Using experts for predicting continuous outcomes

Jyrki Kivinen[1] **Manfred K. Warmuth**[2]

Computer and Information Sciences
University of California, Santa Cruz
Santa Cruz, CA 95064, USA

Abstract

We consider on-line predictions when the inputs to the learner are predictions of a pool of experts. The learning criterion is the difference between the loss of the algorithm and the loss of the best expert, as measured by a loss function, over a worst case sequence of trials. Vovk has previously proven very general results for the case where the outcomes to be predicted are binary. The prediction method is based on representing each expert by a weight that decreases exponentially as a function of the total loss incurred by the expert. We simplify the proofs by making some mild assumptions about the loss function. Using derivative arguments we can then show that for many loss functions this method gives the same upper bounds also in the case where the outcomes are continuous-valued. For the square and logarithmic losses we can use the same prediction algorithm as with binary outcomes. For the absolute loss we need a special algorithm.

1 Introduction

We consider the following on-line prediction model. In each trial the algorithm makes a prediction after receiving an N-component instance. Each component may be seen as a suggested prediction from an expert. At the end of the trial the algorithm receives an outcome and the discrepancy of the algorithm's prediction and the outcome is measured with a loss function. Following a model introduced by Littlestone [1] we seek to obtain total loss bounds that hold for an arbitrary sequence of instances and outcomes. Of course such bounds have to depend in some way on the difficulty

[1]Supported by Emil Aaltonen Foundation, University of Helsinki and the Academy of Finland; e-mail kivinen@cse.ucsc.edu.

[2]Supported by ONR grant NO0014-91-J-1162 and NSF grant IRI-9123692; e-mail manfred@cse.ucsc.edu.

of the sequence itself. As done in [2, 3, 4] we measure this difficulty as the total loss of the best component/expert on the sequence. In particular, as in [4] we are interested in minimizing the additional loss of the algorithm over the total loss of the best expert.

Most of the previous research has focused on the case when the outcomes are binary. The algorithm WMC of [2] for the absolute loss is an exception to this. Here we address the case when the outcome is continuous. For the case of binary outcomes, Vovk [3] introduced an algorithm that generalizes the Weighted Majority Algorithm of [2]. In Vovk's algorithm each expert E_i is represented by a weight of the form $w_{1,i}e^{-\eta L_i}$, where $w_{1,i}$ is E_i's initial weight, L_i is the total loss of E_i incurred in previous trials and $\eta \geq 0$ is a learning rate, which can be tuned to obtain optimal performance. Even though better weighting schemes have recently been devised [5] for the special case when the predictions of the experts, the prediction of the combining algorithm and the outcomes of the trials are all binary and the loss function simply counts the number of wrong predictions, the exponential weighting scheme is appealing because of its general applicability to all loss functions. This general applicability is further exemplified by the results of this paper.

We first introduce some mild monotonicity and continuity assumptions that let us simplify the very general framework considered by Vovk. We give simple proofs for the binary outcome case of the bounds previously obtained for the square loss, the relative entropy loss, and the absolute loss. In our simplified framework we can then use a derivative argument to show that for the square loss and the relative entropy loss, the bounds proven for the binary case also hold for the continuous case. Surprisingly, the algorithm remains unchanged for the square and the relative entropy loss. For the absolute loss, which is not differentiable, we present a special algorithm called the Vee Algorithm. Even though the algorithm needs to be more sophisticated to handle continuous outcomes, we can prove the same worst case bound on the absolute loss of the Vee Algorithm that was previously obtained for the case of binary outputs [3]. The bound is slightly better than the one proven for WMC [2] for the case of continuous outcomes, and essentially matches the lower bounds of [4].

By method similar to those of [2], it is easy to get adversary lower bounds that show the bounds for the square and relative entropy loss to be within a constant factor of optimal. These lower bounds need to be strengthened to show that our bounds are tight as was already done for the absolute loss in [4] using probabilistic methods.

2 Basic notions

In the model we consider, learning proceeds as a sequence of trials. At trial t, the learner is presented with an input, which we assume to be an N-dimensional real vector $\mathbf{x}_t = (x_{t,1}, \ldots, x_{t,N})$. The input component $x_{t,i}$ can be considered the prediction of the ith expert from a pool of N experts. Our analysis will be based on showing that over any sequence of trials, certain learning algorithms predict almost as well as any one of the N experts.

The learner's *prediction* $\hat{y}_t$ at trial t is based on the input and the internal state of the learner before the trial. We assume the internal state to be an N-dimensional real vector $\mathbf{w}_t = (w_{t,1}, \ldots, w_{t,N})$ and the prediction $\hat{y}_t$ to be a real number. The learner's initial state is $\mathbf{w}_1$. Let P be the learner's prediction function, i.e., $\hat{y}_t = P(\mathbf{w}_t, \mathbf{x}_t)$. After making its prediction, the learner receives an *outcome* y_t, which is a real number. It then updates its internal state to $\mathbf{w}_{t+1}$. Let the update function be U, i.e., $\mathbf{w}_{t+1} = U(\mathbf{w}_t, \mathbf{x}_t, y_t)$, or $w_{t+1,i} = U_i(\mathbf{w}_t, \mathbf{x}_t, y_t)$ for all i.

We shall also need the scaled version $\mathbf{v}_t$ of the learner's internal state. Thus, let $W_t = \sum_i w_{t,i}$ and $\mathbf{v}_t = (v_{t,1}, \ldots, v_{t,N})$ where $v_{t,i} = w_{t,i}/W_t$.

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 $[0, +\infty)$. The square loss L_{sq} defined by $L_{\mathrm{sq}}(y, \hat{y}) = (y - \hat{y})^2$ is a typical loss function. Other possibilities include the absolute loss $L_{\mathrm{abs}}(y, \hat{y}) = |y - \hat{y}|$ and, for $y, \hat{y} \in [0, 1]$, the relative entropy loss $L_{\mathrm{ent}}(y, \hat{y}) = y \ln \frac{y}{\hat{y}} + (1 - y) \ln \frac{1-y}{1-\hat{y}}$. (As usual, we define $0 \ln 0 = 0$.) For binary outcomes $y \in \{ 0, 1 \}$, the relative entropy loss simplifies to the logarithmic loss: $L_{\mathrm{ent}}(0, \hat{y}) = -\ln(1 - \hat{y})$ and $L_{\mathrm{ent}}(1, \hat{y}) = -\ln(\hat{y})$.

3 Framework for analysis

We define for the state vectors a function B, which will act as a loss bound. When after the update following the tth trial, the learner's state vector becomes $\mathbf{w}_{t+1}$, the total loss of the learner at the trials $1, \ldots, t$ is allowed to be at most $B(\mathbf{w}_{t+1})$. The function will be defined in such a way that $B(\mathbf{w}_t)$ is nonnegative for all allowed states $\mathbf{w}_t$, and $B(\mathbf{w}_1) = 0$ for the initial state $\mathbf{w}_1$. Thus, define

$$\Delta(\mathbf{w}_t, \mathbf{x}_t, y_t) = B(U(\mathbf{w}_t, \mathbf{x}_t, y_t)) - B(\mathbf{w}_t) \ . \tag{3.1}$$

If we can then prove for every trial t that

$$L(y_t, \hat{y}_t) \leq \Delta(\mathbf{w}_t, \mathbf{x}_t, y_t) \ , \tag{3.2}$$

we obtain the desired bound

$$\sum_{t=1}^{\ell} L(y_t, \hat{y}_t) \leq \sum_{t=1}^{\ell} \Delta(\mathbf{w}_t, \mathbf{x}_t, y_t) = B(\mathbf{w}_{\ell+1}) - B(\mathbf{w}_1) = B(\mathbf{w}_{\ell+1})$$

for the total loss made by the algorithm at the trials $1, \ldots, \ell$.

Two problems now arise. First, how do we choose the update function U and the loss bound function B in order to make the loss bound $B(\mathbf{w}_{\ell+1})$ interesting; and second, after choosing the update and loss bound functions, how do we choose the prediction $\hat{y}_t$ in such a way that (3.2) holds. We first consider the second problem on a general level. This gives us conditions that must be fulfilled by U and B in order to make the condition (3.2) satisfiable by a good choice of $\hat{y}_t$.

We first consider the special case of binary outcomes $y_t \in \{0, 1\}$, studied already by Vovk [3]. The predictions $\hat{y}_t$ of the algorithm, as well as the predictions $x_{t,i}$ of the experts, are allowed to be real numbers from the interval $[0, 1]$. We later see that the results we obtain also hold for continuous-valued outcomes $y_t \in [0, 1]$ if the loss function L satisfies certain assumptions. Scaling arguments allow us to generalize the results for more general ranges.

Consider fixed $\mathbf{w}_t$ and $\mathbf{x}_t$, and write

$$\Delta_0 = \Delta(\mathbf{w}_t, \mathbf{x}_t, 0) \qquad \text{and} \qquad \Delta_1 = \Delta(\mathbf{w}_t, \mathbf{x}_t, 1) \ .$$

Further, for $\hat{y} \in [0, 1]$, let $L_0(\hat{y}) = L(0, \hat{y})$ and $L_1(\hat{y}) = L(1, \hat{y})$. We assume that Δ_0 and Δ_1 are nonnegative, L_0 and L_1 are continuous, and $L_0(0) = L_1(1) = 0$. Now condition (3.2) for binary outcomes reduces to

$$L_0(\hat{y}_t) \le \Delta_0 \quad \text{and} \quad L_1(\hat{y}_t) \le \Delta_1 \ . \tag{3.3}$$

To obtain explicit bounds for $\hat{y}_t$ from (3.3), we need to have some notion of an inverse for L_0 and L_1. Assume that L_0 is strictly increasing and L_1 is strictly decreasing in $[0, 1]$, which is the typical case. We also assume that both L_0 and L_1 are decreasing in $(-\infty, 0]$ and increasing in $[1, +\infty)$, so we do not lose generality by assuming the predictions $\hat{y}_t$ to be always in $[0, 1]$. Then L_0 has a strictly increasing inverse $L_0^{-1}\colon [0, L_0(1)] \to [0, 1]$, and L_1 has a strictly decreasing inverse $L_1^{-1}\colon [0, L_1(0)] \to [0, 1]$. Consider for the moment the case that Δ_0 and Δ_1 are in such ranges where the inverse values $L_0^{-1}(\Delta_0)$ and $L_1^{-1}(\Delta_1)$ are defined. Then (3.3) becomes

$$L_1^{-1}(\Delta_1) \le \hat{y}_t \le L_0^{-1}(\Delta_0) \ . \tag{3.4}$$

A prediction $\hat{y}_t$ that satisfies (3.4) can be found if and only if

$$L_1^{-1}(\Delta_1) \le L_0^{-1}(\Delta_0) \ . \tag{3.5}$$

If (3.5) holds, the prediction $\hat{y}_t$ can be chosen to be an arbitrary number 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 $\hat{y}_t$.

For instance, if L is the square loss L_{sq}, we have $L_0^{-1}(z) = \sqrt{z}$ and $L_1^{-1}(z) = 1 - \sqrt{z}$ for $0 \le z \le 1$, so for values Δ_0 and Δ_1 in the range $[0, 1]$,

(3.5) becomes

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

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

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

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

$$\Delta_0 + \Delta_1 \geq 1 \ .$$

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 $L_0(\hat{y}_t) \leq \Delta_0$ in (3.3) holds for all $\hat{y}_t$. Thus, the equivalence between (3.3) and (3.4) will be maintained for all nonnegative Δ_0 if the inverse L_0^{-1} is extended in such a way that the condition $\hat{y}_t \leq L_0^{-1}(\Delta_0)$ holds for all $\hat{y}_t \in [0,1]$. 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 $L_0^{-1}(\Delta_0) \geq 1$ whenever $\Delta_0 \geq L_0(1)$. 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 $L_1^{-1}(\Delta_1) \leq 0$ whenever $\Delta_1 \geq L_1(0)$. It is easy to see that the expressions given above for L_0^{-1} and L_1^{-1} for the square, relative entropy, and absolute losses define valid generalized inverses in the whole domain $[0, +\infty)$. Our definitions of generalized inverses let us show equivalence between (3.4) and (3.3) for all values of Δ_0 and Δ_1.

Lemma 1. *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 conditions (3.4) and (3.3) are equivalent for all $\hat{y}_t \in [0,1]$.*

We now turn to defining update functions and loss bounds for which the condition (3.5) holds. The state vector $\mathbf{w}_t$ is considered as giving to each experts i a weight $w_{t,i}$ that reflect the credibility enjoyed by the expert just before trial t. We use the update scheme introduced by Vovk [3]. Thus, we let $\mathbf{w}_{t+1} = U(\mathbf{w}_t, \mathbf{x}_t, y_t)$ where the update function U is defined by

$$U_i(\mathbf{w}_t, \mathbf{x}_t, y_t) = w_{t,i} e^{-\eta L(y_t, x_{t,i})} \tag{3.6}$$

for some positive constant η, which we call the *learning rate*. The larger the learning rate, the faster the weights of experts decrease. Initially we could make all the weights equal by setting $w_{1,i} = 1/N$, but also other settings are possible.

Recall that $W_t = \sum_i w_{t,i}$ is the total weight before trial t. Following Vovk, choose the loss bound B to be

$$B(\mathbf{w}_t) = -c \ln \frac{W_t}{W_1} \tag{3.7}$$

where c is another positive constant, called the *loss coefficient*. As the total weight is W_1 before the first trial and can then only decrease as a result of losses incurred by the experts, the value $B(\mathbf{w}_t)$ is always nonnegative. The smaller the value of c, the better our loss bounds will be, and the more difficult it will be to find $\hat{y}_t$ such that (3.2) holds.

For the function Δ given in (3.1) we now get

$$\begin{aligned} \Delta(\mathbf{w}_t, \mathbf{x}_t, y_t) &= -c \ln \frac{W_{t+1}}{W_t} = -c \frac{\sum_i w_{t,i} e^{-\eta L(y_t, x_{t,i})}}{W_t} \\ &= -c \ln \sum_{i=1}^{N} v_{t,i} e^{-\eta L(y_t, x_{t,i})} \end{aligned} \tag{3.8}$$

where $v_{t,i} = w_{t,i}/W_t$ is the normalized ith weight. Then $\Delta(\mathbf{w}_t, \mathbf{x}_t, y_t)$ is always nonnegative, and strictly positive unless the predictions $x_{t,i}$ of all the experts coincide with the outcome y_t. We get explicit formulas for Δ_0 and Δ_1 by substituting onto the right-hand side of (3.8) a particular loss function and the values $y_t = 0$ and $y_t = 1$.

Our interest in defining Δ according to Vovk's work comes from the following result [3], which is the basis for all the loss bounds we obtain.

Theorem 2. *For $t = 1, \ldots, \ell$, let $\mathbf{x}_t$ be an arbitrary instance vector and y_t an arbitrary outcome. Let Δ be as in (3.8). If the predictions $\hat{y}_t$ satisfy (3.2) and the updates satisfy (3.6) for all t, then the total loss satisfies*

$$\sum_{t=1}^{\ell} L(y_t, \hat{y}_t) \le -c \ln \frac{w_{1,i}}{W_1} + c\eta \sum_{t=1}^{\ell} L(y_t, x_{t,i}) \tag{3.9}$$

for all i.

We say that the loss function L is (c, η)-*realizable* if for all $\mathbf{w}_t$ and $\mathbf{x}_t$ it is possible to choose $\hat{y}_t$ such that (3.2) holds for all binary outcomes $y_t \in \{ 0, 1 \}$. We later see that this is often sufficient to make (3.2) hold even when the y_t are chosen in the continuous interval $[0, 1]$. Ideally, we would like to have $c = 1/\eta$, in which case the loss of our algorithm exceeds the loss of the best expert only by an additive term. For instance, if we choose the initial weights $w_{1,i}$ to be equal, this term becomes $c \ln N$. Thus, in order to optimally tune the learning rate η, we would like to know the smallest c such that the loss function is $(c, 1/c)$-realizable, and then use $\eta = 1/c$. As we shall soon see, this leads to choosing $\eta = 2$ for the square loss [3] and $\eta = 1$ for the relative entropy loss [6]. However, the absolute loss is not $(c, 1/c)$-realizable for any c, and more complicated tuning is needed [4].

4 Generic prediction algorithm

We now state our generic algorithm that is based on the condition (3.2) for the predictions $\hat{y}_t$ and the expression (3.8) for Δ. We assume that the experts' predictions $x_{t,i}$ are in $[0,1]$. We first give the loss bounds only for the special of binary outcomes $y_t \in \{0,1\}$. It later turns out that for the square and relative entropy losses, the same bound hold for continuous-valued outcomes $y_t \in [0,1]$. Later we also see how the results generalize for larger ranges.

Algorithm 3. (The Generic Algorithm) Let L be a loss function. The algorithm maintains weights $w_{t,i}$ and $v_{t,i}$ for $i = 1, \ldots, N$, and works as follows.

Assumptions: For $t = 1, \ldots, \ell$, the algorithm receives an input vector $\mathbf{x}_t$ that consists of N components $x_{t,i} \in [0,1]$. After making its tth prediction $\hat{y}_t \in [0,1]$, the algorithm receives an outcome $y_t \in [0,1]$.

Parameters: a positive learning rate η and a positive loss coefficient c such that L is (c,η)-realizable.

Initialization: Set the weights to some initial values $w_{1,i}$, and let $v_{1,i} = w_{1,i} / \sum_i w_{1,i}$.

Prediction: On receiving the tth input $\mathbf{x}_t$, predict with any value $\hat{y}_t$ that satisfies the condition

$$L_1^{-1}(\Delta(\mathbf{w}_t, \mathbf{x}_t, 1)) \leq \hat{y}_t \leq L_0^{-1}(\Delta(\mathbf{w}_t, \mathbf{x}_t, 0)) \tag{4.1}$$

where

$$\Delta(\mathbf{w}_t, \mathbf{x}_t, y_t) = -c \ln \sum_{i=1}^{N} v_{t,i} e^{-\eta L(y_t, x_{t,i})} \quad .$$

Update: After receiving the tth outcome y_t, let

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

and $v_{t+1,i} = w_{t+1,i} / \sum_i w_{t+1,i}$ for $i = 1, \ldots, N$.

By Lemma 1, the prediction $\hat{y}_t$ of the Generic Algorithm satisfies (3.2) if the outcomes are binary. Hence, the bound (3.9) applies in this special case. In Section 5 we show that for the square and relative entropy loss, the Generic Algorithm applies the bound (3.9) even for continuous-valued outcomes $y_t \in [0,1]$. For absolute loss, we give in Section 6 a special algorithm for coping with continuous-valued outcomes.

Example 4. Consider the square loss. Vovk [3] has shown that the square loss is $(1/2, 2)$-realizable. In the full paper, we give for this a simplified proof, which also shows that the square loss is not $(c, 1/c)$-realizable for $c < 1/2$. The condition (4.1) now becomes

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

By numerically substituting random values for $\mathbf{v}_t$ and $\mathbf{x}_t$ we see that the seemingly natural choice $\hat{y}_t = \sum_i v_{t,i} x_{t,i}$ usually does not satisfy this condition. □

Example 5. Consider the relative entropy loss, and choose $c = \eta = 1$. Applying Theorem 2 then gives the bound shown in [6] and [3]. After simple manipulations we get $\Delta(\mathbf{w}_t, \mathbf{x}_t, 0) = -\ln(1-p)$ and $\Delta(\mathbf{w}_t, \mathbf{x}_t, 1) = -\ln p$, where $p = \sum_i v_{t,i} x_{t,i}$. Hence, $L_0^{-1}(\Delta(\mathbf{w}_t, \mathbf{x}_t, 0)) = L_1^{-1}(\Delta(\mathbf{w}_t, \mathbf{x}_t, 1)) = p$, and $\hat{y}_t = p$ is the only prediction for which (4.1) holds with this choice of c and η. □

Example 6. For all $\eta > 0$, the absolute loss is known to be (c, η)-realizable when $c = (2 \ln \frac{2}{1+e^{-\eta}})^{-1}$ [3]. Choosing a value η that makes the loss bound given by (3.9) as low as possible is discussed in [4]. Here we just cite the most basic result. Assume that there is a known upper bound K for the total loss of the best expert, i.e., it is known that $\sum_{t=1}^{\ell} |y_t - x_{t,i}| \leq K$ holds for some i. If all the initial weights $w_{1,i}$ are chosen to be equal, and η is taken to be $-\ln g(K/\ln N)$ where $g(z) = 1 - 2(\sqrt{1+z} - 1)/z$, the Generic Algorithm for absolute loss satisfies

$$\sum_{t=1}^{\ell} |y_t - \hat{y}_t| \leq K + \sqrt{K \ln N} + \frac{\log_2 N}{2} .$$

□

5 Continuous-valued outcomes

We now show that under certain assumptions, the Generic Algorithm 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 6. We also consider the more general situation where the values $x_{t,i}$ and y_t are not in the range $[0, 1]$.

Lemma 7. *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*

$$\frac{\partial^2 G(y, a, b)}{\partial y^2} + \left(\frac{\partial G(y, a, b)}{\partial y} \right)^2 \geq 0 . \tag{5.1}$$

If (3.2) holds for binary outcomes $y_t \in \{0,1\}$, then it holds for all outcomes $y_t \in [0,1]$.

Proof sketch. We write (3.2) as $\exp((L(y_t,\hat{y}_t) - \Delta(\mathbf{w}_t,\mathbf{x}_t,y_t))/c) \le 1$. Inequality (5.1) implies that the second derivative of the left-hand side with respect to y_t is nonnegative, so the worst case is at $y_t \in \{0,1\}$. □

Example 8. For the square loss, as the second derivative of L_{sq} is constant, the second derivative of the function G of Lemma 7 is 0 whenever $c = 1/\eta$, and hence (5.1) trivially holds.

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 2 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, if we choose $\eta = 2$ and the initial weights to be equal,

$$\sum_{i=1}^{\ell}\left(\frac{y_t - \hat{y}_t}{r_t}\right)^2 \le \sum_{i=1}^{\ell}\left(\frac{y_t - x_{t,i}}{r_t}\right)^2 + \frac{\ln N}{2} \tag{5.2}$$

for all i. We can consider (5.2) as giving a loss bound similar to (3.9), 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 (5.2) is equivalent with

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

In the full paper we show that if the range of y_t is not bounded, loss bounds of the above form cannot be attained. □

Example 9. Consider the relative entropy loss. We have $\partial L_{\mathrm{ent}}(y,z)/\partial y = \ln y - \ln(1-y) - \ln z + \ln(1-z)$, so the second derivative $\partial^2 L_{\mathrm{ent}}(y,z)/\partial y^2 = 1/y + 1/(1-y)$ does not depend on z. Hence, if $c = 1/\eta$, the second derivative of the function G of Lemma 7 is 0, and (5.1) holds. □

Since the absolute loss L_{abs} does not even have a first derivative everywhere, the technique of Lemma 7 does not give any results for this loss function. In the next section we devise a new algorithm particularly for this problem.

6 The Vee Algorithm

The Vee Algorithm works for the absolute loss when the outcome is continuous. In choosing the prediction it is now necessary to explicitly also consider other outcomes than just $y = 0$ and $y = 1$.

Algorithm 10. (The Vee Algorithm) As Algorithm 3, except that we have fixed the loss function to be the absolute loss, the loss coefficient to be $c = (2 \ln \frac{2}{1+e^{-\eta}})^{-1}$, and predicting is done as follows:

Prediction: On receiving the tth input $\mathbf{x}_t$, let $Y = \{ 0, 1, x_{t,1}, \ldots, x_{t,N} \}$. Predict with any value $\hat{y}_t$ that satisfies the condition

$$\max_{y \in Y} \{ y - \Delta(\mathbf{w}_t, \mathbf{x}_t, y) \} \leq \hat{y}_t \leq \min_{y \in Y} \{ y + \Delta(\mathbf{w}_t, \mathbf{x}_t, y) \} \quad , \qquad (6.1)$$

where

$$\Delta(\mathbf{w}_t, \mathbf{x}_t, y) = -\frac{\ln(\sum_{i=1}^{N} v_{t,i} e^{-\eta |y - x_{t,i}|})}{2 \ln \frac{2}{1+e^{-\eta}}} \quad .$$

We see in Lemma 11 that there always is a prediction $\hat{y}_t$ that satisfies (6.1) and that (6.1) implies $|y - \hat{y}_t| \leq \Delta(\mathbf{w}_t, \mathbf{x}_t, y)$ for *all* $y \in [0, 1]$. Hence, Theorem 2 now gives for continuous outcomes $y_t \in [0, 1]$ the bound

$$\sum_{t=1}^{\ell} |y_t - \hat{y}_t| \leq \frac{-\ln \frac{w_{1,i}}{W_1} + \eta \sum_{t=1}^{\ell} |y_t - x_{t,i}|}{2 \ln \frac{2}{1+e^{-\eta}}} \qquad (6.2)$$

that was previously obtained for binary outcomes $y_t \in \{ 0, 1 \}$. Note that if (6.2) 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 $\eta / (2 \ln \frac{2}{1+e^{-\eta}})$ that appears in front of $|y_t - x_{t,i}|$ in (6.2) is greater than 1. Again, the parameter η can be tuned as mentioned in Example 6, and the scaling method of Example 8 can be used if the values $x_{t,i}$ are not in the range $[0, 1]$.

For the absolute loss, (3.2) has a simple geometric interpretation. Figure 1 gives an example of the graphs of the left-hand side $|y - \hat{y}|$ and the right-hand side $\Delta(\mathbf{w}, \mathbf{x}, y)$ as functions of y. The left-hand side of the inequality is given by a vee-curve with its tip at $(\hat{y}, 0)$; in Figure 1, we have $\hat{y} = 0.58$. For $\Delta(\mathbf{w}, \mathbf{x}, y)$ we have used $\mathbf{x} = (0.33, 0.83, 0.97, 0.52)$, and we see that the curve has a nondifferentiable tips at each value $y = x_i$. 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, around the value $y = 0.97$. Hence, the value of the maximum on the left-hand side of (6.1) is roughly 0.51. Similarly, the minimum on the right-hand side is about 0.63, since moving the tip

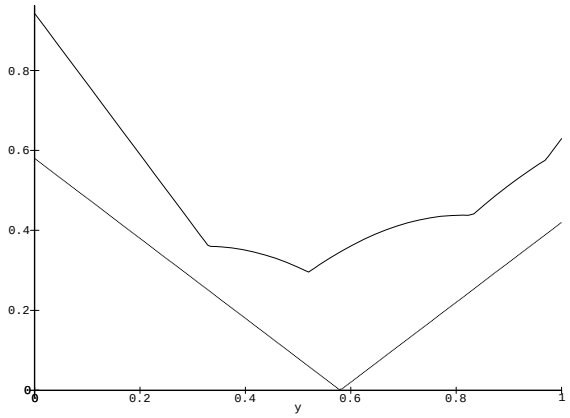

Figure 1. Example graphs of the functions Δ (above) and L_{abs} (below).

of the vee over this value would make its left arm intersect the Δ-curve around $y = 0.33$.

For the case of binary outcomes, the loss bound (6.2) was previously shown for a whole family of algorithms defined by a number of different prediction and update functions. In the continuous case we seem to have less freedom. For example the linearized version of update (3.6) given in [4] does not work any more in the continuous case. The Algorithm WMC of [2] does work for the continuous case. However, its worst case bounds have in the denominator $1 - e^{-\eta}$ instead of $2 \ln \frac{2}{1+e^{-\eta}}$, and hence they are slightly worse than the bounds given here.

We now show that a prediction that satisfies (6.1) always exists and satisfies the conditions of Theorem 2.

Lemma 11. *Let* $\mathbf{w}_t \in [0,1]^N$ *and* $\mathbf{x}_t \in [0,1]^N$, *and let* $\eta > 0$. *Then prediction* $\hat{y}_t$ *that satisfies (6.1) exists. Further, (6.1) implies* $|y - \hat{y}_t| \leq \Delta(\mathbf{w}_t, \mathbf{x}_t, y)$ *for all* $y \in [0,1]$.

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

$$y - \Delta(\mathbf{w}_t, \mathbf{x}_t, y) \leq z + \Delta(\mathbf{w}_t, \mathbf{x}_t, z) \tag{6.3}$$

holds for all y and z. Define

$$g(\mathbf{v}, \mathbf{x}, y, z) \quad = \quad \sum_{i=1}^{N} \sum_{j=1}^{N} v_i v_j \exp\left(-\eta(|y - x_i| + |z - x_j|\right)$$

$$+(y-z)2\ln(2/(1+e^{-\eta}))\big) \quad . \tag{6.4}$$

Then (6.3) is equivalent to $g(\mathbf{v}_t, \mathbf{x}_t, y, z) \leq 1$. The second derivative $\partial^2 g(\mathbf{v}, \mathbf{x}, y, z)/\partial x_i^2$ is defined and positive if $x_i \notin \{0, y, z, 1\}$. Thus it suffices to show $g(\mathbf{v}, \mathbf{x}, y, z) \leq 1$ for $N = 4$ and $\mathbf{x} = \mathbf{x}_a = (0, y, z, 1)$. In this restricted case the second derivative $\partial^2 g(\mathbf{v}, \mathbf{x}_a, y, z)/\partial z^2$ is positive if $z \notin \{0, y, 1\}$. Furthermore, (6.3) trivially holds if $z \geq y$. Thus it suffices to show (6.3) for $z = 0$, $y > 0$ and $\mathbf{x} = \mathbf{x}_b = (0, y, 0, 1)$. Finally, since the second derivative $\partial^2 g(\mathbf{v}, \mathbf{x}_b, y, 0)/\partial y^2$ is positive, we are left with the case $z = 0$, $y = 1$ and $\mathbf{x} \in \{0, 1\}^N$. In this case, the original inequality (6.3) can be rewritten as

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

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

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

Bibliography

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

2. Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. In *Proc. 30th Annual Symposium on Foundations of Computer Science*, pages 256–261, 1989.

3. Volodimir G. Vovk. Aggregating strategies. In *Proc. 3rd Annual Workshop on Computational Learning Theory*, pages 371–383. Morgan Kaufmann, 1990.

4. Nicolò Cesa-Bianchi, Yoav Freund, David P. Helmbold, David Haussler, Robert E. Schapire and Manfred K. Warmuth. How to use expert advice. In *Proc. 25th Annual ACM Symposium on Theory of Computing*, pages 382–391, 1993.

5. Nicolò Cesa-Bianchi, Yoav Freund, David P. Helmbold and Manfred K. Warmuth. On-line prediction and conversion strategies. In *Proc. European Conference on Computational Learning Theory*, 1993.

6. Alfredo DeSantis, George Markowski and Mark N. Wegman. Learning probabilistic prediction functions. In *Proc. 1988 Workshop on Computational Learning Theory*, pages 312–328. Morgan Kaufmann, 1988.