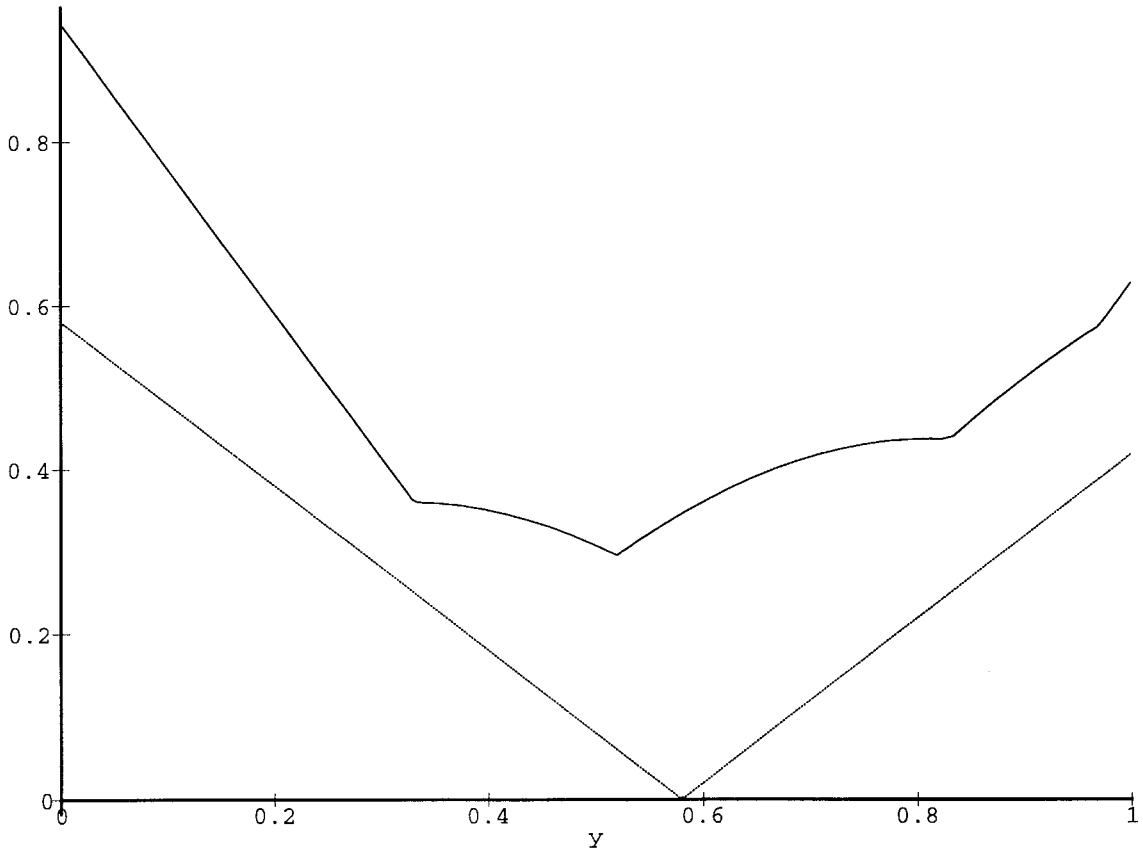

Fig. 1. Example graphs of the functions Δ (above) and L_{abs} (below).

We now show that a prediction that satisfies (4.4) always exists and satisfies the conditions of Theorem 3.8.

Lemma 4.6: Let $\boldsymbol{v}_t \in [0, 1]^N$ with $\sum_i v_{t,i} = 1$ and $\boldsymbol{x}_t \in [0, 1]^N$, and let $\eta > 0$. Then a prediction $\hat{y}_t$ that satisfies (4.4) exists. Further, (4.4) implies $|y - \hat{y}_t| \leq \Delta(y)$ for all $y \in [0, 1]$.

Proof: We prove the existence of $\hat{y}_t$ by showing that

$$y - \Delta(y) \leq z + \Delta(z) \tag{4.5}$$

holds for all y, z, $\boldsymbol{v}_t$, and $\boldsymbol{x}_t$. Define

$$g(\boldsymbol{v}, \boldsymbol{x}, y, z) = \sum_{i=1}^{N} \sum_{j=1}^{N} v_i v_j \exp(-\eta(|y - x_i| + |z - x_j|) + (y - z)2 \ln(2/(1 + e^{-\eta}))). \tag{4.6}$$

Then (4.5) is equivalent to $g(\boldsymbol{v}_t, \boldsymbol{x}_t, y, z) \leq 1$. The second derivative $\partial^2 g(\boldsymbol{v}, \boldsymbol{x}, y, z)/\partial x_i^2$ is defined and positive if $x_i \notin \{0, y, z, 1\}$. Thus it suffices to show $g(\boldsymbol{v}, \boldsymbol{x}, y, z) \leq 1$ for $N = 4$ and $\boldsymbol{x} = \boldsymbol{x}_a = (0, y, z, 1)$. In this restricted case, the second derivative $\partial^2 g(\boldsymbol{v}, \boldsymbol{x}_a, y, z)/\partial z^2$ is positive if $z \notin \{0, y, 1\}$. Furthermore, since $\Delta(z) \geq 0$, (4.5) trivially holds if $z \geq y$. Thus it suffices to show (4.5) for $z = 0$, $y > 0$, and $\boldsymbol{x} = \boldsymbol{x}_b = (0, y, 0, 1)$. Finally, since the second derivative $\partial^2 g(\boldsymbol{v}, \boldsymbol{x}_b, y, 0)/\partial y^2$ is positive, we are left with the case $z = 0$, $y = 1$, and $\boldsymbol{x} \in \{0, 1\}^N$. In this case, the original inequality (4.5) can be rewritten as

$$\frac{\ln((1 - p)e^{-\eta} + p) + \ln(1 - p + pe^{-\eta})}{2} \leq \ln \frac{1 + e^{-\eta}}{2}$$

where $p = \sum_i v_i x_i$. This holds for all $0 \leq p \leq 1$ because the function $\ln$ is concave.

A similar argument based on second derivatives shows that for $y \in [0, 1]$, the value $y - \Delta(y)$ obtains its maximum and the value $y + \Delta(y)$ its minimum when $y \in \{0, 1, x_{t,1}, \cdots, x_{t,N}\}$. □

Lemma 4.6 immediately implies the following result.

Theorem 4.7: Let $S = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_\ell, y_\ell))$ be a trial sequence with $\boldsymbol{x}_t \in [0, 1]^N$ and $y_t \in [0, 1]$ for all t. Let L be the absolute loss and A be the Vee Algorithm 4.5. We then have

$$\text{Loss}_L(A, S) \leq \frac{-\ln \frac{w_{1,i}}{W_1} + \eta \text{Loss}_L(\mathcal{E}_i, S)}{2 \ln \frac{2}{1 + e^{-\eta}}}$$

for all i.

V. Further Work

One of the most challenging open problems is to give tight bounds for the regret of the prediction algorithm compared to the loss of the best expert for even more general classes of loss functions than those considered in this paper. When the outcomes y_t are binary, it might be possible to produce such bounds for arbitrary loss functions. The next challenge is to extend the results for continuous-valued outcomes to more general loss functions. Another direction worth exploring is to let outcomes be discrete valued with more than two

choices. The recent results of Chung [10] address some of these problems.

In this paper we restricted the predictions of the experts to lie between zero and one, except in specific examples where we have indicated how scaling tricks can be used. It would be nice to do a thorough investigation of how scaling the range of the variables affects the results. Bounding some norm of the prediction vector might also lead to interesting problems. Restricting the range of the predictions of individual experts is related to bounding the infinity norm of the prediction vectors.

It would be interesting to see whether the alternative update rules defined by (3.21) for the absolute loss work for other loss functions. As we have seen, it is sometimes possible to obtain the prediction as a function of the weighted average of the experts' predictions. We would like to know exactly when this simplification is possible without weakening our bounds, or with weakening them only slightly.

In this paper we have given bounds of the regret of our algorithms over the loss of the best expert. A more challenging problem is to bound the regret of the algorithms over the best linear combination of experts [9], [23], [24]. The only worst case loss bounds for the latter case that have been obtained were for the square-loss function. Hopefully, some of the results of the present paper can be generalized to the linear combination case. An intermediate case worth exploring is the case of bounding the regret of the algorithm compared with the best "stretched" expert, i.e., an original expert multiplied by some positive constant.

APPENDIX

Lemma A.1: Let P be a probability measure in X and Q a probability measure in Y. For $\ell \in \boldsymbol{N}_+$ and $y \in Y$, let $U^y_{1\ell}, \cdots, U^y_{N\ell}$ be N independent identically distributed random variables such that

$$\mathrm{E}_{x\in P}[U^y_{i\ell}(x)] = 0$$

and

$$\mathrm{Var}_{x\in P}[U^y_{i\ell}(x)] = 1.$$

Assume that there are independent identically distributed random variables $F_1, \cdots, F_N$ such that the sequence $U^y_{i1}, U^y_{i2}, \cdots$ converges in distribution to F_i for all i and y. Further, let $r_1, r_2, \cdots$ be functions on Y such that $\lim_{\ell\to\infty} r_\ell(y) = 1$ holds with probability 1 for y drawn according to Q, and $|r_\ell(y)| \le B$ holds for all y for some constant B. Then

$$\lim_{\ell\to\infty} \mathrm{E}_{y\in Q}\left[r_\ell(y)\mathrm{E}_{x\in P}\left[\min_{1\le i\le N} U^y_{i\ell}(x)\right]\right] = \mathrm{E}\left[\min_{1\le i\le N} F_i\right].$$

Proof: Write

$$U^y_{*\ell} = \min_{1\le i\le N} U^y_{i\ell}$$

and

$$F_* = \min_{1\le i\le N} F_i.$$

We first show that for all y, the sequence $U^y_{*1}, U^y_{*2}, \cdots$ converges in distribution to F_*. For all $a \in \boldsymbol{R}$ we have

$$\begin{aligned}\Pr[F_* \le a] &= 1 - \prod_{i=1}^{N}(1 - \Pr[F_i \le a]) \\ &= 1 - \prod_{i=1}^{N}(1 - \lim_{\ell\to\infty}\Pr[U^y_{i\ell} \le a]) \\ &= \lim_{\ell\to\infty}\left(1 - \prod_{i=1}^{N}(1 - \Pr[U^y_{i\ell} \le a])\right) \\ &= \lim_{\ell\to\infty}\Pr[U^y_{*\ell} \le a],\end{aligned}$$

which proves the claim.

Next we see that

$$\mathrm{E}_{x\in P}[|U^y_{*\ell}(x)|^{1+p}] \le 2N \tag{A.1}$$

holds for all y when $p = 0$ or $p = 1$. To see this, first note that for all $A \subseteq \boldsymbol{R}$, if $U^y_{*\ell}(x) \in A$ then $U^y_{i\ell}(x) \in A$ for at least one value i. As the distribution of $U^y_{i\ell}$ does not depend on i, this implies

$$\Pr_{x\in P}[U^y_{*\ell}(x) \in A] \le N \Pr_{x\in P}[U^y_{1\ell}(x) \in A]$$

if A is measurable. This implies

$$\begin{aligned}\mathrm{E}_{x\in P}[|U^y_{*\ell}(x)|^{1+p}] &\le N\mathrm{E}_{x\in P}[|U^y_{1\ell}(x)|^{1+p}] \\ &= N\int |U^y_{1\ell}|^{1+p}\,dP \\ &\le N\left(1 + \int_{|U^y_{1\ell}|\ge 1} |U^y_{1\ell}|^{1+p}\,dP\right) \\ &\le N(1 + \mathrm{E}_{x\in P}[U^y_{1\ell}(x)^2]) \\ &= 2N.\end{aligned}$$

As the sequence $U^y_{*1}, U^y_{*2}, \cdots$ converges in distribution to F_*, the bound (A.1) with $p = 1$ guarantees [3, Corollary, p. 292]

$$\lim_{\ell\to\infty} \mathrm{E}_{x\in P}[U^y_{*\ell}(x)] = \mathrm{E}[F_*]$$

for all y and, therefore,

$$\lim_{\ell\to\infty} r_\ell(y)\mathrm{E}_{x\in P}[U^y_{*\ell}(x)] = \mathrm{E}[F_*]$$

with probability 1 for y drawn from Q. The bound (A.1) with $p = 0$ implies

$$|r_\ell(y)\mathrm{E}_{x\in P}[U^y_{*\ell}(x)]| \le 2BN$$

and the bounded convergence theorem [3, Theorem 16.5, p. 180]

$$\lim_{\ell\to\infty} \mathrm{E}_{y\in Q}[r_\ell(y)\mathrm{E}_{x\in P}[U^y_{*\ell}(x)]] = \mathrm{E}[F_*]$$

as claimed. □

ACKNOWLEDGMENT

The authors wish to thank David P. Helmbold for his significant help in developing the Vee Algorithm and Andrew Barron for his insightful comments.

REFERENCES

[1] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: The adversarial multi-armed bandit problem," in *Proc. 36th Annu. Symp. Foundations of Computer Science*. Los Alamitos, CA: IEEE Computer Soc. Press, 1995, pp. 322–331.

[2] A. R. Barron and Q. Xie, "Asymptotic minimax loss for data compression, gambling, and prediction," in *Proc. 9th Annu. Conf. Computational Learning Theory*. New York: ACM, 1996.

[3] P. Billingsley, *Probability and Measure*, 2nd ed. New York: Wiley, 1986.

[4] D. Blackwell, "Controlled random walks," in *Proc. Int. Congr. Mathematicians* (Amsterdam, The Netherlands, 1954), vol. III, pp. 336–338.

[5] ———, "An analog of the minimax theorem for vector payoffs," *Pacific J. Math.*, vol. 6, pp. 1–8, 1956.

[6] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth, "How to use expert advice," *J. Assoc. Comput. Mach.*, vol. 44, pp. 427–485, 1997.

[7] N. Cesa-Bianchi, Y. Freund, D. P. Helmbold, and M. K. Warmuth, "On-line prediction and conversion strategies," *Machine Learning*, vol. 25, pp. 71–110, 1996.

[8] N. Cesa-Bianchi, D. P. Helmbold, and S. Panizza, "On Bayes methods for on-line boolean prediction," in *Proc. 9th Annu. Conf. Computational Learning Theory*. New York: ACM, 1996, pp. 314–324.

[9] N. Cesa-Bianchi, P. Long, and M. K. Warmuth, "Worst-case quadratic loss bounds for on-line prediction of linear functions by gradient descent," *IEEE Trans. Neural Networks*, vol. 7, pp. 604–619, 1996.

[10] T. H. Chung, "Approximate methods for sequential decision making using expert advice," in *Proc. 7th Annu. ACM Workshop on Computational Learning Theory*. New York: ACM, 1994, pp. 183–189.

[11] T. Cover, "Behavior of sequential predictors of binary sequences," in *Proc. 4th Prague Conf. Information Theory, Statistical Decision Functions and Random Processes*. Prague, Czechoslovakia: Publishing House of the Czechoslovak Acad. Sci., 1965, pp. 263–272.

[12] T. M. Cover and E. Ordentlich, "Universal portfolios with side information," *IEEE Trans. Inform. Theory*, vol. 42, pp. 348–363, 1996.

[13] A. P. Dawid, "Prequential analysis, stochastic complexity and Bayesian inference," in *Bayesian Statistics 4*. Oxford, U.K.: Clarendon, 1992.

[14] A. DeSantis, G. Markowsky, and M. N. Wegman, "Learning probabilistic prediction functions," in *Proc. 29th Annu. IEEE Symp. Foundations of Computer Science*. Los Alamitos, CA: IEEE Computer Soc. Press, 1988, pp. 110–119.

[15] M. Feder, N. Merhav, and M. Gutman, "Universal prediction of individual sequences," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1258–1270, 1992.

[16] D. P. Foster, "Prediction in the worst case," *Ann. Statist.*, vol. 19, pp. 1084–1090, 1991.

[17] Y. Freund, "Predicting bits almost as well as the optimal biased coin," in *Proc. 9th Annu. Conf. Computational Learning Theory*. New York: ACM, 1996, pp. 89–98.

[18] J. Galambos, *The Asymptotic Theory of Extreme Order Statistics*, 2nd ed. Malabar, FL: Krieger, 1987.

[19] J. Hannan, "The dynamic statistical decision problem when the component problem involves a finite number, m, of distributions," *Ann. Math. Statist.*, vol. 27, p. 212, 1956.

[20] ———, "Approximations to Bayes risk in repeated play," *Ann. Math. Stud.*, vol. 39, pp. 97–139, 1957.

[21] D. Haussler and A. Barron, "How well do Bayes methods work for on-line prediction of $\{+1, -1\}$ values?," in *Proc. 3rd NEC Symp. Computation and Cognition*, 1992, pp. 74–100.

[22] D. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth, "On-line portfolio selection using multiplicative updates," in *Proc. 13th Int. Conf. Machine Learning*. San Francisco, CA: Kaufmann, 1996, pp. 243–251.

[23] J. Kivinen and M. K. Warmuth, "Additive versus exponentiated gradient updates for linear prediction," *Inform. Comput.*, vol. 132, pp. 1–64, 1997.

[24] N. Littlestone, P. M. Long, and M. K. Warmuth, "On-line learning of linear functions," *J. Comput. Complexity*, vol. 5, pp. 1–23, 1995.

[25] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," *Inform. Comput.*, vol. 108, pp. 212–261, 1994.

[26] N. Merhav and M. Feder, "Universal sequential learning and decisions from individual data sequences," in *Proc. 5th Annu. Workshop on Computational Learning Theory*. New York: ACM, 1992, pp. 413–427.

[27] J. Mycielski, "A learning algorithm for linear operators," *Proc. Amer. Math. Soc.*, vol. 103, pp. 547–550, 1988.

[28] M. Opper and D. Haussler, "Worst case prediction over sequences under log loss" in *The Mathematics of Information Coding, Extraction and Distribution*. New York: Springer-Verlag, 1997.

[29] J. Rissanen, "Universal coding, information, prediction and estimation," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 629–636, 1984.

[30] ———, *Stochastic Complexity in Statistical Inquiry*, vol. 15 of *Series in Computer Science*. New York: World Scientific, 1989.

[31] ———, "Fisher information and stochastic complexity," *IEEE Trans. Inform. Theory*, vol. 42, pp. 40–47, 1996.

[32] J. Shtarkov, "Coding of discrete sources with unknown statistics," in *Topics in Information Theory*. Amsterdam, The Netherlands: North Holland, 1975, pp. 559–574.

[33] Y. M. Shtarkov, "Universal sequential coding of single messages," *Probl. Pered. Inform.*, vol. 23, pp. 175–186, 1987.

[34] J. S. Vitter and P. Krishnan, "Optimal prefetching via data compression," in *Proc. 32nd IEEE Symp. Foundations of Computer Science*. Los Alamitos, CA: IEEE Computer Soc., 1991, pp. 121–130.

[35] V. Vovk, "Aggregating strategies," in *Proc. 3rd Annu. Workshop on Computational Learning Theory*. San Mateo, CA: Kaufmann, 1990, pp. 371–383.

[36] ———, "Universal forecasting algorithms," *Inform. Comput.*, vol. 96, pp. 245–277, 1992.

[37] ———, "A game of prediction with expert advice," in *Proc. 8th Annu. Conf. Computational Learning Theory*. New York: ACM, 1995, pp. 51–60.

[38] M. J. Weinberger, A. Lempel, and J. Ziv, "A sequential algorithm for the universal coding of finite memory sources," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1002–1014, 1992.

[39] M. J. Weinberger, N. Merhav, and M. Feder, "Optimal sequential probability assignment for individual sequences," *IEEE Trans. Inform. Theory*, vol. 40, pp. 384–396, 1994.