Online Regression Competitive with Changing Predictors

Steven Busuttil and Yuri Kalnishkan

Computer Learning Research Centre and Department of Computer Science,
Royal Holloway, University of London,
Egham, Surrey, TW20 0EX, United Kingdom.
{steven,yura}@cs.rhul.ac.uk

Abstract. This paper deals with the problem of making predictions in the online mode of learning where the dependence of the outcome y_t on the signal x_t can change with time. The Aggregating Algorithm (AA) is a technique that optimally merges experts from a pool, so that the resulting strategy suffers a cumulative loss that is almost as good as that of the best expert in the pool. We apply the AA to the case where the experts are all the linear predictors that can change with time. KAARCh is the kernel version of the resulting algorithm. In the kernel case, the experts are all the decision rules in some reproducing kernel Hilbert space that can change over time. We show that KAARCh suffers a cumulative square loss that is almost as good as that of any expert that does not change very rapidly.

1 Introduction

We consider the online protocol where on each trial $t = 1, 2, \ldots$ the learner observes a signal $\mathbf{x}_t$ and attempts to predict the outcome y_t, which is shown to the learner later. The performance of the learner is measured by means of the cumulative square loss. The Aggregating Algorithm (AA), introduced by Vovk in [1] and [2], allows us to merge experts from large pools to obtain optimal strategies. Such an optimal strategy performs nearly as good as the best expert from the class in terms of the cumulative loss.

In [3] the AA is applied to merge all constant linear regressors, i.e., experts θ predicting $\theta'\mathbf{x}_t$ (it is assumed that $\mathbf{x}_t$ and θ are drawn from $\mathbb{R}^n$). The resulting Aggregating Algorithm for Regression (AAR) (also known as the Vovk-Azoury-Warmuth forecaster, see [4, Sect. 11.8]) performs almost as well as the best regressor θ. In [5] the kernel version of AAR, known as the Kernel AAR (KAAR), is introduced and a bound on its performance is derived (see also [6, Sect. 8]). From a computational point of view the algorithm is similar to Ridge Regression. We summarise the results concerning AAR and KAAR in Sect. 2.3.

In this paper, AA is applied to merge a wider class of predictors. We let θ vary between trials. Consider a sequence $\theta_1, \theta_2, \ldots$; let it make the prediction $(\theta_1 + \theta_2 + \ldots + \theta_t)'\mathbf{x}_t$ on trial t. We merge all predictors of this type and obtain an algorithm which is again computationally similar to Ridge Regression. We

call the new algorithm Aggregating Algorithm for Regression with Changing dependencies (AARCh) and its kernelised version KAARCh. Clearly, our class of experts is very large and we cannot compete in a reasonable sense with every expert from this class. However in Sects. 4 and 5 we show that KAARCh can perform almost as well as any regressor if the latter is not changing very rapidly, i.e., if each $\|\theta_t\|$ is small or only a few are nonzero.

A similar problem is considered in [7], [8], and [9] for classification and regression. In these publications, this problem is referred to as the non-stationary or shifting target problem and the corresponding bounds are called shifting bounds. The work by Herbster and Warmuth in [7] is closest to ours. However, their methods are based on Gradient Descent and therefore their bounds are of a different type. For instance, since our approach is based on the Aggregating Algorithm we get a coefficent for the term representing the cumulative loss of the experts equal to 1 (see Theorems 3 and 4), whereas those in the bounds of [7, Theorems 14–16] are greater than 1.

In practice, KAARCh can be used to predict parameters that change slowly with time. KAARCh is more computationally expensive than the techniques described in [7], with time and space complexities that grow with time. This is not desirable in an algorithm designed for online learning; however, a practical implementation is described in [10]. Essentially, KAARCh is made to 'forget' older examples that do not affect the prediction too much. In [10] empirical experiments are carried out on an artificial dataset and on the real world problem of predicting the implied volatility of options (the name KAARCh was inspired by the popular GARCH model for predicting volatility in finance).

2 Background

In this section we introduce some preliminaries and related material required for our main results. As usual, all vectors are identified with one-column matrices and $\mathbf{B}'$ stands for the transpose of matrix $\mathbf{B}$. We will not be specifying the size of simple matrices like the identity matrix $\mathbf{I}$ when this is clear from the context.

2.1 Protocol and Loss

We can define online regression by the following protocol. At every moment in time $t = 1, 2, \ldots$, the value of a signal $\mathbf{x}_t \in X$ arrives. Statistician (or Learner) S observes $\mathbf{x}_t$ and then outputs a prediction $\gamma_t \in \mathbb{R}$. Finally, the outcome $y_t \in \mathbb{R}$ arrives. This can be summarised by the following scheme:

for $t = 1, 2, \ldots$ **do**
  S observes $\mathbf{x}_t \in X$
  S outputs $\gamma_t \in \mathbb{R}$
  S observes $y_t \in \mathbb{R}$
end for

The set X is a signal space which is assumed to be known to Statistician in advance. We will be referring to a signal-outcome pair as an example. The performance of S is measured by the sum of squared discrepancies between the predictions and the outcomes (known as square loss). Therefore on trial t Statistician S suffers loss $(y_t - \gamma_t)^2$. Thus after T trials, the total loss of S is

$$\mathrm{L}_T(S) = \sum_{t=1}^{T} (y_t - \gamma_t)^2 \ .$$

Clearly, a smaller value of $\mathrm{L}_T(S)$ means a better predictive performance.

2.2 Linear and Kernel Predictors

If $X \subseteq \mathbb{R}^n$ we can consider simple linear regressors of the form $\theta \in \mathbb{R}^n$. Given a signal $\mathbf{x} \in X$, such a regressor makes a prediction $\theta'\mathbf{x}$. Linear methods are easy to manipulate mathematically but their use in the real world is limited since they can only model simple dependencies. One solution to this could be to map the data to some high dimensional feature space and then find a simple solution there. This however, can lead to what is known as the curse of dimensionality where both the computational and generalisation performance degrades as the number of features grow [11].

The kernel trick (first used in this context in [12]) is now a widely used technique which can make a linear algorithm operate in feature space without the inherent complexities. Informally, a kernel is a dot product in feature space. Typically, to transform a linear method into a nonlinear one, the linear algorithm is first formulated in such a way that all signals appear only in dot products (known as the dual form). Then these dot-products are replaced by kernels.

For a function $k : X \times X \to \mathbb{R}$ to be a kernel it has to be symmetric, and for all ℓ and all $\mathbf{x}_1, \ldots, \mathbf{x}_\ell \in X$, the kernel matrix $\mathbf{K} = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j}$, $i, j = 1, \ldots, \ell$ must be positive semi-definite (have nonnegative eigenvalues). For every kernel there exists a unique reproducing kernel Hilbert space (RKHS) F such that k is the reproducing kernel of F. In fact, there is a mapping $\phi : X \to F$ such that kernels can be defined as

$$k(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle \ .$$

A RKHS on a set X is a (separable and complete) Hilbert space of real valued functions on X comprised of linear combinations of k of the form

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i k(\mathbf{v}_i, \mathbf{x}) \ ,$$

where l is a positive integer, $c_i \in \mathbb{R}$ and $\mathbf{v}_i, \mathbf{x} \in X$, and their limits. We will be referring to any function in the RKHS F as D. Intuitively $D(\mathbf{x})$ is a decision rule in F that produces a prediction for the object $\mathbf{x}$. We will be measuring the complexity of D by its norm $\|D\|$ in F. For more information on kernels and RKHS see, for example, [13] and [14].

2.3 The Aggregating Algorithm (AA)

We now give an overview of the Aggregating Algorithm (AA) mostly following [3, Sects. 1 and 2]. Let Ω be an outcome space, Γ be a prediction space and Θ be a (possibly infinite) pool of experts. We consider the following game between Statistician (or Learner) S, Nature, and Θ:

for $t = 1, 2, \ldots$ **do**
  Every expert $\theta \in \Theta$ makes a prediction $\gamma_t^{(\theta)} \in \Gamma$
  Statistician S observes all $\gamma_t^{(\theta)}$
  Statistician S outputs a prediction $\gamma_t \in \Gamma$
  Nature outputs $\omega_t \in \Omega$
end for

Given a fixed loss function $\lambda : \Omega \times \Gamma \to [0, \infty]$, Statistician aims to suffer a cumulative loss

$$\mathrm{L}_T(S) = \sum_{t=1}^{T} \lambda(\omega_t, \gamma_t)$$

that is not much larger than the loss

$$\mathrm{L}_T(\theta) = \sum_{t=1}^{T} \lambda(\omega_t, \gamma_t^{(\theta)})$$

of the best expert $\theta \in \Theta$. The AA takes two parameters, a prior probability distribution P_0 in the pool of experts Θ and a learning rate $\eta > 0$. Let $\beta = e^{-\eta}$.

We will first describe the Aggregating Pseudo Algorithm (APA) that does not output actual predictions but generalised predictions. A generalised prediction $g : \Omega \to \mathbb{R}$ is a mapping giving a value of loss for each possible outcome. At every step t, the APA updates the experts' weights so that those that suffered large loss during the previous step have their weights reduced:

$$P_t(d\theta) = \beta^{\lambda(\omega_t, \gamma_t^{(\theta)})} P_{t-1}(d\theta) \ , \quad \theta \in \Theta \ .$$

At time t, the APA chooses a generalised prediction by

$$g_t(\omega) = \log_\beta \int_\Theta \beta^{\lambda(\omega, \gamma_t^{(\theta)})} P_{t-1}^*(d\theta) \ ,$$

where $P_{t-1}^*(d\theta)$ are the normalised weights $P_{t-1}^*(d\theta) = P_{t-1}(d\theta)/P_{t-1}(\Theta)$. This guarantees that for any learning rate $\eta > 0$, prior P_0, and $T = 1, 2, \ldots$ (see [3, Lemma 1])

$$\mathrm{L}_T(\mathrm{APA}) = \log_\beta \int_\Theta \beta^{\mathrm{L}_T(\theta)} P_0(d\theta) \ . \tag{1}$$

To get a prediction from the generalised prediction $g_t(\omega)$ (note that we use ω since we do not yet know the real outcome of step t, ω_t) the AA uses a substitution function Σ mapping generalised predictions into Γ. A substitution function may introduce extra loss; however, in many cases perfect substitution is possible.

We say that the loss function λ is η-mixable if there is a substitution function Σ such that

$$\lambda(\omega_t, \Sigma(g_t(\omega))) \le g_t(\omega_t) \tag{2}$$

on every step t, all experts' predictions and all outcomes. The loss function λ is mixable if it is η-mixable for some $\eta > 0$.

Suppose that our loss function is η-mixable. Using (2) and (1) we can obtain the following upper bound on the cumulative loss of the AA:

$$\mathrm{L}_T(\mathrm{AA}) \le \log_\beta \int_\Theta \beta^{\mathrm{L}_T(\theta)} P_0(d\theta) \ .$$

In particular, when the pool of experts is finite and all experts are assigned equal prior weights, we get, for any $\theta \in \Theta$

$$\mathrm{L}_T(\mathrm{AA}) \le \mathrm{L}_T(\theta) + \frac{\ln m}{\eta} \ ,$$

where m is the size of the pool of experts. This bound can be shown to be optimal in a very strong sense for all algorithms attempting to merge experts' predictions (see [2]).

The Square Loss Game. In this paper we are concerned with the (bounded) square loss game (see [3, Sect. 2.4]), where $\Omega = [-Y, Y]$, $Y \in \mathbb{R}$, $\Gamma = \mathbb{R}$, and $\lambda(\omega, \gamma) = (\omega - \gamma)^2$. The square loss game is η-mixable if and only if $\eta \le 1/(2Y^2)$. A perfect substitution function for this game is

$$\gamma = \frac{g(-Y) - g(Y)}{4Y} \ . \tag{3}$$

The Aggregating Algorithm for Regression (AAR). The AA was applied to the problem of linear regression resulting in the Aggregating Algorithm for Regression (AAR). AAR merges all the linear predictors that map signals to outcomes [3, Sect. 3] (a Gaussian prior is assumed on the pool of experts). AAR makes a prediction at time T by

$$\gamma_{\mathrm{AAR}} = \widetilde{\mathbf{y}}' \widetilde{\mathbf{X}} (\widetilde{\mathbf{X}}' \widetilde{\mathbf{X}} + a\mathbf{I})^{-1} \mathbf{x}_T \ ,$$

where $\widetilde{\mathbf{X}} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)'$ and $\widetilde{\mathbf{y}} = (y_1, y_2, \ldots, y_{T-1}, 0)'$.

The main property of AAR is that it is optimal in the sense that the total loss it suffers is only a little worse than that of any linear predictor. By the latter we mean a strategy that predicts $\theta' \mathbf{x}_t$ on every trial t, where $\theta \in \mathbb{R}^n$ is some fixed vector. The set of all linear predictors may be identified with $\mathbb{R}^n$.

Theorem 1 ([3, Theorem 1]). *For any $a > 0$ and any point in time T,*

$$\mathrm{L}_T(\mathrm{AAR}) \le \inf_\theta (\mathrm{L}_T(\theta) + a\|\theta\|^2) + Y^2 \ln \det \left(\frac{1}{a} \sum_{t=1}^{T} \mathbf{x}_t \mathbf{x}_t' + \mathbf{I} \right) \ .$$

The Kernel Aggregating Algorithm for Regression (KAAR). KAAR, the kernel version of AAR introduced in [5], makes a prediction for the signal $\mathbf{x}_T$ by

$$\gamma_{\text{KAAR}} = \widetilde{\mathbf{y}}'(\widetilde{\mathbf{K}} + a\mathbf{I})^{-1}\widetilde{\mathbf{k}} \ ,$$

where

$$\widetilde{\mathbf{K}} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1, \mathbf{x}_T) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_T, \mathbf{x}_1) & \cdots & k(\mathbf{x}_T, \mathbf{x}_T) \end{bmatrix} \ , \text{ and } \widetilde{\mathbf{k}} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_T) \\ \vdots \\ k(\mathbf{x}_T, \mathbf{x}_T) \end{bmatrix} \ .$$

Like AAR, KAAR has an optimality property. KAAR performs little worse than any decision rule D in the RKHS induced by a kernel function k.

Theorem 2 ([5, Theorem 1] and [6, Sect. 8]). *Let k be a kernel on a space X and D be any decision rule in the RKHS induced by k. Then for every $a > 0$ and any point in time T,*

$$\mathrm{L}_T(\text{KAAR}) \leq \left(\mathrm{L}_T(D) + a\|D\|^2\right) + Y^2 \ln \det \left(\frac{1}{a}\widetilde{\mathbf{K}} + \mathbf{I}\right) \ .$$

Corollary 1 ([6, Sect. 8]). *Under the same conditions of Theorem 2 let $c = \sup_{\mathbf{x}\in X} \sqrt{k(\mathbf{x}, \mathbf{x})}$. Then for every $a > 0$, every $d > 0$, every decision rule D such that $\|D\| \leq d$ and any point in time T, we get*

$$\mathrm{L}_T(\text{KAAR}) \leq \mathrm{L}_T(D) + ad^2 + \frac{Y^2c^2T}{a} \ .$$

If, moreover, T is known in advance, one can choose $a = (Yc/d)\sqrt{T}$ and get

$$\mathrm{L}_T(\text{KAAR}) \leq \mathrm{L}_T(D) + 2Ycd\sqrt{T} \ .$$

3 Algorithm

For our new method, we apply the Aggregating Algorithm (AA) to the regression problem where the experts can change with time. We call this method the Aggregating Algorithm for Regression with Changing dependencies (AARCh). Subsequently, we will kernelise this method to get Kernel AARCh (KAARCh). Throughout this section we will be using the lemmas given in the appendix.

3.1 AARCh: Primal Form

The main idea behind AARCh is to apply the Aggregating Algorithm to the case where the pool of experts is made up of all linear predictors that can change independently with time. We assume that outcomes are bounded by Y, therefore, for any t, $y_t \in [-Y, Y]$ (we do not require our algorithm to know Y). We are interested in the square loss, therefore we will be using optimal $\eta = 1/(2Y^2)$ and substitution function (3).

An expert is a sequence $\theta_1, \theta_2, \ldots$, that at time T predicts

$$\mathbf{x}_T'(\theta_1 + \theta_2 + \ldots + \theta_T) \ ,$$

where for any t, $\theta_t \in \mathbb{R}^n$ and $\mathbf{x}_T \in \mathbb{R}^n$. To apply the AA to this problem we need to define a lower triangular block matrix $\mathbf{L}$, and θ which is a concatenation of all the θ_t for $t = 1 \ldots T$, such that[1]

$$\mathbf{L}\theta = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{I} & \mathbf{I} & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{T-1} \\ \theta_T \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_1 + \theta_2 \\ \vdots \\ \theta_1 + \theta_2 + \cdots + \theta_{T-1} \\ \theta_1 + \theta_2 + \cdots + \theta_{T-1} + \theta_T \end{bmatrix} \ .$$

The matrices $\mathbf{I}$ and $\mathbf{0}$ in $\mathbf{L}$ are the $n \times n$ identity and all-zero matrices respectively. We also need to define $\mathbf{z}_t$ which is $\mathbf{x}_t$ padded with zeros in the following way

$$\mathbf{z}_t = \left[\underbrace{0 \cdots 0}_{n(t-1)} \quad \mathbf{x}_t' \quad \underbrace{0 \cdots 0}_{n(T-t)} \right]' \ ,$$

so that

$$\mathbf{z}_t'\mathbf{L}\theta = \mathbf{x}_t'(\theta_1 + \theta_2 + \ldots + \theta_t) \ .$$

Let $a_t > 0$, $t = 1, \ldots, T$, be arbitrary constants. Consider the prior distribution P_0 in the set $\mathbb{R}^{nT}$ of possible weights θ with the Gaussian density

$$\begin{aligned} P_0(d\theta) &= \left(\prod_{t=1}^{T} a_t \right)^{n/2} \left(\frac{\eta}{\pi} \right)^{nT/2} e^{-\eta \sum_{t=1}^{T} a_t \|\theta_t\|^2} d\theta_1 \ldots d\theta_T \\ &= \left(\left(\frac{\eta}{\pi} \right)^{T} \prod_{t=1}^{T} a_t \right)^{n/2} e^{-\eta \theta' \mathbf{A} \theta} d\theta \ , \end{aligned}$$

where, letting $\mathbf{I}$ and $\mathbf{0}$ be as above, we have

$$\mathbf{A} = \begin{bmatrix} a_1\mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & a_2\mathbf{I} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & a_T\mathbf{I} \end{bmatrix} \ .$$

The loss of θ over the first T trials is

$$\begin{aligned} \mathrm{L}_T(\theta) &= \sum_{t=1}^{T} \left(y_t - \mathbf{z}_t'\mathbf{L}\theta \right)^2 \\ &= \theta'\mathbf{L}' \left(\sum_{t=1}^{T} \mathbf{z}_t\mathbf{z}_t' \right) \mathbf{L}\theta - 2 \left(\sum_{t=1}^{T} y_t\mathbf{z}_t' \right) \mathbf{L}\theta + \sum_{t=1}^{T} y_t^2 \ . \end{aligned}$$

[1] The sum $\theta_1 + \ldots + \theta_t$ corresponds to the predictor $\mathbf{u}_t$ in [7].

Therefore, the loss of the APA is (recall that $\beta = e^{-\eta}$)

$$\begin{aligned}
\mathrm{L}_T(\mathrm{APA}) &= \log_\beta \int_{\mathbb{R}^{nT}} \beta^{\mathrm{L}_T(\theta)} P_0(d\theta) \\
&= \log_\beta \int_{\mathbb{R}^{nT}} \left(\left(\frac{\eta}{\pi}\right)^T \prod_{t=1}^T a_t \right)^{n/2} \\
&\qquad \times e^{-\eta\left(\theta' \mathbf{L}' \left(\sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t'\right) \mathbf{L}\theta - 2\left(\sum_{t=1}^T y_t \mathbf{z}_t'\right)\mathbf{L}\theta + \sum_{t=1}^T y_t^2 + \theta' \mathbf{A}\theta\right)} d\theta \\
&= \log_\beta \int_{\mathbb{R}^{nT}} \left(\left(\frac{\eta}{\pi}\right)^T \prod_{t=1}^T a_t \right)^{n/2} \\
&\qquad \times e^{-\eta\theta'\left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A}\right)\theta + 2\eta\left(\sum_{t=1}^T y_t \mathbf{z}_t'\right)\mathbf{L}\theta - \eta \sum_{t=1}^T y_t^2} d\theta \ .
\end{aligned}$$

Given the generalised prediction $g_T(\omega)$ which is the APA's loss with variable $\omega \in \mathbb{R}$ replacing y_T and using substitution function (3), the AA's prediction is

$$\begin{aligned}
\gamma_T &= \frac{1}{4Y} \log_\beta \frac{\beta^{g_T(-Y)}}{\beta^{g_T(Y)}} \\
&= \frac{1}{4Y} \log_\beta \frac{\int_{\mathbb{R}^{nT}} e^{-\eta\theta'\left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A}\right)\theta + 2\eta\left(\sum_{t=1}^{T-1} y_t \mathbf{z}_t' \mathbf{L} - Y \mathbf{z}_T' \mathbf{L}\right)\theta} d\theta}{\int_{\mathbb{R}^{nT}} e^{-\eta\theta'\left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A}\right)\theta + 2\eta\left(\sum_{t=1}^{T-1} y_t \mathbf{z}_t' \mathbf{L} + Y \mathbf{z}_T' \mathbf{L}\right)\theta} d\theta} \ .
\end{aligned}$$

Let

$$\begin{aligned}
Q_1(\theta) &= \theta' \left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A} \right) \theta - 2 \left(\sum_{t=1}^{T-1} y_t \mathbf{z}_t' \mathbf{L} - Y \mathbf{z}_T' \mathbf{L} \right) \theta \ , \text{ and} \\
Q_2(\theta) &= \theta' \left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A} \right) \theta - 2 \left(\sum_{t=1}^{T-1} y_t \mathbf{z}_t' \mathbf{L} + Y \mathbf{z}_T' \mathbf{L} \right) \theta \ .
\end{aligned}$$

By Lemma 1

$$\begin{aligned}
\gamma_T &= \frac{1}{4Y} \log_\beta \frac{e^{-\eta \min_{\theta \in \mathbb{R}^{nT}} Q_1(\theta)}}{e^{-\eta \min_{\theta \in \mathbb{R}^{nT}} Q_2(\theta)}} \\
&= \frac{1}{4Y} \left(\min_{\theta \in \mathbb{R}^{nT}} Q_1(\theta) - \min_{\theta \in \mathbb{R}^{nT}} Q_2(\theta) \right) \ .
\end{aligned}$$

Finally, by using Lemma 2 we get

$$\begin{aligned}
\gamma_T &= \frac{1}{4Y} F \left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A}, \ -2 \sum_{t=1}^{T-1} y_t \mathbf{z}_t' \mathbf{L}, \ 2Y \mathbf{z}_T' \mathbf{L} \right) \\
&= \sum_{t=1}^{T-1} y_t \mathbf{z}_t' \mathbf{L} \left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A} \right)^{-1} \mathbf{L}' \mathbf{z}_T \ . \qquad (4)
\end{aligned}$$

3.2 AARCh: Dual Form

Let us define

$$\widetilde{\mathbf{Z}} = \begin{bmatrix} \mathbf{z}_1' \\ \mathbf{z}_2' \\ \vdots \\ \mathbf{z}_T' \end{bmatrix} \ , \ \sqrt{\mathbf{A}} = \begin{bmatrix} \sqrt{a_1}\mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \sqrt{a_2}\mathbf{I} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \sqrt{a_T}\mathbf{I} \end{bmatrix} \ , \text{ and } \widetilde{\mathbf{y}} = \begin{bmatrix} y_1 \\ \vdots \\ y_{T-1} \\ 0 \end{bmatrix} \ .$$

We can rewrite (4) in matrix notation to get

$$\begin{aligned} \gamma_T &= \widetilde{\mathbf{y}}'\widetilde{\mathbf{Z}}\mathbf{L}\left(\mathbf{L}'\widetilde{\mathbf{Z}}'\widetilde{\mathbf{Z}}\mathbf{L} + \mathbf{A}\right)^{-1}\mathbf{L}'\mathbf{z}_T \\ &= \widetilde{\mathbf{y}}'\widetilde{\mathbf{Z}}\mathbf{L}\left(\sqrt{\mathbf{A}}\left(\sqrt{\mathbf{A}}^{-1}\mathbf{L}'\widetilde{\mathbf{Z}}'\widetilde{\mathbf{Z}}\mathbf{L}\sqrt{\mathbf{A}}^{-1} + \mathbf{I}\right)\sqrt{\mathbf{A}}\right)^{-1}\mathbf{L}'\mathbf{z}_T \\ &= \widetilde{\mathbf{y}}'\widetilde{\mathbf{Z}}\mathbf{L}\sqrt{\mathbf{A}}^{-1}\left(\sqrt{\mathbf{A}}^{-1}\mathbf{L}'\widetilde{\mathbf{Z}}'\widetilde{\mathbf{Z}}\mathbf{L}\sqrt{\mathbf{A}}^{-1} + \mathbf{I}\right)^{-1}\sqrt{\mathbf{A}}^{-1}\mathbf{L}'\mathbf{z}_T \ . \end{aligned}$$

We can now get a dual formulation of this by using Lemma 3:

$$\gamma_T = \widetilde{\mathbf{y}}'\left(\widetilde{\mathbf{Z}}\mathbf{L}\mathbf{A}^{-1}\mathbf{L}'\widetilde{\mathbf{Z}}' + \mathbf{I}\right)^{-1}\widetilde{\mathbf{Z}}\mathbf{L}\mathbf{A}^{-1}\mathbf{L}'\mathbf{z}_T \ . \tag{5}$$

3.3 KAARCh

Since in (5) signals appear only in dot products, we can use the kernel trick to introduce nonlinearity. In this case we get Kernel AARCh (KAARCh) that at time T makes a prediction

$$\gamma_T = \widetilde{\mathbf{y}}'\left(\bar{\mathbf{K}} + \mathbf{I}\right)^{-1}\bar{\mathbf{k}} \ ,$$

where $\bar{\mathbf{K}} = \left(\left(\sum_{t=1}^{\min(i,j)} \frac{1}{a_t}\right) k(\mathbf{x}_i, \mathbf{x}_j)\right)_{i,j}$, for $i, j = 1, \ldots, T$, i.e.

$$\bar{\mathbf{K}} = \begin{bmatrix} \frac{1}{a_1}k(\mathbf{x}_1,\mathbf{x}_1) & \frac{1}{a_1}k(\mathbf{x}_1,\mathbf{x}_2) & \cdots & \frac{1}{a_1}k(\mathbf{x}_1,\mathbf{x}_T) \\ \frac{1}{a_1}k(\mathbf{x}_2,\mathbf{x}_1) & \left(\frac{1}{a_1}+\frac{1}{a_2}\right)k(\mathbf{x}_2,\mathbf{x}_2) & \cdots & \left(\frac{1}{a_1}+\frac{1}{a_2}\right)k(\mathbf{x}_2,\mathbf{x}_T) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_1}k(\mathbf{x}_T,\mathbf{x}_1) & \left(\frac{1}{a_1}+\frac{1}{a_2}\right)k(\mathbf{x}_T,\mathbf{x}_2) & \cdots & \left(\frac{1}{a_1}+\ldots+\frac{1}{a_T}\right)k(\mathbf{x}_T,\mathbf{x}_T) \end{bmatrix} \ ,$$

and $\bar{\mathbf{k}} = \left(\left(\sum_{t=1}^{i} \frac{1}{a_t}\right) k(\mathbf{x}_i, \mathbf{x}_T)\right)_i$, for $i = 1, \ldots, T$, i.e.

$$\bar{\mathbf{k}} = \begin{bmatrix} \frac{1}{a_1}k(\mathbf{x}_1,\mathbf{x}_T) \\ \left(\frac{1}{a_1}+\frac{1}{a_2}\right)k(\mathbf{x}_2,\mathbf{x}_T) \\ \vdots \\ \left(\frac{1}{a_1}+\ldots+\frac{1}{a_T}\right)k(\mathbf{x}_T,\mathbf{x}_T) \end{bmatrix} \ .$$

4 Upper Bounds

In this section we use the Aggregating Algorithm's properties to derive upper bounds on the cumulative square loss suffered by AARCh and KAARCh, compared to that of any expert in the pool.

4.1 AARCh Loss Upper Bound

Theorem 3. *For any point in time T and any $a_t > 0$, $t = 1, \dots, T$,*

$$\mathrm{L}_T(\mathrm{AARCh}) \le \inf_\theta \left(\mathrm{L}_T(\theta) + \sum_{t=1}^T a_t \|\theta_t\|^2 \right) + Y^2 \ln \det \left(\sqrt{\mathbf{A}}^{-1} \mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} \sqrt{\mathbf{A}}^{-1} + \mathbf{I} \right) \quad . \tag{6}$$

Proof. Given the Aggregating Algorithm's properties, we know that

$$\begin{aligned}
\mathrm{L}_T(\mathrm{AARCh}) &\le \log_\beta \int_{\mathbb{R}^{nT}} \beta^{\mathrm{L}_T(\theta)} P_0(d\theta) \\
&= \log_\beta \left(\left(\frac{\eta}{\pi} \right)^T \prod_{t=1}^T a_t \right)^{n/2} \\
&\quad \times \int_{\mathbb{R}^{nT}} e^{-\eta \left(\theta' \left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A} \right) \theta - 2 \left(\sum_{t=1}^T y_t \mathbf{z}_t \right) \mathbf{L} \theta + \sum_{t=1}^T y_t^2 \right)} d\theta \quad .
\end{aligned}$$

By Lemma 1 this is equal to

$$\begin{aligned}
&\inf_\theta \left(\mathrm{L}_T(\theta) + \theta' \mathbf{A} \theta \right) + \log_\beta \left(\left(\left(\frac{\eta}{\pi} \right)^T \prod_{t=1}^T a_t \right)^{n/2} \frac{\pi^{nT/2}}{\sqrt{\det \left(\eta \mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \eta \mathbf{A} \right)}} \right) \\
&= \inf_\theta \left(\mathrm{L}_T(\theta) + \sum_{t=1}^T a_t \|\theta_t\|^2 \right) + \log_\beta \sqrt{\frac{\left(\eta^T \prod_{t=1}^T a_t \right)^n}{\det \left(\eta \mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \eta \mathbf{A} \right)}} \\
&= \inf_\theta \left(\mathrm{L}_T(\theta) + \sum_{t=1}^T a_t \|\theta_t\|^2 \right) + \frac{1}{2} \log_\beta \left(\frac{\prod_{t=1}^T a_t^n}{\det \left(\mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} + \mathbf{A} \right)} \right) \\
&= \inf_\theta \left(\mathrm{L}_T(\theta) + \sum_{t=1}^T a_t \|\theta_t\|^2 \right) \\
&\quad - \frac{1}{2} \log_\beta \left(\frac{\det \left(\sqrt{\mathbf{A}} \left(\sqrt{\mathbf{A}}^{-1} \mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} \sqrt{\mathbf{A}}^{-1} + \mathbf{I} \right) \sqrt{\mathbf{A}} \right)}{\prod_{t=1}^T a_t^n} \right) \\
&= \inf_\theta \left(\mathrm{L}_T(\theta) + \sum_{t=1}^T a_t \|\theta_t\|^2 \right) + Y^2 \ln \det \left(\sqrt{\mathbf{A}}^{-1} \mathbf{L}' \sum_{t=1}^T \mathbf{z}_t \mathbf{z}_t' \mathbf{L} \sqrt{\mathbf{A}}^{-1} + \mathbf{I} \right) \quad .
\end{aligned}$$

4.2 KAARCh Loss Upper Bound

The following generalises Theorem 3. Note that we cannot repeat the proof for the linear case directly since it involves the evaluation of an integral over the space $\mathbb{R}^{nT}$.

Theorem 4. *Let k be a kernel on a space X, let D_t, $t = 1 \dots T$, be any decision rules in the RKHS F induced by k and let $D = (D_1, D_2, \dots, D_T)'$. Then, for any point in time T and every $a_t > 0$, $t = 1, \dots, T$,*

$$\mathrm{L}_T(\mathrm{KAARCh}) \leq \mathrm{L}_T(D) + \sum_{t=1}^{T} a_t \|D_t\|^2 + Y^2 \ln\det\left(\bar{\mathbf{K}} + \mathbf{I}\right) \quad . \tag{7}$$

Proof. It will be sufficient to prove this for D_t of the form

$$f_t(\mathbf{x}) = \sum_{i=1}^{l^{(t)}} c_i^{(t)} k(\mathbf{v}_i^{(t)}, \mathbf{x}) \quad ,$$

where $l^{(t)}$ are positive integers, $c_i^{(t)} \in \mathbb{R}$, and $\mathbf{v}_i^{(t)}, \mathbf{x} \in X$ (we use $^{(t)}$ to show that these parameters can be different for each f_t). This is because such finite sums are dense in the RKHS F. If we take $f = (f_1, f_2, \dots, f_T)'$, (7) becomes

$$\mathrm{L}_T(\mathrm{KAARCh}) \leq \mathrm{L}_T(f) + \sum_{t=1}^{T} a_t \sum_{i,j=1}^{l^{(t)}} c_i^{(t)} c_j^{(t)} k(\mathbf{v}_i^{(t)}, \mathbf{v}_j^{(t)}) + Y^2 \ln\det\left(\bar{\mathbf{K}} + \mathbf{I}\right) \quad , \tag{8}$$

where

$$\mathrm{L}_T(f) = \sum_{t=1}^{T} \left(y_t - \sum_{i=1}^{l^{(t)}} c_i^{(t)} k(\mathbf{v}_i^{(t)}, \mathbf{x}_t) \right)^2 \quad .$$

In the special case when $X = \mathbb{R}^n$ and $k(\mathbf{v}_i, \mathbf{v}_j) = \mathbf{v}_i' \mathbf{v}_j$ for every $\mathbf{v}_i, \mathbf{v}_j \in X$, (8) follows directly from (6). Indeed, a kernel predictor f_t reduces to the linear predictor $\theta_t = \sum_{i=1}^{l^{(t)}} c_i^{(t)} \mathbf{v}_i^{(t)}$ and the term $\sum_{i,j=1}^{l^{(t)}} c_i^{(t)} c_j^{(t)} k\left(\mathbf{v}_i^{(t)}, \mathbf{v}_j^{(t)}\right)$ equals the squared quadratic norm of θ_t. Finally, by Sylvester's determinant identity (see also Lemma 4 for an independent proof of this) we know that

$$\begin{aligned} \det\left(\bar{\mathbf{K}} + \mathbf{I}\right) &= \det\left(\widetilde{\mathbf{Z}}\mathbf{L}\mathbf{A}^{-1}\mathbf{L}'\widetilde{\mathbf{Z}}' + \mathbf{I}\right) \\ &= \det\left(\sqrt{\mathbf{A}}^{-1}\mathbf{L}'\widetilde{\mathbf{Z}}'\widetilde{\mathbf{Z}}\mathbf{L}\sqrt{\mathbf{A}}^{-1} + \mathbf{I}\right) \quad . \end{aligned}$$

The general case can be obtained by using finite dimensional approximations. Recall that inherent in every kernel is a function ϕ that maps objects to the RKHS F, which is isomorphic to $l_2 = \{\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots) | \sum_{i=1}^{\infty} \alpha_i^2 \text{ converges}\}$. Let us consider the sequence on subspaces $R_1 \subseteq R_2 \subseteq \ldots \subseteq F$. The set $R_s = \{(\alpha_1, \alpha_2, \dots, \alpha_s, 0, 0, \dots)\}$ may be identified with $\mathbb{R}^s$. Let $p_s : F \to R_s$ be the

projection operator $p_s(\boldsymbol{\alpha}) = (\alpha_1, \alpha_2, \ldots, \alpha_s, 0, 0, \ldots)$, $\phi_s : X \to R_s$ be $\phi_s = p_s(\phi)$, and k_s be given by $k_s(\mathbf{v}_1, \mathbf{v}_2) = \langle \phi_s(\mathbf{v}_1), \phi_s(\mathbf{v}_2) \rangle$, where $\mathbf{v}_1, \mathbf{v}_2 \in X$.

Inequality (8) holds for k_s since R_s has a finite dimension. If (8) is violated, then its counterpart with some large s is violated too and this observation completes the proof.

5 Discussion

In this section we shall analyse upper bound (7) in order to obtain an equivalent of Corollary 1. Our goal is to show that KAARCh's cumulative loss is less or equal to that of a wide class of experts plus a term of the order $o(T)$.

Estimating the determinant of a positive definite matrix by the product of its diagonal elements (see [15, Sect. 2.10, Theorem 7]) and using the inequality $\ln(1+x) \le x$ (in our case x is small, and therefore the resulting bound is tight), we get

$$\begin{aligned} Y^2 \ln\det\left(\bar{\mathbf{K}} + \mathbf{I}\right) &\le Y^2 \sum_{t=1}^{T} \ln\left(1 + c^2 \sum_{i=1}^{t} \frac{1}{a_i}\right) \\ &\le Y^2 c^2 \sum_{t=1}^{T} \sum_{i=1}^{t} \frac{1}{a_i} \\ &= Y^2 c^2 \sum_{t=1}^{T} \frac{T-t+1}{a_t} \quad , \end{aligned}$$

where $c = \sup_{\mathbf{x} \in X} \sqrt{k(\mathbf{x}, \mathbf{x})}$.

It is natural to single out the first decision rule D_1 and the corresponding coefficient a_1 from the rest. We may think of it as corresponding to the choice of the 'principal' dependency; let the rest of D_t $(t = 2, \ldots, T)$ be small correction terms. Let us take equal $a_2 = \ldots = a_t = a$. We get

$$\begin{aligned} \mathrm{L}_T(\mathrm{KAARCh}) \le \mathrm{L}_T(D) &+ \left(a_1 \|D_1\|^2 + \frac{Y^2 c^2 T}{a_1}\right) \\ &+ \left(a \sum_{t=2}^{T} \|D_t\|^2 + \frac{Y^2 c^2 T(T-1)}{2a}\right) \quad . \end{aligned} \tag{9}$$

If we bound the norm of D_1 by d_1 and assume that T is known in advance, a_1 may be chosen as in Corollary 1. The second term in the right hand side of (9) can thus be bounded by $O\left(\sqrt{T}\right)$. If we assume that $\sum_{t=2}^{T} \|D_t\|^2 \le s(T)$, then the estimate is minimised by $a = \sqrt{Y^2 c^2 T(T-1)/(2s(T))}$ and the third term in the right hand side of (9) can be bounded by $O\left(T\sqrt{s(T)}\right)$. We therefore get the following corollary:

Corollary 2. *Under the conditions of Theorem 4, let T be known in advance and $c = \sup_{\mathbf{x}\in X} \sqrt{k(\mathbf{x},\mathbf{x})}$. For every every $d_1 > 0$ and every function $s(T)$, if $\|D_1\| \le d_1$ and $\sum_{t=2}^{T} \|D_t\|^2 \le s(T)$, then a_t, for $t = 1,\dots,T$, can be chosen so that*

$$\mathrm{L}_T(\mathrm{KAARCh}) \le \mathrm{L}_T(D) + 2Ycd_1\sqrt{T} + 2Yc\sqrt{s(T)T(T-1)/2} \ .$$

If $s(T) = o(1)$, then $\mathrm{L}_T(\mathrm{KAARCh}) \le \mathrm{L}_T(D) + o(T)$.

The estimate $s(T) = o(1)$ can be achieved in two natural ways. First, one can assume that each $\|D_t\|$, for $t = 2,\dots,T$, is small.

Corollary 3. *Under the conditions of Theorem 4, let T be known in advance. For every positive d, d_1, and ε, if $\|D_1\| \le d_1$ and, for $t = 2,\dots,T$,*

$$\|D_t\| \le \frac{d}{T^{0.5+\varepsilon}} \ ,$$

then

$$\begin{aligned}\mathrm{L}_T(\mathrm{KAARCh}) &\le \mathrm{L}_T(D) + O\left(T^{\max(0.5,(1-\varepsilon))}\right)\\ &= \mathrm{L}_T(D) + o(T) \ .\end{aligned}$$

Secondly, one may assume that there are only a few nonzero D_t, for $t = 2,\dots,T$. In this case, the nonzero D_t can have greater flexibility.

Acknowledgements. We thank Volodya Vovk and Alex Gammerman for valuable discussions. We are grateful to Michael Vyugin for suggesting the problem of predicting implied volatility which inspired this work.

References

1. Vovk, V.: Aggregating strategies. In Fulk, M., Case, J., eds.: Proceedings of the 3rd Annual Workshop on Computational Learning Theory, Morgan Kaufmann (1990) 371–383
2. Vovk, V.: A game of prediction with expert advice. Journal of Computer and System Sciences **56** (1998) 153–173
3. Vovk, V.: Competitive on-line statistics. International Statistical Review **69**(2) (2001) 213–248
4. Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning, and Games. Cambridge University Press (2006)
5. Gammerman, A., Kalnishkan, Y., Vovk, V.: On-line prediction with kernels and the complexity approximation principle. In: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, AUAI Press (2004) 170–176
6. Vovk, V.: On-line regression competitive with reproducing kernel Hilbert spaces. Technical Report arXiv:cs.LG/0511058 (version 2), arXiv.org (2006)
7. Herbster, M., Warmuth, M.K.: Tracking the best linear predictor. Journal of Machine Learning Research **1** (2001) 281–309

8. Kivinen, J., Smola, A.J., Williamson, R.C.: Online learning with kernels. IEEE Transactions on Signal Processing **52**(8) (2004) 2165–2176
9. Cavallanti, G., Cesa-Bianchi, N., Gentile, C.: Tracking the best hyperplane with a simple budget perceptron. Machine Learning (to appear)
10. Busuttil, S., Kalnishkan, Y.: Weighted kernel regression for predicting changing dependencies. In: Proceedings of the 18th European Conference on Machine Learning (ECML 2007). (to appear)
11. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines (and Other Kernel-Based Learning Methods). Cambridge University Press, UK (2000)
12. Aizerman, M., Braverman, E., Rozonoer, L.: Theoretical foundations of the potential function method in pattern recognition learning. Automation and Remote Control **25** (1964) 821–837
13. Aronszajn, N.: Theory of reproducing kernels. Transactions of the American Mathematical Society **68** (1950) 337–404
14. Schölkopf, B., Smola, A.J.: Learning with Kernels — Support Vector Machines, Regularization, Optimization and Beyond. The MIT Press, USA (2002)
15. Beckenbach, E.F., Bellman, R.: Inequalities. Springer (1961)

Appendix

Lemma 1. *Let $Q(\theta) = \theta'\mathbf{A}\theta + \mathbf{b}'\theta + c$, where $\theta, \mathbf{b} \in \mathbb{R}^n$, c is a scalar and $\mathbf{A}$ is a symmetric positive definite $n \times n$ matrix. Then*

$$\int_{\mathbb{R}^n} e^{-Q(\theta)} d\theta = e^{-Q_0} \frac{\pi^{n/2}}{\sqrt{\det \mathbf{A}}} \ ,$$

where $Q_0 = \min_{\theta \in \mathbb{R}^n} Q(\theta)$.

Proof. Let $\theta_0 \in \arg\min Q$. Take $\xi = \theta - \theta_0$ and $\widetilde{Q}(\xi) = Q(\xi + \theta_0)$. It is easy to see that the quadratic part of $\widetilde{Q}$ is $\xi'\mathbf{A}\xi$. Since $0 \in \arg\min \widetilde{Q}$, the form has no linear term. Indeed, in the vicinity of 0 the linear term dominates over the quadratic term; if $\widetilde{Q}$ has a non-zero linear term, it cannot have a minimum at 0. Since $Q_0 = \min_{\xi \in \mathbb{R}^n} \widetilde{Q}(\xi)$, we can conclude that the constant term in $\widetilde{Q}$ is Q_0. Thus $\widetilde{Q}(\xi) = \xi'\mathbf{A}\xi + Q_0$.

It remains to show that $\int_{\mathbb{R}^n} e^{-\xi'\mathbf{A}\xi} d\xi = \pi^{n/2}/\sqrt{\det \mathbf{A}}$. This can be proved by considering a basis where $\mathbf{A}$ diagonalises (or see [15, Sect. 2.7, Theorem 3]).

Lemma 2. *Let*

$$F(\mathbf{A}, \mathbf{b}, \mathbf{x}) = \min_{\theta \in \mathbb{R}^n} (\theta'\mathbf{A}\theta + \mathbf{b}'\theta + \mathbf{x}'\theta) - \min_{\theta \in \mathbb{R}^n} (\theta'\mathbf{A}\theta + \mathbf{b}'\theta - \mathbf{x}'\theta) \ ,$$

where $\mathbf{b}, \mathbf{x} \in \mathbb{R}^n$ and $\mathbf{A}$ is a symmetric positive definite $n \times n$ matrix. Then $F(\mathbf{A}, \mathbf{b}, \mathbf{x}) = -\mathbf{b}'\mathbf{A}^{-1}\mathbf{x}$.

Proof. It can be shown by differentiation that the first minimum is achieved at $\theta_1 = -\frac{1}{2}\mathbf{A}^{-1}(\mathbf{b} + \mathbf{x})$ and the second minimum at $\theta_2 = -\frac{1}{2}\mathbf{A}^{-1}(\mathbf{b} - \mathbf{x})$. The substitution proves the lemma.

Lemma 3. *Given a matrix* $\mathbf{A}$*, a scalar* a *and* $\mathbf{I}$ *identity matrices of the appropriate size,*

$$(\mathbf{A}\mathbf{A}' + a\mathbf{I})^{-1}\mathbf{A} = \mathbf{A}(\mathbf{A}'\mathbf{A} + a\mathbf{I})^{-1} \ .$$

Proof.

$$\begin{aligned}(\mathbf{A}\mathbf{A}' + a\mathbf{I})^{-1}\mathbf{A} &= (\mathbf{A}\mathbf{A}' + a\mathbf{I})^{-1}\mathbf{A}(\mathbf{A}'\mathbf{A} + a\mathbf{I})(\mathbf{A}'\mathbf{A} + a\mathbf{I})^{-1} \\ &= (\mathbf{A}\mathbf{A}' + a\mathbf{I})^{-1}(\mathbf{A}\mathbf{A}'\mathbf{A} + a\mathbf{A})(\mathbf{A}'\mathbf{A} + a\mathbf{I})^{-1} \\ &= (\mathbf{A}\mathbf{A}' + a\mathbf{I})^{-1}(\mathbf{A}\mathbf{A}' + a\mathbf{I})\mathbf{A}(\mathbf{A}'\mathbf{A} + a\mathbf{I})^{-1} \\ &= \mathbf{A}(\mathbf{A}'\mathbf{A} + a\mathbf{I})^{-1}\end{aligned}$$

Lemma 4. *For every matrix* $\mathbf{M}$ *the equality* $\det(\mathbf{I} + \mathbf{M}'\mathbf{M}) = \det(\mathbf{I} + \mathbf{M}\mathbf{M}')$ *holds (where* $\mathbf{I}$ *are identity matrices of the correct size).*

Proof. Suppose that $\mathbf{M}$ is an $n \times m$ matrix. Thus $(\mathbf{I} + \mathbf{M}\mathbf{M}')$ and $(\mathbf{I} + \mathbf{M}'\mathbf{M})$ are $n \times n$ and $m \times m$ matrices respectively. Without loss of generality, we may assume that $n \geq m$ (otherwise we swap $\mathbf{M}$ and $\mathbf{M}'$). Let the columns of $\mathbf{M}$ be m vectors $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbb{R}^n$.

We have $\mathbf{M}\mathbf{M}' = \sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i'$. Let us see how the operator $\mathbf{M}\mathbf{M}'$ acts on a vector $\mathbf{x} \in \mathbb{R}^n$. By associativity, $\mathbf{x}_i\mathbf{x}_i'\mathbf{x} = (\mathbf{x}_i'\mathbf{x})\mathbf{x}_i$, where $\mathbf{x}_i'\mathbf{x}$ is a scalar. Therefore, if U is the span of $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$, then $\mathbf{M}\mathbf{M}'(\mathbb{R}^n) \subseteq U$. In a similar way, it follows that $(\mathbf{I} + \mathbf{M}\mathbf{M}')(U) \subseteq U$. On the other hand, if $\mathbf{x}$ is orthogonal to $\mathbf{x}_i$, then $\mathbf{x}_i\mathbf{x}_i'\mathbf{x} = (\mathbf{x}_i'\mathbf{x})\mathbf{x}_i = 0$. Hence $\mathbf{M}\mathbf{M}'(U^\perp) = 0$, where $U^\perp$ is the orthogonal complement to U with respect to $\mathbb{R}^n$. Consequently, $(\mathbf{I} + \mathbf{M}\mathbf{M}')|_{U^\perp} = \mathbf{I}$ (by $\mathbf{B}|_V$ we denote the restriction of an operator $\mathbf{B}$ to a subspace V). Therefore $(\mathbf{I} + \mathbf{M}\mathbf{M}')(U^\perp) \subseteq U^\perp$.

One can see that both U and $U^\perp$ are invariant subspaces of $(\mathbf{I} + \mathbf{M}\mathbf{M}')$. If we choose bases in U and in $U^\perp$ and then concatenate them, we get a basis of $\mathbb{R}^n$. In this basis the matrix of $(\mathbf{I} + \mathbf{M}\mathbf{M}')$ has the form

$$\begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \ ,$$

where $\mathbf{A}$ is the matrix of $(\mathbf{I} + \mathbf{M}\mathbf{M}')|_U$. It remains to evaluate $\det(\mathbf{A})$.

First let us consider the case of linearly independent $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$. They form a basis of U and we may use it to calculate the determinant of the operator $(\mathbf{I} + \mathbf{M}\mathbf{M}')|_U$. However,

$$(\mathbf{I} + \mathbf{M}\mathbf{M}')\mathbf{x}_i = \mathbf{x}_i + \sum_{j=1}^{m} (\mathbf{x}_j'\mathbf{x}_i)\mathbf{x}_j$$

and thus the matrix of the operator $(\mathbf{I} + \mathbf{M}\mathbf{M}')|_U$ in the basis $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ is $(\mathbf{I} + \mathbf{M}'\mathbf{M})$.

The case of linearly dependent $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ follows by continuity. Indeed, m vectors in an n-dimensional space with $n \geq m$ may be approximated by m independent vectors to any degree of precision and the determinant is a continuous function of the elements of a matrix.