The{dollar}p{dollar}-Norm Generalization of the LMS Algorithm for Adaptive Filtering

Kivinen, Jyrki

2006

Kivinen , J , Warmuth , M K & Hassibi , B 2006 , ' The{dollar}p{dollar}-Norm Generalization of the LMS Algorithm for Adaptive Filtering ' , IEEE Transactions on Signal Processing , vol. 54 , no. 5 , pp. 1782-1793 . https://doi.org/10.1109/TSP.2006.872551

http://hdl.handle.net/10138/27989
https://doi.org/10.1109/TSP.2006.872551

publishedVersion

The p-Norm Generalization of the LMS Algor for Adaptive Filtering

Jyrki Kivinen, Manfred K. Warmuth, and Babak Hassibi

Abstract—Recently much work has been done analyzing online machine learning algorithms in a worst case setting, where no probabilistic assumptions are made about the data. This is analogous to the H^∞ setting used in adaptive linear filtering. Bregman divergences have become a standard tool for analyzing online machine learning algorithms. Using these divergences, we motivate a generalization of the least mean squared (LMS) algorithm. The loss bounds for these so-called -norm algorithms involve other norms than the standard 2-norm. The bounds can be significantly better if a large proportion of the input variables are irrelevant, i.e., if the weight vector we are trying to learn is sparse. We also prove results for nonstationary targets. We only know how to apply kernel methods to the standard LMS algorithm (i.e., $= 2$). However, even in the general -norm case, we can handle generalized linear models where the output of the system is a linear function combined with a nonlinear transfer function (e.g., the logistic sigmoid).

Index Terms—Adaptive filtering, Bregman divergences, H^∞ optimality, least mean squares, online learning.

I. INTRODUCTION

WE focus on the following linear model of adaptive filtering:

$$ \qquad (1)$$

Here is the unknown target, is a known input, is unknown noise, and is the known output signal. We are interested in algorithms that maintain a weight vector based on the past examples $, \quad 1 \quad t$, and, over a sequence of T trials, get as close as possible to the target. As we shall see, closely related online problems have also been studied in machine learning.

More specifically, at trial the algorithm receives and (in order) and has to commit to a weight vector at some point after seeing . We consider three problems depending on whether the algorithm needs to commit to its weight vector before or after seeing and depending on how the loss of the algori is measured.

- A priori filtering: Here we are interested in predicting noncorrupted output before the signal is received. Therefore the algorithm needs to commit to its w vector $_{-1}$ right before seeing and our loss is the energy of the a priori filtering error $- \quad _{-1}$, i.e.,

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \qquad (2)$$

- A posteriori filtering: Here we assume that for estimatin the noncorrupted output , we also have access to the measurement. Thus, the algorithm needs to commit its weight vector only after seeing and the loss is the square of the a posteriori error

$$\sum_{=1}^{T} \quad - \quad ^2 \qquad (3)$$

Note that as in a priori filtering, the algorithm does n know when it produces weight vector . It only knows the past instances and outputs.

- Prediction: Here we are interested in predicting the next observation before receiving it. Thus the algori needs to commit to its weight vector before seeing . The prediction error is $- \quad _{-1}$ and the loss

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \qquad (4)$$

The prediction problem of minimizing (4) is also stu machine learning. Note that in the filtering problems, $-$ is regarded as a disturbance, so we are i ested in estimating the true output of the linear system for the input . In the prediction problem we consider th as the true outcome of some event we are interested dicting. In that case there is no particular value in m prediction at those times when it is inaccurate.

We could also define the notion of a posteriori prediction, i.e., trying to minimize

$$\sum_{=1}^{T} \quad - \quad ^2 \qquad (5)$$

However, since is known when is chosen, the loss (5) is trivially minimized by just choosing such that .

Manuscript received December 1, 2004; revised June 26, 2005. This work was supported by the National Science Foundation under Grant CCR 9821087, the Australian Research Council, the Academy of Finland under Decision 210796, and the IST Programme of the European Community under PASCAL Network of Excellence IST-2002-506778. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Dominic K. C. Ho.

J. Kivinen is with the Department of Computer Science, University of Helsinki, FI-00014 Helsinki, Finland (e-mail: jyrki.kivinen@cs.helsinki.).

M. K. Warmuth is with the Department of Computer Science, University of California Santa Cruz, Santa Cruz, CA 95064 USA (e-mail: manfred@cse.ucsc.edu).

B. Hassibi is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: hassibi@systems.caltech.edu).

Digital Object Identifier 10.1109/TSP.2006.872551

Although there are algorithms that do satisfy in some limiting cases, taking this condition as the primary design principle does not seem to add anything. Hence, we do not further consider the loss (5).

In contrast to the loss function used by the prediction problem, the loss functions for the two ltering problems include the target that is unknown. Because the algorithm cannot even evaluate its own loss, we need to be careful about setting a reasonable performance criterion. We next set the performance criteria we use in this paper, starting with *a priori* ltering and its connection to recent work in machine learning.

Clearly the quality of output depends on the amount of noise, which can be de ned, for example, as $\sum_{=1}^{T} \quad - \quad ^2$. Additionally, even with no noise, the loss (2) for any given algorithm can be made arbitrarily large by scaling . To have a well-de ned choice of , we consider the *regularized loss* $\sum_{=1}^{T} \quad - \quad ^2 \quad 1/\eta \, \| \, \|_2^2$ where $\eta > 0$ is a tradeoff parameter. We then normalize the algorithm s loss (2) with respect to the regularized loss. Since we wish to avoid assumptions about , we consider the worst case choice, leading us to the quantity

$$\max_{\boldsymbol{u}} \frac{\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2}{\sum_{=1}^{T} \quad - \quad ^2 \quad \frac{1}{\eta} \| \, \|_2^2} \tag{6}$$

Given the data and an algorithm for producing , the quantity (6) is always well de ned. In control theory, (6) is seen as a maximum energy gain and called the H$^\infty$ *norm*. (For the above, and as done throughout this paper, we assumed $_0 \quad \mathbf{0}$; if $_0 \; / \; \mathbf{0}$, then $\| \, \|_2^2$ must be replaced by $\| \quad - \quad _0\|_2^2$.)

To get a reference point, consider the least mean squares (LMS) algorithm [2] (also known as the Widrow–Hoff algorithm), de ned by the update rule

$$_{-1} - \eta \quad _{-1} \quad - \tag{7}$$

where $\eta > 0$ is now a parameter of the algorithm and called the learning rate. According to the basic result for *a priori* ltering [3], if $\eta \leq 1/\max \| \quad \|_2^2$, then the LMS algorithm satis es

$$\max_{\boldsymbol{u}} \frac{\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2}{\sum_{=1}^{T} \quad - \quad ^2 \quad \frac{1}{\eta} \| \, \|_2^2} \leq 1 \tag{8}$$

In other words, LMS has H$^\infty$ norm at most 1. (Notice that the learning rate parameter of the algorithm becomes the tradeoff parameter for the regularized loss.) Further, no algorithm can have H$^\infty$ norm less than 1. Therefore, we say that LMS is H$^\infty$ *optimal*.

To compare this with results from machine learning, assume there is a known upper bound X_2 such that $\| \quad \|_2 \leq X_2$ for all t, and write $\eta \quad \alpha/X_2^2$. Then Cesa–Bianchi *et al.* [4] have shown that for $0 < \alpha < 1$

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \leq \frac{1}{1-\alpha} \sum_{=1}^{T} \quad - \quad ^2 \quad \frac{1}{\alpha} X_2^2 \| \quad \|_2^2 \tag{9}$$

To compare prediction with ltering, we write (6) as

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \leq \sum_{=1}^{T} \quad - \quad ^2 \quad \frac{1}{\alpha} X_2^2 \| \quad \|_2^2 \tag{10}$$

where X_2 and η are as above and $0 < \alpha \leq 1$. We see that the bounds are similar in form, except for the factor $1/(1-\alpha)$ in (9).

The factor $1/(1-\alpha)$ in (9) is a source of many dif culties in machine learning, where the goal is to tune the learning rate so as to obtain the smallest possible bound. However, the ltering bound (10) is optimized at $\alpha \quad 1$. Thus we omit the α parameter from the ltering bounds when the norm of instances is bounded.

Motivated by the similarity between (9) and (10), we are going to take machine learning techniques that have recently been used to generalize the LMS algorithm and apply them in the ltering setting. This leads to generalizations of (10) and new interpretations of the ltering algorithms. Techniques we are interested in include:

1) motivating algorithms in terms of minimization problems based on Bregman divergences [5], [6];
2) replacing the 2-norms in the bounds by other norms [5], [7], [8];
3) allowing for nonstationary targets [9] and nonlinear predictors [10].

Before going on with the above program, let us have a brief look at the *a posteriori* model. The H$^\infty$ norm for *a posteriori* ltering is

$$\max_{\boldsymbol{u}} \frac{\sum_{=1}^{T} \quad - \quad ^2}{\sum_{=1}^{T} \quad - \quad ^2 \quad \frac{1}{\eta} \| \, \|_2^2}$$

Notice that since is available when choosing , we can trivially obtain H$^\infty$ norm at most 1 by any choice that satis es . One particular way of doing this would be to let the learning rate go to in nity in the *normalized LMS* algorithm [3]. However, there are other criteria that are minimized by using a nite learning rate, while still retaining the H$^\infty$ norm at most 1. For example, this is the case if the data points are generated by the model (1) with the noise variables independent and Gaussian [3, Theorem 9]. Thus, while requiring the H$^\infty$ norm to be at most 1 is a good robustness guarantee, in the *a posteriori* case such a worst case measure is not by itself a suf cient criterion for choosing a good algorithm. In the following we will state all our bounds both for *a priori* and *a posteriori* ltering, but they must be read with this caveat in mind.

Our H$^\infty$-based performance criteria do not directly address convergence. If the data are generated by the model (1) with the noise variables $_i$ independent and Gaussian, then one could hope that the weights would converge toward the target . However, if we do not wish to make such assumptions about noise, the issue becomes less clear. An algorithm geared toward fast convergence under zero-mean independent noise may fail badly if, say, the early data points have large amounts of biased and correlated noise. We aim for results that are not sensitive to probabilistic assumptions and develop bounds like (6) and (10), which hold for every sequence of examples. Such worst case bounds are rather stringent. If the examples are independent identically distributed (i.i.d.), an averaging technique can be used to convert worst case loss bounds to bounds on the expected loss (see, e.g., [5, Section 8]) or bounds on the probability of high loss [11]. Clearly the choice of algorithm should depend on the assumptions. In particular, even with independent noise, updates like (7) with xed learning rate do not typically lead to convergence but remain oscillating around the optimal weight setting.

In Section III, we introduce Bregman divergences and show how a Bregman divergence can be used to derive two subtly different updates: the *implicit* and *explicit* update. When the squared Euclidean distance is used as the Bregman divergence, these updates give the standard *LMS* and *normalized LMS* algorithm [3], respectively. In Section IV, we give ltering loss bounds for the explicit and implicit updates in the case of Bregman divergences based on squared q-norms [7]. These bounds generalize the results of Hassibi *et al.* [3] about the H^∞ optimality of LMS and normalized LMS for the *a priori* and *a posteriori* ltering problems. The generalization replaces the product $\|\ \|_2\|\ \|_2$ in the bound by another product of dual norms $\|\ \|_p\|\ \|_q$, where p and q are such that $1/p \quad 1/q \quad 1$ and $2 \leq p < \infty$. The new bounds are signi cantly stronger when the target is sparse, i.e., has few nonzero components. In Section V, we generalize the q-norm based algorithms to allow for nonstationary targets . The loss bounds in the nonstationary case include an extra term that depends on the total distance travels during the whole sequence, as measured by the q-norm. Again there are no distribution assumptions about this movement. Section VI gives bounds for generalized linear regression where the linear predictor is fed through a nonlinear transfer function (such as the logistic sigmoid). Some simulations are reported in Section VII, and our conclusions presented in Section VIII.

Some preliminary results of this paper were presented at the 13th IFAC Symposium on System Identi cation [1]. This paper includes some additional algorithms and new simulation results, as well as full proofs of the theoretical results.

II. The LMS Bound

As an introduction to our methods, we rederive the basic result of [3]. Later we will see how the algorithm and proof generalize from the Euclidean to other p-norms.

Theorem 1 [3]: Assume that $\|\ \|_2 \leq X_2$ for all t, and choose $\eta \quad 1/X_2^2$. Then the LMS algorithm (7) satis es

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \leq \sum_{=1}^{T} \quad - \quad ^2 \quad X_2^2\|\ \|_2^2$$

for any $\in \mathbf{R}^n$.

Proof: Following [4], we analyze the *progress* $d \quad 1/2\ \| \ - \ \|_2^2 - 1/2\ \| \ - \ _{-1}\|_2^2$ made at update t toward the *comparison vector* . Direct calculation gives us

$$d \quad \eta \quad - \quad _{-1} \qquad - \quad _{-1}$$
$$-\frac{\eta^2}{2} \quad - \quad _{-1} \quad ^2\|\ \|_2^2$$

By estimating $\|\ \|_2 \leq X_2$ and rearranging terms, we get

$$d \geq \frac{\eta}{2}s^2 - \frac{\eta}{2}r^2 \quad \frac{\eta}{2}\ s \ - r \ ^2\left(1-\eta X_2^2\right)$$

where $s \qquad - \quad _{-1}$ and $r \qquad -$. Since $\eta X_2^2 \quad 1$ and $_0 \quad \mathbf{0}$, we can apply $\| \ - \ _0\|_2 \quad \|\ \|_2$ and $\| \ - \ _{T+1}\|_2 \geq 0$ to get

$$\frac{1}{2}\|\ \|_2^2 \geq \frac{1}{2}\| \ - \ _0\|_2^2 - \frac{1}{2}\| \ - \ _{T+1}\|_2^2$$
$$\sum_{=1}^{T} d$$
$$\geq \frac{1}{2X_2^2}\left(\sum_{=1} s^2 - \sum_{=1} r^2\right)$$

from which the claim follows. ∎

III. Derivation of Algorithms

In this section we give the basic de nitions of Bregman divergences and explain their use in deriving generalizations of the LMS algorithm. (See [12] and references therein for more background on these divergences.) Later the same Bregman divergences will be used to prove bounds for these new algorithms. Note that the bound for the LMS algorithm involves the 2-norms of the inputs and target . The bounds for the new algorithm will depend on norms $\|\ \|_p$ and $\|\ \|_q$ where in general $p, q \ / \ 2$.

Assume that F is a strictly convex twice differentiable function from a subset of $\mathbf{R}^n$ to $\mathbf{R}$. Denote its gradient by $\boldsymbol{f} \quad \nabla F$; notice that $\boldsymbol{f}$ is one-to-one. The *Bregman divergence* Δ_F [13] is de ned for $\ ,\ \in \mathbf{R}^n$ as the error in approximating F by its rst order Taylor polynomial around . More formally

$$\Delta_F \qquad F \quad - F \quad - \quad - \quad \boldsymbol{f}$$

The Bregman divergence Δ_F is always nonnegative, and zero only for . It is (strictly) convex in but might not be convex in . Usually, Δ_F is not symmetric.

Example 1: For $q > 1$, de ne $F \qquad 1/2\ \|\ \|_q^2$, where $\|\ \|_q$ denotes the q-norm de ned as $\|\ \|_q \quad \sum_i |\ _i|^q\ ^{1/q}$. We denote the corresponding Bregman divergence by Δ_q. Thus

$$\Delta_q \qquad \frac{1}{2}\|\ \|_q^2 - \frac{1}{2}\|\ \|_q^2 - \quad - \quad \boldsymbol{f}$$

where the gradient is given by

$$f_i \qquad \frac{\mathrm{sign}\ \ _i\ |\ _i|^{q-1}}{\|\ \|_q^{q-2}}$$

A second important family of Bregman divergences is the relative entropy and its variants.

Example 2: Assume $_i \geq 0$ for all i and de ne $F \quad \sum_i \ _i \ln \ _i - \ _i$, with the usual convention $0 \ln 0 \quad 0$. Then

$$\Delta_F \qquad \sum_i \left(\ _i \ln \frac{_i}{_i} - \ _i \quad _i\right)$$

is the unnormalized relative entropy. (When $\sum_i \quad _i \quad \sum_i \quad _i$ 1, this gives the standard relative entropy.) The gradient is given by $f_i \qquad \ln \quad _i$.

The following generalization of the Pythagorean theorem follows directly from the de nition of a Bregman divergence:

$$\Delta_F \quad ' \quad \Delta_F \qquad \Delta_F \quad ' \qquad - \quad \boldsymbol{f} \quad ' - \boldsymbol{f} \tag{11}$$

Since the dot product $\quad - \quad \boldsymbol{f} \quad ' - \boldsymbol{f}$ can be positive, this shows in particular that Δ_F does not satisfy the triangle inequality. We recover the standard Pythagorean theorem when the divergence is the squared Euclidean distance (i.e., $\boldsymbol{f}$ is identity) and the dot product is zero (i.e., $\quad ' -$ and $-$ are orthogonal).

We now use a Bregman divergence Δ_F as a regularizer for deriving an update rule. This framework for motivating updates was introduced in [5] in the prediction setting. In the following, we are mainly interested in Bregman divergences based on the squared q-norm. They were introduced in [7] to analyze algorithms for learning linear threshold functions.

Suppose an example has been observed and we wish to update our hypothesis $_{-1}$ based on this example. We wish to decrease the squared loss $\quad - \qquad ^2$ (other convex loss functions can also be considered; see Section VI). However, we should not make big changes based on just a single example. Thus, we de ne

$$C \qquad \Delta_F \qquad _{-1} \quad \frac{1}{2}\eta \quad - \qquad ^2$$

where $\eta > 0$ is a tradeoff parameter, and tentatively set $\arg\min_{\boldsymbol{w}} C$. Since C is convex, we can minimize by setting $\nabla C \qquad 0$. By substituting the de nition of Δ_F, this becomes

$$\boldsymbol{f}^{-1} \ \boldsymbol{f} \quad _{-1} - \eta \qquad - \tag{12}$$

Since appears on both sides of (12), we call the update rule de ned by this equality the *implicit update* for divergence Δ_F. Notice that (12) can be solved numerically by a line search since $\boldsymbol{f}^{-1} \ \boldsymbol{f} \quad _{-1} \quad \alpha$ for some scalar α, and the inverse $\boldsymbol{f}^{-1}$ is easy to compute in the cases we consider. Also in the special case of 2-norm $\Delta_F \quad \Delta_2$, with $\boldsymbol{f}$ the identity function, we can solve (12) in closed form to get

$$_{-1} - \frac{\eta}{1 \quad \eta\| \ \|_2^2} \quad _{-1} \quad - \tag{13}$$

This is the algorithm called *normalized LMS* in [3].

Instead of solving (12) numerically, we often nd it suf cient to notice that for reasonable values of η, the values and $_{-1}$ should be fairly close to each other. Thus, we may approximate the solution of (12) by

$$\boldsymbol{f}^{-1} \ \boldsymbol{f} \quad _{-1} - \eta \quad _{-1} \quad - \tag{14}$$

We call this the *explicit update* for divergence Δ_F. The special case $\Delta_F \quad \Delta_2$ gives the usual LMS algorithm.

Note that the explicit update uses the gradient of the square loss evaluated at the old weight vector $_{-1}$, whereas the implicit update is based on the gradient at the updated parameter vector . For a discussion of taking the old gradient versus the future gradient in for the prediction problem, and a derivation of the implicit LMS algorithm, see [5]. In [14], an implicit update was derived as an alternate to the TD λ algorithm. In this case the implicit de nition was crucial for producing an improved algorithm.

IV. Bounds in Terms of Different Norms

Our interest in considering the generalization of LMS to the p-norm based algorithms comes from the fact that for these algorithms, the term $\| \ \|_2\| \ \|_2$ in the LMS bound is replaced by another product of dual norms $\| \ \|_p\| \ \|_q$ (i.e., $1/p \quad 1/q \quad 1$). We discuss the implications of this after giving the main result, which is a direct generalization of Theorem 1.

We consider the explicit (14) and implicit (12) updates for the divergence Δ_q given in Example 1. The special case $q \quad 2$ gives the classic LMS and Theorem 1. For the updates, we need the gradient $\boldsymbol{f}$, which was given in Example 1, and also its inverse $\boldsymbol{f}^{-1}$, which is easily seen to be

$$f_i^{-1} \ \boldsymbol{\theta} \qquad \frac{\text{sign} \ \theta_i \ |\theta_i|^{p-1}}{\|\boldsymbol{\theta}\|_2^{p-2}}$$

where $1/p \quad 1/q \quad 1$.

We assume the relationship $1/p \quad 1/q \quad 1$ throughout this paper. It means that we can apply *H lder s inequality* $| \qquad | \leq \| \ \|_q\| \ \|_p$. As a further convention, we assume $q \leq p$, so $1 < q \leq 2 \leq p < \infty$. The important special case $p \quad q \quad 2$ gives $\Delta_2 \qquad 1/2 \ \| \quad - \quad \|_2^2$, with $\boldsymbol{f}$ the identity function.

We use the following inequality for proving bounds for the updates:

$$\Delta_q\left(\quad \boldsymbol{f}^{-1} \ \boldsymbol{f} \qquad \right) \leq \frac{p-1}{2}\| \ \|_p^2 \tag{15}$$

This inequality is implied by derivations given in [7] and was stated explicitly in [8, Lemma 2]. For completeness, we give the proof in Appendix I.

Theorem 2: Fix p and q such that $1/p \quad 1/q \quad 1$ and $2 \leq p < \infty$. Assume that $\| \ \|_p \leq X_p$ for all t. Then the explicit update (14) for Δ_q with learning rate $\eta \quad 1/ \ p-1 \ X_p^2$ satis es

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \leq \sum_{=1}^{T} \quad - \quad ^2 \quad p-1 \ X_p^2\| \ \|_q^2$$

for any $\in \mathbf{R}^n$.

Proof: Following [5], we analyze the *progress* $d \qquad \Delta_q \quad _{-1} - \Delta_q$ made at update t toward the *comparison vector* . By substituting (14) into (11) and then using (15), we get

$$\begin{aligned} d \quad \eta \quad - \quad _{-1} \qquad - \quad _{-1} - \Delta_q \quad _{-1} \\ \geq \eta \quad - \quad _{-1} \qquad - \quad _{-1} \\ - \frac{p-1}{2}\eta^2 \quad - \quad _{-1} \quad ^2 X_p^2 \end{aligned}$$

By rearranging terms, we can write this as

$$d \ \geq \frac{\eta}{2}s^2 - \frac{\eta}{2}r^2 \quad \frac{\eta}{2} \ s \ - r \quad ^2\left(1 - \eta \ p-1 \ X_p^2\right)$$

where $s \quad - \quad_{-1}$ and $r \quad - \quad$. Since $\eta\ p-1\ X_p^2 \quad 1$ and $\quad_0 \quad \mathbf{0}$, we can apply $\Delta_q \quad_0 \quad 1/2\ \|\ \|_q^2$ and $\Delta_q \quad_{T+1} \geq 0$ to get

$$\frac{\|\ \|_q^2}{2} \geq \Delta_q \quad_0 - \Delta_q \quad_{T+1}$$
$$\sum^{T}_{=1} d$$
$$\geq \frac{1}{2\ p-1\ X_p^2}\left(\sum_{=1} s^2 - \sum_{=1} r^2\right)$$

from which the claim follows. ■

The main intuitive implication of Theorem 2 (and later Theorem 3, which will deal with the implicit update) is that the bound favors large p when the target is sparse. To make this more precise, we compare the bound for $p \quad 2$ (i.e., classic LMS) against $p \quad 2\ln n$ (i.e., fairly large p). Gentile and Littlestone [8, Corollary 7] have shown that for the particular choice $p \quad 2\ln n$, we have

$$p-1\ \|\ \|_p^2\|\ \|_q^2 \leq\ 2\mathrm{e}\ln n\ \|\ \|_\infty^2\|\ \|_1^2 \qquad (16)$$

(where $\|\ \|_\infty \quad \max_i |\ _i|$). Thus, we compare the bound $\|\ \|_2^2\|\ \|_2^2$ (for LMS) with the bound $2\mathrm{e}\ln n\ \|\ \|_\infty^2\|\ \|_1^2$ (for large p).

Since the p-norm is decreasing in p, we have $\|\ \|_2 \leq \|\ \|_1$ and $\|\ \|_2 \geq \|\ \|_\infty$, with equality if the vector has only one nonzero component. Hence, the dependence on favors $p \quad 2$, but the advantage gets smaller if is very sparse. Similarly, the dependence on favors large p, but the advantage gets smaller if is very sparse.

To get a concrete picture of the tradeoff, let us consider two extreme cases. In the rst case, we choose $\in \{-1\ 1\}^n$ and $\in \{-1\ 0\ 1\}^n$ such that exactly one component $_i$ is nonzero. Then $\|\ \|_2^2 \quad n$, $\|\ \|_1^2 \quad n^2$, and $\|\ \|_2 \quad \|\ \|_\infty \quad 1$. The LMS bound becomes simply n, while the large p bound becomes $2\mathrm{e}n^2\ln n$. Hence, the LMS bound is clearly better for large n. In the second case, choose $\in \{-1\ 0\ 1\}^n$ such that exactly one component $_i$ is nonzero, and choose $\in \{-1\ 1\}^n$. Then $\|\ \|_2 \quad \|\ \|_1 \quad 1$, $\|\ \|_2^2 \quad n$, and $\|\ \|_\infty \quad 1$. The LMS bound is n as in the rst case, but the large p bound drops to $2\mathrm{e}\ln n$. Notice that the dependence on n in this last bound is only logarithmic, so for large p the difference to LMS can be quite large.

The above two example scenarios were of course unrealistically extreme. In a typical application, one would expect the components $_i$ of the inputs to have roughly the same magnitude, so the inputs would be relatively dense. Then a large p would be favored if $\|\ \|_1$ is close to $\|\ \|_2$, which is the case if most of the weight in is concentrated on only few components. One should also notice that the upper bounds might not re ect the actual behavior of the algorithms. However, simulations suggest that the picture given here is at least qualitatively correct: the algorithms for $p \quad 2$ and large p are incomparable, and large p is better if the target is sparse. See Section VII for some examples.

In the context of prediction, much attention has been paid to *multiplicative algorithms* such as Winnow [15] and EG [5], which have bounds similar to the p-norm algorithms for $p \quad O\ \log n$. In addition to upper bounds and simulations [5], there are also some lower bounds [16] showing that in certain situations LMS-style algorithms cannot perform as well as multiplicative ones. The multiplicative EG algorithm can be seen as applying the update (14) with $f_i \quad \ln \quad_i$ (with a further normalization step). The analysis of EG can also be lifted to the ltering setting, resulting in the bound

$$\sum^{T}_{=1} \quad - \quad_{-1} \quad ^2 \leq \sum^{T}_{=1} \quad - \quad ^2 \quad \ln 2n\ \|\ \|_\infty^2\|\ \|_1^2$$

for a scaled explicit version. See Appendix II for details and notice the improved constant of $\ln 2n$ over $2e\ln n$ appearing in (16). Multiplicative algorithms are closely related to L_1 regularization, which can be seen as a form of feature selection [17].

We now consider the *a posteriori* case. The following theorem generalizes the result about normalized LMS in [3]. However, our result has an additional restriction on the learning rate, which we believe to be an artefact of the proof technique. We shall discuss this after giving the theorem and its proof.

Theorem 3: Fix p and q such that $1/p \quad 1/q \quad 1$ and $2 \leq p < \infty$. Assume that $\|\ \|_p \leq X_p$ for all t. Then the implicit update for Δ_q with learning rate $\eta \quad 1/\ \ p-1\ X_p^2$ satis es

$$\sum^{T}_{=1} \quad - \quad ^2 \leq \sum^{T}_{=1} \quad - \quad ^2 \quad p-1\ X_p^2\|\ \|_q^2$$

Proof: Again let $d \quad \Delta_q \quad_{-1} - \Delta_q$. By substituting (12) into (11) and applying (15), we get

$$d \quad \eta \quad - \quad\quad - \quad_{-1} - \Delta_q \quad_{-1}$$
$$\geq \eta \quad - \quad\quad - \quad_{-1} \quad - \eta \quad -$$
$$\times \quad - \quad\quad - \frac{p-1}{2}\eta^2 \quad - \quad ^2 X_p^2$$

Since minimizes C, it is easy to show that $\quad_{-1} \leq \quad \leq$ or $\leq \quad \leq \quad_{-1}$; that is, the update moves to the right direction but not too far. This implies

$$- \quad\quad - \quad_{-1} \quad \geq \quad - \quad ^2 \qquad (17)$$

so we get

$$d \geq \eta \quad - \quad ^2 - \eta \quad - \quad\quad -$$
$$- \frac{p-1}{2}\eta^2 \quad - \quad ^2 X_p^2$$

We can rewrite this as

$$d \geq \eta\ s \ - r \quad ^2 \quad \eta\ s \ - r \quad r - \frac{p-1}{2}\eta^2 X_p^2\ s \ - r \quad ^2$$

where $s \quad - \quad$ and $r \quad - \quad$. By rearranging terms, this becomes

$$d \geq \frac{\eta}{2}s^2 - \frac{\eta}{2}r^2 \quad \frac{\eta}{2}\ s \ - r \quad ^2 \left(1 - \eta\ p-1\ X_p^2\right)$$

The rest follows as in the proof of Theorem 2. ■

Our proof actually implies

$$\sum^{T}_{=1} \quad - \quad ^2 \leq \sum^{T}_{=1} \quad - \quad ^2 \quad \frac{1}{\eta}\|\ \|_q^2 \qquad (18)$$

for any learning rate $0 < \eta \leq 1/ \; p-1 \; X_p^2$. For the case $p \quad 2$, Hassibi *et al.* [3] actually show (18) for any $\eta > 0$. Notice that the estimate (17) in our proof can equivalently be written as $\quad _{-1} \quad - \quad / \quad - \quad \geq 0$. This holds as equality for $\eta \quad 0$, but becomes very loose as η approaches in nity (so $\quad - \quad$ approaches zero). In the case $p \quad 2$, we can use the closed form (13) of the normalized LMS algorithm to obtain $\quad _{-1} \quad - \quad / \quad - \quad \eta\| \quad \|_2^2$. Using this tighter estimate allows the proof to go through for arbitrary $\eta > 0$. Unfortunately, we have not been able to obtain a similar bound for the case $p > 2$, with nonlinear $\boldsymbol{f}$ in the update (12).

As discussed in [5], whenever a learning rate η needs to be tuned, then the tuned choice should be of the correct type. As we shall see, this is indeed the case in the above two theorems. We denote the type of the weight vectors as [ ] and the type of the instances as [ ]. The type of the outputs must then be [ ] [ ][ ]. It is easy to check that the transformations $\boldsymbol{f}$ and $\boldsymbol{f}^{-1}$ for Δ_p do not change the type of a weight vector. So now the type of η in the implicit and explicit update for Δ_q must be $[\ ]^{-2}$ and the tunings prescribed in the theorems indeed choose an η of this type. Throughout this paper, our tunings of η always x the type of η for all the updates discussed.

V. NONSTATIONARY TARGETS

Following [9], we now consider a variant of the algorithm that keeps the q-norm of the weight vector bounded by U_q, where $U_q > 0$ is a parameter to the algorithm. We call this two-step update the *bounded explicit update* for Δ_F.

- *Explicit update step:* Let $\quad ' \quad \boldsymbol{f}^{-1} \; \boldsymbol{f} \quad _{-1} - \eta \quad _{-1} \quad -$
- *Out-of-bound update step:* If $\| \quad '\|_q > U_q$, then $U_q \quad '/\| \quad '\|_q$; otherwise $\quad '$.

Thus if the update tries to increase the q-norm of its weight vector above U_q, then we scale it back.

We now let the target vary with time (nonstationary model):

$$ \tag{19}$$

As previously, our bound will include a penalty for the (maximum) norm of . Additionally, there is now also a penalty for the total distance the target moves during the process.

Theorem 4: Fix p and q such that $1/p \quad 1/q \quad 1$ and $2 \leq p < \infty$. Assume $\| \quad \|_p \leq X_p$ and $\| \quad \|_q \leq U_q$ for all t. Then the bounded explicit update for Δ_q with learning rate $\eta \quad 1/ \; p - 1 \; X_p^2$ and parameter U_q satis es

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \leq \sum_{=1}^{T} \quad - \quad ^2 \quad p-1 \; X_p^2 U_q^2$$
$$2 \; p-1 \; X_p^2 U_q \sum_{=1}^{T-1} \| \quad _{+1} - \quad \|_q$$

Proof: We apply the proof technique introduced in the prediction setting in [9]. We de ne the progress at trial t as the sum of three parts $d \quad d^1 \quad d^2 \quad d^3$, where

$$d^1 \quad \Delta_q \quad _{-1} - \Delta_q \quad '$$
$$d^2 \quad \Delta_q \quad ' - \Delta_q$$
$$d^3 \quad \Delta_q \quad - \Delta_q \quad _{+1}$$

Then $d \quad \Delta_q \quad _{-1} - \Delta_q \quad _{+1}$. (For notational convenience we de ne $\quad _{T+1} \quad _T$ for the last time step.)

For $\eta \leq 1/ \; p-1 \; X_p^2$, the proof of Theorem 2 gives directly

$$d^1 \geq \frac{\eta}{2} s^2 - \frac{\eta}{2} r^2 \tag{20}$$

where $s \quad - \quad _{-1}$ and $r \quad - \quad$.

For estimating d^2, rst note that the out-of-bound step can be expressed as

$$\arg\min_{\|\boldsymbol{w}\|_q \leq U_q} \Delta_q \quad '$$

In other words, is the *projection* of $'$ into the closed convex set $B \quad \{ \quad | \; \| \quad \|_q \leq U_q\}$ with respect to Δ_q. Well-known properties of such projections [9], [13] imply that for any $\in B$, we have $\Delta_q \quad \leq \Delta_q \quad '$ and thus $d^2 \geq 0$.

From the de nition of Δ_q, we get

$$d^3 \quad \frac{1}{2}\| \quad \|_q^2 - \frac{1}{2}\| \quad _{+1}\|_q^2 \quad _{+1} - \quad \boldsymbol{f}$$

By H lder s inequality, $| \quad _{+1} - \quad \boldsymbol{f} \quad | \leq \| \quad _{+1} - \quad \|_q \|\boldsymbol{f} \quad \|_p$. Since $\|\boldsymbol{f} \quad \|_p \quad \| \quad \|_q \leq U_q$, we get

$$d^3 \geq \frac{1}{2}\| \quad \|_q^2 - \frac{1}{2}\| \quad _{+1}\|_q^2 - U_q\| \quad _{+1} - \quad \|_q$$

By summing over $t \quad 1 \quad T$ and substituting the value of η, we obtain

$$\Delta_q \quad _1 \quad _0 - \Delta_q \quad _{T+1}$$
$$\sum_{=1}^{T} d$$
$$\geq \frac{1}{2 \; p-1 \; X_p^2} \sum_{=1}^{T} s^2 - \frac{1}{2 \; p-1 \; X_p^2} \sum_{=1}^{T} r^2$$
$$\frac{1}{2}\| \quad _1\|_q^2 - \frac{1}{2}\| \quad _{T+1}\|_q^2 \quad U_q \sum_{=1}^{T} \| \quad _{+1} - \quad \|_q$$

For $\quad _0 \quad \mathbf{0}$, we have $\Delta_q \quad _1 \quad _0 \quad \| \quad _1\|_q^2/2$. Estimating $\Delta_q \quad _{T+1} \quad \geq 0$ and $\| \quad _{T+1}\|_q \leq U_q$ gives the claim. ■

In the special case $\quad _{+1}$ for all t, the result becomes Theorem 2 with the exception that the norm bound U_q must be xed in advance.

The same technique can be applied to the *a posteriori* problem. Given $U_q > 0$, we de ne the *bounded implicit update* for Δ_F with the following two-step update.

- *Implicit update step:* Let be such that

$$\boldsymbol{f}^{-1}\left(\boldsymbol{f}\left(\quad '_{-1}\right) - \eta \quad - \quad \right)$$

- *Out-of-bound update step:* If $\| \quad \|_q > U_q$, then $' \quad U \quad /\| \quad \|_q$; otherwise $'$.

Thus, we swapped the notation from the explicit update and use $'$ for the bounded and for the unbounded weight. Basically we now want to predict with the unbounded weights. The bound is as expected.

Theorem 5: Fix p and q such that $1/p \quad 1/q \quad 1$ and $2 \leq p < \infty$. Assume $\| \quad \|_p \leq X_p$ and $\| \quad \|_q \leq U_q$ for all t. Then the bounded implicit update for Δ_q with learning rate $\eta \quad 1/ \quad p - 1 \; X_p^2$ and parameter U_q satis es

$$\sum_{=1}^{T} \quad - \quad ^2 \leq \sum_{=1}^{T} \quad - \quad ^2$$

$$p-1 \; X_p^2 U_q^2 \quad 2 \; p-1 \; X_p^2 U_q \sum_{=1}^{T-1} \| \quad _{+1} - \quad \|_q$$

Proof: We mimic the proof of Theorem 4. This time we set

$$d^1 \quad \Delta_q \left(\quad '_{-1} \right) - \Delta_q$$
$$d^2 \quad \Delta_q \quad - \Delta_q \quad '$$
$$d^3 \quad \Delta_q \quad ' - \Delta_q \quad _{+1} \quad '$$

For $\eta \leq 1/ \quad p-1 \; X_p^2$, the proof of Theorem 3 gives

$$d^1 \geq \frac{\eta}{2} s^2 - \frac{\eta}{2} r^2$$

where $s \quad -$ and $r \quad -$. We estimate d^2 and d^3 and sum over t exactly as in the proof of Theorem 4. ∎

All the previous bounds are for algorithms that use a constant learning rate that needs to be set at the beginning, and the optimal choice depends on the norms of the instances, which may not be known in advance. We close this section by considering a variant where we use a variable learning rate based on the norms of instances seen thus far. For simplicity, we deal only with the explicit update case.

Thus, de ne the *explicit update with variable learning rate* as

$$' \quad \boldsymbol{f}^{-1} \; \boldsymbol{f} \quad _{-1} - \eta \quad _{-1} \quad -$$

where now η is a time-dependent learning rate. The out-of-bound update is as before.

The bound proven below is identical to the xed η version given in Theorem 4 except for an additional factor of ve in the second term on the right-hand side.

Theorem 6: Fix p and q such that $1/p \quad 1/q \quad 1$ and $2 \leq p < \infty$. Let $\eta \quad 1/ \quad p-1 \; X_{p,}^2$ where $X_{p,} \quad \max_{\leq} \| \quad \|_p$. Assume $\| \quad \|_q \leq U_q$ for all t. Then the bounded explicit update for Δ_q with the variable learning rate $\eta \quad 1/ \quad p-1 \; X_{p,}^2$ and parameter U_q satis es

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \leq \sum_{=1}^{T} \quad - \quad ^2 \quad 5 \; p-1 \; X_{p,T}^2 U_q^2$$

$$2 \; p-1 \; X_{p,T}^2 U_q \sum_{=1}^{T-1} \| \quad _{+1} - \quad \|_q$$

Proof: We modify the proof of Theorem 4 using the method of [18] for handling the variable learning rate. Fortunately, in ltering, the technicalities are much easier than in the prediction setting.

Thus, we consider the quantity $\Delta_q \quad _{-1} \; /\eta$. By replacing η with η in (20), we see that the proof of Theorem 4 implies

$$\frac{\Delta_q \quad _{-1}}{\eta} - \frac{\Delta_q \quad _{+1}}{\eta} \geq \frac{s^2}{2} - \frac{r^2}{2} \quad \frac{1}{2\eta} \| \quad \|_q^2$$
$$- \frac{1}{2\eta} \| \quad _{+1} \|_q^2 - \frac{U_q}{\eta} \| \quad _{+1} - \quad \|_q$$

where $s \quad - \quad _{-1}$ and $r \quad -$. (Again we set $_{T+1} \quad _T$; also let $X_{p,T+1} \quad X_{p,T}$.) By substituting $\eta \quad 1/ \quad p-1 \; X_{p,}^2$ and then noticing that $X_{p,} \leq X_{p,+1} \leq X_{p,T}$, we get

$$p-1 \; X_{p,}^2 \quad \Delta_q \quad _{-1} - \Delta_q \quad _{+1}$$
$$\geq \frac{s^2}{2} - \frac{r^2}{2} \quad \frac{p-1}{2} X_{p,}^2 \| \quad \|_q^2 - \frac{p-1}{2} X_{p,+1}^2 \| \quad _{+1} \|_q^2$$
$$- \; p-1 \; X_{p,T}^2 U_q \| \quad _{+1} - \quad \|_q$$

By [18, Lemma 3.2], we have $\Delta_q \quad ' \leq 2V^2$ whenever $\| \quad \|_q \leq V$ and $\| \quad ' \|_q \leq V$, so in particular $\Delta_q \quad _{+1} \leq 2U_q^2$. Remembering that $X_{p,}^2 - X_{p,+1}^2 \leq 0$, we get

$$p-1 \; X_{p,}^2 \; \Delta_q \quad _{-1} - \; p-1 \; X_{p,+1}^2 \Delta_q \quad _{+1}$$
$$p-1 \; X_{p,}^2 \quad \Delta_q \quad _{-1} - \Delta_q \quad _{+1}$$
$$p-1 \; \left(X_{p,}^2 - X_{p,+1}^2 \right) \Delta_q \quad _{+1}$$
$$\geq \frac{s^2}{2} - \frac{r^2}{2} \quad \frac{p-1}{2} X_{p,}^2 \| \quad \|_q^2 - \frac{p-1}{2} X_{p,+1}^2 \| \quad _{+1} \|_q^2$$
$$- \; p-1 \; X_{p,T}^2 U_q \| \quad _{+1} - \quad \|_q$$
$$2 \; p-1 \; \left(X_{p,}^2 - X_{p,+1}^2 \right) U_q^2$$

By summing over $t \quad 1 \quad T$, we get

$$p-1 \; X_{p,1}^2 \Delta_q \quad _1 \quad _0 - \; p-1 \; X_{p,T+1}^2 \Delta_q \quad _{T+1} \quad _T$$
$$\geq \frac{1}{2} \sum_{=1}^{T} s^2 - \frac{1}{2} \sum_{=1}^{T} r^2 \quad \frac{p-1}{2} X_{p,1}^2 \| \quad _1 \|_q^2$$
$$- \frac{p-1}{2} X_{p,T+1}^2 \| \quad _{T+1} \|_q^2 - \; p-1 \; X_{p,T}^2 U_q$$
$$\times \sum_{=1}^{T} \| \quad _{+1} - \quad \|_q \quad 2 \; p-1 \; \left(X_{p,1}^2 - X_{p,T+1}^2 \right) U_q^2$$

The result follows by solving for $\sum_{=1}^{T} s^2$, noticing $\Delta_q \quad _1 \quad _0 \quad \| \quad _1 \|_q^2 / 2$ and then ignoring the negative terms $- \; p-1 \; X_{p,T+1}^2 \Delta_q \quad _{T+1} \quad _T$ and $4 \; p-1 \; X_{p,1}^2 U_q^2$. ∎

VI. Generalized Linear Models

We extended framework slightly to cover *generalized linear regression*. Here we replace the model (1) by

$$h \tag{21}$$

where h is a continuous, strictly increasing *transfer function*. The logistic sigmoid $h \; r \quad 1/ \; 1 \quad \exp -r$ is a typical example. In the prediction setting (where the learner tries to match), the prediction becomes $\hat{} \quad h \quad _{-1}$. In the ltering setting, we would naturally also include the transfer function in the prediction, giving $\hat{} \quad h \quad _{-1}$ for the *a priori* and $\hat{} \quad h$ for the *a posteriori* case. The algorithm then tries to match $\hat{}$ to h . One could in principle still use the squared error $h \quad - h \quad ^2$ as the performance measure, but this is nonconvex in and and actually leads to a very badly behaved optimization problem [19]. We obtain a better behaved problem by using the *matching loss* for h [19], de ned for and $'$ in the range of h as

$$L \quad ' \quad \int_{h^{-1}(y)}^{h^{-1}(y')} h \; r \; - \quad dr \tag{22}$$

(Notice that by our assumptions h is one-to-one.) It is easy to see that for the identity transfer function $h \; r \quad r$, we get

$L \quad ' \quad - \; '\,^2/2$; and for the logistic sigmoid $h\; r$ $1/\;1 \quad \exp\, -r$, we get the logarithmic loss

$$L \quad ' \qquad \ln \frac{}{'} \qquad 1 - \quad \ln \frac{1-}{1-\;'}$$

The de nition (22) may seem arbitrary, but it is actually a one-dimensional Bregman divergence: if we let $H\; r \qquad \int h\; r\; dr$, then

$$L\;\; h\; a \quad h\; a' \qquad \Delta_H\;\; a'\;\; a \tag{23}$$

Using a Bregman divergence as a loss naturally generalizes to multidimensional outputs [6], but we shall not pursue that here.

Directly from (22), we obtain a simple expression for its gradient

$$\nabla_{\boldsymbol{w}} L \qquad h \qquad\qquad h \qquad - \tag{24}$$

Therefore, the explicit update (14) naturally generalizes to

$$\boldsymbol{f}^{-1} \;\; \boldsymbol{f} \qquad _{-1} - \eta \;\; \hat{} \;\; -$$

where $\hat{}\quad h \quad _{-1}$. The implicit update can be generalized similarly; for it we use $\hat{} \quad h$. For these updates we can now prove bounds that have as an additional factor an upper bound on the slope of the transfer function. The techniques are essentially those introduced by [10].

Theorem 7: Fix p and q such that $1/p \quad 1/q \quad 1$ and $2 \le p < \infty$. Let h be strictly increasing and continuously differentiable with c such that $0 < h\; r \;\; \le c$ holds for all r, and let L be the matching loss for h. Assume that $\| \quad \|_p \le X_p$ for all t. Then both the explicit update and implicit update for Δ_q with learning rate $\eta \quad 1/\;\; p-1\;\; cX_p^2$ satisfy

$$\sum_{=1}^{T} L \;\; \hat{} \;\; h \qquad \le \sum_{=1}^{T} L \qquad h \qquad\qquad p-1\;\; cX_p^2 \| \quad \|_q^2$$

for any $\quad \in \mathbf{R}^n$.

Proof: Consider rst the explicit update. As in the proof of Theorem 2, let

$$d \quad \Delta_q \quad _{-1} \; - \Delta_q \qquad \eta \quad - \;\hat{}\; \qquad - \quad _{-1} \qquad - \Delta_q \quad _{-1}$$

Using (23), we get

$$- \;\hat{}\; \qquad - \quad _{-1} \qquad L \;\; \hat{}\;\; h \qquad -L \quad h \qquad\quad L \quad \hat{}$$

Simple calculus shows that $L \quad \hat{} \;\; \ge \;\; 1/2c \quad - \;\hat{}\;^2$ for all and $\hat{}$. By combining this with (15), we get

$$d \;\; \ge \eta \;\; L \;\; \hat{} \;\; h \qquad -L \quad h \qquad \frac{\eta}{2} \quad - \;\hat{}\;\; ^2 \left(\frac{1}{c} - \eta \;\; p-1 \;\; X_p^2 \right)$$

The claim follows by summing over t as usual.

Consider now the implicit update. We have

$$d \quad \Delta_q \quad _{-1} \; - \Delta_q \qquad \eta \quad - \;\hat{}\; \qquad - \quad _{-1} \qquad - \Delta_q \quad _{-1}$$

where now $\hat{} \quad h$. We write

$$- \;\hat{}\; \qquad - \quad _{-1} \qquad - \;\hat{}\; \qquad - \qquad - \;\hat{}\; \qquad _{-1} \quad -$$

Like above, we have

$$- \;\hat{}\; \qquad - \qquad L \;\; \hat{} \;\; h \qquad -L \quad h \qquad L \quad \hat{}$$

Also, since is the solution to

$$\arg\min_{\boldsymbol{w}} \;\; \Delta_q \qquad _{-1} \qquad \eta L \qquad h$$

we have either $\le \qquad \le \quad _{-1}$ or $_{-1} \quad \le \qquad \le$. In either case, $- \;\hat{}\; \quad _{-1} \quad - \qquad \le 0$. Hence, we have established

$$d \;\; \ge \eta \;\; L \;\; \hat{} \;\; h \qquad -L \quad h \qquad L \quad \hat{} \quad - \Delta_q \quad _{-1}$$

and can proceed as with the explicit update. ∎

Because of how we de ned $\hat{}$, the theorem gives an *a priori* ltering bound for the explicit update and *a posteriori* bound for the implicit update.

When h is the identity function, we get the results of Section IV with $c \quad 1$. For the logistic sigmoid, $c \quad 1/4$. Thresholded transfer functions, such as $h\; r \quad \text{sign}\; r$, correspond to the limiting case $c \to \infty$, which makes the bound vacuous.

This result generalizes to the nonstationary case (Section V) in the obvious manner; we omit the details.

Our main motivation for considering loss functions other than square loss was that they make the problem involving a nonlinear transfer function computationally simpler, which also allows strong worst case bounds. One might also prefer different loss functions if one assumes a non-Gaussian noise distribution [20]. This is quite different from our framework, where no statistical assumptions are made.

VII. Simulation Results

The discussion following Theorem 2 suggests that having a sparse target favors having a large p. We illustrate this with a simple ltering simulation.

At time t, the sender sends a bit $\quad \in \{-1 \;\; 1\}$ over a channel. The recipient is required to produce a binary prediction $\hat{} \;\; \in \{-1 \;\; 1\}$ about the sent bit. If $\hat{} \;\; / \quad$, we say that an error occurred. What the recipient actually observes is

$$r \qquad \sum_{i=0}^{k-1} \quad _{i+1} \quad _{-i}$$

where $\quad \in \mathbf{R}^k$ for some k describes the channel and is zero-mean Gaussian noise. The prediction is then $\hat{} \quad \text{sign} \quad _{-1}$, where $\quad r \;\; _{-m} \qquad r \qquad r \;\; _{+m} \;\; \in \mathbf{R}^{2m+1}$ and n $2m \quad 1$ is the lter length.

Notice that this setting is not quite the same as introduced earlier, since we are now considering discrete errors but still using the update rules based on square loss. The purpose of this is to illustrate how the algorithms work on binary prediction, which often is the problem one is really interested in.

For choosing , we considered two different distributions. In the rst experiment, is from a Gaussian with unit variance. In the second experiment, $\quad _i \quad s_i e^{r_i}$, where $s_i \in \{-1 \;\; 1\}$ and $r_i \in [-10 \;\; 10]$ are distributed uniformly. In both cases, we then renormalize to make $\| \quad \|_2 \quad 1$. The targets from the second distribution are sparse in the sense that most of the weight is concentrated on only few components, whereas the targets from the rst distribution are dense. In both experiments, we used $k = 10$, $m = 15$ and a signal-to-noise ratio of 10 dB. We compared the explicit update algorithm with $p = 2$ against $p = 2 \ln n \approx 6.9$. (As we remarked after Theorem 2, for $p = 2 \ln n$ we can estimate $(p-1)\| \|_p^2 \leq 2e \ln n \| \|_\infty^2$.)

Notice that due to the constant learning rate, the weight vectors of the algorithms end up oscillating around the optimum, so the algorithms converge to a nonzero error rate. By using a smaller learning rate, one can reduce the oscillations and thus achieve a smaller final error rate, but this makes the initial convergence slower. The choice of learning rate is thus not straightforward.

We used for $p = 2 \ln n$ the value $\eta = 1/((p-1)X_p^2)$ as suggested by Theorem 2. This gave final error rates 0.02 in the first experiment and 0.01 in the second one. For $p = 2$ we then chose η so that these same final error rates were achieved. For the first experiment, this resulted in $\eta = 0.45/X_2^2$, and for the second one, $\eta = 0.4/X_2^2$.

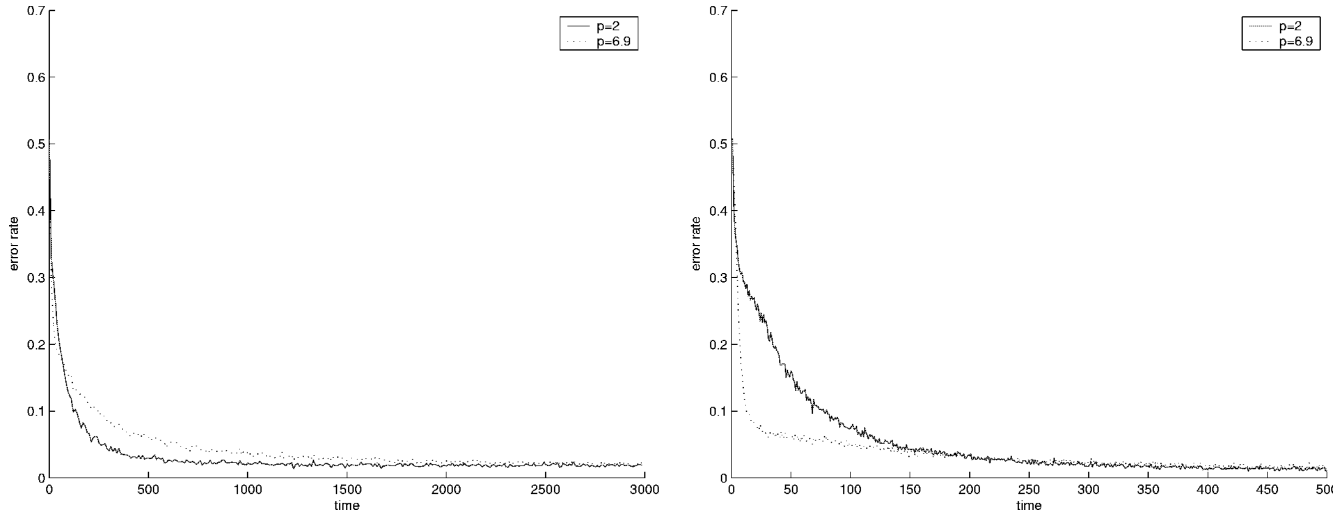
The development of the error rates over time is shown in Fig. 1. As expected, $p = 2$ gives a faster convergence for dense targets and $p = 2 \ln n$ for sparse targets. The differences here are not large, but they become more apparent if the filter length (i.e., dimensionality of inputs) is increased.

We did not include the implicit updates in this comparison. In other experiments we noticed that for any fixed p and η, the implicit update has slower initial convergence and smaller final error rate than the explicit one. This can be understood by noticing that by (17), the implicit update always makes a smaller step. Hence, as a crude first approximation, the implicit update is similar to the explicit update with a smaller η.

VIII. Discussion and Conclusion

We have shown how Bregman divergences based on p-norms can be used to derive generalizations of the classical LMS algorithm. This is a direct application of methods recently introduced in machine learning. The resulting p-norm algorithms have for large p quite different behavior from the LMS, which is the special case $p = 2$. In particular, both theoretical bounds and preliminary simulations suggest that the large p version has better performance when the target weight vector is sparse. We apply further methods from machine learning to show that also in filtering, the p-norm algorithms can be made robust against target shift and can be adapted for generalized linear systems.

The question of applying these techniques to genuinely nonlinear problems remains unsolved. Recently much work has been done in machine learning on applying linear algorithm to nonlinear problems using the so-called kernel trick. This trick works for a large class of algorithms, such as LMS, the support vector machine, or more generally any *rotation invariant* algorithm [16], [17], [21]. The p-norm algorithm for $p \neq 2$ is not rotation invariant, and it remains an open problem whether it can be efficiently nonlinearized with some technique analogous to the kernel trick. For algorithms with similar performance to the p-norm algorithm with large p, efficient techniques have been found for some kernels [22], but for other kernels the problem is known to be intractable [23]. Further, the computational requirements in signal processing applications may even rule out kernel-style approaches that rely on storing a large number of data points. Thus, the prospects of finding a general nonlinear version of the p-norm algorithms do not seem good.



(a) (b)

Fig. 1. Error rates as function of time for $p = 2$ (solid line) and $p \approx 6.9$ (dotted line) in the ltering simulation. The error rates are averages over 5000 runs. The experiments were run for 30000 time steps to make sure the algorithms converged to same error rate; the plots show only the initial part. (a) Dense target and (b) sparse target.

APPENDIX I
PROOF OF (15)

Since $1/p \quad 1/q \quad 1$, a straightforward calculation shows that $\| \quad \|_p \quad \|\boldsymbol{f} \quad \|_q$ and $\quad \boldsymbol{f} \qquad \| \quad \|_q^2$ for all $\quad \in \mathbf{R}^n$ [8, Lemma 1]. Fix now $\boldsymbol{\theta} \quad \boldsymbol{f}$ and $\boldsymbol{\theta}' \quad \boldsymbol{f} \quad '$, with $\boldsymbol{\theta}' - \boldsymbol{\theta}$. Based on the above, it is easy to verify that $\Delta_q \qquad '$ $\Delta_p \ \boldsymbol{\theta}' \ \boldsymbol{\theta}$. (Notice the order of the arguments.) Let $G \ \boldsymbol{\theta}$ $1/2 \ \|\boldsymbol{\theta}\|_p^2$. Since Δ_p is de ned as the error of a rst-order Taylor approximation for G, we can write

$$\Delta_p \ \boldsymbol{\theta} \qquad \boldsymbol{\theta} \quad \frac{1}{2} \ {}^{\mathrm{T}}H \tag{25}$$

where $H_{ij} \quad \partial^2 G \ \boldsymbol{\xi} \ /\partial\xi_i\partial\xi_j$ and the derivatives are evaluated at some point $\boldsymbol{\xi}$ on the line between $\boldsymbol{\theta}$ and $\boldsymbol{\theta}$. We now estimate the right-hand side of (25) as in [7, Theorem 7.1]. We have

$$\begin{aligned} H_{ij} \quad & 2 - p \ \mathrm{sign} \ \xi_i \ |\xi_i|^{p-1} \mathrm{sign} \ \xi_j \ |\xi_j|^{p-1} \|\boldsymbol{\xi}\|_p^{2-2p} \\ & \qquad \delta_{ij} \ p - 1 \ |\xi_i|^{p-2} \|\boldsymbol{\xi}\|_p^{2-p} \end{aligned}$$

Since we assume $p \geq 2$, we get

$$\begin{aligned} {}^{\mathrm{T}}H \quad & 2 - p \ \|\boldsymbol{\xi}\|_p^{2-2p} \left(\sum_i \mathrm{sign} \ \xi_i \ |\xi_i|^{p-1} \ {}_i \right)^2 \\ & p - 1 \ \|\boldsymbol{\xi}\|_p^{2-p} \left(\sum_i |\xi_i|^{p-2} \ {}_i^2 \right) \\ \leq \ & p - 1 \ \|\boldsymbol{\xi}\|_p^{2-p} \tilde{\boldsymbol{\xi}} \quad \tilde{} \end{aligned}$$

where $\tilde{\xi}_i \quad |\xi_i|^{p-2}$ and $\tilde{}_i \quad {}_i^2$. Since $1/ \ p/ \ p - 2$ $1/ \ p/2 \quad 1$, H lder s inequality gives us

$$|\tilde{\boldsymbol{\xi}} \quad \tilde{}| \leq \|\tilde{\boldsymbol{\xi}}\|_{\frac{p}{(p-2)}} \|\tilde{}\|_{\frac{p}{2}} \quad \|\boldsymbol{\xi}\|_p^{p-2} \| \quad \|_p^2$$

and the claim follows.

APPENDIX II
EXPONENTIATED GRADIENT

As in Example 2, the relative entropy can be seen as a Bregman divergence. The constraint $\sum_i \ {}_i \quad 1$ requires some additional technicalities. We present here a fairly straightforward method. For a more general framework allowing potential functions F that are not *strictly* convex, see [6].

For and $'$ with ${}_i$, ${}'_i \geq 0$, and $\sum_i \ {}_i \quad \sum_i \ {}'_i \quad 1$, de ne the relative entropy

$$\Delta_{\mathrm{re}} \quad ' \qquad \sum_{i=1}^n \ {}_i \ln \frac{{}_i}{{}'_i}$$

(We take $0 \ln 0 \quad 0$ and $\ln 0 \quad \infty$ otherwise.) Notice that $\Delta_{\mathrm{re}} \quad '$ is convex in . Consider minimizing

$$C \qquad \Delta_{\mathrm{re}} \qquad {}_{-1} \quad \frac{1}{2}\eta \quad - \qquad {}^2$$

subject to $\sum_i \ {}_i \quad 1$, ${}_i \geq 0$ for all i. The problem is convex, so we solve it by setting the gradient of the Lagrangian

$$A \qquad \Delta_{\mathrm{re}} \qquad {}_{-1} \quad \lambda \left(\sum_{i=1}^n \ {}_i - 1 \right) \ \frac{1}{2}\eta \quad - \qquad {}^2$$

to zero. This yields

$$\ln \frac{{}_i}{{}_{-1,i}} \quad 1 \quad \lambda \quad \eta \qquad - \quad {}_i \quad 0$$

or (after substituting λ such that $\sum_i \ {}_i \quad 1$)

$${}_i \qquad \frac{{}_{-1,i} \exp \ -\eta \qquad - \qquad {}_{,i}}{Z} \tag{26}$$

where $Z \quad \sum_{i=1}^n \ {}_{-1,i} \exp \ -\eta \qquad - \qquad {}_{,i}$. Notice that ${}_{,i} > 0$ implies ${}_i > 0$.

For $\boldsymbol{z} \in \mathbf{R}^n$, de ne now $\boldsymbol{g}\ \boldsymbol{z}$ by

$$g_i\ \boldsymbol{z} \quad \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \tag{27}$$

Let $\boldsymbol{z}\ _{-1}$ be such that $\boldsymbol{g}\ \boldsymbol{z}\ _{-1}$ $_{-1}$. It is easy to see that such a $\boldsymbol{z}\ _{-1}$ exists assuming $_{-1,i} > 0$ for all i and $\sum_{i=1}^{n}$ $_{-1,i}$ 1. Further, if $\boldsymbol{z}'_{-1}$ is another vector satisfying $\boldsymbol{g}\ \boldsymbol{z}'_{-1}$ $_{-1}$, then $\boldsymbol{z}\ _{-1}$ and $\boldsymbol{z}'_{-1}$ are the same up to an additive constant, i.e., $z\ _{-1,i}$ z'_{-1} b for some b that does not depend on i. Equation (26) can now be written as $\boldsymbol{g}\ \boldsymbol{z}$, where $\boldsymbol{z}$ $\boldsymbol{z}\ _{-1} - \eta$ $-$. Notice that because of the normalization, the choice of the representative $\boldsymbol{z}\ _{-1}$ (i.e., the constant b) makes no difference.

Again, we de ne the implicit and explicit version of the update. We use an additional parameter vector $\boldsymbol{z}$ to present the algorithm, the actual weights being given by $\boldsymbol{g}\ \boldsymbol{z}$. In both cases, we start with $\boldsymbol{z}_1$ $\mathbf{0}$. For the *implicit exponentiated gradient* algorithm, we de ne $\boldsymbol{z}$ by

$$\boldsymbol{z} \quad \boldsymbol{z}\ _{-1} - \eta \qquad -$$

and for *explicit exponentiated gradient* (EG) algorithm by

$$\boldsymbol{z} \quad \boldsymbol{z}\ _{-1} - \eta \quad _{-1} \qquad -$$

Thus the implicit update uses as the minimizer of C , while the explicit update uses an approximation thereof. These updates are analogous to the implicit and explicit updates given previously, with $\boldsymbol{g}$ now replacing $\boldsymbol{f}^{-1}$. However, in this case $\boldsymbol{g}$ is not one-to-one, so we write the update in terms of $\boldsymbol{z}$ (which corresponds to $\boldsymbol{f}$ in the previous setting) and not directly in terms of .

The following lemma gives the analogues of (11) and (15) for relative entropy.

Lemma 1: Let $\boldsymbol{g}\ \boldsymbol{z}$ and $'$ $\boldsymbol{g}\ \boldsymbol{z}'$ for some $\boldsymbol{z}, \boldsymbol{z}' \in \mathbf{R}^n$. Then

$$\Delta_{\rm re} \quad ' \le \frac{1}{8}\left(\max_i\ z'_i - z_i\ - \min_i\ z'_i - z_i\ \right)^2 \tag{28}$$

and for any $\in \mathbf{R}^n$ with $_i \ge 0$ and $\sum_i$ $_i$ 1, we have

$$\Delta_{\rm re} \quad ' \quad \Delta_{\rm re} \qquad \Delta_{\rm re} \quad ' \quad \boldsymbol{z}' - \boldsymbol{z} \qquad - \tag{29}$$

Proof: Equation (29) follows directly from the de nition. To prove (28), we rst write

$$\Delta_{\rm re} \quad ' \quad G\ \boldsymbol{z}'\ -\ G\ \boldsymbol{z} \quad \boldsymbol{g}\ \boldsymbol{z} \quad \boldsymbol{z}' - \boldsymbol{z}$$

where $G\ \boldsymbol{z}$ $\ln \sum_i e^{z_i}$. Notice that $\boldsymbol{g}$ ∇G. Therefore, $\Delta_{\rm re}$ $'$ is the error in the rst-order Taylor approximation of $G\ \boldsymbol{z}'$ around $G\ \boldsymbol{z}$, and we have $\Delta_{\rm re}$ $'$ $1/2$ $\boldsymbol{z}' - \boldsymbol{z}\ ^T H\ \boldsymbol{z}' - \boldsymbol{z}$, where H is the Hessian of G evaluated at some point between $\boldsymbol{z}$ and $\boldsymbol{z}'$. We have

$$\frac{\partial^2 G\ \boldsymbol{z}}{\partial z_i \partial z_j} \quad \frac{\partial g_i\ \boldsymbol{z}}{\partial z_j} \quad \delta_{ij} g_i\ \boldsymbol{z}\ - g_i\ \boldsymbol{z}\ g_j\ \boldsymbol{z}$$

Therefore we can write H_{ij} $\delta_{ij} p_i - p_i p_j$ for some $\boldsymbol{p}$ that satis es $p_i > 0$ and $\sum_i p_i$ 1. Denote now by X a random variable that is obtained by choosing the value $_i$ $z'_i - z_i$ with probability p_i. Then

$$\begin{aligned} \boldsymbol{z}' - \boldsymbol{z}\ ^T H\ \boldsymbol{z}' - \boldsymbol{z} \quad & \sum_i p_i\ {}_i^2 - \sum_{i,j}\ {}_i\ {}_j p_i p_j \\ & \mathrm{E}[X^2] - \mathrm{E}[X]^2 \\ & \mathrm{Var}[X] \\ & \le \frac{1}{4}\ \max_i\ {}_i - \min_i\ {}_i\ ^2 \end{aligned}$$

■

Theorem 8: Assume that $\max_i\ {}_{,i} - \min_i\ {}_{,i} \le R$ for all t. Then for any $\in \mathbf{R}^n$ with $_i \ge 0$ and $\sum_i$ $_i$ 1, the explicit EG algorithm with learning rate η $4/R^2$ satis es

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \le \sum_{=1}^{T} \quad - \quad ^2 \quad \frac{1}{4} R^2 \Delta_{\rm re} \qquad _0$$

where $_0$ $\boldsymbol{g}\ \mathbf{0}$ is the uniform weight vector.

Proof: We analyze the progress d $\Delta_{\rm re}$ $_{-1}$ $-$ $\Delta_{\rm re}$. By substituting the explicit EG update into (29) and then using (28), we get

$$\begin{aligned} d \quad & \eta \quad - \quad _{-1} \qquad\qquad - \quad _{-1} - \Delta_{\rm re} \quad _{-1} \\ & \ge \eta \quad - \quad _{-1} \qquad\qquad - \quad _{-1} \\ & \quad - \frac{1}{8}\eta^2 \quad - \quad _{-1} \qquad ^2 R^2 \end{aligned}$$

By rearranging terms, we can write this as

$$d \ \ge \frac{\eta}{2} s^2 - \frac{\eta}{2} r^2 \quad \frac{\eta}{2}\ s\ \ - r\ \ ^2 \left(1 - \frac{\eta R^2}{4}\right)$$

where s $-$ $_{-1}$ and r $-$. Since $\eta R^2/4$ 1, we can apply $\Delta_{\rm re}$ $_{T+1}$ ≥ 0 to get

$$\begin{aligned} \Delta_{\rm re} \quad _0 \ & \ge \Delta_{\rm re} \quad _0\ - \Delta_{\rm re} \quad _{T+1} \\ & \sum_{=1}^{T} d \\ & \ge \frac{4}{R^2}\left(\sum_{=1}^{T} s^2 - \sum_{=1}^{T} r^2\right) \end{aligned}$$

from which the claim follows. ■

The above theorem assumes the comparison vector is a probability vector. To deal with arbitrary vectors with $\|\ \|_1 \le U_1$ for some given bound $U_1 > 0$, we de ne the *scaled* explicit EG$^{\pm}$ algorithm as explicit EG with each input replaced by $'$ U $_{,1}$ U $_{,n}$ $-U$ $_{,1}$ $-U$ $_{,n}$ $\in \mathbf{R}^{2n}$.

Corollary 1: Assume $\|\ \|_\infty \le X_\infty$ for all t. Then for any $\in \mathbf{R}^n$ with $\|\ \|_1 \le U_1$, the scaled explicit EG$^{\pm}$ algorithm satis es

$$\sum_{=1}^{T} \quad - \quad _{-1} \quad ^2 \le \sum_{=1}^{T} \quad - \quad ^2 \quad \ln\ 2n\ X_\infty^2 U_1^2$$

Proof: There is some $' \in \mathbf{R}^{2n}$ with $'_i \ge 0$ for all i and $\sum_i\ '_i$ 1 such that $'$ $'$ for all t. Thus we can apply Theorem 8 with this $'$. We have $\max_i\ '_{,i} - \min_i\ '_{,i}$ $2U_1\|\ \|_\infty$. Since $_0$ is the uniform $2n$-dimensional probability vector, we have $\Delta_{\rm re}$ $'$ $_0$ $\le \ln\ 2n$. ■

Bounds for the implicit EG algorithm can be proven analogously.

REFERENCES

[1] J. Kivinen, M. K. Warmuth, and B. Hassibi, The p-norm generalization of the LMS algorithm for adaptive filtering, presented at the 13th IFAC Symp. System Identification, P. M. J. V. den Hof, B. Wahlberg, and S. Weilands, Eds., Rotterdam, The Netherlands, Aug. 27–29, 2003.

[2] B. Widrow and M. E. Hoff, Adaptive switching circuits, in *1960 IRE WESCON Convention Rec.*, 1960, pp. 96–104.

[3] B. Hassibi, A. H. Sayed, and T. Kailath, H^∞ optimality of the LMS algorithm, *IEEE Trans. Signal Process.*, vol. 44, no. 2, pp. 267–280, Feb. 1996.

[4] N. Cesa-Bianchi, P. Long, and M. Warmuth, Worst-case quadratic loss bounds for on-line prediction of linear functions by gradient descent, *IEEE Trans. Neural Netw.*, vol. 7, no. 3, pp. 604–619, May 1996.

[5] J. Kivinen and M. K. Warmuth, Additive versus exponentiated gradient updates for linear prediction, *Inf. Comput.*, vol. 132, no. 1, pp. 1–64, Jan. 1997.

[6] ——, Relative loss bounds for multidimensional regression problems, *Mach. Learn.*, vol. 45, no. 3, pp. 301–329, Dec. 2001.

[7] A. J. Grove, N. Littlestone, and D. Schuurmans, General convergence results for linear discriminant updates, *Mach. Learn.*, vol. 43, no. 3, pp. 173–210, Jun. 2001.

[8] C. Gentile and N. Littlestone, The robustness of the p-norm algorithms, in *12th Annu. Conf. Comput. Learn. Theory*. New York: ACM Press, Jul. 1999, pp. 1–11.

[9] M. Herbster and M. K. Warmuth, Tracking the best linear predictor, *J. Mach. Learn. Res.*, vol. 1, pp. 281–309, Sep. 2001.

[10] D. P. Helmbold, J. Kivinen, and M. K. Warmuth, Relative loss bounds for single neurons, *IEEE Trans. Neural Netw.*, vol. 10, no. 6, pp. 1291–1304, Nov. 1999.

[11] N. Cesa-Bianchi, A. Conconi, and C. Gentile, On the generalization ability of on-line learning algorithms, *IEEE Trans. Inf. Theory*, vol. 50, no. 9, pp. 2050–2057, Sep. 2004.

[12] K. S. Azoury and M. K. Warmuth, Relative loss bounds for on-line density estimation with the exponential family of distributions, *Mach. Learn.*, vol. 43, no. 3, pp. 211–246, Jun. 2001.

[13] L. Bregman, The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming, *USSR Comp. Math. Phys.*, vol. 7, pp. 200–217, 1967.

[14] R. E. Schapire and M. K. Warmuth, On the worst-case analysis of temporal-difference learning algorithms, *Mach. Learn.*, vol. 22, no. 1/2/3, pp. 95–121, Jan. 1996.

[15] N. Littlestone, Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm, *Mach. Learn.*, vol. 2, no. 4, pp. 285–318, Apr. 1988.

[16] J. Kivinen, M. K. Warmuth, and P. Auer, The Perceptron algorithm vs. Winnow: Linear vs. logarithmic mistake bounds when few input variables are relevant, *Artif. Intell.*, vol. 97, pp. 325–343, Dec. 1997.

[17] A. Y. Ng, Feature selection, L_1 vs. L_2 regularization, and rotational invariance, in *Proc. 21st Int. Conf. Machine Learning*, R. Greiner and D. Schuurmans, Eds., Jul. 2004, pp. 615–622.

[18] P. Auer, N. Cesa-Bianchi, and C. Gentile, Adaptive and self-confident on-line learning algorithms, *J. Comp. Syst. Sci.*, vol. 64, no. 1, pp. 48–75, Feb. 2002.

[19] P. Auer, M. Herbster, and M. K. Warmuth, Exponentially many local minima for single neurons, in *Advances in Neural Information Processing Systems 8*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds. Cambridge, MA: MIT Press, June 1996, pp. 316–317.

[20] T. Y. Al-Naffouri, A. H. Sayed, and T. Kailath, On the selection of optimal nonlinearities for stochastic gradient adaptive algorithms, in *Proc. 2000 IEEE Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 1, Jun. 2000, pp. 464–467.

[21] M. K. Warmuth and S. V. N. Vishwanathan, Leaving the span, in *Proc. 18th Annu. Conf. Learning Theory*, P. Auer and R. Meir, Eds. Berlin, Germany: Springer, Jun. 2005, pp. 366–381.

[22] E. Takimoto and M. K. Warmuth, Path kernels and multiplicative updates, *J. Mach. Learn. Res.*, vol. 4, pp. 773–818, Oct. 2003.

[23] R. Khardon, D. Roth, and R. A. Servedio, Efficiency versus convergence of boolean kernels for on-line learning algorithms, *J. Artif. Intell. Res.*, vol. 24, pp. 341–356, Sep. 2005.

Jyrki Kivinen received the M.Sc. and Ph.D. degrees in computer science from the University of Helsinki, Finland, in 1989 and 1992, respectively.

He has held various teaching and research appointments at the University of Helsinki and visited the University of California at Santa Cruz and Australian National University as a Postdoctoral Fellow. Since 2003, he has been a Professor at the University of Helsinki. His scientific interests include machine learning and algorithms theory.

Manfred K. Warmuth received the undergraduate degree in computer science from Friedrich Alexander Universität, Germany, in 1978. He received the M.S. and Ph.D. degrees in computer science from the University of Colorado at Boulder in 1980 and 1981, respectively.

Since then he has spent most of his time teaching at the University of California in Santa Cruz. His current research interests are machine learning, online learning, statistical decision theory, and game theory.

Dr. Warmuth received a Fulbright fellowship.

Babak Hassibi was born in Tehran, Iran, in 1967. He received the B.S. degree from the University of Tehran in 1989 and the M.S. and Ph.D. degrees from Stanford University, Stanford, CA, in 1993 and 1996, respectively, all in electrical engineering.

From October 1996 to October 1998, he was a Research Associate with the Information Systems Laboratory, Stanford University. From November 1998 to December 2000, he was a Member of Technical Staff with the Mathematical Sciences Research Center, Bell Laboratories, Murray Hill, NJ. Since January 2001, he has been with the Department of Electrical Engineering, California Institute of Technology, Pasadena, where he is currently an Associate Professor. He has also held short-term appointments at Ricoh California Research Center, the Indian Institute of Science, and Linköping University, Sweden. His research interests include wireless communications, robust estimation and control, adaptive signal processing, and linear algebra. He is the coauthor of *Indefinite Quadratic Estimation and Control: A Unified Approach to H^2 and H^∞ Theories* (New York: SIAM, 1999) and *Linear Estimation* (Englewood Cliffs, NJ: Prentice Hall, 2000).

Prof. Hassibi received an Alborz Foundation Fellowship, the 1999 O. Hugo Schuck Best Paper Award from the American Automatic Control Council, the 2002 National Science Foundation Career Award, the 2002 Okawa Foundation Research Grant for Information and Telecommunications, the 2003 David and Lucille Packard Fellowship for Science and Engineering, and the 2003 Presidential Early Career Award for Scientists and Engineers. He was a Guest Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY special issue on Space-Time Transmission, Reception, Coding and Signal Processing and is currently an Associate Editor for Communications of the IEEE TRANSACTIONS ON INFORMATION THEORY.