Journal of Machine Learning Research 1 (2001) 281-309 Submitted 05/01; Published 09/01

Tracking the Best Linear Predictor*

Mark Herbster M.HERBSTER@CS.UCL.AC.UK
Department of Computer Science
University College London
Gower Street
London, WC1E 6BT, UK

Manfred K. Warmuth MANFRED@CS.UCSC.EDU
Department of Computer Science
University of California at Santa Cruz
Baskin School of Engineering
Santa Cruz, CA 95064, USA

Editor: Peter Bartlett

Abstract

In most on-line learning research the total on-line loss of the algorithm is compared to the total loss of the best off-line predictor $\boldsymbol{u}$ from a comparison class of predictors. We call such bounds *static bounds*. The interesting feature of these bounds is that they hold for an arbitrary sequence of examples. Recently some work has been done where the predictor $\boldsymbol{u}_t$ at each trial t is allowed to change with time, and the total on-line loss of the algorithm is compared to the sum of the losses of $\boldsymbol{u}_t$ at each trial plus the total "cost" for shifting to successive predictors. This is to model situations in which the examples change over time, and different predictors from the comparison class are best for different segments of the sequence of examples. We call such bounds *shifting bounds*. They hold for arbitrary sequences of examples and arbitrary sequences of predictors.

Naturally shifting bounds are much harder to prove. The only known bounds are for the case when the comparison class consists of a sequences of experts or boolean disjunctions. In this paper we develop the methodology for lifting known static bounds to the shifting case. In particular we obtain bounds when the comparison class consists of linear neurons (linear combinations of experts). Our essential technique is to *project* the hypothesis of the static algorithm at the end of each trial into a suitably chosen convex region. This keeps the hypothesis of the algorithm well-behaved and the static bounds can be converted to shifting bounds.

Keywords: on-line learning, amortized analysis, shifting, switching, bregman divergence, projection

1. Introduction

Consider the following by now standard on-line learning model which is a generalization of a model introduced by Littlestone (1989; 1988).

*. The authors were supported by the NSF grants CCR 9700201 and CCR 9821087. Mark Herbster was also supported by ESPRC grant GR/M15972. An extended abstract appeared in (Herbster and Warmuth, 1998b).

Learning proceeds in trials $t = 1, 2, \ldots, \ell$. The algorithm maintains a parameter vector (hypothesis), denoted by $\boldsymbol{w}_t \in \Re^n$. In each trial the algorithm receives an *instance* $\mathbf{x}_t$. It then produces some action or a prediction for $\mathbf{x}_t$ based on $\boldsymbol{w}_t$. Finally, the algorithm receives an *outcome* y_t and incurs a loss describing how well the hypothesis $\boldsymbol{w}_t$ "performed" on the *example* $(\mathbf{x}_t, y_t)$. This loss is some non-negative function $L(\boldsymbol{w}_t, (\mathbf{x}_t, y_t))$.

For example, the algorithm might predict with $\sigma(\boldsymbol{w}_t \cdot \mathbf{x}_t)$, where σ is the logistic function, and the loss might be the square loss, i.e., $\frac{1}{2}(\sigma(\boldsymbol{w}_t \cdot \mathbf{x}_t) - y_t)^2$. Thus in this example setup the weight vector $\boldsymbol{w}$ represents a hypothesis $h_{\boldsymbol{w}}(\boldsymbol{x}) = \sigma(\boldsymbol{w} \cdot \boldsymbol{x})$. By choosing a different function σ, the hypotheses class of the algorithm changes. We abbreviate the loss in trial t by $L_t(\boldsymbol{w}_t)$ and the total loss of the algorithm A on a sequence $S = \langle (\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\ell, y_\ell) \rangle$ of examples by $L(A, S) = \sum_{t=1}^{\ell} L_t(\boldsymbol{w}_t)$.

In the methodology of worst-case loss bounds the total loss of the algorithm is expressed as a function of the total loss of any member in a comparison class of predictors, which is usually a subset of the hypotheses class. Such a predictor $\boldsymbol{u}$ also incurs a loss $L_t(\boldsymbol{u})$ in each trial and the total loss of $\boldsymbol{u}$ on the entire sequence is abbreviated as $L(\boldsymbol{u}, S) = \sum_{t=1}^{\ell} L_t(\boldsymbol{u})$. In the simplest case the bounds have the form

$$L(A, S) \leq c_1 L(\boldsymbol{u}, S) + c_2 \ \mathrm{size}(\boldsymbol{u}). \tag{1}$$

Here c_1 and c_2 are small constants, $\boldsymbol{u}$ is any predictor in the comparison class, and $\mathrm{size}(\boldsymbol{u})$ is a measurement of the size or complexity of $\boldsymbol{u}$. We call such bounds *static* bounds, because the predictor $\boldsymbol{u}$ does not change with time. Surprisingly, such bounds are achievable even when there are no probabilistic assumptions made on the sequence of examples (Littlestone, 1988, Mycielski, 1988, Vovk, 1990, Cesa-Bianchi et al., 1996, Haussler et al., 1998, Kivinen and Warmuth, 1997, Bylander, 1997, Helmbold et al., 1999). In this article we allow the predictor $\boldsymbol{u}$ to shift with time. For a sequence S of examples of length ℓ and a schedule of predictors $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle$ from the comparison class, we seek an upper bound of the form

$$L(A, S) \leq c_1 L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) + c_2 \ \mathrm{size}(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle) + \delta. \tag{2}$$

Here $L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S)$ is the loss of the schedule of predictors on S, i.e., $L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) = \sum_{t=1}^{\ell} L_t(\boldsymbol{u}_t)$; $\mathrm{size}(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle)$ measures, intuitively, the amount of shifting that occurs in the schedule and δ is a small additional term. We call such bounds *shifting* bounds. In this article our bounds use the following measure of $\mathrm{size}(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle)$:

$$|\!|\!| \langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle |\!|\!|_p = \sum_{t=1}^{\ell-1} \|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_p,$$

where $\|\cdot\|_p$ is a p-norm. This is in accord with other work (Cesa-Bianchi et al., 1996, Kivinen and Warmuth, 1997, Grove et al., 2001, Gentile and Littlestone, 1999) on worst case loss bounds, where the static bound also grows with a p-norm of the predictor.

A preliminary shifting result was developed by Littlestone and Warmuth (1994) in the mistake counting model. The new shifting bounds presented in this article build on previous work of the authors (Herbster and Warmuth, 1998a) where the loss of the algorithm was compared against the loss of the best shifting expert (see also Vovk 1997) or the best shifting disjunction (Auer and Warmuth, 1998).

The work on shifting experts has been applied to predicting disk idle times (Helmbold et al., 2000), and load balancing problems (Blum and Burch, 2000), as well as predicting TCP packet inter-arrival times (Scott, 1998). Vovk (1997) developed a quasi-probabilistic interpretation of the shifting expert algorithms by Herbster and Warmuth (1998a). Kalai et al. (1999) and Singer (1998) significantly expanded the shifting expert algorithms to quasi-probabilistic methods for combining language models and managing portfolios, respectively.

The rest of this article is outlined as follows. In Section 2, we give a general overview of how a certain class of on-line algorithms with static bounds may be transformed to algorithms with shifting bounds. In order to obtain a shifting bound we constrain the hypothesis of the algorithm to a suitably chosen convex region. In Section 3 we discuss how constraints are chosen to ensure good shifting bounds. In order to maintain the constraints, we *project* the hypothesis into the convex region. In Section 4, we discuss how to compute these projections. Finally, in Section 5 we give the formal bounds for our technique.

2. Shifting bounds for General Additive Regression Algorithms

We focus on a class of Algorithms, called *General Additive Regression Algorithms* (Jagota and Warmuth, 1998, Kivinen and Warmuth, 2001), that is characterized by a strictly convex function F. The function F is used to define a Bregman divergence $D_F(\boldsymbol{u},\boldsymbol{v})$ (Bregman, 1967, Censor and Lent, 1981, Csiszar, 1991). Bregman used these divergences in convex programming. In the context of on-line learning, various formulations of these divergence functions were first used in (Auer et al., 1995, Kivinen and Warmuth, 2001, Grove et al., 2001, Jagota and Warmuth, 1998, Azoury and Warmuth, 2001). At this point we define Bregman divergence $D_F(\boldsymbol{u},\boldsymbol{v})$ without fully listing the technical conditions on F specified in Definition 20 of the Appendix. Given a strictly convex differentiable function $F : E \rightarrow \Re$, where $E \subseteq \Re^n$ is a closed convex set and $\mathrm{ri}\, E$ denotes the relative interior[1] of E, the Bregman divergence $D_F : E \times \mathrm{ri}\, E \rightarrow [0,\infty)$ is defined as

$$D_F(\boldsymbol{u},\boldsymbol{w}) = F(\boldsymbol{u}) - F(\boldsymbol{w}) - (\boldsymbol{u}-\boldsymbol{w}) \cdot \nabla F(\boldsymbol{w}). \tag{3}$$

The Bregman divergence $D_F(\boldsymbol{u},\boldsymbol{w})$ is an important tool in this article. The main update (see (4)) of a General Additive Regression Algorithm is motivated (Kivinen and Warmuth, 1997) by such a divergence. The static loss bounds for these algorithms are proven with an amortized analysis (Cesa-Bianchi et al., 1996, Kivinen and Warmuth, 1997, Helmbold et al., 1999, Bylander, 1997, Kivinen and Warmuth, 2001) using a D_F divergence as a potential function. As part of the new methodology of this article we "project" (see Definition 1) via a D_F divergence to define a new update. This new update, when used in conjunction with the General Additive Regression Algorithm, allows shifting bounds to be proven.

In this article we focus on three General Additive Regression algorithms: the GD Algorithm (Cesa-Bianchi et al., 1996), the Un-normalized Exponentiated Gradient (EGU) Algorithm, and the Normalized Exponentiated Gradient (EG) Algorithm (Kivinen and Warmuth, 1997). The convex functions corresponding to these algorithms are denoted as $\mathrm{sq}(\boldsymbol{w})$, $\mathrm{ne}(\boldsymbol{w})$, and $\overline{\mathrm{ne}}(\boldsymbol{w})$, respectively. The convex function $\mathrm{sq}(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^n {w_i}^2$ with domain $\Re^n$ leads to the divergence $D_{\mathrm{sq}}(\boldsymbol{u},\boldsymbol{w}) = \sum_{i=1}^n \frac{1}{2}(u_i - w_i)^2$, which is half the

1. Intuitively, the relative interior of a set E corresponds to the "inside" of a set. For example, $\mathrm{ri}[0,1] = (0,1)$. The precise definitions are given in (Rockafellar, 1970).

squared Euclidean distance. The convex function $\mathrm{ne}(\boldsymbol{w}) = \sum_{i=1}^{n} w_i \ln w_i - w_i$ with domain $[0,\infty)^n$ leads to the un-normalized relative entropy as the divergence function $D_{\mathrm{ne}}(\boldsymbol{u},\boldsymbol{w}) = \sum_{i=1}^{n} u_i \ln \frac{u_i}{w_i} + w_i - u_i$. The convex function $\overline{\mathrm{ne}}(\boldsymbol{w}) = \sum_{i=1}^{n} w_i \ln w_i - w_i$ with domain $\mathcal{P}_n = \{\boldsymbol{w} : \sum_{i=1}^{n} w_i = 1 \text{ and } w_i \geq 0\}$ gives rise to the relative entropy as the divergence function $D_{\overline{\mathrm{ne}}}(\boldsymbol{u},\boldsymbol{w}) = \sum_{i=1}^{n} u_i \ln \frac{u_i}{w_i}$. This is summarized in Figure 1.

For any convex function F and loss function L, the General Additive Regression Algorithm uses the following update (Kivinen and Warmuth, 1997, Helmbold et al., 1997, 1998):

$$\boldsymbol{w}_{t+1} = \operatorname*{arg\,min}_{\boldsymbol{w} \in E} [D_F(\boldsymbol{w},\boldsymbol{w}_t) + \eta(L(\boldsymbol{w}_t \cdot \mathbf{x}_t, y_t) + \nabla_{\boldsymbol{v}} L(\boldsymbol{v} \cdot \mathbf{x}_t, y_t)|_{\boldsymbol{v}=\boldsymbol{w}_t} \cdot (\boldsymbol{w} - \boldsymbol{w}_t))]. \tag{4}$$

We call this the *general gradient descent update.* It is an approximation to the following more difficult to compute update

$$\boldsymbol{w}_{t+1} = \operatorname*{arg\,min}_{\boldsymbol{w} \in E} D_F(\boldsymbol{w},\boldsymbol{w}_t) + \eta L(\boldsymbol{w} \cdot \mathbf{x}_t, y_t),$$

which is motivated in (Kivinen and Warmuth, 1997). The explicit solutions to (4) for the convex functions $\mathrm{sq}(\boldsymbol{w})$, $\mathrm{ne}(\boldsymbol{w})$, and $\overline{\mathrm{ne}}(\boldsymbol{w})$ are summarized in Figure 1.

The update (4) may also be expressed in the simpler form

$$\boldsymbol{w}_{t+1} = f^{-1}\left(f(\boldsymbol{w}_t) - \eta \nabla L_t(\boldsymbol{w}_t)\right), \tag{5}$$

where $f = \nabla F$ for the GD and EGU algorithms. For the GD algorithm f is the identity function, and hence in this case the GD update is the standard online gradient descent update. In (Kivinen and Warmuth, 2001) the update (4) is also shown to be expressible in the form (5) for the EG Algorithm by choosing an alternate convex function F which is closely related to $\overline{\mathrm{ne}}$, except that it is expressed as a function of $n-1$ dimensions.

We now outline how worst-case loss bounds are obtained in the static case. At the center of all the static proofs (Kivinen and Warmuth, 1997, Helmbold et al., 1999, Bylander, 1997, Kivinen and Warmuth, 2001) for the General Additive Regression Algorithms lies the following type of inequality:

$$a(\eta)\ L_t(\boldsymbol{w}_t) - b(\eta)\ L_t(\boldsymbol{u}) \leq D_F(\boldsymbol{u},\boldsymbol{w}_t) - D_F(\boldsymbol{u},\boldsymbol{w}_{t+1}). \tag{6}$$

Here a and b are non-negative functions that depend on the learning rate η and upper bounds on the norms of the instances. Note that $D_F(\boldsymbol{u},\boldsymbol{w}_t) - D_F(\boldsymbol{u},\boldsymbol{w}_{t+1})$ may be seen as the progress towards the off-line weight vector $\boldsymbol{u}$. Since this inequality holds for each trial, we can sum Equation (6) over trials. The progresses towards $\boldsymbol{u}$ that appear on the right-hand sides form a telescoping sum, and only the first and last terms survive:

$$a(\eta)L(A,S) - b(\eta)L(\boldsymbol{u},S) \leq D_F(\boldsymbol{u},\boldsymbol{w}_1) - D_F(\boldsymbol{u},\boldsymbol{w}_{\ell+1}). \tag{7}$$

The last term, $D_F(\boldsymbol{u},\boldsymbol{w}_{\ell+1})$, can be dropped, since the divergence is non-negative and we are forming an upper bound. The first term, $D_F(\boldsymbol{u},\boldsymbol{w}_1)$, plays the role of the size measure of $\boldsymbol{u}$. Rearranging the above and picking a good choice for η gives bounds of the form (1).

The new approach (which allows us to prove bounds for the shifting case) starts with picking a suitable convex region Γ. After the usual update, the hypothesis vector is projected into this convex region. This gives us two benefits. First, we can show that the General Additive Regression Algorithms may be modified to maintain convex constraints on the hypothesis vector. Second, we show that shifting bounds may readily be obtained. The convex region plays a key role in bounding the size of the schedule of predictors.

Definition and properties of the Projection update

The divergence D_F may also be used to define a type of *projection* (Bregman, 1967). As is the case with Euclidean distance, the projection of a point onto a hyperplane allows a Pythagorean-type theorem (Bregman, 1967) to be proven. The Generalized Pythagorean Theorem 2 will be the key to proving shifting bounds.

The projection of a point x onto a closed convex set Γ is simply the point in Γ closest to x w.r.t. the divergence D_F.

Definition 1 *The projection of a point* $\boldsymbol{w} \in \mathrm{ri}\, E$ *w.r.t. divergence* $D_F : E \times \mathrm{ri}\, E \rightarrow [0, \infty)$*, where* $E \subseteq \Re^n$*, onto a closed convex set* Γ *is defined by:*

$$P_{(\Gamma, D_F)}(\boldsymbol{w}) = \operatorname*{arg\,min}_{\boldsymbol{u} \in \Gamma \cap E} D_F(\boldsymbol{u}, \boldsymbol{w}). \tag{8}$$

In the Appendix we show that the projection exists and is unique, provided that F is strictly convex and differentiable (see also Bregman 1967 and Csiszar 1991).

Recall the standard Pythagorean Theorem for the divergence associated with the convex function $\mathrm{sq}(\boldsymbol{w})$:

$$\|\boldsymbol{u} - \boldsymbol{w}\|_2^2 = \|\boldsymbol{u} - \boldsymbol{p}\|_2^2 + \|\boldsymbol{p} - \boldsymbol{w}\|_2^2, \text{ when } (\boldsymbol{u} - \boldsymbol{p}) \perp (\boldsymbol{p} - \boldsymbol{w}).$$

The orthogonality condition can be rewritten as $\boldsymbol{p}$ being a projection of $\boldsymbol{w}$ onto a hyperplane that contains $\boldsymbol{u}$. Here projections are w.r.t. the (squared) Euclidean distance. Surprisingly, a generalization of the Pythagorean Theorem holds for projections w.r.t. a large class of D_F divergences (see the Appendix for a list of technical restrictions on F).

Theorem 2 (Pythagoras generalized (Bregman, 1967)) *Given a divergence* $D_F : E \times \mathrm{ri}\, E \rightarrow [0, \infty)$*, a closed convex set* $\Gamma \subseteq \Re^n$ *such that* $\Gamma \cap \mathrm{ri}\, E \neq \emptyset$*, and points* $\boldsymbol{w} \in \mathrm{ri}\, E$ *and* $\boldsymbol{u} \in \Gamma$*, then*

$$D_F(\boldsymbol{u}, \boldsymbol{w}) \geq D_F(\boldsymbol{u}, P_{(\Gamma, D_F)}(\boldsymbol{w})) + D_F(P_{(\Gamma, D_F)}(\boldsymbol{w}), \boldsymbol{w}). \tag{9}$$

In the special case where Γ *is an affine set the above becomes an equality.*

A hyperplane is an example of an affine set. Any affine set in $\Re^n$ may be represented as an intersection of k hyperplanes $\cap_{i=1}^{k} \{\boldsymbol{v} : \boldsymbol{v} \cdot \boldsymbol{x}_i = y_i\}$ for $k \leq n$. Formally a set A is affine if for any $\boldsymbol{v}, \boldsymbol{w} \in A$ and every $\lambda \in \Re$ the point $\lambda \boldsymbol{v} + (1 - \lambda) \boldsymbol{w}$ is in A.

Note that the above Inequality (9) is opposite to the triangle inequality that holds for a (distance) metric. For this reason, we call D_F a divergence instead of a distance.

The above theorem has been proven many times under a variety of assumptions, e.g. (Bregman, 1967, Csiszar, 1991, Jones and Byrne, 1990, Bauschke and Borwein, 1997). For the sake of completeness we provide a streamlined proof in the Appendix. The following corollary of the Generalized Pythagorean Theorem will be used repeatedly in the analyses of our Algorithms.

Corollary 3 *Given a divergence $D_F : E \times \operatorname{ri} E \to [0,\infty)$, a closed convex set Γ such that $\Gamma \cap \operatorname{ri} E \neq \emptyset$, and points $\boldsymbol{w} \in \operatorname{ri} E$ and $\boldsymbol{u} \in \Gamma$, then*

$$D_F(\boldsymbol{u},\boldsymbol{w}) - D_F(\boldsymbol{u}, P_{(\Gamma,D_F)}(\boldsymbol{w})) \geq 0. \tag{10}$$

Using the above definition of projection, we now introduce our modification of the General Additive Regression Algorithms:

`General gradient descent update:`

$$\boldsymbol{w}_t^m = \operatorname*{arg\,min}_{\boldsymbol{w}\in E} \left[D_F(\boldsymbol{w},\boldsymbol{w}_t) + \eta(L(\boldsymbol{w}_t \cdot \mathbf{x}_t, y_t) + \nabla_{\boldsymbol{v}} L(\boldsymbol{v}\cdot\mathbf{x}_t, y_t)|_{\boldsymbol{v}=\boldsymbol{w}_t} \cdot (\boldsymbol{w} - \boldsymbol{w}_t))\right]$$

`Projection update:`

$$\boldsymbol{w}_{t+1} = P_{(\Gamma,D_F)}(\boldsymbol{w}_t^m) \tag{11}$$

Since we now have two updates, we will refer to the intermediate weight vector following the generalized gradient update as $\boldsymbol{w}_t^m$. We call this algorithm the Constrained General Additive Regression Algorithm. We use C-GD(η,Γ), C-EGU(η,Γ), and C-EG(η,Γ) for the example algorithms of this article.

In order to obtain shifting bounds for C-GD(η,Γ), C-EGU(η,Γ), and C-EG(η,Γ), we need to choose a constraint set for each of the algorithms. The constraint sets $\Gamma_{\mathrm{sq},2,\gamma}$, $\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}$, and $\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}$ (see Figure 1) are chosen to obtain shifting bounds for GD, EGU, and EG, respectively. The three parameters of a constraint set $\Gamma_{F,p,\gamma}$ describe the shape of the constraint set. The first parameter refers to the convex function F, which also determines the main update (see 11) of the algorithm. The second parameter $p \in [1,\infty]$ is associated with the norm of the instances of the loss bounds proven for the algorithm. The third parameter $\gamma \in [0,\infty)$ essentially "scales" the constraint set.

In (Kivinen and Warmuth, 1997) it is observed that the bounds of GD depend on the 2-norm of both the instances $\mathbf{x}_t$ and the predictor $\boldsymbol{u}$, while in EGU (as well as EG) the bounds depend on the ∞-norm of the instances $\mathbf{x}_t$ and the 1-norm of the predictor $\boldsymbol{u}$. In the shifting versions of these algorithms the constraint set $\Gamma_{\mathrm{sq},2,\gamma}$ is the level set of the 2-norm, and the constraint sets $\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}$ and $\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}$ are shifted level sets of the ∞-norm. Our worst-case loss bounds for C-GD$(\eta,\Gamma_{\mathrm{sq},2,\gamma})$ thus depend on $\|\langle \boldsymbol{u}_1,\ldots,\boldsymbol{u}_\ell\rangle\|_2$ and $\|\mathbf{x}_t\|_2$, and the bounds for C-EGU$(\eta,\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$ and C-EG$(\eta,\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}})$ depend on $\|\langle \boldsymbol{u}_1,\ldots,\boldsymbol{u}_\ell\rangle\|_1$ and $\|\mathbf{x}_t\|_\infty$. In Section 3 we discuss the definition and choice of constraint set in more detail.

Calculating an arbitrary projection $P_{(\Gamma,D_F)}(\boldsymbol{w})$ may be computationally expensive. However, the projections $P_{(\Gamma_{\mathrm{sq},2,\gamma},D_{\mathrm{sq}})}(\boldsymbol{w})$, $P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}},D_{\mathrm{ne}})}(\boldsymbol{w})$, and $P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}},D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$ are computable (see Figure 1) in $O(n)$ time. In particular, the projection update $P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}},D_{\mathrm{ne}})}(\boldsymbol{w})$ corresponds to simple weight clipping. Similar updates were used in the Fixed-share Algorithm (Herbster and Warmuth, 1998a), the WML Algorithm (Littlestone and Warmuth, 1994), and the algorithms for tracking disjunctions (Auer and Warmuth, 1998). In Section 4 we discuss the computations of the projections in more detail.

<table>
<tr><td>C-GD$(\eta, \Gamma_{\mathrm{sq},2,\gamma})$</td><td>C-EGU$(\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$</td><td>C-EG$(\eta, \Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}})$</td></tr>
<tr><td colspan="3">Convex Function $F(\boldsymbol{w})$</td></tr>
<tr><td>$\mathrm{sq}(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n} {w_i}^2$, $\boldsymbol{w} \in \Re^n$</td><td>$\mathrm{ne}(\boldsymbol{w}) = \sum_{i=1}^{n} w_i \ln w_i - w_i$, $\boldsymbol{w} \in [0,\infty)^n$</td><td>$\overline{\mathrm{ne}}(\boldsymbol{w}) = \sum_{i=1}^{n} w_i \ln w_i - w_i$, $\boldsymbol{w} \in \mathcal{P}_n$</td></tr>
<tr><td colspan="3">Divergence Function $D_F(\boldsymbol{u}, \boldsymbol{w})$</td></tr>
<tr><td>$D_{\mathrm{sq}}(\boldsymbol{u}, \boldsymbol{w}) = \sum_{i=1}^{n} \frac{1}{2}(u_i - w_i)^2$</td><td>$D_{\mathrm{ne}}(\boldsymbol{u}, \boldsymbol{w}) = \sum_{i=1}^{n} u_i \ln \frac{u_i}{w_i} + w_i - u_i$</td><td>$D_{\overline{\mathrm{ne}}}(\boldsymbol{u}, \boldsymbol{w}) = \sum_{i=1}^{n} u_i \ln \frac{u_i}{w_i}$</td></tr>
<tr><td colspan="3">General Gradient Descent Update</td></tr>
<tr><td>$$w^m_{t+1,i} = w_{t,i} - \eta \nabla_{t,i} \quad (12)$$</td><td>$$w^m_{t+1,i} = w_{t,i} e^{-\eta \nabla_{t,i}} \quad (13)$$</td><td>$$w^m_{t+1,i} = \frac{w_{t,i} e^{-\eta \nabla_{t,i}}}{\sum_{i=1}^{n} w_{t,i} e^{-\eta \nabla_{t,i}}} \quad (14)$$</td></tr>
<tr><td colspan="3">Constraint Set $\Gamma_{F,p,\gamma}$</td></tr>
<tr><td>$\Gamma_{\mathrm{sq},2,\gamma} = \{\boldsymbol{u} : \|\boldsymbol{u}\|_2 \le \gamma\}$</td><td>$\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}} = [\frac{\alpha}{n}, \frac{n}{\alpha}]^n$</td><td>$\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}} = [\frac{\alpha}{n}, 1]^n \cap \mathcal{P}_n$</td></tr>
<tr><td colspan="3">Projection Update $\boldsymbol{w}_{t+1} = P_{(\Gamma, D_F)}(\boldsymbol{w}^m_t)$</td></tr>
<tr><td>$$\boldsymbol{w}_{t+1} = \begin{cases} \boldsymbol{w}^m_t & \boldsymbol{w}^m_t \in \Gamma_{\mathrm{sq},2,\gamma} \\ \gamma \frac{\boldsymbol{w}^m_t}{\|\boldsymbol{w}^m_t\|_2} & \boldsymbol{w}^m_t \notin \Gamma_{\mathrm{sq},2,\gamma} \end{cases} \quad (15)$$</td><td>$$w_{t+1,i} = \begin{cases} w^m_{t,i} & w^m_{t,i} \in [\frac{\alpha}{n}, \frac{n}{\alpha}] \\ \frac{\alpha}{n} & w^m_{t,i} < \frac{\alpha}{n} \\ \frac{n}{\alpha} & w^m_{t,i} > \frac{n}{\alpha} \end{cases} \quad (16)$$</td><td>See Figure 3.</td></tr>
</table>

Figure 1: Summary information for C-GD$(\eta, \Gamma_{\mathrm{sq},2,\gamma})$, C-EGU$(\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$, and C-EG$(\eta, \Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}})$. Here $\nabla_{t,i}$ is shorthand for $\frac{\partial L_t(\boldsymbol{w}_t)}{\partial w_{t,i}}$. In the case of the square loss, $\nabla_{t,i} = 2x_{t,i}(\boldsymbol{w}_t \cdot \mathbf{x}_t - y_t)$.

An overview of proof techniques based on projections

We have three results that follow almost immediately from properties of projections and the static bounds. First, we show that if the predictor $\boldsymbol{u}$ from the comparison class lies in the constraint set, the previous static loss bounds hold for the constraint algorithm. Second, we prove a shifting bound on a predictor schedule $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle$ whose predictors must all lie in Γ. Third, in Section 5.1, we extend the shifting bound so that the predictor schedule may contain predictors that lie outside of Γ.

For the first part recall that $\boldsymbol{w}_t$ is the weight vector at the start of a trial, and $\boldsymbol{w}_t^m$ the weight vector between the two updates. With this notation, Inequality (6) becomes

$$a(\eta)L_t(\boldsymbol{w}_t) - b(\eta)L_t(\boldsymbol{u}_t) \leq D_F(\boldsymbol{u}, \boldsymbol{w}_t) - D_F(\boldsymbol{u}, \boldsymbol{w}_t^m). \tag{17}$$

Note that this inequality no longer telescopes. By Corollary 3 we have

$$0 \leq D_F(\boldsymbol{u}, \boldsymbol{w}_t^m) - D_F(\boldsymbol{u}, \boldsymbol{w}_{t+1}),$$

provided that $\boldsymbol{u}$ lies in the constraint set. The sum of the two inequalities is again a telescoping inequality with the same functions a and b:

$$a(\eta)L_t(\boldsymbol{w}_t) - b(\eta)L_t(\boldsymbol{u}_t) \leq D_F(\boldsymbol{u}, \boldsymbol{w}_t) - D_F(\boldsymbol{u}, \boldsymbol{w}_{t+1}). \tag{19}$$

Thus the constraint algorithms have the same static bounds, provided that the predictor vector $\boldsymbol{u}$ lies in the constraint set.

For the second part we assume that all the predictors of predictor schedule $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle$ lie in the constraint set. Thus Equation (19) holds for each trial:

$$a(\eta)L_t(\boldsymbol{w}_t) - b(\eta)L_t(\boldsymbol{u}_t) \leq D_F(\boldsymbol{u}_t, \boldsymbol{w}_t) - D_F(\boldsymbol{u}_t, \boldsymbol{w}_{t+1}). \tag{20}$$

This again does not telescope. We fix this by adding

$$G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) := D_F(\boldsymbol{u}_t, \boldsymbol{w}_{t+1}) - D_F(\boldsymbol{u}_{t+1}, \boldsymbol{w}_{t+1}). \tag{21}$$

Intuitively, $G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1})$ measures the cost for shifting from $\boldsymbol{u}_t$ to $\boldsymbol{u}_{t+1}$. Since the sequence of predictors $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle$ is arbitrary (the shifting loss bound holds for all sequences of predictors), this cost may be positive or negative, i.e., $\boldsymbol{u}_{t+1}$ may be closer to or farther from the algorithm's current hypothesis $\boldsymbol{w}_{t+1}$. By adding (20) and (21) and by summing over all trials we get the following:

$$L(A,S) \leq \frac{b(\eta)}{a(\eta)} L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) + \frac{1}{a(\eta)} \left[D_F(\boldsymbol{u}_1, \boldsymbol{w}_1) - D_F(\boldsymbol{u}_{\ell+1}, \boldsymbol{w}_{\ell+1}) - \sum_{t=1}^{\ell} G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) \right].$$

By expanding $G_F(\boldsymbol{u}_\ell, \boldsymbol{u}_{\ell+1})$ the bound reduces to

$$L(A,S) \leq \frac{b(\eta)}{a(\eta)} L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) + \frac{1}{a(\eta)} \left[D_F(\boldsymbol{u}_1, \boldsymbol{w}_1) - D_F(\boldsymbol{u}_\ell, \boldsymbol{w}_{\ell+1}) - \sum_{t=1}^{\ell-1} G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) \right]. \tag{22}$$

This would be a good bound if $G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1})$ was lower bounded. Unfortunately this is not true for arbitrary $\boldsymbol{w}_{t+1}$. For example, for the potential function $sq(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n} w_i^2$,

$$G_{\mathrm{sq}}(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) = \frac{1}{2}\|\boldsymbol{u}_t\|_2^2 - \frac{1}{2}\|\boldsymbol{u}_{t+1}\|_2^2 - \boldsymbol{w}_{t+1} \cdot (\boldsymbol{u}_t - \boldsymbol{u}_{t+1}). \tag{23}$$

Observe that unless we have a bound on some norm of $\boldsymbol{w}_{t+1}$, the cost $G_{\mathrm{sq}}(\boldsymbol{u}_t, \boldsymbol{u}_{t+1})$ cannot be lower bounded solely in terms of $\boldsymbol{u}_t$ and $\boldsymbol{u}_{t+1}$. However, $\boldsymbol{w}_{t+1} \in \Gamma_{\mathrm{sq},2,\gamma}$ and thus by our choice of the constraint set $\Gamma_{\mathrm{sq},2,\gamma}$, $\|\boldsymbol{w}_{t+1}\|_2 \leq \gamma$. Through an application of Hölder's inequality, we have

$$G_{\mathrm{sq}}(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) \geq \frac{1}{2}\|\boldsymbol{u}_t\|_2^2 - \frac{1}{2}\|\boldsymbol{u}_{t+1}\|_2^2 - \gamma\|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_2, \tag{24}$$

and plugging into Equation (22) we obtain a reasonable upper bound of the form (2):

$$L(\text{C-GD}(\eta, \Gamma_{\mathrm{sq},2,\gamma}), S) \leq \frac{b(\eta)}{a(\eta)} L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) + \frac{1}{a(\eta)} \Big[D_{\mathrm{sq}}(\boldsymbol{u}_1, \boldsymbol{w}_1) - D_{\mathrm{sq}}(\boldsymbol{u}_\ell, \boldsymbol{w}_{\ell+1}) + \frac{1}{2}\|\boldsymbol{u}_\ell\|_2^2 - \frac{1}{2}\|\boldsymbol{u}_1\|_2^2 + \gamma \sum_{t=1}^{\ell-1} \|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_2 \Big]. \tag{25}$$

In this section, we have seen that a convex function F may be used to define a General Additive Regression Algorithm. Such an algorithm may then be transformed to an algorithm for which a shifting loss bound may be proven. The transformed algorithm contains an additional update, which projects the hypothesis vector of the algorithm onto a constraint set Γ. In the following sections we prove bounds based on the techniques outlined above.

3. Constraint sets for shifting

In this section, we develop a method for choosing Γ in terms of the convex function F. To obtain a shifting loss bound it is necessary to choose a constraint region that keeps the hypothesis vector $\boldsymbol{w}$ bounded. The constraint region is chosen so that we may bound the cost of shifting on trial t independently of the hypothesis vector $\boldsymbol{w}_t$. Below we recall (21) where we defined $G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1})$ to measure the cost of shifting from predictor $\boldsymbol{u}_t$ to predictor $\boldsymbol{u}_{t+1}$:

$$G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) = D_F(\boldsymbol{u}_t, \boldsymbol{w}_{t+1}) - D_F(\boldsymbol{u}_{t+1}, \boldsymbol{w}_{t+1}). \tag{26}$$

This cost must be bounded to obtain a shifting loss bound. Expanding the definition of G_F we have

$$G_F(\boldsymbol{u}_t, \boldsymbol{u}_{t+1}) = F(\boldsymbol{u}_t) - F(\boldsymbol{u}_{t+1}) - (\boldsymbol{u}_t - \boldsymbol{u}_{t+1}) \cdot \nabla F(\boldsymbol{w}_{t+1}). \tag{27}$$

We bound the last term with Hölder's inequality, which states that $\boldsymbol{a} \cdot \boldsymbol{b} \leq \|\boldsymbol{a}\|_p \|\boldsymbol{b}\|_{\frac{p}{p-1}}$ for $\boldsymbol{a}, \boldsymbol{b} \in \Re^n$ and $p \in [1, \infty]$. Hence the magnitude of the last term is now bounded by $\|\nabla F(\boldsymbol{w}_{t+1})\|_p \|\boldsymbol{u}_{t+1} - \boldsymbol{u}_t\|_{\frac{p}{p-1}}$. In the following definition we define a constraint set $\Gamma_{F,p,\gamma}$ that directly bounds $\|\nabla F(\boldsymbol{w}_{t+1})\|_p$.

Definition 4 *Given a convex function $F : E \to \Re$, parameters $p \in [1, \infty]$, and $\gamma \in [0, \infty)$ we define the $\Gamma_{F,p,\gamma}$ constraint set to be*

$$\Gamma_{F,p,\gamma} = E \cap \{\boldsymbol{w} : \|\nabla F(\boldsymbol{w})\|_p \leq \gamma\}. \tag{28}$$

The above constraint set is not necessarily convex. In order to define a unique projection to a set, that set must be convex. Therefore the primary qualification for a constraint set $\Gamma_{F,p,\gamma}$ to be useful is convexity. Now we consider the choice of the parameters p and γ for constraint sets $\Gamma_{\mathrm{sq},p,\gamma}$, $\Gamma_{\mathrm{ne},p,\gamma}$, and $\Gamma_{\overline{\mathrm{ne}},p,\gamma}$.

The constraint set $\Gamma_{\mathrm{sq},p,\gamma}$ (for constrained GD) is convex for parameters $p \in [1,\infty]$, and $\gamma \in [0,\infty)$. We fix $p=2$ and use origin-centered hyperspheres as our constraint sets:

$$\Gamma_{\mathrm{sq},2,\gamma} = \{\boldsymbol{w} : \|\boldsymbol{w}\|_2 \leq \gamma\}.$$

There are two reasons for choosing origin-centered hyperspheres. First, the projection $P_{(\Gamma_{\mathrm{sq},2,\gamma}, D_{\mathrm{sq}})}(\boldsymbol{w})$ is simple to compute (see Theorem 5): if $\boldsymbol{w}$ is in $\Gamma_{\mathrm{sq},2,\gamma}$ then the projection equals $\boldsymbol{w}$; otherwise we project by multiplying $\boldsymbol{w}$ by the scalar $\frac{\gamma}{\|\boldsymbol{w}\|_2}$ (see Equation (29)) so that it lies on the boundary of the hypersphere. Second, because the 2-norm is the only p-norm that may be identified with an inner product, it is the only $\Gamma_{\mathrm{sq},p,\gamma}$ constraint set that generalizes to arbitrary Hilbert spaces.

The GD Algorithm and its loss bound is shown in (Cesa-Bianchi et al., 1996) to hold not only for $\Re^n$ but also for arbitrary Hilbert spaces. If we choose constraint sets of the form $\{\boldsymbol{w} : \|\boldsymbol{w}\| \leq \gamma\}$ the projection is still well defined for Hilbert spaces. This is a benefit since the technique developed in (Aizerman et al., 1964) for the Perceptron Algorithm (Rosenblatt, 1958) and in (Boser et al., 1992) for the "optimal separating hyperplane" Algorithm (Vapnik and Chervonenkis, 1974), may be applied to constrained GD. This technique allows the transform of the comparison class from the set of linear predictors in $\Re^n$ to the set of all functions in some *reproducing kernel* Hilbert space (Aronszajn, 1950), e.g., polynomials of degree d, or linear combinations of Gaussians. The shifting bounds continue to hold for these expanded comparison classes.

The constraint sets[2] $\Gamma_{\mathrm{ne},p=\infty,\gamma}$ and $\Gamma_{\overline{\mathrm{ne}},p=\infty,\gamma}$, for constrained EGU and EG, respectively, are convex for all $\gamma \in [0,\infty)$. For each $p \in [1,\infty)$ there exists a γ such that the constraint set $\Gamma_{\mathrm{ne},p,\gamma}$ is nonconvex. The constraint set $\Gamma_{\mathrm{ne},\infty,\gamma}$ is the simple rectangular region $[e^{-\gamma}, e^{\gamma}]^n$ and the constraint set $\Gamma_{\overline{\mathrm{ne}},\infty,\gamma}$ is $[e^{-\gamma}, e^{\gamma}]^n \cap \mathcal{P}_n$.

In the papers (Herbster and Warmuth, 1998a, Vovk, 1997, Blum and Burch, 2000) the hypothesis vector was implicitly constrained to the region $\Gamma_{\overline{\mathrm{ne}},\infty,\gamma=\ln\frac{n}{\alpha}} = [\frac{\alpha}{n}, 1] \cap \mathcal{P}_n$. Thus to maintain continuity in notation with those papers we parameterize the negative entropy-based constraint sets by $\ln\frac{n}{\alpha}$ rather than γ. In (Herbster and Warmuth, 1998a) and (Vovk, 1997), the $\ln\frac{n}{\alpha}$ parameterization is shown to have a probabilistic interpretation.

4. Computing projections

The computation of the projection $P_{(\Gamma, D_F)}(\boldsymbol{w})$ (as defined in Definition 1) divides into two cases. For the first case, when $\boldsymbol{w}$ is contained in Γ, the projection of $\boldsymbol{w}$ to Γ is simply $\boldsymbol{w}$. This follows directly from the facts $D_F(\boldsymbol{u}, \boldsymbol{w}) \geq 0$ and $\boldsymbol{u} = \boldsymbol{w} \Leftrightarrow D_F(\boldsymbol{u}, \boldsymbol{w}) = 0$ (Proposition 21). For the second case, when $\boldsymbol{w}$ is not in Γ, computing the projection may be nontrivial. However, we show that the projections $P_{(\Gamma_{\mathrm{sq},2,\gamma}, D_{\mathrm{sq}})}(\boldsymbol{w})$, $P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}, D_{\mathrm{ne}})}(\boldsymbol{w})$, and $P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}, D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$ can be computed in $O(n)$ time.

The constraint set $\Gamma_{\mathrm{sq},2,\gamma}$ corresponds to an origin-centered sphere, and intuitively the projection of $\boldsymbol{w}$ when $\boldsymbol{w}$ is not on the sphere is simply to "radially scale" $\boldsymbol{w}$ so that it is on the boundary of $\Gamma_{\mathrm{sq},2,\gamma}$. The following proof confirms the intuition.

2. The use of the gradient in Definition 4 is slightly problematic for some convex functions F if the dimension of the affine hull of the domain of F is less than n. In particular, for $\overline{\mathrm{ne}}$ the gradient is only determined up to a constant, i.e., $\nabla\overline{\mathrm{ne}}(\boldsymbol{w}) = (\ln(\boldsymbol{w}) + c, \ldots, \ln(\boldsymbol{w}) + c)$ for all $c \in \Re$. For simplicity we set $c = 0$.

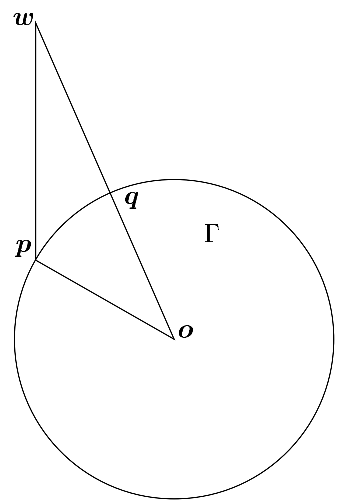

Figure 2: Illustration for the proof of Theorem 5

Theorem 5 *The projection* $P_{(\Gamma_{sq,2,\gamma}, D_{sq})}(\boldsymbol{w})$ *is computed by*

$$P_{(\Gamma_{sq,2,\gamma}, D_{sq})}(\boldsymbol{w}) = \begin{cases} \boldsymbol{w} & \boldsymbol{w} \in \Gamma_{sq,2,\gamma} \\ \frac{\gamma \boldsymbol{w}}{\|\boldsymbol{w}\|_2} & \boldsymbol{w} \notin \Gamma_{sq,2,\gamma} \end{cases} \tag{29}$$

Proof For this proof we abbreviate $P_{(\Gamma_{\mathrm{sq},2,\gamma}, D_{\mathrm{sq}})}(\boldsymbol{w})$ to $\boldsymbol{p}$, $\Gamma_{\mathrm{sq},2,\gamma}$ to Γ, $\frac{\gamma \boldsymbol{w}}{\|\boldsymbol{w}\|_2}$ to $\boldsymbol{q}$, and we let $\boldsymbol{o}$ denote the origin. Recall that $D_{\mathrm{sq}}(\boldsymbol{a}, \boldsymbol{b}) = \frac{1}{2}\|\boldsymbol{a} - \boldsymbol{b}\|_2^2$; for the sake of brevity we omit the factors of $\frac{1}{2}$ in this proof.

Assume $\boldsymbol{w} \notin \Gamma$. The point $\boldsymbol{p}$ is on the boundary of Γ, since if $\boldsymbol{p}$ is not on the boundary of Γ then there exists an $\epsilon > 0$ such that the point $\boldsymbol{p}' = \boldsymbol{p} + \epsilon(\boldsymbol{w} - \boldsymbol{p})$ is contained in Γ. But $\|\boldsymbol{p}' - \boldsymbol{w}\|_2^2 = (1-\epsilon)^2 \|\boldsymbol{p} - \boldsymbol{w}\|_2^2$ and this contradicts the assumption that $\boldsymbol{p}$ is the projection $\boldsymbol{w}$. Suppose $\boldsymbol{p} \neq \boldsymbol{q}$; then Figure 2 correctly illustrates the relations between $\boldsymbol{o}$, $\boldsymbol{p}$, $\boldsymbol{q}$, and $\boldsymbol{w}$, since $\|\boldsymbol{q}\|_2 = \gamma$ and $\boldsymbol{q}$ is a convex combination of $\boldsymbol{w}$ and $\boldsymbol{o}$. By the triangle inequality,

$$\|\boldsymbol{o} - \boldsymbol{p}\|_2 + \|\boldsymbol{p} - \boldsymbol{w}\|_2 > \|\boldsymbol{o} - \boldsymbol{w}\|_2 = \|\boldsymbol{o} - \boldsymbol{q}\|_2 + \|\boldsymbol{q} - \boldsymbol{w}\|_2.$$

Since $\|\boldsymbol{o} - \boldsymbol{p}\|_2 = \|\boldsymbol{o} - \boldsymbol{q}\|_2 = \gamma$, we have that $\|\boldsymbol{p} - \boldsymbol{w}\|_2 > \|\boldsymbol{q} - \boldsymbol{w}\|_2$. This implies that $D_{\mathrm{sq}}(\boldsymbol{p}, \boldsymbol{w}) > D_{\mathrm{sq}}(\boldsymbol{q}, \boldsymbol{w})$, which contradicts the assumption that $\boldsymbol{p}$ is the projection of $\boldsymbol{w}$. We conclude that $\boldsymbol{p} = \boldsymbol{q}$. ■

The constraint set $\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}$ corresponds to the box constraint $[\frac{\alpha}{n}, \frac{n}{\alpha}]^n$. The projection onto the constraint set corresponds to "clipping" each coordinate that is not in $[\frac{\alpha}{n}, \frac{n}{\alpha}]$. The proof, that this projection is clipping, follows directly from the fact that the definition of the constraint set $\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}$ and the divergence D_{ne} treat each component independently. Thus the proof reduces to the single component case.

Theorem 6 *The projection* $\boldsymbol{p} = P_{(\Gamma_{ne,\infty,\ln\frac{n}{\alpha}}, D_{ne})}(\boldsymbol{w})$ *is computed by*

$$\forall i\!:\!1,\ldots,n : p_i = \begin{cases} w_i & w_i \in [\frac{\alpha}{n}, \frac{n}{\alpha}] \\ \frac{\alpha}{n} & w_i < \frac{\alpha}{n} \\ \frac{n}{\alpha} & w_i > \frac{n}{\alpha} \end{cases} \tag{30}$$

Proof Suppose $\boldsymbol{p} = P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}, D_{\mathrm{ne}})}(\boldsymbol{w})$ and $\boldsymbol{p}$ does not equal the r.h.s. of (30). Then there exists a component p_i such that either $w_i < \frac{\alpha}{n}$ and $p_i > \frac{\alpha}{n}$; $w_i > \frac{n}{\alpha}$ and $p_i < \frac{n}{\alpha}$; or $w_i \in [\frac{\alpha}{n}, \frac{n}{\alpha}]$ and $p_i \neq w_i$. Without loss of generality, assume $w_i < \frac{\alpha}{n}$ and $p_i > \frac{\alpha}{n}$. Let $\boldsymbol{p}'$ equal $\boldsymbol{p}$, but with the ith component set to $\frac{\alpha}{n}$, i.e., $\boldsymbol{p}' = (p_1, \ldots, p_{i-1}, \frac{\alpha}{n}, p_{i+1}, \ldots, p_n)$. Since the function $g_b(a) = a \ln \frac{a}{b} + b - a$ is strictly convex and is minimized when $a = b$, the following inequality holds since $\frac{\alpha}{n}$ is a convex combination of w_i and p_i:

$$p_i \ln \frac{p_i}{w_i} + w_i - p_i > \frac{\alpha}{n} \ln \frac{\frac{\alpha}{n}}{w_i} + w_i - \frac{\alpha}{n} > w_i \ln \frac{w_i}{w_i} + w_i - w_i.$$

Therefore

$$D_{\mathrm{ne}}(\boldsymbol{p}, \boldsymbol{w}) - D_{\mathrm{ne}}(\boldsymbol{p}', \boldsymbol{w}) = p_i \ln \frac{p_i}{w_i} + w_i - p_i - \left[\frac{\alpha}{n} \ln \frac{\frac{\alpha}{n}}{w_i} + w_i - \frac{\alpha}{n}\right] > 0,$$

which contradicts the supposition that $\boldsymbol{p}$ is the projection of $\boldsymbol{w}$. ∎

The constraint set $\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}$ corresponds to the region $[\frac{\alpha}{n}, 1]^n \cap \mathcal{P}_n$. Unlike the computations of the previous two projections, the projection $\boldsymbol{v}^* = P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}, D_{\mathrm{ne}})}(\boldsymbol{w})$ does not lend itself to simple description. On first inspection, this projection should be similar to $P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}, D_{\mathrm{ne}})}(\boldsymbol{w})$, since the form of the divergences D_{ne} and $D_{\overline{\mathrm{ne}}}$ are the same except for the restriction of the domain to $\mathcal{P}_n$. However, the additional constraint $\sum_{i=1}^n v_i^* = 1$ introduces complications, as the independence between components is lost. For instance, consider a simple case for the computation of $P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}, D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$. The weight vector $\boldsymbol{w}$ to be projected contains components which are less than $\frac{\alpha}{n}$. In the projected vector those components are fixed to $\frac{\alpha}{n}$, and the remaining components are normalized (multiplied by some fraction) so that all the components now sum to one. The action of normalizing some components could also result in some components dropping to less than $\frac{\alpha}{n}$. Therefore, the projection algorithm chooses the set of components with least cardinality (the elements of this set must also be smaller than the remaining components of the weight vector) such that when these components are set to $\frac{\alpha}{n}$, the remaining components may be normalized without any component falling below $\frac{\alpha}{n}$. A straightforward algorithm is as follows: a) sort the components of the weight vector $\boldsymbol{w}$; b) for each $k = 0, \ldots, n-1$, generate a candidate solution by fixing the k smallest components to $\frac{\alpha}{n}$, then normalizing the remaining components; c) from those n candidate solutions choose the solution with minimal k that satisfies the constraints. Given an efficient implementation, this algorithm takes $O(n \ln n)$ time. In Figure 3 we present an algorithm that computes the projection $P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}, D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$ in $O(n)$ time. This algorithm avoids sorting the components. Instead it recursively uses an $O(n)$ algorithm for finding the median.

Theorem 7 *The algorithm in Figure 3 calculates the projection $P_{(\Gamma_{\overline{ne},\infty,\ln\frac{n}{\alpha}},D_{\overline{ne}})}(\boldsymbol{w})$ in $O(n)$ time.*

Before we prove Theorem 7, we give an overview of the "problem" and then prove three claims (Claims 1, 2, and 3) which provide a simple characterization of the projection $P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}},D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$. We then present the proof of Theorem 7 which combines the three claims.

Computing the projection $P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}},D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$ is a convex programming problem. The convex program is as follows:

`given a` $\boldsymbol{w} \in \mathrm{ri}\,\mathcal{P}_n$ `and a` $\alpha \in [0,1]$`, find the` $\boldsymbol{v}$ `that minimizes` $D_{\overline{\mathbf{ne}}}(\boldsymbol{v},\boldsymbol{w})$`,`
`such that` $\forall i : 1,\ldots,n : v_i \geq \frac{\alpha}{n}$ `and` $\sum_{i=1}^{n} v_i = 1$.

Let I be an $n \times n$ identity matrix, let $\vec{x}$ denote the n-dimensional vector $(x,\ldots,x)$, let κ be a Lagrange multiplier, and finally let $\boldsymbol{\lambda} = (\lambda_1,\ldots,\lambda_n)$ be a vector of n Lagrange multipliers. The Lagrangian for our minimization problem is

$$L(\boldsymbol{\lambda},\kappa,\boldsymbol{v}) = D_{\overline{\mathrm{ne}}}(\boldsymbol{v},\boldsymbol{w}) + \boldsymbol{\lambda}\cdot(I\boldsymbol{v} - \frac{\vec{\alpha}}{n}) + \kappa((\boldsymbol{v}\cdot\vec{1}) - 1). \tag{31}$$

Solving $\nabla_{\boldsymbol{v}} L(\boldsymbol{\lambda},\kappa,\boldsymbol{v}) = \vec{0}$ gives

$$\forall i : 1,\ldots,n : v_i = e^{-\lambda_i-\kappa} w_i.$$

Let $m_0 = e^{-\kappa}$ and let $m_i = e^{-\lambda_i}$. Thus the above n equations may be rewritten as

$$\forall i : 1,\ldots,n : v_i = m_0 m_i w_i. \tag{32}$$

The above n equations and the $n+1$ constraint equations must be satisfied by any $\boldsymbol{v}$ that is the minimizer of the convex program. The solution is unique by Proposition 23. We proceed to prove that the $\boldsymbol{v}$ calculated by the algorithm in Figure 3 is the minimizer of the convex program. We also introduce the following notation: $\boldsymbol{v}^* = P_{(\Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}},D_{\overline{\mathrm{ne}}})}(\boldsymbol{w})$, $\Psi = \{i : \boldsymbol{v}_i^* = \frac{\alpha}{n}\}$ (called an index set), and $\varsigma = \sum_{i\in\Psi} w_i$.

Claim 1

$$\forall i \notin \Psi : \boldsymbol{v}_i^* = \frac{1 - |\Psi|\frac{\alpha}{n}}{1-\varsigma} w_i.$$

Proof The Kuhn-Tucker complementary principle (Rockafellar, 1970, Theorem 28.3, Condition (a)) states that for each slack inequality (i.e., $\boldsymbol{v}_i^* > \frac{\alpha}{n}$), the corresponding Lagrange multiplier is 0. Thus for all $i \notin \Psi$, $v_i^* = m_0 w_i$. Since $\sum_{i\notin\Psi} v_i^* = 1 - |\Psi|\frac{\alpha}{n}$ and $\sum_{i\notin\Psi} w_i = 1-\varsigma$ we conclude

$$m_0 = \frac{\sum_{i\notin\Psi} v_i^*}{\sum_{i\notin\Psi} w_i} = \frac{1-|\Psi|\frac{\alpha}{n}}{1-\varsigma}.$$

■

Claim 2 *Without loss of generality for $i < j$ we assume $w_i \leq w_j$. Let $m_0 = \frac{1-|\Psi|\frac{\alpha}{n}}{1-\varsigma}$; then*

$$\boldsymbol{v}^* = \{\frac{\alpha}{n}, \ldots, \frac{\alpha}{n}, m_0 w_{|\Psi|+1}, \ldots, m_0 w_n\}. \tag{33}$$

Proof By Claim 1 the projection $\boldsymbol{v}^*$ is a "permutation" of Equation (33); either $v_i^* = \frac{\alpha}{n}$ or $v_i^* = m_0 w_i$ with exactly $|\Psi|$ terms equal to $\frac{\alpha}{n}$. Suppose $\boldsymbol{v}^*$ is not in the form of Equation (33); then there exists $p < q$ such $v_p^* = m_0 w_p$ and $v_q^* = \frac{\alpha}{n}$. Set $\boldsymbol{v}' = \boldsymbol{v}^*$, except $v_p' = \frac{\alpha}{n}$ and $v_q' = m_0 w_p$. Clearly $\boldsymbol{v}'$ lies within the constraints, and thus is a feasible solution to the convex programming problem. Then

$$\begin{aligned}
&D_{\overline{\text{ne}}}(\boldsymbol{v}^*, \boldsymbol{w}) - D_{\overline{\text{ne}}}(\boldsymbol{v}', \boldsymbol{w}) \\
&= m_0 w_p \ln \frac{m_0 w_p}{w_p} + \frac{\alpha}{n} \ln \frac{\frac{\alpha}{n}}{w_q} - \frac{\alpha}{n} \ln \frac{\frac{\alpha}{n}}{w_p} - m_0 w_p \ln \frac{m_0 w_p}{w_q} \\
&= m_0 w_p \ln \frac{w_q}{w_p} - \frac{\alpha}{n} \ln \frac{w_p}{w_q} \\
&= (m_0 w_p - \frac{\alpha}{n}) \ln \frac{w_q}{w_p}.
\end{aligned}$$

Since $m_0 w_p > \frac{\alpha}{n}$ and $w_q \geq w_p > 0$ we have $D_{\overline{\text{ne}}}(\boldsymbol{v}^*, \boldsymbol{w}) - D_{\overline{\text{ne}}}(\boldsymbol{v}', \boldsymbol{w}) \geq 0$. This contradicts our assumption that $\boldsymbol{v}^*$ is the unique minimizer of the convex program. Hence our supposition that $\boldsymbol{v}^*$ is not in the form of Equation 33 is false. ■

Claim 3 *Let $m_0' = \frac{1-k\frac{\alpha}{n}}{1-\sum_{i=1}^{k} w_i}$, $m_0'' = \frac{1-(k+1)\frac{\alpha}{n}}{1-\sum_{i=1}^{k+1} w_i}$,*

$$\boldsymbol{v}' = \{\overbrace{\frac{\alpha}{n}, \ldots, \frac{\alpha}{n}}^{k}, m_0' w_{k+1}, \ldots, m_0' w_n\},$$

and

$$\boldsymbol{v}'' = \{\overbrace{\frac{\alpha}{n}, \ldots, \frac{\alpha}{n}}^{k+1}, m_0'' w_{k+2}, \ldots, m_0'' w_n\};$$

then $D_{\overline{ne}}(\boldsymbol{v}', \boldsymbol{w}) \leq D_{\overline{ne}}(\boldsymbol{v}'', \boldsymbol{w})$.

Proof The vector $\boldsymbol{v}'$ is the minimum of the following convex program:

> `given a` $\boldsymbol{w} \in \operatorname{ri}\mathcal{P}_n$ `and a` $\alpha \in [0,1]$ `find the` $\boldsymbol{v}$ `that minimizes` $D_{\overline{\text{ne}}}(\boldsymbol{v}, \boldsymbol{w})$`, such that` $v_1 = \cdots = v_k = \frac{\alpha}{n}$ `and` $\sum_{i=1}^{n} v_i = 1$.

The vector $\boldsymbol{v}''$ is a feasible point of the above convex program, hence $D_{\overline{\text{ne}}}(\boldsymbol{v}', \boldsymbol{w}) \leq D_{\overline{\text{ne}}}(\boldsymbol{v}'', \boldsymbol{w})$. ■

Proof [Proof of Theorem 7] We first prove that the projection of $\boldsymbol{w}$ may be computed by choosing the set of components of $\boldsymbol{w}$ with the least cardinality (the elements of this set must also be smaller than the remaining components of the vector) such that when these components are set to $\frac{\alpha}{n}$, the remaining components may then be normalized without any

1. $\mathcal{W} = \{1, \ldots, n\}$
2. $C_{\#} = 0$; $C_{\%} = 0$
3. while $\mathcal{W} \neq \emptyset$ do
4. $\quad\omega = \text{findmedian}(\{w_i : i \in \mathcal{W}\})$
5. $\quad\mathcal{L} = \{i : w_i < \omega,\, i \in \mathcal{W}\}$; $L_{\#} = |\mathcal{L}|$; $L_{\%} = \sum_{i \in \mathcal{L}} w_i$
6. $\quad\mathcal{M} = \{i : w_i = \omega,\, i \in \mathcal{W}\}$; $M_{\#} = |\mathcal{M}|$; $M_{\%} = \sum_{i \in \mathcal{M}} w_i$
7. $\quad\mathcal{H} = \{i : w_i > \omega,\, i \in \mathcal{W}\}$
8. $\quad m_0 = \frac{1-(C_{\#}+L_{\#})\frac{\alpha}{n}}{1-(C_{\%}+L_{\%})}$
9. $\quad$if $\omega m_0 < \frac{\alpha}{n}$ then
10. $\quad\quad C_{\#} = C_{\#} + L_{\#} + M_{\#}$; $C_{\%} = C_{\%} + L_{\%} + M_{\%}$
11. $\quad\quad$if $\mathcal{H} = \emptyset$ then $\omega = \min(\{w_i : w_i > \omega,\, 1 \leq i \leq n\})$
12. $\quad\quad\mathcal{W} = \mathcal{H}$
13. $\quad$else
14. $\quad\quad\mathcal{W} = \mathcal{L}$
15. $m_0 = \frac{1-C_{\#}\frac{\alpha}{n}}{1-C_{\%}}$
16. $\forall i : 1, \ldots, n : v_i^* = \begin{cases} \frac{\alpha}{n} & w_i < \omega \\ w_i m_0 & w_i \geq \omega \end{cases}$

Figure 3: Algorithm to compute $\boldsymbol{v}^* = P_{(\Gamma_{\overline{\text{ne}},\infty,\ln\frac{n}{\alpha}}, D_{\overline{\text{ne}}})}(\boldsymbol{w})$.

component falling below $\frac{\alpha}{n}$. Second, we argue that the algorithm in Figure 3 correctly computes that projection $\boldsymbol{v}^*$ in $O(n)$ time.

Claim 1 proves that the projection $\boldsymbol{v}^*$ vector consists of a "index" set of components Ψ of $\boldsymbol{w}$ fixed to $\frac{\alpha}{n}$, while the remaining components are normalized. By identifying a potential projection vector with an index set Ψ one can narrow the choice of the potential projection vector $\boldsymbol{v}^*$ from any vector in $\Gamma_{\overline{\text{ne}},\infty,\ln\frac{n}{\alpha}}$ to the 2^n vectors that correspond to the 2^n sets $\Psi \subseteq \{1, \ldots, n\}$. Claim 2 proves that the magnitude of a component to be fixed is smaller than the magnitude of a component to be normalized, i.e., if $i \in \Psi$ and $j \notin \Psi$, then $w_i \leq w_j$. This further narrows the number of potential index sets to n. These n sets correspond to the index sets containing the $k = 0, \ldots, n-1$ smallest components of $\boldsymbol{w}$. Finally, Claim 3 shows that among these n index sets, if $\Psi' \subseteq \Psi''$ with corresponding potential projection vectors $\boldsymbol{v}'$ and $\boldsymbol{v}''$, then $D_{\overline{\text{ne}}}(\boldsymbol{v}', \boldsymbol{w}) \leq D_{\overline{\text{ne}}}(\boldsymbol{v}'', \boldsymbol{w})$. Thus to compute the projection we need to choose the index set of least cardinality whose corresponding potential projection is also contained in the constraint set $\Gamma_{\overline{\text{ne}},\infty,\ln\frac{n}{\alpha}}$.

We have shown thus far that the projection is characterized by a particular "minimal" index set Ψ. The "minimal" index set may be specified uniquely by a single component ω of the weight vector $\boldsymbol{w}$. The *threshold component* ω is the component which is just larger than the components of the index set Ψ, i.e., $\Psi = \{i : w_i < \omega\}$. The algorithm thus finds the projection by finding the threshold component ω.

The algorithm seeks the *threshold component* ω; when ω is found, the projection is computed by fixing all the components less than ω to $\frac{\alpha}{n}$ and normalizing the remaining components (line 16). We proceed to discuss how the algorithm finds ω, though we ignore some details of the control structure. On each iteration of the algorithm (lines 3-14) a new value for the *threshold component* ω is examined[3]. The values chosen for ω are determined from the index set variable $\mathcal{W}$. The variable $\mathcal{W}$ is equal initially to $\{1,\ldots,n\}$. On each iteration the median[4] of the component values of $\mathcal{W}$ (line 4) is chosen as a potential threshold component ω. The elements of $\mathcal{W}$ are then sorted into two sets, $\mathcal{L}$ and $\mathcal{H}$ (lines 5,7); the set $\mathcal{L}$ contains the indices of components of $\mathcal{W}$ which are smaller than ω, and $\mathcal{H}$ contains correspondingly the indices of components larger than ω. The value m_0, the normalizing constant, is then calculated (line 8). If $m_0\omega < \frac{\alpha}{n}$, then by Claims 2 and 3 the true threshold component must be larger than the current ω and is thus contained in $\mathcal{H}$. Otherwise, the true threshold component must equal ω or be contained in $\mathcal{L}$. Since ω was the median element, the algorithm now iterates (lines 3-14) with either $\mathcal{W} = \mathcal{H}$ or $\mathcal{W} = \mathcal{L}$, as appropriate (note that $\max\{|\mathcal{L}|,|\mathcal{H}|\} \leq \frac{1}{2}|\mathcal{W}|$). When $\mathcal{W} = \emptyset$ the iteration of lines 3-14 completes, and the threshold component ω has been found. There are a maximum of $\lceil \log n + 1\rceil$ iterations. The ith iterate takes $O(\frac{n}{2^i})$ time, thus the algorithm spends $O(n)$ time in lines 3-14. Consequently, the time complexity of the algorithm is $O(n)$. ■

5. Applications

In Section 2 we sketched a general technique for proving shifting relative loss bounds. The technique consisted of modifying a General Additive Regression Algorithm by adding a projection update that projects the algorithm's hypothesis onto a constraint set. The analysis of the projection update was then easily combined with the amortized analysis of the original algorithm to prove a shifting loss bound. In Definition 8 we explicitly define the type of amortized analysis that is necessary to apply our technique. In Definition 9 we give the general technique for transforming an algorithm with an amortized analysis to an algorithm for which we can prove a shifting relative loss bound. A general bound is then given in Theorem 10 and this bound is applied in theorems 14, 15, and 16 to the shifting analysis of EG, GD, and EGU, respectively. Finally, we show in Theorem 18 that the shifting loss bound analysis of EGU may be extended to predictor schedules with predictors not in the constraint set.

Definition 8 *Consider an on-line algorithm $A_F(\eta)$, based on a convex function $F : E \rightarrow \Re$, with constant learning rate η and associated nonnegative functions $a(\eta)$ and $b(\eta)$. Let $\boldsymbol{w} \in \operatorname{ri} E \subseteq \Re^n$ denote the "weight vector" of the algorithm at the start of a trial; let $\boldsymbol{w}'$ denote the updated weight vector at the end of the trial. If for all $\boldsymbol{w} \in \operatorname{ri} E$ and all $\boldsymbol{u} \in E$ the following inequality holds*

$$a(\eta)\ L(\boldsymbol{w}) - b(\eta)\ L(\boldsymbol{u}) \leq D_F(\boldsymbol{u},\boldsymbol{w}) - D_F(\boldsymbol{u},\boldsymbol{w}') \tag{34}$$

then we say that the algorithm $A_F(\eta)$ has an **amortized analysis.**

3. Even though on each iteration a new ω is examined, it is possible that a previously examined ω will eventually become the threshold component (see line 11).
4. The median of a set of n numbers may be found in $O(n)$ time (Blum et al., 1973). For our purposes if n is even, it does not matter if the algorithm chooses the $\frac{n}{2}$ or $(\frac{n}{2}+1)$ largest element.

We sketched in Section 2 how the inequality of the definition (see also Equation 6) can be applied. Also, the three General Additive Regression Algorithms GD, EGU, and EG have an *amortized analysis* according to above definition (see Cesa-Bianchi et al. 1996 and Kivinen and Warmuth 1997).

Given an on-line algorithm $A_F(\eta)$ with an amortized analysis and a convex set $\Gamma_{F,p,\gamma} \subseteq E$, we may transform algorithm $A_F(\eta)$ to an algorithm with a shifting loss bound.

Definition 9 *The constrained online algorithm $C\text{-}A_F(\eta, \Gamma_{F,p,\gamma})$ is defined by transforming the algorithm $A_F(\eta)$ with an amortized analysis as follows.*

> Choose a convex constraint set $\Gamma_{F,p,\gamma}$ (see Definition 4). Then, let $\boldsymbol{w}_t^m$ denote the weight vector at the end of every trial t in algorithm $A_F(\eta)$; at the end of every trial t add the update
>
> $$\boldsymbol{w}_{t+1} = P_{(\Gamma_{F,p,\gamma}, D_F)}(\boldsymbol{w}_t^m)$$
>
> to algorithm $A_F(\eta)$, completing the transformation.

We now have the requisite definitions to give a general shifting loss bound for a broad class of algorithms which have an amortized analysis.

Theorem 10 *The shifting loss for algorithm $C\text{-}A_F(\eta, \Gamma_{F,p,\gamma})$ is bounded by*

$$\begin{aligned} L(C\text{-}A_F(\eta, \Gamma_{F,p,\gamma}), S) \leq \frac{b(\eta)}{a(\eta)} L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) + \\ \frac{1}{a(\eta)}[D_F(\boldsymbol{u}_1, \boldsymbol{w}_1) - D_F(\boldsymbol{u}_\ell, \boldsymbol{w}_{\ell+1}) + F(\boldsymbol{u}_\ell) - F(\boldsymbol{u}_1) + \gamma \|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle\|_{\frac{p}{p-1}}] \end{aligned} \quad (35)$$

for all $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle \in \Gamma^\ell_{F,p,\gamma}$.

Proof We sum the following three inequalities,

$$a(\eta)\ L_t(\boldsymbol{w}_t) - b(\eta)\ L_t(\boldsymbol{u}_t) \leq D_F(\boldsymbol{u}_t, \boldsymbol{w}_t) - D_F(\boldsymbol{u}_t, \boldsymbol{w}_t^m) \quad (36)$$

$$0 \leq D_F(\boldsymbol{u}_t, \boldsymbol{w}_t^m) - D_F(\boldsymbol{u}_t, \boldsymbol{w}_{t+1}) \quad (37)$$

$$F(\boldsymbol{u}_t) - F(\boldsymbol{u}_{t+1}) - \gamma \|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_{\frac{p}{p-1}} \leq D_F(\boldsymbol{u}_t, \boldsymbol{w}_{t+1}) - D_F(\boldsymbol{u}_{t+1}, \boldsymbol{w}_{t+1}) \quad (38)$$

to obtain

$$a(\eta) L_t(\boldsymbol{w}_t) - b(\eta) L_t(\boldsymbol{u}_t) + F(\boldsymbol{u}_t) - F(\boldsymbol{u}_{t+1}) - \gamma \|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_{\frac{p}{p-1}} \leq D_F(\boldsymbol{u}_t, \boldsymbol{w}_t) - D_F(\boldsymbol{u}_{t+1}, \boldsymbol{w}_{t+1}). \quad (39)$$

The first inequality (36) is equivalent to (34); only the notation has changed. The second equality follows from Corollary 3, since $\boldsymbol{w}_{t+1} = P_{(\Gamma_{F,p,\gamma}, D_F)}(\boldsymbol{w}_t^m)$ and $\boldsymbol{u}_t \in \Gamma_{F,p,\gamma}$. We prove the third inequality (38) by expanding $D_F(\boldsymbol{u}_t, \boldsymbol{w}_t) - D_F(\boldsymbol{u}_{t+1}, \boldsymbol{w}_{t+1})$ to $F(\boldsymbol{u}_t) - F(\boldsymbol{u}_{t+1}) + \nabla F(\boldsymbol{w}_{t+1}) \cdot (\boldsymbol{u}_{t+1} - \boldsymbol{u}_t)$. Then from Hölder's inequality, $\nabla F(\boldsymbol{w}_{t+1}) \cdot (\boldsymbol{u}_{t+1} - \boldsymbol{u}_t) \geq -\|\nabla F(\boldsymbol{w}_{t+1})\|_p \|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_{\frac{p}{p-1}}$. Thus the lower bound is completed, since $\|\nabla F(\boldsymbol{w}_{t+1})\|_p < \gamma$ by the definition of $\Gamma_{F,p,\gamma}$ (see Definition 4). We then obtain a shifting loss bound (35) by summing (39) over trials $t = 1, \ldots, \ell - 1$, and then adding to the sum on trial $t = \ell$ only the inequalities (36) and (37). The bound then holds by straightforward algebraic manipulations of the sum. ■

Observe in the above theorem that when choosing the constraint set $\Gamma_{F,p,\gamma}$ there is a tradeoff between the size of the constraint set (feasible predictors) and the size of bound (35). For example, if $r < s$, $\Gamma_{F,r,\gamma}$ leads to a smaller constraint set and loss bound than does $\Gamma_{F,s,\gamma}$, since $\Gamma_{F,r,\gamma} \subseteq \Gamma_{F,s,\gamma}$ and $\|\boldsymbol{w}\|_{\frac{r}{r-1}} \leq \|\boldsymbol{w}\|_{\frac{s}{s-1}}$.

In order to apply Theorem 10 to GD, EGU and EG we need to use the original Lemmas from (Cesa-Bianchi et al., 1996, Kivinen and Warmuth, 1997) that determine the functions $a(\eta)$ and $b(\eta)$ as in (34). In the following Lemmas we expand $L_t(\boldsymbol{w})$ and $L_t(\boldsymbol{u})$ to $(\boldsymbol{w}\cdot\boldsymbol{x}-y)^2$ and $(\boldsymbol{u}\cdot\boldsymbol{x}-y)^2$, respectively, to emphasize the implicit dependence in these Lemmas on the magnitudes of $\boldsymbol{x}$ and y.

Lemma 11 ((Cesa-Bianchi et al., 1996, Lemma 4.3)) *For some $X > 0$, let $\|\boldsymbol{x}\|_2 < X$ and let $\eta = \frac{1}{4X^2}$. Then for any $\boldsymbol{w} \in \Re^n$ let*

$$\boldsymbol{w}' = \boldsymbol{w} - 2\eta(\boldsymbol{w}\cdot\boldsymbol{x} - y)\boldsymbol{x};$$

then for any $\boldsymbol{u} \in \Re^n$

$$\frac{1}{4X^2}(y - \boldsymbol{w}\cdot\boldsymbol{x})^2 - \frac{1}{2X^2}(y - \boldsymbol{u}\cdot\boldsymbol{x})^2 \leq D_{sq}(\boldsymbol{u},\boldsymbol{w}) - D_{sq}(\boldsymbol{u},\boldsymbol{w}'). \tag{40}$$

Lemma 12 ((Kivinen and Warmuth, 1997, Lemma 5.14)) *Let $\boldsymbol{x}\in[0,X]^n$, $y\in[0,Y]$ for some $X, Y > 0$, and let $\eta = \frac{1}{3XY}$. Then for any $\boldsymbol{w} \in [0,\infty)^n$ let*

$$\forall i : 1,\ldots,n : \boldsymbol{w}'_i = \boldsymbol{w}_i e^{-\eta x_i(\boldsymbol{w}\cdot\boldsymbol{x}-y)};$$

then for any $\boldsymbol{u} \in [0,\infty)^n$

$$\frac{1}{3XY}(y - \boldsymbol{w}\cdot\boldsymbol{x})^2 - \frac{1}{XY}(y - \boldsymbol{u}\cdot\boldsymbol{x})^2 \leq D_{ne}(\boldsymbol{u},\boldsymbol{w}) - D_{ne}(\boldsymbol{u},\boldsymbol{w}'). \tag{41}$$

Lemma 13 ((Kivinen and Warmuth, 1997, Lemma 5.8)) *Let $\boldsymbol{x} \in [a_1, a_1+X]\times\ldots\times[a_n, a_n+X]$ for some $X > 0$ and $\boldsymbol{a} \in \Re^n$, and let $\eta = \frac{2}{3X^2}$. Then for any $\boldsymbol{w} \in \mathcal{P}_n$ let*

$$\forall i : 1,\ldots,n : w'_i = \frac{w_i e^{-\eta x_{t,i}(\boldsymbol{x}\cdot\boldsymbol{w}-y)}}{\sum_{j=1}^n w_j e^{-\eta x_{t,j}(\boldsymbol{x}\cdot\boldsymbol{w}-y)}};$$

then for any $\boldsymbol{u} \in \mathcal{P}_n$

$$\frac{2}{3X^2}(y - \boldsymbol{w}\cdot\boldsymbol{x})^2 - \frac{1}{X^2}(y - \boldsymbol{u}\cdot\boldsymbol{x})^2 \leq D_{\overline{ne}}(\boldsymbol{u},\boldsymbol{w}) - D_{\overline{ne}}(\boldsymbol{u},\boldsymbol{w}'). \tag{42}$$

We now use the above lemmas in conjunction with Theorem 10 to prove shifting loss bounds for C-GD($\eta, \Gamma_{\mathrm{sq},2,\gamma}$), C-EGU($\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}$), and C-EG($\eta, \Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}}$).

Theorem 14 *Consider a trial sequence $S = \langle(\boldsymbol{x}_1, y_1),\ldots,(\boldsymbol{x}_\ell, y_\ell)\rangle$ with $\|\mathbf{x}_t\|_2 < X$ and $y_t \in \Re$ for all t for some constant $X > 0$. Let $\eta = \frac{1}{4X^2}$. For some constant $\gamma > 0$, let $\Gamma_{sq,2,\gamma}$ be the constraint set and let $w_1 = (0,\ldots,0)$. Then for all predictor sequences $\langle\boldsymbol{u}_1,\ldots,\boldsymbol{u}_\ell\rangle \in \Gamma^\ell_{sq,2,\gamma}$ we have the bound*

$$L(S, \text{C-GD}(\eta,\Gamma_{sq,2,\gamma})) \leq 2L(\langle\boldsymbol{u}_1,\ldots,\boldsymbol{u}_\ell\rangle, S) + 4X^2[\frac{1}{2}\|\boldsymbol{u}_\ell\|_2^2 + \gamma\|\langle\boldsymbol{u}_1,\ldots,\boldsymbol{u}_\ell\rangle\|_2]. \tag{43}$$

Proof We combine Lemma 11 with Theorem 10 and exclude the negative terms since we are upper bounding. ■

In the following theorem, the notation $H(\boldsymbol{x}) = \sum_{i=1}^{n} x_i \ln \frac{1}{x_i}$ for the "entropy" of a vector in $[0,\infty)^n$ is used.

Theorem 15 *Consider a trial sequence* $S = \langle(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\ell, y_\ell)\rangle$ *with* $\mathbf{x}_t \in [0, X]^n$ *and* $y_t \in [0, Y]$ *for all t for some constants* $X > 0$ *and* $Y > 0$*. Let* $\eta = \frac{1}{3XY}$*. For some constant* $\alpha \in [0,1]$*, let* $\Gamma_{ne,\infty,\ln\frac{n}{\alpha}}$ *be the constraint set and let* $\boldsymbol{w}_1 = (\frac{1}{n}, \ldots, \frac{1}{n})$*. Then for all predictor sequences* $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle \in \Gamma^{\ell}_{ne,\infty,\ln\frac{n}{\alpha}}$ *we have the bound*

$$L(S, \textit{C-EGU}(\eta, \Gamma_{ne,\infty,\ln\frac{n}{\alpha}})) \leq 3L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle, S) + 3XY\left[\|\boldsymbol{u}_1\|_1 \ln n + \|\boldsymbol{u}_\ell\|_1 + H(\boldsymbol{u}_\ell) + 1 + \ln(\frac{n}{\alpha})\|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1\right]. \quad (44)$$

Proof We combine Lemma 12 with Theorem 10 and exclude negative terms since we are upper bounding. ■

Theorem 16 *Consider a trial sequence* $S = \langle(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\ell, y_\ell)\rangle$ *with* $\mathbf{x}_t \in [a_1, a_1 + X] \times \ldots \times [a_n, a_n + X]$ *for some constants* $X > 0$ *and* $\boldsymbol{a} \in \Re^n$*. Let* $\eta = \frac{2}{3X^2}$*. For some constant* $\alpha \in [0,1]$*, let* $\Gamma_{\overline{ne},\infty,\ln\frac{n}{\alpha}}$ *be the constraint set and let* $\boldsymbol{w}_1 = (\frac{1}{n}, \ldots, \frac{1}{n})$*. Then for all predictor sequences* $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle \in \Gamma^{\ell}_{\overline{ne},\infty,\ln\frac{n}{\alpha}}$ *we have the bound*

$$L(S, \textit{C-EG}(\eta, \Gamma_{\overline{ne},\infty,\ln\frac{n}{\alpha}})) \leq \frac{3}{2}L(\boldsymbol{u}, S) + \frac{3}{2}X^2\left[\ln n + \frac{1}{2}\ln(\frac{n}{\alpha})\|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1\right]. \quad (45)$$

Proof Since the predictors and the weights are constrained within a subset of $\mathcal{P}_n$ we may obtain a slightly tighter bound than the straightforward combination of Lemma 13 and Theorem 10. Observe that if $\boldsymbol{p} \in \mathcal{P}_n$ and $\sum_{i=1}^{n} q_i = 0$, then $\boldsymbol{p} \cdot \boldsymbol{q} \leq \frac{1}{2}\|\boldsymbol{p}\|_\infty \|\boldsymbol{q}\|_1$.[5] ■

5.1 An extension of the analysis of C-EGU$(\eta, \Gamma_{\mathbf{ne},\infty,\ln\frac{n}{\alpha}})$ to sparse predictor vectors

One of the key strengths of the EG and EGU Algorithms is that the loss bound may be exponentially smaller (Kivinen and Warmuth, 1997) than that of the GD Algorithm in terms of the dimension of the instances ($\mathbf{x}_t$). This occurs when the predictor vector $\boldsymbol{u}$ is sparse ($\boldsymbol{u}$ contains $O(1)$ non-zero components) for typical $\mathbf{x}_t$ (e.g., $\mathbf{x}_t \in \{-1, 1\}^n$). A lower bound[6] for this sparse case is also shown by an adversary argument in (Kivinen et al., 1997).

Given that a major strength of the EGU Algorithm is the case when the predictor vector $\boldsymbol{u}$ is sparse then the bound of C-EGU$(\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$ in the Theorem 15 is limited in that

5. Note that by Hölder's inequality one can obtain the weaker bound of $\boldsymbol{p} \cdot \boldsymbol{q} \leq \|\boldsymbol{p}\|_\infty \|\boldsymbol{q}\|_1$, i.e., the factor of $\frac{1}{2}$ is missing.

6. In (Kivinen et al., 1997) this is shown for the perceptron vs. winnow. This argument can be extended to the case of GD vs. EGU.

each predictor vector $\boldsymbol{u}_t$ in the sequence of predictor vectors $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle$ is non-sparse since it is contained within $\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}} = [\frac{\alpha}{n}, \frac{n}{\alpha}]^n$.

In this section we present an analysis of C-EGU$(\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$ that allows sparse predictor vectors. Thus the set of allowable predictor vectors will be extended to $[0, \frac{n}{\alpha}]^n$. The analysis of C-EG$(\eta, \Gamma_{\overline{\mathrm{ne}},\infty,\ln\frac{n}{\alpha}})$ in Theorem 16 is similarly limited but we do not consider its extension as its analysis is parallel to that of C-EGU$(\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$.

Since we are changing the analysis of C-EGU$(\eta, \Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}})$ but not the algorithm the weight vectors are still restricted to $[\frac{\alpha}{n}, \frac{n}{\alpha}]^n$. For the simpler case when the comparison class is a sequence of experts chosen from a set of n "experts",[7] the Variable-share Algorithm (Herbster and Warmuth, 1998a) allows the components of the weight vectors to approach zero.

Lemma 17 *Let $\boldsymbol{w} \in [0,\infty)^n$ and let $\boldsymbol{u}_t, \boldsymbol{u}_{t+1} \in [0, \frac{n}{\alpha}]^n$. Then*

$$D_{ne}(\boldsymbol{u}_t, \boldsymbol{w}) - D_{ne}(\boldsymbol{u}_{t+1}, P_{(\Gamma_{ne,\infty,\ln\frac{n}{\alpha}}, D_{ne})}(\boldsymbol{w})) \geq \\ - H(\boldsymbol{u}_t) + H(\boldsymbol{u}_{t+1}) - \|\boldsymbol{u}_t\|_1 + \|\boldsymbol{u}_{t+1}\|_1 - \ln(\frac{n}{\alpha})\|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_1 - \alpha. \quad (46)$$

Proof Note that the divergence D_{ne} is a simple sum over coordinates, and the constraint set $\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}$ is the same in each coordinate. A simple but lengthy argument can show that

$$P_{(\Gamma_{\mathrm{ne},\infty,\ln\frac{n}{\alpha}}, D_{\mathrm{ne}})}(\boldsymbol{w}) = P_{([\frac{\alpha}{n},\infty)^n, D_{\mathrm{ne}})}(P_{([0,\frac{n}{\alpha}]^n, D_{\mathrm{ne}})}(\boldsymbol{w})).$$

Let $\boldsymbol{w'} = P_{([0,\frac{n}{\alpha}]^n, D_{\mathrm{ne}})}(\boldsymbol{w})$, and let $\boldsymbol{w''} = P_{([\frac{\alpha}{n},\infty)^n, D_{\mathrm{ne}})}(\boldsymbol{w'})$. The sum of the following three inequalities gives Inequality (46), thus proving the lemma:

$$D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w}) - D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w'}) \geq 0 \quad (47)$$

$$D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w'}) - D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w''}) \geq -\alpha \quad (48)$$

$$D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w''}) - D_{\mathrm{ne}}(\boldsymbol{u}_{t+1}, \boldsymbol{w''}) \geq \\ - H(\boldsymbol{u}_t) + H(\boldsymbol{u}_{t+1}) - \|\boldsymbol{u}_t\|_1 + \|\boldsymbol{u}_{t+1}\|_1 - \ln(\frac{n}{\alpha})\|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_1. \quad (49)$$

Inequality (47) holds by Corollary 3 when all $\boldsymbol{u}_t$ lie in the constraint set $[0, \frac{n}{\alpha}]^n$. Recall that the latter convex set was used as the constraint set for defining $\boldsymbol{w'}$ as a projection.

For Inequality (48) we expand the definition of D_{ne}:

$$D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w'}) - D_{\mathrm{ne}}(\boldsymbol{u}_t, \boldsymbol{w''}) = \sum_{i=1}^{n}(\boldsymbol{w}'_i - \boldsymbol{w}''_i) + \sum_{i=1}^{n} u_{t,i} \ln \frac{w''_i}{w'_i}$$

From the definition of $\boldsymbol{w'}$ and $\boldsymbol{w''}$ it follows that $w'_i - w''_i \geq -\frac{\alpha}{n}$ and $w''_i - w'_i \geq 0$. Thus the first sum is lower bounded by $-\alpha$ and the second sum is lower bounded by zero, giving us Inequality (48).

7. The comparison vector corresponding to the i-th expert in the set is the i-th unit vector, i.e., one in component i and zero otherwise.

For the proof of Inequality (49) we expand $D_{\rm ne}$, then apply Hölder's inequality and use the fact that $w''_i \in [\frac{\alpha}{n}, \frac{n}{\alpha}]$:

$$\begin{aligned} D_{\rm ne}(\boldsymbol{u}_t, \boldsymbol{w}'') - D_{\rm ne}(\boldsymbol{u}_{t+1}, \boldsymbol{w}'') = \\ & - H(\boldsymbol{u}_t) + H(\boldsymbol{u}_{t+1}) - \|\boldsymbol{u}_t\|_1 + \|\boldsymbol{u}_{t+1}\|_1 + \sum_{i=1}^{n}\left[(u_{t,i} - u_{t+1,i}) \ln \frac{1}{w''_i}\right] \\ & \geq -H(\boldsymbol{u}_t) + H(\boldsymbol{u}_{t+1}) - \|\boldsymbol{u}_t\|_1 + \|\boldsymbol{u}_{t+1}\|_1 - \ln(\frac{n}{\alpha})\|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_1. \end{aligned}$$

■

Theorem 18 *Consider a trial sequence $S = \langle(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\ell, y_\ell)\rangle$ with $\mathbf{x}_t \in [0, X]^n$ and $y_t \in [0, Y]$ for all t for some constants $X > 0$ and $Y > 0$. Let $\eta = \frac{1}{3XY}$. For some constant $\alpha \in [0, 1]$, let $\Gamma_{ne,\infty,\ln\frac{n}{\alpha}}$ be the constraint set and let $\boldsymbol{w}_1 = (\frac{1}{n}, \ldots, \frac{1}{n})$. Then for all predictor sequences $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle \in \{[0, \frac{n}{\alpha}]\}^\ell$ we have the bound*

$$\begin{aligned} L(S, \textit{C-EGU}(\eta, \Gamma_{ne,\infty,\ln\frac{n}{\alpha}})) \leq 3L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle, S) + \\ 3XY\left[\|\boldsymbol{u}_1\|_1 \ln n + \ln(\frac{n}{\alpha})\|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1 + \alpha(\ell - 1) + \|\boldsymbol{u}_\ell\|_1 + H(\boldsymbol{u}_\ell) + 1\right]. \quad (50) \end{aligned}$$

Proof The proof is essentially the same as Theorem 15, except that we cannot directly apply Theorem 10. The same summing and rearranging of inequalities occurs as in Theorem 10 except that Lemma 17 replaces the sum of inequalities (37) and (38). ■

Finally, we give a tuning of α for the purpose of minimizing the above loss bound[8]. The tuning introduce two parameters, $\hat{\ell}$ and $\hat{U}$, where $\hat{\ell}$ is an upper bound on the length of the trial sequence and $\hat{U}$ is an approximation to $\|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1$. We set $\alpha = \min\{\frac{\hat{U}}{\hat{\ell}-1}, 1\}$, we choose this value of α since if $\hat{\ell}$ and $\hat{U}$ equals ℓ and $\|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1$ respectively then this tuning minimizes the loss bound with respect to α.

Corollary 19 *Consider a trial sequence $S = \langle(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\ell, y_\ell)\rangle$ with $\mathbf{x}_t \in [0, X]^n$ and $y_t \in [0, Y]$ for all t for some constants $X > 0$ and $Y > 0$. Let $\eta = \frac{1}{3XY}$. Let $\hat{\ell} \geq \ell$ and let $\hat{U} \geq 0$ then set $\alpha = \min\{\frac{\hat{U}}{\hat{\ell}-1}, 1\}$. Let $\Gamma_{ne,\infty,\ln\frac{n}{\alpha}}$ be the constraint set and let $\boldsymbol{w}_1 = (\frac{1}{n}, \ldots, \frac{1}{n})$. Then for all predictor sequences $\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle \in \{[0, \frac{n}{\alpha}]\}^\ell$ when $\alpha = \frac{\hat{U}}{\hat{\ell}-1}$ we have the bound*

$$\begin{aligned} & L(S, \textit{C-EGU}(\eta, \Gamma_{ne,\infty,\ln\frac{n}{\alpha}})) \leq 3L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle, S) + \\ & 3XY\left[(\|\boldsymbol{u}_1\|_1 + \|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1) \ln n + \|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell\rangle|\!\|_1 \ln \frac{\hat{\ell}-1}{\hat{U}} + \hat{U} + \|\boldsymbol{u}_\ell\|_1 + H(\boldsymbol{u}_\ell) + 1\right], \end{aligned} \quad (51)$$

8. However, the tuning ignores the issue that as we increase α the comparison class shrinks. However, the comparison class never shrinks smaller than $\{[0, n]^n\}^\ell$.

and when $\alpha = 1$ we have the bound

$$L(S, C\text{-}EGU(\eta, \Gamma_{ne,\infty,\ln\frac{n}{\alpha}})) \leq 3L(\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle, S) \\ + 3XY\left[(\|\boldsymbol{u}_1\|_1 + \|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle|\!\|_1) \ln n + (\ell - 1) + \|\boldsymbol{u}_\ell\|_1 + H(\boldsymbol{u}_\ell) + 1 \right]. \quad (52)$$

The term $(\ell - 1)$ indicates that (52) is a fairly weak performance guarantee. This is not surprising since "large" α indicates that we expect a great deal of shifting between predictor vectors as measured by $\|\!|\langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell \rangle|\!\|_1$ when compared to the sequence length.

6. Conclusion

We are developing important methods based on projections that can be used for proving worst-case loss bounds when the predictor from the comparison class is allowed to shift over time. These methods apply to such algorithms as the WM Algorithm (Littlestone and Warmuth, 1994), the Aggregating Algorithm (Vovk, 1995), the Hedge Algorithm (Freund and Schapire, 1997), and various exponentiated gradient algorithms (Kivinen and Warmuth, 1997, Helmbold et al., 1999, Bylander, 1997, Kivinen and Warmuth, 2001), as well as Winnow (Littlestone, 1988). The application of the projection update to the static case is also interesting. Prior knowledge may be represented with convex constraints. The constraints may then be maintained without any increase in the loss bound. For example this methodology can be applied to the on-line portfolio prediction problem (Cover, 1991, Helmbold et al., 1998). In this case linear inequality constraints may be used to express relations that must be maintained between the instruments of the portfolio.

In future work the methodology developed in this paper needs to be applied to other families of algorithms such as the p-norm algorithms (Grove et al., 2001, Gentile and Littlestone, 1999). Also, the tightness of our bounds should be investigated by proving matching lower bounds. Lower bounds in the expert setting for the entropic loss have been proven in (Herbster and Warmuth, 1998a).

Acknowledgements We would like to thank Claudio Gentile, Deborah Lewis, Andrew Klinger, and Harold Widom for valuable discussions. We also the thank the anonymous referees for their helpful comments.

References

M. A. Aizerman, E. M. Braverman, and L. I. Rozonoér. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 68:337–404, 1950.

P. Auer, M. Herbster, and M. K. Warmuth. Exponentially many local minima for single neurons. In *Proc. 1995 Neural Information Processing Conference*, pages 316–317. MIT Press, Cambridge, MA, November 1995.

P. Auer and M. K. Warmuth. Tracking the best disjunction. *Journal of Machine Learning*, 32(2):127–150, August 1998. Special issue on concept drift.

K. Azoury and M. K. Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Journal of Machine Learning*, 43(3):211–246, June 2001.

H. H. Bauschke and J. M. Borwein. Legendre functions and the method of random Bregman projections. *J. Convex Anal.*, 4(1):27–67, 1997. ISSN 0944-6532.

A. Blum and C. Burch. On-line learning and the metrical task system problem. *Machine Learning*, 39(1):35–58, 2000.

M. Blum, R. W. Floyd, V. Pratt, R. L. Rivest, and R. E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448–461, August 1973.

B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. 5th Annu. Workshop on Comput. Learning Theory*, pages 144–152. ACM Press, New York, NY, 1992.

L.M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Physics*, 7:200–217, 1967.

T. Bylander. The binary exponentiated gradient algorithm for learning linear functions. In *Proc. 10th Annu. Conf. on Comput. Learning Theory*, pages 184–192. ACM Press, New York, NY, 1997.

Y. Censor and A. Lent. An iterative row-action method for interval convex programming. *Journal of Optimization Theory and Applications*, 34(3):321–353, July 1981.

Y. Censor and S. A. Zenios. *Parallel Optimization.* Oxford, New York, 1997.

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 Transactions on Neural Networks*, 7(2):604–619, May 1996.

T. M. Cover. Universal portfolios. *Mathematical Finance*, 1(1):1–29, 1991.

I. Csiszar. Why least squares and maximum entropy? An axiomatic approach for linear inverse problems. *The Annals of Statistics*, 19(4):2032–2066, 1991.

Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.

C. Gentile and N. Littlestone. The robustness of the p-norm algorithms. In *Proc. 12th Annu. Conf. on Comput. Learning Theory*, pages 1–11. ACM Press, New York, NY, 1999.

A. Grove, N. Littlestone, and D. Schuurmans. General convergence results for linear discriminant updates. *Journal of Machine Learning*, 43(3):173–210, 2001.

D. Haussler, J. Kivinen, and M. K. Warmuth. Sequential prediction of individual sequences under general loss functions. IEEE Transactions on Information Theory, 44(2):1906–1925, September 1998.

D. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth. A comparison of new and old algorithms for a mixture estimation problem. *Machine Learning*, pages 97–119, 1997.

D. P. Helmbold, J. Kivinen, and M. K. Warmuth. Relative loss bounds for single neurons. *IEEE Transactions on Neural Networks*, 10(6):1291–1304, November 1999.

D. P. Helmbold, D. D. E. Long, T. L. Sconyers, and B. Sherrod. Adaptive disk spin-down for mobile computers. *Mobile Networks and Applications*, 5(4):285–297, December 2000.

D. P. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth. On-line portfolio selection using multiplicative updates. *Mathematical Finance*, 8(4):325–347, 1998.

M. Herbster and M. K. Warmuth. Tracking the best expert. *Journal of Machine Learning*, 32(2):151–178, August 1998a. Special issue on concept drift.

M. Herbster and M. K. Warmuth. Tracking the best regressor. In *Proc. 11th Annu. Conf. on Comput. Learning Theory*, pages 24–31. ACM Press, New York, NY, 1998b.

A. Jagota and M. K. Warmuth. Continuous and discrete time nonlinear gradient descent: relative loss bounds and convergence. In R. Greiner E. Boros, editor, *Electronic Proceedings of Fifth International Symposium on Artificial Intelligence and Mathematics*. Electronic,http://rutcor.rutgers.edu/~amai, 1998.

L. Jones and C. Byrne. General entropy criteria for inverse problems, with applications to data compression, pattern classification and cluster analysis. *IEEE Transactions on Information Theory*, 36(1):23–30, 1990.

A. Kalai, S. Chen, A. Blum, and R. Rosenfeld. On-line algorithms for combining language models. In *Proceedings of the International Conference on Accoustics, Speech, and Signal Processing (ICASSP '99)*, 1999.

J. Kivinen and M. K. Warmuth. Additive versus exponentiated gradient updates for linear prediction. *Information and Computation*, 132(1):1–64, January 1997.

J. Kivinen and M. K. Warmuth. Relative loss bounds for multidimensional regression problems. *Journal of Machine Learning*, 45(3):301–329, July 2001.

J. Kivinen, M. K. Warmuth, and P. Auer. The perceptron algorithm vs. winnow: linear vs. logarithmic mistake bounds when few input variables are relevant. *Artificial Intelligence*, 97:325–343, December 1997.

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

N. Littlestone. *Mistake Bounds and Logarithmic Linear-threshold Learning Algorithms*. PhD thesis, Technical Report UCSC-CRL-89-11, University of California Santa Cruz, 1989.

N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

J. Mycielski. A learning algorithm for linear operators. *Proceedings of the American Mathematical Society*, 103(2):547–550, 1988.

R. Rockafellar. *Convex Analysis.* Princeton University Press, 1970.

F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psych. Rev.*, 65:386–407, 1958. (Reprinted in *Neurocomputing* (MIT Press, 1988).).

W. Rudin. *Functional Analysis.* McGraw-Hill, 1991.

S. D. Scott. *Exploring Applications of Learning Theory to Pattern Matching and Dynamic Adjustment of TCP Acknowledgment Delays.* PhD thesis, Washington University in St. Louis, August 1998.

Y. Singer. Switching portfolios. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI98)*, San Francisco, CA, 1998. Morgan Kaufmann.

V. N. Vapnik and A. Y. Chervonenkis. *Teoriya raspoznavaniya obrazov. Statisticheskie problemy obucheniya. [Theory of Pattern Recognition].* Izdat. "Nauka", Moscow, 1974.

V. Vovk. Aggregating strategies. In *Proc. 3rd Annu. Workshop on Comput. Learning Theory*, pages 371–383. Morgan Kaufmann, 1990.

V. Vovk. A game of prediction with expert advice. In *Proc. 8th Annu. Conf. on Comput. Learning Theory*, pages 51–60. ACM Press, New York, NY, 1995.

V. Vovk. Derandomizing stochastic prediction strategies. In *Proc. 10th Annu. Conf. on Comput. Learning Theory*, pages 32–44. ACM Press, New York, NY, 1997.

Appendix: Bregman Divergences and the Generalized Pythagorean Theorem

We prove a series of simple propositions culminating in a generalized Pythagorean Theorem. These results have been proven by many others with slightly varying sets of assumptions. See for example (Bregman, 1967, Csiszar, 1991, Jones and Byrne, 1990, Bauschke and Borwein, 1997, Censor and Zenios, 1997). We begin by giving a more detailed definition of Bregman divergences. Here ri and bd denote the relative interior and relative boundary[9] of a set (Rockafellar, 1970).

Definition 20 *Let F be a function from a closed convex set $E \subseteq \Re^n$ into $\Re$ such that the following three conditions hold:*

(C1) *F is strictly convex.*

9. Intuitively, the relative interior of a set E corresponds to the "inside" of a set, and the relative boundary to the "boundary" of a set. For example, $\mathrm{ri}[0,1] = (0,1)$, and $\mathrm{bd}[0,1] = \{0,1\}$.

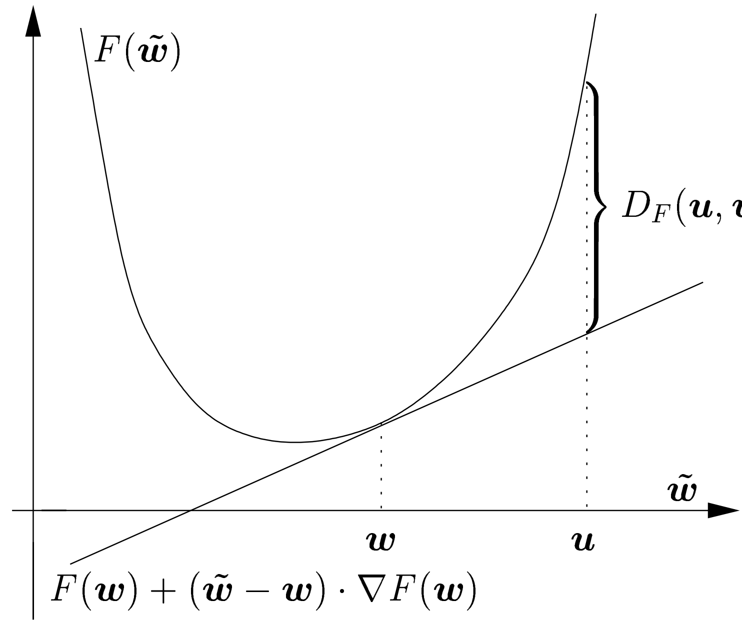

Figure 4: The Bregman divergence

(C2) *F is differentiable.*

(C3) $\forall \boldsymbol{r} \in \mathrm{bd}\, E, \forall \boldsymbol{s}, \boldsymbol{t} \in \mathrm{ri}\, E : (\nabla F(\boldsymbol{r}) - \nabla F(\boldsymbol{s})) \cdot (\boldsymbol{t} - \boldsymbol{r}) < 0.$

Then the Bregman divergence $D_F : E \times \mathrm{ri}\, E \to [0, \infty)$ is defined as

$$D_F(\boldsymbol{u}, \boldsymbol{w}) = F(\boldsymbol{u}) - F(\boldsymbol{w}) - (\boldsymbol{u} - \boldsymbol{w}) \cdot \nabla F(\boldsymbol{w}). \tag{53}$$

The technical Condition (C3) guarantees that any projection of a point in $\mathrm{ri}\, E$ (Definition 1) w.r.t. to the divergence will not lie on the boundary of E. This condition is not required to prove the uniqueness and existence of projections (Proposition 23). However, then the condition is necessary to prove the generalized Pythagorean Theorem 2. Note that the inner product in Condition (C3) is the directional derivative[10] of $D_F(\cdot, \boldsymbol{s})$ at the point $\boldsymbol{r}$ in the direction $\boldsymbol{t} - \boldsymbol{r}$. The direction $\boldsymbol{t} - \boldsymbol{r}$ is the direction from $\boldsymbol{r}$ (any point in $\mathrm{bd}\, E$) to $\boldsymbol{t}$ (any point in $\mathrm{ri}\, E$). Also note that if the domain of F is $\Re^n$, then the condition is trivially satisfied since the $\mathrm{bd}\, \Re^n$ is the empty set.

The divergence $D_F(\boldsymbol{u}, \boldsymbol{w})$ is strictly convex in $\boldsymbol{u}$, since it is the sum of a strictly convex function and a linear function. Similarly, $D_F(\boldsymbol{u}, \boldsymbol{w})$ is differentiable in $\boldsymbol{u}$. Since $D_F(\boldsymbol{u}, \boldsymbol{w})$ is $F(\boldsymbol{u})$ minus the tangent plane of F at $\boldsymbol{w}$ evaluated on $\boldsymbol{u}$ (see Figure 4) the following proposition holds.

Proposition 21 *For all $\boldsymbol{u} \in E$ and $\boldsymbol{w} \in \mathrm{ri}\, E$:*

1. $D_F(\boldsymbol{u}, \boldsymbol{w}) \geq 0$.

10. The directional derivative may be $-\infty$.

2. $\boldsymbol{u} = \boldsymbol{w} \Leftrightarrow D_F(\boldsymbol{u}, \boldsymbol{w}) = 0$.

We define the open and closed balls relative to a point and a divergence as

$$B_\epsilon(\boldsymbol{w}) = \{\boldsymbol{v} : D_F(\boldsymbol{v}, \boldsymbol{w}) < \epsilon\} \text{ and } \overline{B}_\epsilon(\boldsymbol{w}) = \{\boldsymbol{v} : D_F(\boldsymbol{v}, \boldsymbol{w}) \leq \epsilon\}.$$

Since $D_F(\boldsymbol{u}, \boldsymbol{w})$ is continuous in $\boldsymbol{u}$, the closure $B_\epsilon(\boldsymbol{w})$ is equal to $\overline{B}_\epsilon(\boldsymbol{w})$. It is straightforward to check that these balls are strictly convex sets since $D_F(\cdot, \boldsymbol{w})$ is strictly convex.

Proposition 22 *For any point $\boldsymbol{w} \in \operatorname{ri} E$, $\overline{B}_\epsilon(\boldsymbol{w})$ is bounded.*

Proof Suppose $\overline{B}_\epsilon(\boldsymbol{w})$ is unbounded. Then there exists an unbounded sequence of points $S = \langle \boldsymbol{w}+k_1\theta_1, \ldots \boldsymbol{w}+k_i\theta_i, \ldots\rangle$ such that $\boldsymbol{w}+k_i\theta_i \in \overline{B}_\epsilon(\boldsymbol{w})$, $\theta_i \in \{\theta : \|\theta\|_2 = 1\}$, $k_i \in (1, \infty)$, $k_i < k_{i+1}$, and $\lim_{i\to\infty} k_i = \infty$. Consider the related sequence $T = \langle \boldsymbol{w}+\theta_1, \ldots, \boldsymbol{w}+\theta_i, \ldots\rangle$; since the sequence is contained within a compact set it has a convergent subsequence. Now, in order to avoid double subscripts we simply assume that the sequence S has the additional property that the related sequence T is convergent. Let $\theta^*_{\boldsymbol{w}} = \boldsymbol{w} + \lim_{i\to\infty} \theta_i$. Note that $\boldsymbol{w} + \theta_i \in \overline{B}_\epsilon(\boldsymbol{w})$, since it is a convex combination of $\boldsymbol{w}$ and $\boldsymbol{w} + k_i\theta_i$; also, $\theta^*_{\boldsymbol{w}}$ is in $\overline{B}_\epsilon(\boldsymbol{w})$, since it is a limit point of a sequence in the compact set $\overline{B}_\epsilon(\boldsymbol{w}) \cap \{\boldsymbol{w} + \theta : \|\theta\|_2 = 1\}$. We proceed by computing a bound on $D_F(\boldsymbol{w} + \theta_i, \boldsymbol{w})$,

$$\begin{aligned}
D_F(\boldsymbol{w}+\theta_i, \boldsymbol{w}) &= D_F((1 - \frac{1}{k_i})\boldsymbol{w} + \frac{\boldsymbol{w} + k_i\theta_i}{k_i}, \boldsymbol{w}) \\
&< (1 - \frac{1}{k_i}) D_F(\boldsymbol{w}, \boldsymbol{w}) + \frac{1}{k_i} D_F(\boldsymbol{w} + k_i\theta_i, \boldsymbol{w}) \\
&\leq \frac{\epsilon}{k_i}.
\end{aligned}$$

By taking limits we have that

$$D_F(\theta^*_{\boldsymbol{w}}, \boldsymbol{w}) = \lim_{i\to\infty} D_F(\boldsymbol{w} + \theta_i, \boldsymbol{w}) \leq \lim_{i\to\infty} \frac{\epsilon}{k_i} = 0,$$

where $D_F(\theta^*_{\boldsymbol{w}}, \boldsymbol{w}) = \lim_{i\to\infty} D_F(\boldsymbol{w} + \theta_i, \boldsymbol{w})$ since D_F is continuous in the first argument. However, we now have that $D_F(\theta^*_{\boldsymbol{w}}, \boldsymbol{w}) \leq 0$, which contradicts Proposition 21. ■

Recall in Definition 1 the projection of a point $\boldsymbol{w}$ onto a convex set Γ w.r.t. divergence $D_F : E \times \operatorname{ri} E \to [0, \infty)$ is defined as

$$P_{(\Gamma, D_F)}(\boldsymbol{w}) = \operatorname*{arg\,min}_{\boldsymbol{u} \in \Gamma \cap E} D_F(\boldsymbol{u}, \boldsymbol{w}).$$

The uniqueness of the projection essentially follows from the boundedness and strict convexity of the balls, and the convexity of the constraint set.

Proposition 23 *Given $\boldsymbol{w} \in \operatorname{ri} E$ and a closed convex set Γ such that $\Gamma \cap \operatorname{ri} E \neq \emptyset$, then $P_{(\Gamma, D_F)}(\boldsymbol{w})$ exists and is unique.*

Proof Without loss of generality, assume $\Gamma \subseteq E$. We first show existence. Let $A = D_F(\Gamma, \boldsymbol{w})$. Since all points in A are lower bounded by 0, $a = \inf A$ exists. We now prove that a lies in A, which proves the existence of a projection. Let $\delta > 0$. The ball $\overline{B}_{a+\delta}(\boldsymbol{w})$ is closed and bounded. Therefore the intersection $B = \overline{B}_{a+\delta}(\boldsymbol{w}) \cap \Gamma$ is closed and bounded. Since we are in $\Re^n$, this implies that B is compact. Let $C = D_F(B, \boldsymbol{w})$. Since B is compact and $D_F(\cdot, \boldsymbol{w})$ is continuous, it follows that C is compact. Since C is compact, $\inf C$ must lie in C. It follows that a projection exists and all projections lie in $D = \{\boldsymbol{v} : D_F(\boldsymbol{v}, \boldsymbol{w}) = \inf C\} \cap \Gamma$. We now show that the set D has a single element, and hence the projection is unique. Let $\boldsymbol{s}, \boldsymbol{t}$ be distinct elements of D and let $\boldsymbol{u} = \frac{\boldsymbol{s}+\boldsymbol{t}}{2}$. Then $\boldsymbol{u} \in \Gamma$, since Γ is convex. Since $D_F(\cdot, \boldsymbol{w})$ is strictly convex, $D_F(\boldsymbol{u}, \boldsymbol{w}) < \frac{D_F(\boldsymbol{s},\boldsymbol{w})}{2} + \frac{D_F(\boldsymbol{t},\boldsymbol{w})}{2} = \inf C$, which is a contradiction. Hence $P_{(\Gamma, D_F)}(\boldsymbol{w})$ is unique. ∎

We prove a sequence of three propositions. The first and third comprise Theorem 2, the generalization of the Pythagorean Theorem. The following proposition treats the special case of this theorem when Γ is an affine set (second part of Theorem 2).

Proposition 24 *Given a divergence $D_F : E \times \operatorname{ri} E \to [0, \infty)$, an affine set H such that $H \cap \operatorname{ri} E \neq \emptyset$, and points $\boldsymbol{w} \in \operatorname{ri} E$ and $\boldsymbol{u} \in H$, then*

$$D_F(\boldsymbol{u}, \boldsymbol{w}) = D_F(\boldsymbol{u}, P_{(H,D_F)}(\boldsymbol{w})) + D_F(P_{(H,D_F)}(\boldsymbol{w}), \boldsymbol{w}). \tag{54}$$

Proof An affine set in $\Re^n$ may be represented as the intersection of $k \leq n$ hyperplanes. Thus let $H = \cap_{i=1}^{k} \{\boldsymbol{v} : \boldsymbol{v} \cdot \boldsymbol{x}_i = y_i\}$. If $\boldsymbol{w} \in H$, then $P_{(H,D_F)}(\boldsymbol{w}) = \boldsymbol{w}$ and the proposition trivially holds. Assume $\boldsymbol{w} \notin H$. By expanding the D_F (cf. Definition 20) we get the following equivalent form of Equation (54):

$$(\nabla F(\boldsymbol{w}) - \nabla F(P_{(H,D_F)}(\boldsymbol{w}))) \cdot (\boldsymbol{u} - P_{(H,D_F)}(\boldsymbol{w})) = 0. \tag{55}$$

We now proceed to prove an implicit form of the projection. Computing the projection is a convex programming problem, i.e., the computation of $\arg\min_{\boldsymbol{v}} D_F(\boldsymbol{v}, \boldsymbol{w})$ subject to the constraint that $\boldsymbol{v}$ lies in the convex set $E \cap H$. We may ignore the constraint $\boldsymbol{v} \in E$, since technical Condition (C3) ensures that $\boldsymbol{v}$ is in $\operatorname{ri} E$. We introduce a vector of Lagrange multipliers $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_k)$ for the equality constraint that $\boldsymbol{v} \in H$:

$$L(\boldsymbol{\lambda}, \boldsymbol{v}) = D_F(\boldsymbol{v}, \boldsymbol{w}) + \sum_{i=1}^{k} \lambda_i (\boldsymbol{x}_i \cdot \boldsymbol{v} - y_i). \tag{56}$$

Thus for the projection the following equation holds:

$$\begin{aligned} 0 &= \nabla_{\boldsymbol{v}} L(\boldsymbol{\lambda}, \boldsymbol{v}) \mid_{\boldsymbol{v} = P_{(H,D_F)}(\boldsymbol{w})} \\ &= \nabla F(P_{(H,D_F)}(\boldsymbol{w})) - \nabla F(\boldsymbol{w}) + \sum_{i=1}^{k} \lambda_i \boldsymbol{x}_i. \end{aligned}$$

Rewriting the above equation, we have

$$\nabla F(\boldsymbol{w}) - \nabla F(P_{(H,D_F)}(\boldsymbol{w})) = \sum_{i=1}^{k} \lambda_i \boldsymbol{x}_i, \tag{57}$$

and the left-hand-side of (55) becomes $(\boldsymbol{u}-P_{(H,D_F)}(\boldsymbol{w}))\cdot\sum_{i=1}^{k}\lambda_i\boldsymbol{x}_i$. Since $\boldsymbol{x}_i\cdot P_{(H,D_F)}(\boldsymbol{w}) = y_i$ and $\boldsymbol{x}_i\cdot\boldsymbol{u} = y_i$ for all i, this inner product is zero. ■

We next show the generalized Pythagorean Theorem for projections onto halfspaces.

Proposition 25 *Given a divergence $D_F : E\times \mathrm{ri}\, E \to [0,\infty)$, a closed halfspace $\mathcal{H} = \{\boldsymbol{v} : \boldsymbol{x}\cdot\boldsymbol{v}\le y\}$ such that $\mathcal{H}\cap \mathrm{ri}\, E \neq \emptyset$, and points $\boldsymbol{w}\in \mathrm{ri}\, E$ and $\boldsymbol{u}\in\mathcal{H}$, then*

$$D_F(\boldsymbol{u},\boldsymbol{w}) \ge D_F(\boldsymbol{u}, P_{(\mathcal{H},D_F)}(\boldsymbol{w})) + D_F(P_{(\mathcal{H},D_F)}(\boldsymbol{w}),\boldsymbol{w}). \tag{58}$$

Proof The case $\boldsymbol{w}\in\mathcal{H}$ is trivially true. Assume $\boldsymbol{w}\notin\mathcal{H}$. Equation (58) has the following equivalent form:

$$(\nabla F(\boldsymbol{w}) - \nabla F(P_{(\mathcal{H},D_F)}(\boldsymbol{w})))\cdot(\boldsymbol{u} - P_{(\mathcal{H},D_F)}(\boldsymbol{w})) \le 0. \tag{59}$$

The projection of $\boldsymbol{w}$ clearly lies on the boundary of the halfspace, i.e., the hyperplane $H = \{\boldsymbol{v} : \boldsymbol{x}\cdot\boldsymbol{v} = y\}$. Thus $P_{(\mathcal{H},D_F)}(\boldsymbol{w}) = P_{(H,D_F)}(\boldsymbol{w})$. Finding the projection is a convex programming problem subject to an inequality constraint. Since we are now considering only one constraint, Equation 56 is now

$$L(\lambda,\boldsymbol{v}) = D_F(\boldsymbol{v},\boldsymbol{w}) + \lambda(\boldsymbol{x}\cdot\boldsymbol{v} - y).$$

Equation (57) now holds for a single hyperplane H. This allows us to rewrite the left-hand-side of (59) as $(\boldsymbol{u}-P_{(H,D_F)}(\boldsymbol{w}))\cdot\lambda\boldsymbol{x}$. We complete the proof by showing that this expression is at most zero. First note that the projection lies on the boundary H of $\mathcal{H}$, and thus the Lagrange multiplier is positive by the Kuhn-Tucker complementary principle (Rockafellar, 1970). Finally, $\boldsymbol{x}\cdot P_{(H,D_F)}(\boldsymbol{w}) = y$ and $\boldsymbol{x}\cdot\boldsymbol{u}\le y$, so we have that

$$(\boldsymbol{u} - P_{(H,D_F)}(\boldsymbol{w}))\cdot\lambda\boldsymbol{x} \le 0.$$

■

We now prove the first part of the generalized Pythagorean Theorem:

Proposition 26 *Given a divergence $D_F : E\times\mathrm{ri}\, E\to[0,\infty)$, a closed convex set Γ such that $\Gamma\cap\mathrm{ri}\, E\neq\emptyset$, and points $\boldsymbol{w}\in\mathrm{ri}\, E$ and $\boldsymbol{u}\in\Gamma$, then*

$$D_F(\boldsymbol{u},\boldsymbol{w}) \ge D_F(\boldsymbol{u}, P_{(\Gamma,D_F)}(\boldsymbol{w})) + D_F(P_{(\Gamma,D_F)}(\boldsymbol{w}),\boldsymbol{w}). \tag{60}$$

Proof The case $\boldsymbol{w}\in\Gamma$ is again trivially true. Assume $\boldsymbol{w}\notin\Gamma$. Let $\epsilon = D_F(P_{(\Gamma,D_F)}(\boldsymbol{w}),\boldsymbol{w})$. Consider the balls $B_\epsilon(\boldsymbol{w})$ and $\overline{B}_\epsilon(\boldsymbol{w})$. Note that $B_\epsilon(\boldsymbol{w})\cap\Gamma = \emptyset$ and $\overline{B}_\epsilon(\boldsymbol{w})\cap\Gamma = P_{(\Gamma,D_F)}(\boldsymbol{w})$. Let $\mathcal{H}$ be a halfspace that contains Γ but does not intersect with $B_\epsilon(\boldsymbol{w})$. The existence of such a halfspace is implied by the Hahn-Banach separation Theorem (Rudin, 1991). (This theorem says that for any disjoint open and closed sets there exists a halfspace containing the closed set such that this halfspace is disjoint with the open set.)

We now claim that $P_{(\Gamma,D_F)}(\boldsymbol{w}) = P_{(\mathcal{H},D_F)}(\boldsymbol{w})$. Provided this claim is true, the current proposition clearly follows from the previous one. Since $\mathcal{H}\supseteq\Gamma$,

$$D_F(P_{(\mathcal{H},D_F)}(\boldsymbol{w}),\boldsymbol{w}) \le D_F(P_{(\Gamma,D_F)}(\boldsymbol{w}),\boldsymbol{w}) = \epsilon.$$

On the other hand $<$ is impossible because $B_\epsilon(\boldsymbol{w})$ and $\mathcal{H}$ are disjoint. Thus the above inequality must be an equality and the claim follows from the uniqueness of projections. ■