
Minimax Time Series Prediction

Wouter M. Koolen
Centrum Wiskunde & Informatica
wmkoolen@cwi.nl

Alan Malek
UC Berkeley
malek@berkeley.edu

Peter L. Bartlett
UC Berkeley & QUT
bartlett@cs.berkeley.edu

Yasin Abbasi-Yadkori
QUT & UC Berkeley
yasin.abbasi@gmail.com

Abstract

We consider an adversarial formulation of the problem of predicting a time series with square loss. The aim is to predict an arbitrary sequence of vectors almost as well as the best smooth comparator sequence in retrospect. Our approach allows natural measures of smoothness such as the squared norm of increments. More generally, we consider a linear time series model and penalize the comparator sequence through the energy of the implied driving noise terms. We derive the minimax strategy for all problems of this type and show that it can be implemented efficiently. The optimal predictions are linear in the previous observations. We obtain an explicit expression for the regret in terms of the parameters defining the problem. For typical, simple definitions of smoothness, the computation of the optimal predictions involves only sparse matrices. In the case of norm-constrained data, where the smoothness is defined in terms of the squared norm of the comparator's increments, we show that the regret grows as $T/\sqrt{\lambda_T}$, where T is the length of the game and λ_T is an increasing limit on comparator smoothness.

1 Introduction

In *time series prediction*, *tracking*, and *filtering* problems, a learner sees a stream of (possibly noisy, vector-valued) data and needs to predict the future path. One may think of robot poses, meteorological measurements, stock prices, etc. Popular stochastic models for such tasks include the auto-regressive moving average (ARMA) model in time series analysis, Brownian motion models in finance, and state space models in signal processing.

In this paper, we study the time series prediction problem in the regret framework; instead of making assumptions on the data generating process, we ask: can we predict the data sequence online almost as well as the best offline prediction method in some comparison class (in this case, offline means that the comparator only needs to model the data sequence after seeing all of it)? Our main contribution is computing the exact minimax strategy for a range of time series prediction problems. As a concrete motivating example, let us pose the simplest nontrivial such minimax problem

$$\min_{\mathbf{a}_1} \max_{\mathbf{x}_1 \in \mathcal{B}} \cdots \min_{\mathbf{a}_T} \max_{\mathbf{x}_T \in \mathcal{B}} \underbrace{\sum_{t=1}^T \|\mathbf{a}_t - \mathbf{x}_t\|^2}_{\text{Loss of Learner}} - \min_{\hat{\mathbf{a}}_1, \dots, \hat{\mathbf{a}}_T} \left\{ \underbrace{\sum_{t=1}^T \|\hat{\mathbf{a}}_t - \mathbf{x}_t\|^2}_{\text{Loss of Comparator}} + \lambda_T \underbrace{\sum_{t=1}^{T+1} \|\hat{\mathbf{a}}_t - \hat{\mathbf{a}}_{t-1}\|^2}_{\text{Comparator Complexity}} \right\}. \quad (1)$$

This notion of regret is standard in online learning, going back at least to [1] in 2001, which views it as the natural generalization of L_2 regularization to deal with non-stationarity comparators. We offer two motivations for this regularization. First, one can interpret the complexity term as the magnitude

of the noise required to generate the comparator using a multivariate Gaussian random walk, and, generalizing slightly, as the energy of the innovations required to model the comparator using a single, fixed linear time series model (e.g. specific ARMA coefficients). Second, we can view the comparator term in Equation (1) as akin to the Lagrangian of a constrained optimization problem. Rather than competing with the comparator sequence $\hat{\mathbf{a}}_1, \dots, \hat{\mathbf{a}}_T$ that minimizes the cumulative loss subject to a hard constraint on the complexity term, the learner must compete with the comparator sequence that best trades off the cumulative loss and the smoothness. The Lagrange multiplier, λ_T , controls the trade-off. Notice that it is natural to allow λ_T to grow with T , since that penalizes the comparator’s change per round more than the loss per round.

For the particular problem (1) we obtain an efficient algorithm using amortized $O(d)$ time per round, where d is the dimension of the data; there is no nasty dependence on T as often happens with minimax algorithms. Our general minimax analysis extends to more advanced complexity terms. For example, we may regularize instead by higher-order smoothness (magnitude of increments of increments, etc.), or more generally, we may consider a fixed linear process and regularize the comparator by the energy of its implied driving noise terms (innovations). We also deal with arbitrary sequences of rank-one quadratic constraints on the data.

We show that the minimax algorithm is of a familiar nature; it is a linear filter, with a twist. Its coefficients are not time-invariant but instead arise from the intricate interplay between the regularization and the range of the data, combined with shrinkage. Fortunately, they may be computed in a pre-processing step by a simple recurrence. An unexpected detail of the analysis is the following. As we will show, the regret objective in (1) is a convex quadratic function of all data, and the sub-problem objectives that arise from the backward induction steps in the minimax analysis remain quadratic functions of the past. However, they may be either concave or convex. Changing direction of curvature is typically a source of technical difficulty: the minimax solution is different in either case. Quite remarkably, we show that one can determine a priori which rounds are convex and which are concave and apply the appropriate solution method in each.

We also consider what happens when the assumptions we need to make for the minimax analysis to go through are violated. We will show that the obtained minimax algorithm is in fact highly robust. Simply applying it unlicensed anyway results in adaptive regret *bounds* that scale naturally with the realized data magnitude (or, more generally, its energy).

1.1 Related Work

There is a rich history of tracking problems in the expert setting. In this setting, the learner has some finite number of actions to play and must select a distribution over actions to play each round in such a way as to guarantee that the loss is almost as small as the best single action in hindsight. The problem of tracking the best expert forces the learner to compare with sequences of experts (usually with some fixed number of switches). The fixed-share algorithm [2] was an early solution, but there has been more recent work [3, 4, 5, 6]. Tracking experts has been applied to other areas; see e.g. [7] for an application to sequential allocation. An extension to linear combinations of experts where the expert class is penalized by the p -norm of the sequence was considered in [1].

Minimax algorithms for squared Euclidean loss have been studied in several contexts such as Gaussian density estimation [8] and linear regression [9]. In [10], the authors showed that the minimax algorithm for quadratic loss is Follow the Leader (i.e. predicting the previous data mean) when the player is constrained to play in a ball around the previous data mean. Additionally, Moroshko and Krammer [11, 12] propose a weak notion of non-stationarity that allows them to apply the last-step minimax approach to a regression-like framework.

The tracking problem in the regret setting has been considered previously, e.g. [1], where the authors studied the best linear predictor with a comparison class of all sequences with bounded smoothness $\sum_t \|\mathbf{a}_t - \mathbf{a}_{t-1}\|^2$ and proposed a general method for converting regret bounds in the static setting to ones in the shifting setting (where the best expert is allowed to change).

Outline We start by presenting the formal setup in Section 2 and derive the optimal offline predictions. In Section 3 we zoom in to single-shot quadratic games, and solve these both in the convex and concave case. With this in hand, we derive the minimax solution to the time series prediction problem by backward induction in Section 4. In Section 5 we focus on the motivating problem

(1) for which we give a faster implementation and tightly sandwich the minimax regret. Section 6 concludes with discussion, conjectures and open problems.

2 Protocol and Offline Problem

The game protocol is described in Figure 1 and is the usual online prediction game with squared Euclidean loss. The goal of the learner is to incur small regret, that is, to predict the data almost as well as the best complexity-penalized sequence $\hat{\mathbf{a}}_1 \cdots \hat{\mathbf{a}}_T$ chosen in hindsight. Our motivating problem (1) gauged complexity by the sum of squared norms of the increments, thus encouraging smoothness. Here we generalize to complexity terms defined by a complexity matrix $\mathbf{K} \succeq \mathbf{0}$, and charge the comparator $\hat{\mathbf{a}}_1 \cdots \hat{\mathbf{a}}_T$ by $\sum_{s,t} K_{s,t} \hat{\mathbf{a}}_s^\top \hat{\mathbf{a}}_t$. We recover the smoothness penalty of (1) by taking \mathbf{K} to be the $T \times T$ tridiagonal matrix

<p>For $t = 1, 2, \dots, T$:</p> <ul style="list-style-type: none"> • Learner predicts $\mathbf{a}_t \in \mathbb{R}^d$ • Environment reveals $\mathbf{x}_t \in \mathbb{R}^d$ • Learner suffers loss $\ \mathbf{a}_t - \mathbf{x}_t\ ^2$. 	$\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & & \ddots & & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}, \quad (2)$
--	---

Figure 1: Protocol

but we may also regularize by e.g. the sum of squared norms ($\mathbf{K} = \mathbf{I}$), the sum of norms of higher

order increments, or more generally, we may consider a fixed linear process and take $\mathbf{K}^{1/2}$ to be the matrix that recovers the driving noise terms from the signal, and then our penalty is exactly the energy of the implied noise for that linear process. We now turn to computing the identity and quality of the best competitor sequence in hindsight.

Theorem 1. *For any complexity matrix $\mathbf{K} \succeq \mathbf{0}$, regularization scalar $\lambda_T \geq 0$, and $d \times T$ data matrix $\mathbf{X}_T = [\mathbf{x}_1 \cdots \mathbf{x}_T]$ the problem*

$$L^* := \min_{\hat{\mathbf{a}}_1, \dots, \hat{\mathbf{a}}_T} \sum_{t=1}^T \|\hat{\mathbf{a}}_t - \mathbf{x}_t\|^2 + \lambda_T \sum_{s,t} K_{s,t} \hat{\mathbf{a}}_s^\top \hat{\mathbf{a}}_t$$

has linear minimizer and quadratic value given by

$$[\hat{\mathbf{a}}_1 \cdots \hat{\mathbf{a}}_T] = \mathbf{X}_T (\mathbf{I} + \lambda_T \mathbf{K})^{-1} \quad \text{and} \quad L^* = \text{tr} (\mathbf{X}_T (\mathbf{I} - (\mathbf{I} + \lambda_T \mathbf{K})^{-1}) \mathbf{X}_T^\top).$$

Proof. Writing $\hat{\mathbf{A}} = [\hat{\mathbf{a}}_1 \cdots \hat{\mathbf{a}}_T]$ we can compactly express the offline problem as

$$L^* = \min_{\hat{\mathbf{A}}} \text{tr} \left((\hat{\mathbf{A}} - \mathbf{X}_T)^\top (\hat{\mathbf{A}} - \mathbf{X}_T) + \lambda_T \mathbf{K} \hat{\mathbf{A}}^\top \hat{\mathbf{A}} \right).$$

The $\hat{\mathbf{A}}$ derivative of the objective is $2(\hat{\mathbf{A}} - \mathbf{X}_T) + 2\lambda_T \hat{\mathbf{A}} \mathbf{K}$. Setting this to zero yields the minimizer $\hat{\mathbf{A}} = \mathbf{X}_T (\mathbf{I} + \lambda_T \mathbf{K})^{-1}$. Back-substitution and simplification result in value $\text{tr} (\mathbf{X}_T (\mathbf{I} - (\mathbf{I} + \lambda_T \mathbf{K})^{-1}) \mathbf{X}_T^\top)$. \square

Note that for the choice of \mathbf{K} in (2) computing the optimal $\hat{\mathbf{A}}$ can be performed in $O(dT)$ time by solving the linear system $\mathbf{A}(\mathbf{I} + \lambda_T \mathbf{K}_T) = \mathbf{X}_T$ directly. This system decomposes into d (one per dimension) independent tridiagonal systems, each in T (one per time step) variables, which can each be solved in linear time using Gaussian elimination.

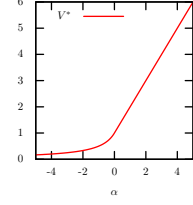
This theorem shows that the objective of our minimax problem is a quadratic function of the data. In order to solve a T round minimax problem with quadratic regret objective, we first solve simple single round quadratic games.

3 Minimax Single-shot Squared Loss Games

One crucial tool in the minimax analysis of our tracking problem will be solving particular single-shot min-max games. In such games, the player and adversary play prediction \mathbf{a} and data \mathbf{x} resulting in payoff given by the following square loss plus a quadratic in \mathbf{x} :

$$V(\mathbf{a}, \mathbf{x}) := \|\mathbf{a} - \mathbf{x}\|^2 + (\alpha - 1) \|\mathbf{x}\|^2 + 2\mathbf{b}^\top \mathbf{x}. \quad (3)$$

The quadratic and linear terms in \mathbf{x} have coefficients $\alpha \in \mathbb{R}$ and $\mathbf{b} \in \mathbb{R}^d$. Note that $V(\mathbf{a}, \mathbf{x})$ is convex in \mathbf{a} and either convex or concave in \mathbf{x} as decided by the sign of α . The following result, proved in Appendix B.1 and illustrated for $\|\mathbf{b}\| = 1$ by the figure to the right, gives the minimax analysis for both cases.



Theorem 2. Let $V(\mathbf{a}, \mathbf{x})$ be as in (3). If $\|\mathbf{b}\| \leq 1$, then the minimax problem

$$V^* := \min_{\mathbf{a} \in \mathbb{R}^d} \max_{\mathbf{x} \in \mathbb{R}^d: \|\mathbf{x}\| \leq 1} V(\mathbf{a}, \mathbf{x})$$

has value $V^* = \begin{cases} \frac{\|\mathbf{b}\|^2}{1-\alpha} & \text{if } \alpha \leq 0, \\ \|\mathbf{b}\|^2 + \alpha & \text{if } \alpha \geq 0, \end{cases}$ and minimizer $\mathbf{a} = \begin{cases} \frac{\mathbf{b}}{1-\alpha} & \text{if } \alpha \leq 0, \\ \mathbf{b} & \text{if } \alpha \geq 0. \end{cases}$ (4)

We also want to look at the performance of this strategy when we do not impose the norm bound $\|\mathbf{x}\| \leq 1$ nor make the assumption $\|\mathbf{b}\| \leq 1$. By evaluating (3) we obtain an adaptive expression that scales with the actual norm $\|\mathbf{x}\|^2$ of the data.

Theorem 3. Let \mathbf{a} be the strategy from (4). Then, for any data $\mathbf{x} \in \mathbb{R}^d$ and any $\mathbf{b} \in \mathbb{R}^d$,

$$V(\mathbf{a}, \mathbf{x}) = \frac{\|\mathbf{b}\|^2}{1-\alpha} + \alpha \left\| \frac{\mathbf{b}}{1-\alpha} - \mathbf{x} \right\|^2 \leq \frac{\|\mathbf{b}\|^2}{1-\alpha} \quad \text{if } \alpha \leq 0, \text{ and}$$

$$V(\mathbf{a}, \mathbf{x}) = \|\mathbf{b}\|^2 + \alpha \|\mathbf{x}\|^2 \quad \text{if } \alpha \geq 0.$$

These two theorems point out that the strategy in (4) is amazingly versatile. The former theorem establishes minimax optimality under data constraint $\|\mathbf{x}\| \leq 1$ assuming that $\|\mathbf{b}\| \leq 1$. Yet the latter theorem tells us that, even without constraints and assumptions, this strategy is still an extremely useful heuristic. For its actual regret is bounded by the minimax regret we would have incurred if we would have known the scale of the data $\|\mathbf{x}\|$ (and $\|\mathbf{b}\|$) in advance. The norm bound we imposed in the derivation induces the complexity measure for the data to which the strategy adapts. This robustness property will extend to the minimax strategy for time series prediction.

Finally, it remains to note that we present the theorems in the canonical case. Problems with a constraint of the form $\|\mathbf{x} - \mathbf{c}\| \leq \beta$ may be canonized by re-parameterizing by $\mathbf{x}' = \frac{\mathbf{x} - \mathbf{c}}{\beta}$ and $\mathbf{a}' = \frac{\mathbf{a} - \mathbf{c}}{\beta}$ and scaling the objective by β^{-2} . We find

Corollary 4. Fix $\beta \geq 0$ and $\mathbf{c} \in \mathbb{R}^d$. Let $V^*(\alpha, \mathbf{b})$ denote the minimax value from (4) with parameters α, \mathbf{b} . If $\|(\alpha - 1)\mathbf{c} + \mathbf{b}\| \leq \beta$ then

$$\min_{\mathbf{a}} \max_{\mathbf{x}: \|\mathbf{x} - \mathbf{c}\| \leq \beta} V(\mathbf{a}, \mathbf{x}) = \beta^2 V^* \left(\alpha, \frac{(\alpha - 1)\mathbf{c} + \mathbf{b}}{\beta} \right) + 2\mathbf{b}^\top \mathbf{c} + (\alpha - 1) \|\mathbf{c}\|^2.$$

With this machinery in place, we continue the minimax analysis of time series prediction problems.

4 Minimax Time Series Prediction

In this section, we give the minimax solution to the online prediction problem. Recall that the evaluation criterion, the regret, is defined by

$$\mathcal{R} := \sum_{t=1}^T \|\mathbf{a}_t - \mathbf{x}_t\|^2 - \min_{\hat{\mathbf{a}}_1, \dots, \hat{\mathbf{a}}_T} \sum_{t=1}^T \|\hat{\mathbf{a}}_t - \mathbf{x}_t\|^2 + \lambda_T \text{tr} \left(\mathbf{K} \hat{\mathbf{A}}^\top \hat{\mathbf{A}} \right) \quad (5)$$

where $\mathbf{K} \succeq \mathbf{0}$ is a fixed $T \times T$ matrix measuring the complexity of the comparator sequence. Since all the derivations ahead will be for a fixed T , we drop the T subscript on the λ . We study the minimax problem

$$\mathcal{R}^* := \min_{\mathbf{a}_1} \max_{\mathbf{x}_1} \cdots \min_{\mathbf{a}_T} \max_{\mathbf{x}_T} \mathcal{R} \quad (6)$$

under the constraint on the data that $\|\mathbf{X}_t \mathbf{v}_t\| \leq 1$ in each round t for some fixed sequence $\mathbf{v}_1, \dots, \mathbf{v}_T$ such that $\mathbf{v}_t \in \mathbb{R}^t$. This constraint generalizes the norm bound constraint from the motivating problem (1), which is recovered by taking $\mathbf{v}_t = \mathbf{e}_t$. This natural generalization allows us to also consider bounded norms of increments, bounded higher order discrete derivative norms etc.

We compute the minimax regret and get an expression for the minimax algorithm. We show that, at any point in the game, the value is a quadratic function of the past samples and the minimax algorithm is linear: it always predicts with a weighted sum of all past samples.

Most intriguingly, the value function can either be a convex or concave quadratic in the last data point, depending on the regularization. We saw in the previous section that these two cases require a different minimax solution. It is therefore an extremely fortunate fact that the particular case we find ourselves in at each round is *not* a function of the past data, but just a property of the problem parameters \mathbf{K} and \mathbf{v}_t .

We are going to solve the sequential minimax problem (6) one round at a time. To do so, it is convenient to define the value-to-go of the game from any state $\mathbf{X}_t = [\mathbf{x}_1 \cdots \mathbf{x}_t]$ recursively by

$$V(\mathbf{X}_T) := -L^* \quad \text{and} \quad V(\mathbf{X}_{t-1}) := \min_{\mathbf{a}_t} \max_{\mathbf{x}_t: \|\mathbf{X}_t \mathbf{v}_t\| \leq 1} \|\mathbf{a}_t - \mathbf{x}_t\|^2 + V(\mathbf{X}_t).$$

We are interested in the minimax algorithm and minimax regret $\mathcal{R}^* = V(\mathbf{X}_0)$. We will show that the minimax value and strategy are a quadratic and linear function of the observations. To express the value and strategy and state the necessary condition on the problem, we will need a series of scalars d_t and matrices $\mathbf{R}_t \in \mathbb{R}^{t \times t}$ for $t = 1, \dots, T$, which, as we will explain below, arises naturally from the minimax analysis. The matrices, which depend on the regularization parameter λ , comparator complexity matrix \mathbf{K} and data constraints \mathbf{v}_t , are defined recursively back-to-front. The base case is $\mathbf{R}_T := (\mathbf{I} + \lambda_T \mathbf{K})^{-1}$. Using the convenient abbreviations $\mathbf{v}_t = w_t \begin{pmatrix} \mathbf{u}_t \\ 1 \end{pmatrix}$ and $\mathbf{R}_t = \begin{pmatrix} \mathbf{A}_t & \mathbf{b}_t \\ \mathbf{b}_t^\top & c_t \end{pmatrix}$ we then recursively define \mathbf{R}_{t-1} and set d_t by

$$\mathbf{R}_{t-1} := \mathbf{A}_t + (\mathbf{b}_t - c_t \mathbf{u}_t)(\mathbf{b}_t - c_t \mathbf{u}_t)^\top - c_t \mathbf{u}_t \mathbf{u}_t^\top, \quad d_t := \frac{c_t}{w_t^2} \quad \text{if } c_t \geq 0, \quad (7a)$$

$$\mathbf{R}_{t-1} := \mathbf{A}_t + \frac{\mathbf{b}_t \mathbf{b}_t^\top}{1 - c_t}, \quad d_t := 0 \quad \text{if } c_t \leq 0. \quad (7b)$$

Using this recursion for d_t and \mathbf{R}_t , we can perform the exact minimax analysis under a certain condition on the interplay between the data constraint and the regularization. We then show below that the obtained algorithm has a condition-free data-dependent regret bound.

Theorem 5. *Assume that \mathbf{K} and \mathbf{v}_t are such that any data sequence \mathbf{X}_T satisfying the constraint $\|\mathbf{X}_t \mathbf{v}_t\| \leq 1$ for all rounds $t \leq T$ also satisfies $\|\mathbf{X}_{t-1}((c_t - 1)\mathbf{u}_t - \mathbf{b}_t)\| \leq 1/w_t$ for all rounds $t \leq T$. Then the minimax value of and strategy for problem (6) are given by*

$$V(\mathbf{X}_t) = \text{tr}(\mathbf{X}_t(\mathbf{R}_t - \mathbf{I})\mathbf{X}_t^\top) + \sum_{s=t+1}^T d_s \quad \text{and} \quad \mathbf{a}_t = \mathbf{X}_{t-1} \begin{cases} \frac{\mathbf{b}_t}{1 - c_t} & \text{if } c_t \leq 0, \\ \mathbf{b}_t - c_t \mathbf{u}_t & \text{if } c_t \geq 0, \end{cases}$$

In particular, this shows that the minimax regret (6) is given by $\mathcal{R}^* = \sum_{t=1}^T d_t$.

Proof. By induction. The base case $V(\mathbf{X}_T)$ is Theorem 1. For any $t < T$ we apply the definition of $V(\mathbf{X}_{t-1})$ and the induction hypothesis to get

$$\begin{aligned} V(\mathbf{X}_{t-1}) &= \min_{\mathbf{a}_t} \max_{\mathbf{x}_t: \|\mathbf{X}_t \mathbf{v}_t\| \leq 1} \|\mathbf{a}_t - \mathbf{x}_t\|^2 + \text{tr}(\mathbf{X}_t(\mathbf{R}_t - \mathbf{I})\mathbf{X}_t^\top) + \sum_{s=t+1}^T d_s \\ &= \text{tr}(\mathbf{X}_{t-1}(\mathbf{A}_t - \mathbf{I})\mathbf{X}_{t-1}^\top) + \sum_{s=t+1}^T d_s + C \end{aligned}$$

where we abbreviated

$$C := \min_{\mathbf{a}_t} \max_{\mathbf{x}_t: \|\mathbf{X}_t \mathbf{v}_t\| \leq 1} \|\mathbf{a}_t - \mathbf{x}_t\|^2 + (c_t - 1)\mathbf{x}_t^\top \mathbf{x}_t + 2\mathbf{x}_t^\top \mathbf{X}_{t-1} \mathbf{b}_t.$$

Without loss of generality, assume $w_t > 0$. Now, as $\|\mathbf{X}_t \mathbf{v}_t\| \leq 1$ iff $\|\mathbf{X}_{t-1} \mathbf{u}_t + \mathbf{x}_t\| \leq 1/w_t$, application of Corollary 4 with $\alpha = c_t$, $\mathbf{b} = \mathbf{X}_{t-1} \mathbf{b}_t$, $\beta = 1/w_t$ and $\mathbf{c} = -\mathbf{X}_{t-1} \mathbf{u}_t$ followed by Theorem 2 results in optimal strategy

$$\mathbf{a}_t = \begin{cases} \frac{\mathbf{X}_{t-1} \mathbf{b}_t}{1 - c_t} & \text{if } c_t \leq 0, \\ -c_t \mathbf{X}_{t-1} \mathbf{u}_t + \mathbf{X}_{t-1} \mathbf{b}_t & \text{if } c_t \geq 0. \end{cases}$$

and value

$$C = (c_t - 1) \|\mathbf{X}_{t-1} \mathbf{u}_t\|^2 - 2\mathbf{b}_t^\top \mathbf{X}_{t-1}^\top \mathbf{X}_{t-1} \mathbf{u}_t + \begin{cases} \|\mathbf{X}_{t-1} ((c_t - 1)\mathbf{u}_t - \mathbf{b}_t)\|^2 / (1 - c_t) & \text{if } c_t \leq 0, \\ \|\mathbf{X}_{t-1} ((c_t - 1)\mathbf{u}_t - \mathbf{b}_t)\|^2 + c_t / w_t^2 & \text{if } c_t \geq 0, \end{cases}$$


Expanding all squares and rearranging (cycling under the trace) completes the proof. \square

On the one hand, from a technical perspective the condition of Theorem 5 is rather natural. It guarantees that the prediction of the algorithm will fall within the constraint imposed on the data. (If it would not, we could benefit by clipping the prediction. This would be guaranteed to reduce the loss, and it would wreck the backwards induction.) Similar clipping conditions arise in the minimax analyses for linear regression [9] and square loss prediction with Mahalanobis losses [13].

In practice we typically do not have a hard bound on the data. Sill, by running the above minimax algorithm obtained for data complexity *bounds* $\|\mathbf{X}_t \mathbf{v}_t\| \leq 1$, we get an adaptive regret bound that scales with the *actual data complexity* $\|\mathbf{X}_t \mathbf{v}_t\|^2$, as can be derived by replacing the application of Theorem 2 in the proof of Theorem 5 by an invocation of Theorem 3.

Theorem 6. *Let $\mathbf{K} \succeq \mathbf{0}$ and \mathbf{v}_t be arbitrary. The minimax algorithm obtained in Theorem 5 keeps the regret (5) bounded by $\mathcal{R} \leq \sum_{t=1}^T d_t \|\mathbf{X}_t \mathbf{v}_t\|^2$ for any data sequence \mathbf{X}_T .*

4.1 Computation, sparsity

In the important special case (typical application) where the regularization \mathbf{K} and data constraint \mathbf{v}_t are encoding some order of smoothness, we find that \mathbf{K} is banded diagonal and \mathbf{v}_t only has a few tail non-zero entries. It hence is the case that $\mathbf{R}_{T-1}^{-1} = \mathbf{I} + \lambda \mathbf{K}$ is sparse. We now argue that the recursive updates (7) preserve sparsity of the inverse \mathbf{R}_t^{-1} . In Appendix C we derive an update for \mathbf{R}_{t-1}^{-1} in terms of \mathbf{R}_t^{-1} . For computation it hence makes sense to tabulate \mathbf{R}_t^{-1} directly. We now argue (proof in Appendix B.2) that all \mathbf{R}_t^{-1} are sparse. 

Theorem 7. *Say the \mathbf{v}_t are V -sparse (all but their tail V entries are zero). And say that \mathbf{K} is D -banded (all but the main and $D - 1$ adjacent diagonals to either side are zero). Then each \mathbf{R}_t^{-1} is the sum of the D -banded matrix $\mathbf{I} + \lambda \mathbf{K}_{1:t,1:t}$ and a $(D + V - 2)$ -blocked matrix (i.e. all but the lower-right block of size $D + V - 2$ is zero).*

So what does this sparsity argument buy us? We only need to maintain the original D -banded matrix \mathbf{K} and the $(D + V - 2)^2$ entries of the block perturbation. These entries can be updated backwards from $t = T, \dots, 1$ in $\mathcal{O}((D + V - 2)^3)$ time per round using block matrix inverses. This means that the run-time of the entire pre-processing step is *linear* in T . For updates and prediction we need c_t and \mathbf{b}_t , which we can compute using Gaussian elimination from \mathbf{R}_t^{-1} in $\mathcal{O}(t(D + V))$ time. In the next section we will see a special case in which we can update and predict in constant time.

5 Norm-bounded Data with Increment Squared Regularization

We return to our motivating problem (1) with complexity matrix $\mathbf{K} = \mathbf{K}_T$ given by (2) and norm constrained data, i.e. $\mathbf{v}_t = \mathbf{e}_t$. We show that the \mathbf{R}_t matrices are very simple: their inverse is $\mathbf{I} + \lambda \mathbf{K}_t$ with its lower-right entry perturbed. Using this, we show that the prediction is a linear combination of the past observations with weights decaying exponentially backward in time. We derive a constant-time update equation for the minimax prediction and tightly sandwich the regret.

Here, we will calculate a few quantities that will be useful throughout this section. The inverse $(\mathbf{I} + \lambda \mathbf{K}_T)^{-1}$ can be computed in closed form as a direct application of the results in [14]:

Lemma 8. *Recall that $\sinh(x) = \frac{e^x - e^{-x}}{2}$ and $\cosh(x) = \frac{e^x + e^{-x}}{2}$. For any $\lambda \geq 0$:*

$$(\mathbf{I} + \lambda \mathbf{K}_T)_{i,j}^{-1} = \frac{\cosh((T + 1 - |i - j|)\nu) - \cosh((T + 1 - i - j)\nu)}{2\lambda \sinh(\nu) \sinh((T + 1)\nu)},$$

where $\nu = \cosh^{-1}(1 + \frac{1}{2\lambda})$.

We need some control on this inverse. We will use the abbreviations

$$\mathbf{z}_t := (\mathbf{I} + \lambda \mathbf{K}_t)^{-1} \mathbf{e}_t, \quad (8)$$

$$h_t := \mathbf{e}_t^\top (\mathbf{I} + \lambda \mathbf{K}_t)^{-1} \mathbf{e}_t = \mathbf{e}_t^\top \mathbf{z}_t, \text{ and} \quad (9)$$

$$h := \frac{2}{1 + 2\lambda + \sqrt{1 + 4\lambda}}. \quad (10)$$

We now show that these quantities are easily computable (see Appendix B for proofs).

Lemma 9. *Let ν be as in Lemma 8. Then, we can write*

$$h_t = \frac{1 - (\lambda h)^{2t}}{1 - (\lambda h)^{2t+2}} h,$$

and $\lim_{t \rightarrow \infty} h_t = h$ from below, exponentially fast.

A direct application of block matrix inversion (Lemma 12) results in

Lemma 10. *We have*

$$h_t = \frac{1}{1 + 2\lambda - \lambda^2 h_{t-1}} \quad \text{and} \quad \mathbf{z}_t = h_t \begin{pmatrix} \lambda \mathbf{z}_{t-1} \\ 1 \end{pmatrix}.$$

Intriguingly, following the optimal algorithm for all T rounds can be done in $O(Td)$ computation and $O(d)$ memory. These resource requirements are surprising as playing weighted averages typically requires $O(T^2 d)$. We found that the weighted averages are similar between rounds and can be updated cheaply.

We are now ready to state the main result of this section, proved in Appendix B.3.

Theorem 11. *Let \mathbf{z}_t and h_t be as in (8) and \mathbf{K}_t as in (2). For the minimax problem (1) we have*

$$\mathbf{R}_t^{-1} = \mathbf{I} + \lambda \mathbf{K}_t + \gamma_t \mathbf{e}_t \mathbf{e}_t^\top$$

and the minimax prediction in round t is given by

$$\mathbf{a}_t = \lambda c_t \mathbf{X}_{t-1} \mathbf{z}_{t-1}$$

where $\gamma_t = \frac{1}{c_t} - \frac{1}{h_t}$ and c_t satisfy the recurrence $c_T = h_T$ and $c_{t-1} = h_{t-1} + \lambda^2 h_{t-1}^2 c_t (1 + c_t)$.

5.1 Implementation

Theorem 11 states that the minimax prediction is $\mathbf{a}_t = \lambda c_t \mathbf{X}_{t-1} \mathbf{z}_{t-1}$. Using Lemma 10, we can derive an incremental update for \mathbf{a}_t by defining $\mathbf{a}_1 = \mathbf{0}$ and

$$\begin{aligned} \mathbf{a}_{t+1} &= \lambda c_{t+1} \mathbf{X}_t \mathbf{z}_t = \lambda c_{t+1} [\mathbf{X}_{t-1} \mathbf{x}_t] h_t \begin{pmatrix} \lambda \mathbf{z}_{t-1} \\ 1 \end{pmatrix} = \lambda c_{t+1} h_t (\mathbf{X}_{t-1} \lambda \mathbf{z}_{t-1} + \mathbf{x}_t) \\ &= \lambda c_{t+1} h_t \left(\frac{\mathbf{a}_t}{c_t} + \mathbf{x}_t \right). \end{aligned}$$

This means we can predict in constant time $O(d)$ per round.

5.2 Lower Bound

By Theorem 5, using that $w_t = 1$ so that $d_t = c_t$, the minimax regret equals $\sum_{t=1}^T c_t$. For convenience, we define $r_t := 1 - (\lambda_T h)^{2t}$ (and $r_{T+1} = 1$) so that $h_t = h r_t / r_{t+1}$. We can obtain a lower bound on c_t from the expression given in Theorem 11 by ignoring the (positive) c_t^2 term to obtain: $c_{t-1} \geq h_{t-1} + \lambda_T^2 h_{t-1}^2 c_t$. By unpacking this lower bound recursively, we arrive at

$$c_t \geq h \sum_{k=t}^T (\lambda_T h)^{2(k-t)} \frac{r_t^2}{r_k r_{k+1}}.$$

Since $r_t^2/(r_i r_{i+1})$ is a decreasing function in i for every t , we have $\frac{r_t^2}{r_i r_{i+1}} \geq \frac{r_t}{r_{t+1}}$ which leads to

$$\sum_{t=1}^T c_t \geq h \sum_{t=1}^T \sum_{k=t}^T (\lambda_T h)^{2(k-t)} \frac{r_t}{r_{t+1}} \geq h \int_0^{T-1} \int_{t+1}^T (\lambda_T h)^{2(k-t)} \frac{r_t}{r_{t+1}} dk dt = \Omega\left(-\frac{hT}{2 \log(\lambda_T h)}\right)$$

where we have exploited the fact that the integrand is monotonic and concave in k and monotonic and convex in t to lower bound the sums with an integral. See Claim 14 in the appendix for more details. Since $-\log(\lambda_T h) = O(1/\sqrt{\lambda_T})$ and $h = \Omega(1/\lambda_T)$, we have that $\sum_{t=1}^T c_t = \Omega(\frac{T}{\sqrt{\lambda_T}})$, matching the upper bound below.

5.3 Upper Bound

As $h \geq h_t$, the alternative recursion $c'_{T+1} = 0$ and $c'_{t-1} = h + \lambda^2 h^2 c'_t (1 + c'_t)$ satisfies $c'_t \geq c_t$. A simple induction¹ shows that c'_t is increasing with decreasing t , and it must hence have a limit. This limit is a fixed-point of $c \mapsto h + \lambda^2 h^2 c(1 + c)$. This results in a quadratic equation, which has two solutions. Our starting point $c'_{T+1} = 0$ lies below the half-way point $\frac{1 - \lambda^2 h^2}{2\lambda^2 h^2} > 0$, so the sought limit is the smaller solution:

$$c = \frac{-\lambda^2 h^2 + 1 - \sqrt{(\lambda^2 h^2 - 1)^2 - 4\lambda^2 h^3}}{2\lambda^2 h^2}.$$

This is monotonic in h . Plugging in the definition of h , we find

$$c = \frac{\sqrt{4\lambda + 1}(2\lambda + 1) + 4\lambda + 1 - \sqrt{2}\sqrt{2\lambda(\lambda(2\sqrt{4\lambda + 1} + 7) + 3\sqrt{4\lambda + 1} + 4) + \sqrt{4\lambda + 1} + 1}}{4\lambda^2}.$$

Series expansion around $\lambda \rightarrow \infty$ results in $c \leq (1 + \lambda)^{-1/2}$. So all in all, the bound is

$$\mathcal{R}^* = \mathcal{O}\left(\frac{T}{\sqrt{1 + \lambda_T}}\right),$$

where we have written the explicit T dependence of λ . As discussed in the introduction, allowing λ_T to grow with T is natural and necessary for sub-linear regret. If λ_T were constant, the regret term and complexity term would grow with T at the same rate, effectively forcing the learner to compete with sequences that could track the \mathbf{x}_t sequence arbitrarily well.

6 Discussion

We looked at obtaining the minimax solution to simple tracking/filtering/time series prediction problems with square loss, square norm regularization and square norm data constraints. We obtained a computational method to get the minimax result. Surprisingly, the problem turns out to be a mixture of per-step quadratic minimax problems that can be either concave or convex. These two problems have different solutions. Since the type of problem that is faced in each round is *not* a function of the past data, but only of the regularization, the coefficients of the value-to-go function can still be computed recursively. However, extending the analysis beyond quadratic loss and constraints is difficult; the self-dual property of the 2-norm is central to the calculations.

Several open problems arise. The stability of the coefficient recursion is so far elusive. For the case of norm bounded data, we found that the c_t are positive and essentially constant. However, for higher order smoothness constraints on the data (norm bounded increments, increments of increments, ...) the situation is more intricate. We find negative c_t and oscillating c_t , both diminishing and increasing. Understanding the behavior of the minimax regret and algorithm as a function of the regularization \mathbf{K} (so that we can tune λ appropriately) is an intriguing and elusive open problem.

¹For the base case, $c_{T+1} = 0 \leq c_T = h$. Then $c'_{t-1} = h + \lambda^2 h^2 c'_t (1 + c'_t) \geq h + \lambda^2 h^2 c'_{t+1} (1 + c'_{t+1}) = c'_t$.

References

- [1] Mark Herbster and Manfred K Warmuth. Tracking the best linear predictor. *The Journal of Machine Learning Research*, 1:281–309, 2001.
- [2] Mark Herbster and Manfred K. Warmuth. Tracking the best expert. *Machine Learning*, 32:151–178, 1998.
- [3] Claire Monteleoni. Online learning of non-stationary sequences. Master’s thesis, MIT, May 2003. Artificial Intelligence Report 2003-11.
- [4] Kamalika Chaudhuri, Yoav Freund, and Daniel Hsu. An online learning-based framework for tracking. *arXiv preprint arXiv:1203.3471*, 2012.
- [5] Olivier Bousquet and Manfred K Warmuth. Tracking a small set of experts by mixing past posteriors. *The Journal of Machine Learning Research*, 3:363–396, 2003.
- [6] Nicolò Cesa-bianchi, Pierre Gaillard, Gabor Lugosi, and Gilles Stoltz. Mirror descent meets fixed share (and feels no regret). In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 980–988. Curran Associates, Inc., 2012.
- [7] Avrim Blum and Carl Burch. On-line learning and the metrical task system problem. *Machine Learning*, 39(1):35–58, 2000.
- [8] Eiji Takimoto and Manfred K. Warmuth. The minimax strategy for Gaussian density estimation. In *13th COLT*, pages 100–106, 2000.
- [9] Peter L. Bartlett, Wouter M. Koolen, Alan Malek, Manfred K. Warmuth, and Eiji Takimoto. Minimax fixed-design linear regression. Accepted to COLT.
- [10] Jacob Abernethy, Peter L. Bartlett, Alexander Rakhlin, and Ambuj Tewari. Optimal strategies and minimax lower bounds for online convex games. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT 2008)*, pages 415–423, December 2008.
- [11] Edward Moroshko and Koby Crammer. Weighted last-step min-max algorithm with improved sub-logarithmic regret. In Nader H. Bshouty, Gilles Stoltz, Nicolas Vayatis, and Thomas Zeugmann, editors, *Algorithmic Learning Theory - 23rd International Conference, ALT 2012, Lyon, France, October 29-31, 2012. Proceedings*, volume 7568 of *Lecture Notes in Computer Science*, pages 245–259. Springer, 2012.
- [12] Edward Moroshko and Koby Crammer. A last-step regression algorithm for non-stationary online learning. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2013, Scottsdale, AZ, USA, April 29 - May 1, 2013*, volume 31 of *JMLR Proceedings*, pages 451–462. JMLR.org, 2013.
- [13] Wouter M. Koolen, Alan Malek, and Peter L. Bartlett. Efficient minimax strategies for square loss games. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems (NIPS) 27*, pages 3230–3238, December 2014.
- [14] G. Y. Hu and Robert F. O’Connell. Analytical inversion of symmetric tridiagonal matrices. *Journal of Physics A: Mathematical and General*, 29(7):1511, 1996.

A Technical lemmas

Here, we gather the version of block matrix inversion and Sherman Morrison formulas used previously.

Lemma 12 (Block Matrix Inverse). *We have*

$$\begin{pmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{b}^\top & c \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{b}(c - \mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b})^{-1}\mathbf{b}^\top\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{b}(c - \mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b})^{-1} \\ -(c - \mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b})^{-1}\mathbf{b}^\top\mathbf{A}^{-1} & (c - \mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b})^{-1} \end{pmatrix}.$$

Lemma 13 (Sherman-Morrison).

$$(\mathbf{A} + \mathbf{b}\mathbf{b}^\top)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{b}\mathbf{b}^\top\mathbf{A}^{-1}}{1 + \mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b}} \quad \text{and} \quad (\mathbf{A} - \mathbf{b}\mathbf{b}^\top)^{-1} = \mathbf{A}^{-1} + \frac{\mathbf{A}^{-1}\mathbf{b}\mathbf{b}^\top\mathbf{A}^{-1}}{1 - \mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b}}.$$

B Proofs

Proof of Lemma 9. Recall that

$$h := \frac{2}{1 + 2\lambda + \sqrt{1 + 4\lambda}}.$$

This allows us to write

$$\begin{aligned} h_t &= \frac{\cosh((t+1)\nu) - \cosh((1-t)\nu)}{2\lambda \sinh(\nu) \sinh((t+1)\nu)} \\ &= \frac{1 - \left(\frac{2\lambda}{1+2\lambda+\sqrt{4\lambda+1}}\right)^{2t}}{\sqrt{4\lambda+1} + \frac{2\lambda^2}{1+2\lambda+\sqrt{4\lambda+1}} \left(1 - \left(\frac{2\lambda}{1+2\lambda+\sqrt{4\lambda+1}}\right)^{2t}\right)} \\ &= \frac{1 - (\lambda h)^{2t}}{\sqrt{4\lambda+1} + \lambda^2 h (1 - (\lambda h)^{2t})}. \end{aligned}$$

Verifying that $1/h = \sqrt{4\lambda+1} + \lambda^2 h$ and plugging this in yields

$$h_t = \frac{1 - (\lambda h)^{2t}}{1 - (\lambda h)^{2t+2}} h,$$

as desired. □

Claim 14. *We claimed that*

$$\sum_{t=1}^T c_t \geq h \int_0^{T-1} \int_{t+1}^T (\lambda h)^{2(k-t)} \frac{1 - (\lambda h)^{2t}}{1 - (\lambda h)^{2(t+1)}} dk dt = O\left(\frac{-hT}{(\lambda h)^2 \log(\lambda h)}\right).$$

Evaluating the integral gives

$$\begin{aligned} &\int_1^{T-1} \int_{t+1}^T (\lambda h)^{2(k-t)} \frac{1 - (\lambda h)^{2t}}{1 - (\lambda h)^{2(t+1)}} dk dt \\ &= -\frac{T}{\log((\lambda h)^2)} + \frac{(\lambda h)^2 - \log((\lambda h)^2)}{\log^2((\lambda h)^2)} + \frac{1}{\log((\lambda h)^2)} \\ &+ \frac{(\lambda h)^{2T} + ((\lambda h)^2 - 1)(\lambda h)^{2T}((\lambda h)^{2T} - 1) \log((\lambda h)^{-2} - (\lambda h)^2)}{(\lambda h)^2 \log^2((\lambda h)^2)} \\ &- \frac{((\lambda h)^2 - 1)(\lambda h)^{2T} - 1 \log((\lambda h)^{-2T+2} - (\lambda h)^2)}{\log^2((\lambda h)^2)} \\ &= -\frac{T}{2 \log(\lambda h)} + o(1). \end{aligned}$$

B.1 Proof of Theorem 2

Proof. First, we consider $\alpha \leq 0$. The objective is concave in \mathbf{x} so the inner maximum is found at zero derivative, i.e. at $\mathbf{x} = \frac{\mathbf{a}-\mathbf{b}}{\alpha}$ (it remains to check this is feasible) where the value equals $V^* = \min_{\mathbf{a} \in \mathbb{R}^d} \|\mathbf{a}\|^2 - \frac{\|\mathbf{a}-\mathbf{b}\|^2}{\alpha}$. This optimization is convex in \mathbf{a} (again owing to the sign of α) and again minimized at vanishing derivative. That is, $\mathbf{a} = \frac{\mathbf{b}}{1-\alpha}$, (back substitution reveals $\mathbf{x} = \frac{\mathbf{b}}{1-\alpha}$, which is indeed feasible since $\|\mathbf{b}\| \leq 1$ by assumption and we are in the case $\alpha \leq 0$). Thus, the value equals $V^* = \frac{\|\mathbf{b}\|^2}{1-\alpha}$, as claimed.

Next, we consider $\alpha \geq 0$. The objective is convex in \mathbf{x} and must take its maximum on the boundary. Specifically, it is maximized at the point of the unit sphere maximizing the linear term, i.e. at $\mathbf{x} = \frac{\mathbf{b}-\mathbf{a}}{\|\mathbf{b}-\mathbf{a}\|}$. Plugging this in, we find value equal to $V^* = \min_{\mathbf{a} \in \mathbb{R}^d} \|\mathbf{a}\|^2 + \alpha + 2\|\mathbf{b}-\mathbf{a}\|$. This is a concave function of \mathbf{a} , although it is not differentiable due to the square-less norm. Yet it is minimized where zero is a sub-gradient (the unit ball is sub-gradient to the norm at $\mathbf{0}$). In our case this happens at $\mathbf{a} = \mathbf{b}$, where we used the assumption that $\|\mathbf{b}\| \leq 1$. We conclude that the value is $V^* = \|\mathbf{b}\|^2 + \alpha$ as desired. \square

B.2 Proof of Theorem 7

Proof. By induction. For the base case notice that \mathbf{R}_T is the D -banded matrix $\mathbf{I} + \lambda\mathbf{K}$. Now assume that \mathbf{R}_{t-1}^{-1} is a sum of a D -banded matrix and an N -blocked matrix. Naturally $D, V \geq 1$ and $N \geq 0$. Then \mathbf{b}'_t is $\max\{D-1, N-1\}$ -sparse, and hence $\mathbf{b}'_t \mathbf{b}'_t{}^\top$ is $\max\{D-1, N-1\}$ -blocked. Also \mathbf{u}_t is $(V-1)$ -sparse, and \mathbf{A}'_t is D -banded plus $(N-1)$ -blocked. So $\mathbf{A}'_t \mathbf{u}_t$ is $\max\{D+V-2, N-1\}$ -sparse. Now $\mathbf{A}'_t \mathbf{u}_t \mathbf{b}'_t{}^\top$ is $\max\{N-1, D+V-2\}$ -blocked. So \mathbf{R}_{t-1}^{-1} is D -banded plus $\max\{N-1, D+V-2\}$ -blocked. So if $N \geq D+V-2$ it remains D -banded plus N -blocked. (Note that strict inequality is wasteful: after a few pullbacks \mathbf{R}_s^{-1} will be D -banded plus $(D+V-2)$ -blocked). \square

B.3 Proof of Theorem 11

Proof. By induction. The base case $t = T$ follows by Theorem 1 and noticing that $\gamma_T = 0$. Then

$$\mathbf{R}_t = \begin{pmatrix} \mathbf{I} + \lambda\mathbf{K}_{t-1} & -\lambda\mathbf{e}_{t-1} \\ -\lambda\mathbf{e}_{t-1}^\top & 1 + 2\lambda + \gamma_t \end{pmatrix}^{-1} = \begin{pmatrix} (\mathbf{I} + \lambda\mathbf{K}_{t-1})^{-1} + \lambda^2 c_t \mathbf{z}_{t-1} \mathbf{z}_{t-1}^\top & \lambda c_t \mathbf{z}_{t-1} \\ \lambda c_t \mathbf{z}_{t-1}^\top & c_t \end{pmatrix}$$

where used that $c_t = \frac{1}{\frac{1}{h_t} + \gamma_t} = \frac{1}{1+2\lambda+\gamma_t-\lambda^2 h_{t-1}}$. Now using that $c_t \geq 0$, Theorem 5 gives optimal prediction $\mathbf{a}_t = \lambda c_t \mathbf{X}_{t-1} \mathbf{z}_{t-1}$ and the update equation (7) gives $\mathbf{R}_{t-1} = (\mathbf{I} + \lambda\mathbf{K}_{t-1})^{-1} + \lambda^2 c_t \mathbf{z}_{t-1} \mathbf{z}_{t-1}^\top + \lambda^2 c_t^2 \mathbf{z}_{t-1} \mathbf{z}_{t-1}^\top$. By Sherman-Morrison (Lemma 13), using that $(\mathbf{I} + \lambda\mathbf{K}_t) \mathbf{z}_t = \mathbf{e}_t$,

$$\begin{aligned} \mathbf{R}_{t-1}^{-1} &= \mathbf{I} + \lambda\mathbf{K}_{t-1} - \frac{(\lambda^2 c_t + \lambda^2 c_t^2) (\mathbf{I} + \lambda\mathbf{K}_{t-1}) \mathbf{z}_{t-1} \mathbf{z}_{t-1}^\top (\mathbf{I} + \lambda\mathbf{K}_{t-1})}{1 + (\lambda^2 c_t + \lambda^2 c_t^2) \mathbf{z}_{t-1}^\top (\mathbf{I} + \lambda\mathbf{K}_{t-1}) \mathbf{z}_{t-1}} \\ &= \mathbf{I} + \lambda\mathbf{K}_{t-1} - \frac{(\lambda^2 c_t + \lambda^2 c_t^2) \mathbf{e}_{t-1} \mathbf{e}_{t-1}^\top}{1 + (\lambda^2 c_t + \lambda^2 c_t^2) h_{t-1}} \end{aligned}$$

witnessing the expression for

$$\gamma_{t-1} = - \frac{\lambda^2 c_t (1 + c_t)}{1 + \lambda^2 c_t (1 + c_t) h_{t-1}}.$$

We finally check, using $h_t = \frac{1}{1+2\lambda-\lambda^2 h_{t-1}}$, that

$$c_{t-1} = \frac{1}{\frac{1}{h_{t-1}} - \frac{\lambda^2 c_t (1+c_t)}{1+\lambda^2 c_t (1+c_t) h_{t-1}}} = h_{t-1} + \lambda^2 h_{t-1}^2 c_t (1 + c_t). \quad \square$$

C Inverse Update Equation

The update equation (7) for \mathbf{R}_t is simple, but results in dense matrices \mathbf{R}_t , even though \mathbf{R}_t^{t-1} is sparse. In this section we derive a recursion for \mathbf{R}_t^{-1} purely in terms of inverses. Let us decompose

$\mathbf{R}_t^{-1} = \begin{pmatrix} \mathbf{A}'_t & \mathbf{b}'_t \\ \mathbf{b}'_t{}^\top & c'_t \end{pmatrix}$. Then

$$\mathbf{A}_t = \mathbf{A}'_t{}^{-1} + \mathbf{A}'_t{}^{-1} \mathbf{b}'_t (c'_t - \mathbf{b}'_t{}^\top \mathbf{A}'_t{}^{-1} \mathbf{b}'_t)^{-1} \mathbf{b}'_t{}^\top \mathbf{A}'_t{}^{-1} \quad (11a)$$

$$\mathbf{b}_t = -\mathbf{A}'_t{}^{-1} \mathbf{b}'_t (c'_t - \mathbf{b}'_t{}^\top \mathbf{A}'_t{}^{-1} \mathbf{b}'_t)^{-1} \quad (11b)$$

$$c_t = (c'_t - \mathbf{b}'_t{}^\top \mathbf{A}'_t{}^{-1} \mathbf{b}'_t)^{-1} \quad (11c)$$

If $c_t \geq 0$ it follows from (7) that

$$\mathbf{R}_{t-1} = \mathbf{A}'_t{}^{-1} + [\mathbf{b}_t \mathbf{u}_t] \begin{pmatrix} 1 + \frac{1}{c_t} & -c_t \\ -c_t & c_t(c_t - 1) \end{pmatrix} [\mathbf{b}_t \mathbf{u}_t]^\top$$

and hence

$$\begin{aligned} \mathbf{R}_{t-1}^{-1} &= \mathbf{A}'_t - \mathbf{A}'_t [\mathbf{b}_t \mathbf{u}_t] \left(\begin{pmatrix} 1 + \frac{1}{c_t} & -c_t \\ -c_t & c_t(c_t - 1) \end{pmatrix}^{-1} + [\mathbf{b}_t \mathbf{u}_t]^\top \mathbf{A}'_t [\mathbf{b}_t \mathbf{u}_t] \right)^{-1} [\mathbf{b}_t \mathbf{u}_t]^\top \mathbf{A}'_t \\ &= \mathbf{A}'_t - [-c_t \mathbf{b}'_t \mathbf{A}'_t \mathbf{u}_t] \left(\begin{pmatrix} c_t(1 - c_t) & -c_t \\ -c_t & -(1 + \frac{1}{c_t}) \end{pmatrix} + [\mathbf{b}_t \mathbf{u}_t]^\top \mathbf{A}'_t [\mathbf{b}_t \mathbf{u}_t] \right)^{-1} [-c_t \mathbf{b}'_t \mathbf{A}'_t \mathbf{u}_t]^\top \\ &= \mathbf{A}'_t - \frac{[-c_t \mathbf{b}'_t \mathbf{A}'_t \mathbf{u}_t] \begin{pmatrix} \mathbf{u}_t^\top \mathbf{A}'_t \mathbf{u}_t - (1 + \frac{1}{c_t}) & c_t (\mathbf{b}'_t{}^\top \mathbf{u}_t + 1) \\ c_t (\mathbf{b}'_t{}^\top \mathbf{u}_t + 1) & c_t^2 (c'_t - 1) \end{pmatrix} [-c_t \mathbf{b}'_t \mathbf{A}'_t \mathbf{u}_t]^\top}{\left(\mathbf{u}_t^\top \mathbf{A}'_t \mathbf{u}_t - (1 + \frac{1}{c_t}) \right) c_t^2 (c'_t - 1) - c_t^2 (\mathbf{b}'_t{}^\top \mathbf{u}_t + 1)^2} \\ &= \mathbf{A}'_t - \frac{[\mathbf{b}'_t \mathbf{A}'_t \mathbf{u}_t] \begin{pmatrix} \mathbf{u}_t^\top \mathbf{A}'_t \mathbf{u}_t - (1 + \frac{1}{c_t}) & -\mathbf{b}'_t{}^\top \mathbf{u}_t - 1 \\ -\mathbf{b}'_t{}^\top \mathbf{u}_t - 1 & c'_t - 1 \end{pmatrix} [\mathbf{b}'_t \mathbf{A}'_t \mathbf{u}_t]^\top}{\left(\mathbf{u}_t^\top \mathbf{A}'_t \mathbf{u}_t - (1 + \frac{1}{c_t}) \right) (c'_t - 1) - (\mathbf{b}'_t{}^\top \mathbf{u}_t + 1)^2} \end{aligned}$$

Which expands to

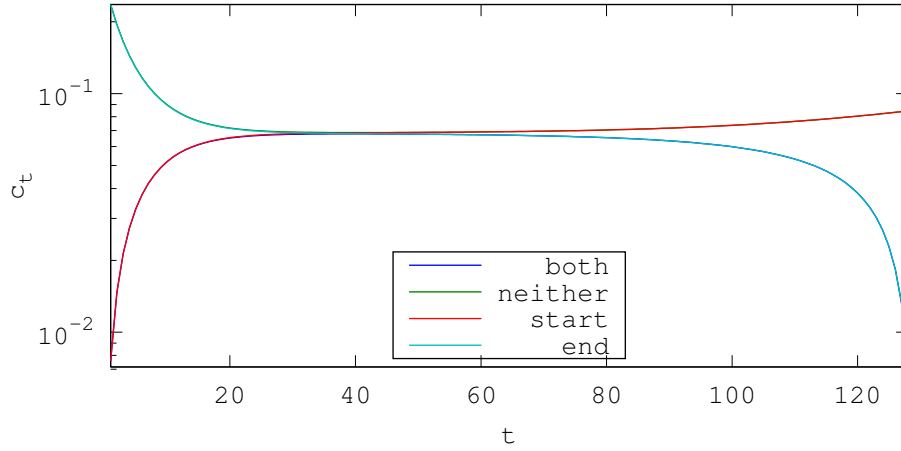
$$\begin{aligned} \mathbf{R}_{t-1}^{-1} &= \mathbf{A}'_t - \frac{\left(\mathbf{u}_t^\top \mathbf{A}'_t \mathbf{u}_t - (1 + \frac{1}{c_t}) \right) \mathbf{b}'_t \mathbf{b}'_t{}^\top}{Z} \\ &\quad + \frac{(\mathbf{b}'_t{}^\top \mathbf{u}_t + 1) (\mathbf{b}'_t \mathbf{u}_t^\top \mathbf{A}'_t + \mathbf{A}'_t \mathbf{u}_t \mathbf{b}'_t{}^\top)}{Z} - \frac{(c'_t - 1) \mathbf{A}'_t \mathbf{u}_t \mathbf{u}_t^\top \mathbf{A}'_t}{Z} \end{aligned}$$

where we abbreviated $Z = \left(\mathbf{u}_t^\top \mathbf{A}'_t \mathbf{u}_t - (1 + \frac{1}{c_t}) \right) (c'_t - 1) - (\mathbf{b}'_t{}^\top \mathbf{u}_t + 1)^2$. On the other hand for $c_t \leq 0$ we find

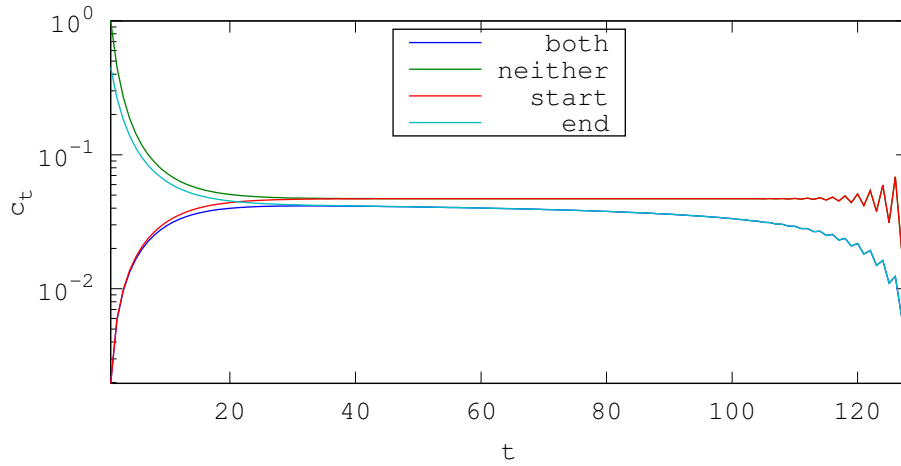
$$\mathbf{R}_{t-1}^{-1} = \left(\mathbf{A}'_t{}^{-1} + \frac{\mathbf{b}_t \mathbf{b}_t{}^\top}{c_t} + \frac{\mathbf{b}_t \mathbf{b}_t{}^\top}{1 - c_t} \right)^{-1} = \mathbf{A}'_t + \frac{\mathbf{b}'_t \mathbf{b}'_t{}^\top}{1 - c'_t}.$$

C.1 The c_t

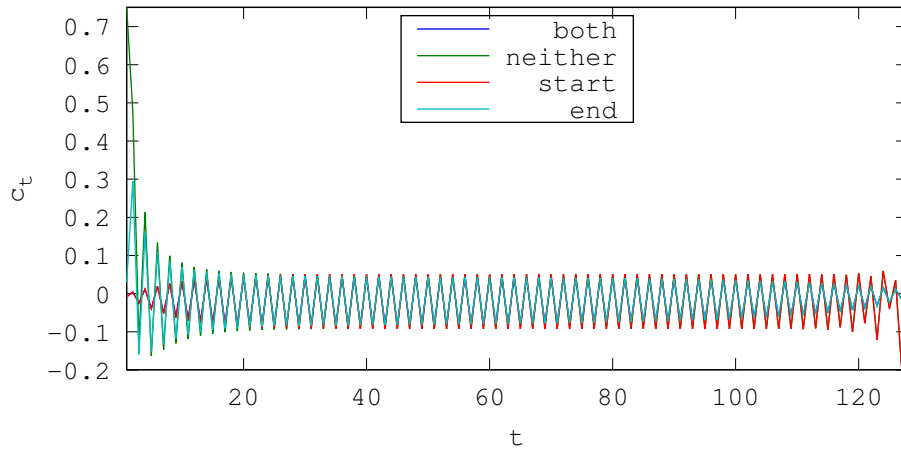
In Figure 2 we graph the c_t , which are crucial in determining the regret, for 4×3 natural choices the regularization matrix \mathbf{K} and the data constraints \mathbf{v}_t . We consider minor variations of the standard second-order regularization (2), where we choose to regularize either end-point down to zero. We consider first order $\mathbf{v}_t = \mathbf{e}_t$, second order $\mathbf{v} = \mathbf{e}_t - \mathbf{e}_{t-1}$ and third order $\mathbf{v}_t = \mathbf{e}_t - 2\mathbf{e}_{t-1} + \mathbf{e}_{t-2}$ data constraints. We see that the first two never hit negative c_t , whereas the third option consistently runs into them.



(a) Norm of data $v_t = e_t$



(b) Norm of increments $v = e_t - e_{t-1}$



(c) Norm of increments of increments $v_t = e_t - 2e_{t-1} + e_{t-2}$

Figure 2: Graphs of the c_t as a function of t for $T = \lambda = 128$ under the assumption that the data are norm bounded (a), increment bounded (b) and increments of increments bounded (c). Note the logarithmic scale on the first two plots and the linear scale on the last plot, where the c_t go negative. Each plot shows four lines, one for each possibility of regularizing the comparator down to zero at either end-point.