S@MR

СИБИРСКИЕ ЭЛЕКТРОННЫЕ
МАТЕМАТИЧЕСКИЕ ИЗВЕСТИЯ
Siberian Electronic Mathematical Reports
http://semr.math.nsc.ru
ISSN 1813-3304

Vol. 20, No. 2, pp. 144–145 (2023) УДК 004.852
https://doi.org/10.33048/semi.2023.20.??? MSC 68Q32,
68W27, 68W20

RANDOM FEATURE-BASED DOUBLE VOVK-AZOURY-WARMUTH ALGORITHM FOR ONLINE MULTI-KERNEL LEARNING

D.B. ROKHLIN AND **O.V. GURTOVAYA**

Communicated by ...

Abstract: We introduce a novel multi-kernel learning algorithm, VAW^2, for online least squares regression in reproducing kernel Hilbert spaces (RKHS). VAW^2 leverages random Fourier feature-based functional approximation and the Vovk-Azoury-Warmuth (VAW) method in a two-level procedure: VAW is used to construct expert strategies from random features generated for each kernel at the first level, and then again to combine their predictions at the second level. A theoretical analysis yields a regret bound of $O(T^{1/2} \ln T)$ in expectation with respect to artificial randomness, when the number of random features scales as $T^{1/2}$. Empirical results on some benchmark datasets demonstrate that VAW^2 achieves superior performance compared to the existing online multi-kernel learning algorithms: Raker and OMKL-GF, and to other theoretically grounded method methods involving convex combination of expert predictions at the second level.

Keywords: Vovk-Azoury-Warmuth algorithm, online multi-kernel learning, RKHS, random Fourier features, regret bounds.

Rokhlin, D.B., Gortovaya, O.V., Random feature-based double Vovk-Azoury-Warmuth algorithm for online multi-kernel learning.

The research of D.B. Rokhlin was supported by the Regional Mathematical Center of the Southern Federal University with the Agreement no. 075-02-2025-1720 of the Ministry of Science and Higher Education of Russia.

Received January, 1, 2023, Published December, 31, 2023.

1 Introduction

Kernel methods [15, 7] allow to extend the scope of the linear models to the analysis of complex nonlinear dependencies by working in reproducing kernel Hilbert spaces (RKHS). They combine high expressive power, formalized as the universality property (see, e.g. [18]), with the possibility of using tools from the convex analysis to establish global optimality results. However, the computational complexity of these methods grows as T^3, where T is the number of examples in a classical batch supervised learning problem.

The gradient descent algorithm, being applied in an RKHS in the online mode [9], at each iteration increases the complexity of the linear combination of kernels by adding a new "support vector" (SV) to a dictionary. There is a lot of techniques for dealing with this phenomenon, called the curse of kernelization [23]. These techniques can be broadly categorized into budget maintenance strategies and functional approximation strategies [8]. The budget maintenance strategies include SV removal, SV projection and SV merging families of algorithms. Similar kernel adaptive filtering algorithms were developed for signal processing [17, 20].

In this paper, we follow the functional approximation strategy [10] based on random Fourier features (RFF) [12]. This approach allows to work in a fixed dimensional space in each iteration. However, to get sublinear regret with respect to a ball in an RKHS, the dimension of this space, which is equal to the number of random features, should grow with T.

Besides the computational complexity, another issue with kernel methods concerns the kernel selection, which essentially influences the results. Multi-kernel methods try to address this issue by choosing a kernel combination from a large preselected dictionary [6]. In the online learning setting multi-kernel methods in conjunction with the RFF-based functional approximation were used in [14, 16]. These papers apply the online gradient descent method to random feature vectors related to each kernel to generated "expert" strategies, and then combine their predictions by an exponential weight update rule (used in both papers [14, 16]), or by the online gradient descent algorithm (used in [14]).

In this paper we are interested in the online least squares regression problem in the RKHS spaces. In the finite dimensional case the Vovk-Azoury-Warmuth (VAW) algorithm [21, 1] provides an optimal regret bound $O(\ln T)$ (see also [4]). In the general case the regret w.r.t. a predictor in a ball of an RKHS can be bounded by $O(T^{1/2})$ [22], and this bound is not improvable. We consider the problem in the multi-kernel setting and prove a loss bound $O(T^{1/2} \ln T)$ in expectation w.r.t. an artificial randomness. This bound is obtained via computationally feasible algorithms. We apply VAW algorithm for construction of expert strategies and either VAW or exponentially weighted average (EWA) forecasting algorithm for combining expert predictions. Note that the regret bounds of [14, 16] are not applicable due to the lack of the global Lipschitz condition.

The paper is organized as follows. In Section 2 we recall the definition of an RKHS space and fix a class of RKHS spaces with translation invariant kernels as in [13]. We also provide a simple result related to approximation by a linear combination of random features (Lemma 1), and recall the basic regret bound of the VAW algorithm.

The main results are contained Section 3. We consider a dictionary, containing N kernels k_i and related RKHS spaces $\mathcal{H}_i$. For each kernel we generate m random features and apply the VAW algorithm either to the concatenated Nm-dimensional vector (Theorem 1), or to each m-dimensional vector separately. In the second case we combine prediction of the "expert" VAW algorithms either by the VAW algorithm (Theorem 2), or by the EWA algorithm (Theorem 3). The first approach (adopted in Theorem 1) provides the regret bounds w.r.t. elements of a ball in the large RKHS space $\mathcal{H}$ with the kernel $k = k_1 + \cdots + k_N$, while the second approach provides the same bound only w.r.t. the elements of a ball in each $\mathcal{H}_i$. At the same time, the second approach has lower computational and spatial complexity, and we consider it to be the primary one.

In Section 4, we provide computer experiments on several benchmark datasets. We compare the performance of VAW^2 against state-of-the-art online multi-kernel learning algorithms, and to other traditional methods of combining VAW expert predictions. The results demonstrate the effectiveness of VAW^2 in achieving superior prediction accuracy. Section 5 concludes.

2 Preliminaries

Recall that a reproducing kernel Hilbert space (RKHS) is a Hilbert space $\mathcal{H}$ of functions $f : \mathcal{X} \to \mathbb{R}$ such that any evaluation functional $x \mapsto f(x)$, $x \in \mathcal{X}$ is bounded. If $\mathcal{H}$ is an RKHS on $\mathcal{X}$, then by the Riesz representation theorem for each $x \in \mathcal{X}$ there exists a unique element, $k_x \in \mathcal{H}$, such that for every $f \in \mathcal{H}$,

$$f(x) = \langle f, k_x \rangle_{\mathcal{H}}.$$

The function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined by $k(x, x') = k_x(x')$ is called the reproducing kernel of $\mathcal{H}$. The kernel can be expressed via the feature map $x \mapsto k_x$: $k(x, x') = \langle k_x, k_{x'} \rangle_{\mathcal{H}}$.

Consider a continuous function $\phi : \mathbb{R}^d \times \Theta \to [-a, a]$, where Θ is a closed subset of a finite dimensional space. Following [13], we will consider only reproducing kernel Hilbert spaces of the form

$$\mathcal{H} = \left\{ x \mapsto f(x) = \int_\Theta \alpha(\theta)\phi(x;\theta)\, d\theta : \int_\Theta \frac{\alpha^2(\theta)}{p(\theta)}\, d\theta < \infty \right\} \tag{1}$$

with the inner product

$$\langle f, g \rangle_{\mathcal{H}} = \int_\Theta \frac{\alpha(\theta)\beta(\theta)}{p(\theta)}\, d\theta,$$

where $g(x) = \int \beta(\theta)\phi(x;\theta)\,d\theta$. In [13, Proposition 4.1] it is proved that $\mathcal{H}$ is an RKHS with the reproducing kernel

$$k(x,y) = \int_\Theta p(\theta)\phi(x;\theta)\phi(y;\theta)\,d\theta. \tag{2}$$

In particular,

$$\begin{aligned}\langle f, k(x,\cdot)\rangle_{\mathcal{H}} &= \left\langle \int_\Theta \alpha(\theta)\phi(\cdot;\theta)\,d\theta, \int_\Theta p(\theta)\phi(x;\theta)\phi(\cdot;\theta)\,d\theta \right\rangle_{\mathcal{H}} \\ &= \int_\Theta \alpha(\theta)\phi(x;\theta)\,d\theta = f(x).\end{aligned}$$

Assume that the kernel k is translation invariant: $k(x,y) = \kappa(x-y)$. Then by the Bochner theorem

$$\kappa(z) = \int_{\mathbb{R}^d} e^{i\langle\omega,z\rangle}\Lambda(d\omega)$$

for some non-negative σ-additive measure Λ on the Borel σ-algebra $\mathscr{B}(\mathbb{R}^d)$. For our purposes it is enough to assume that Λ is absolutely continuous w.r.t. the Lebesgue measure:

$$\kappa(z) = \int_{\mathbb{R}^d} e^{i\langle\omega,z\rangle}q(\omega)\,d\omega = \int_{\mathbb{R}^d} q(\omega)\cos\langle\omega,z\rangle\,d\omega,$$

where q is a probability density function. In particular, for Gaussian kernels:

$$k(x,y) = e^{-\|x-y\|_2^2/(2\sigma^2)}, \quad q(\omega) = \left(\frac{\sigma}{\sqrt{2\pi}}\right)^d e^{-\sigma^2\|\omega\|_2^2/2}, \tag{3}$$

for Laplacian kernels:

$$k(x,y) = e^{-\|x-y\|_1/\sigma}, \quad q(\omega) = \frac{\sigma^d}{\pi^d}\prod_{j=1}^d \frac{1}{1+\sigma^2\omega_j^2}. \tag{4}$$

We see that here q are products of Gaussian and Cauchy distributions respectively (see [12]).

For such kernels formula (2) holds true with $\Theta = \mathbb{R}^d \times [0,2\pi]$, $\theta = (\omega, b)$,

$$\begin{aligned}p(\theta) &= q(\omega)r(b), \quad r(b) = 1/(2\pi), \\ \phi(x;\theta) &= \sqrt{2}\cos(\langle\omega,x\rangle + b),\end{aligned}$$

(see [12]). We have,

$$\begin{aligned}\int_\Theta p(\theta)\phi(x;\theta)\phi(y;\theta)\,d\theta &= \frac{1}{2\pi}\int_0^{2\pi}\int_{\mathbb{R}^d} 2\cos(\langle\omega,x\rangle + b)\cos(\langle\omega,y\rangle + b)q(\omega)d\omega db \\ &= \int_{\mathbb{R}^d} \cos\langle\omega, x-y\rangle q(\omega)d\omega = \kappa(x-y) = k(x,y).\end{aligned}$$

Consider the vector $\Phi_\theta(x) = (\phi(x,\theta_k))_{k=1}^m = (\sqrt{2}\cos(\langle\omega_k,x\rangle + b_k))_{k=1}^m$ of random Fourier features, generated from the distributions p. Here $\omega_k \sim q$,

$b_k \sim U(0, 2\pi)$ are i.i.d. random variables. Denote by E_θ the expectation w.r.t. to the joint distribution of $\theta_1, \dots, \theta_m$.

The following simple result shows that any element of $\mathcal{H}$, defined by (1), can be approximated by a linear combination of random Fourier features.

Lemma 1. *For any $f = \int \gamma(\theta)\phi(\cdot, \theta)\, d\theta \in \mathcal{H}$ put*

$$\widehat{w} = \frac{1}{m}\left(\frac{\gamma(\theta_1)}{p(\theta_1)}, \dots, \frac{\gamma(\theta_m)}{p(\theta_m)}\right),$$

where $\theta_i \sim p$ are i.i.d. random variables. Then

$$\mathsf{E}_\theta(\langle \widehat{w}, \Phi_\theta(x)\rangle - f(x))^2 \le 2\frac{\|f\|^2_{\mathcal{H}}}{m}, \qquad \mathsf{E}_\theta\|\widehat{w}\|_2^2 = \frac{\|f\|^2_{\mathcal{H}}}{m}. \tag{5}$$

Proof. The random estimate $\langle \widehat{w}, \Phi_\theta(x)\rangle$ of $f(x)$ is unbiased:

$$\mathsf{E}_\theta\langle \widehat{w}, \Phi_\theta(x)\rangle = \frac{1}{m}\sum_{i=1}^m \mathsf{E}_{\theta_i}\left(\frac{\gamma(\theta_i)}{p(\theta_i)}\phi(x, \theta_i)\right) = f(x). \tag{6}$$

Compute the variance of this estimate:

$$\mathsf{E}_\theta\left(\frac{1}{m}\sum_{k=1}^m \frac{\gamma(\theta_k)}{p(\theta_k)}\phi(x;\theta_k) - f(x)\right)^2 = \frac{1}{m}\mathsf{E}_{\theta_1}\left(\frac{\gamma(\theta_1)}{p(\theta_1)}\phi(x;\theta_1) - f(x)\right)^2$$
$$\le \frac{1}{m}\mathsf{E}_{\theta_1}\left(\frac{\gamma(\theta_1)}{p(\theta_1)}\phi(x;\theta_1)\right)^2 = \frac{1}{m}\int \frac{\gamma^2(\theta_1)}{p(\theta_1)}\phi^2(x;\theta)\, d\theta_1$$
$$\le \frac{2}{m}\int \frac{\gamma^2(\theta_1)}{p(\theta_1)}\, d\theta_1 \le 2\frac{\|f\|^2_{\mathcal{H}}}{m}.$$

The proof of the equality in (5) is also elementary:

$$\mathsf{E}_\theta\|\widehat{w}\|_2^2 = \frac{1}{m^2}\sum_{i=1}^m \mathsf{E}_{\theta_i}\left(\frac{\gamma^2(\theta_i)}{p^2(\theta_i)}\right) = \frac{\|f\|^2_{\mathcal{H}}}{m}. \qquad \square$$

Let $(x_t, y_t) \in \mathbb{R}^d \times \mathbb{R}$ be an arbitrary sequence. Assuming that the dependence between features x_t and labels y_t can be described sufficiently well by a function $f \in \mathcal{H}$, consider the least squares problem

$$\sum_{t=1}^T (y_t - f(x_t))^2 \to \min_{f \in \mathcal{H}}. \tag{7}$$

Lemma 1 allows to pass to its parametric form:

$$\sum_{t=1}^T (y_t - \langle w, \Phi_\theta(x_t)\rangle)^2 \to \min_{w \in \mathbb{R}^m}.$$

More precisely, we will consider the online learning problem, where the goal is to find a sequence w_t with "small" cumulative expected loss:

$$\mathsf{E}_\theta\sum_{t=1}^T (y_t - \langle w_t, \Phi_\theta(x_t)\rangle)^2, \quad \text{where} \quad w_t = w_t(\Phi_\theta(x_1), \dots, \Phi_\theta(x_t), y_1, \dots, y_{t-1}),$$

compared to the loss (7) of any element $f \in \mathcal{H}$.

We allow the weight w_t to depend on the feature mapping $\Phi_\theta(x_t)$, indicating that features x_t are available at time t before predicting the label y_t. This natural assumption is important in the Vovk-Azoury-Warmuth (VAW) algorithm [2, Section 11.8], defined by

$$w_t = \operatorname*{argmin}_{w\in\mathbb{R}^d} \left\{ \frac{\lambda}{2}\|w\|_2^2 + \frac{1}{2}\sum_{i=1}^{t-1}(\langle \Phi_\theta(x_i), w\rangle - y_i)^2 + \frac{1}{2}\langle \Phi_\theta(x_t), w\rangle^2 \right\}.$$

Explicitly,

$$w_t = S_t^{-1}\sum_{i=1}^{t-1} y_i \Phi_\theta(x_i), \quad S_t = \lambda I_d + \sum_{i=1}^{t} \Phi_\theta(x_i)\Phi_\theta(x_i)^\top. \tag{8}$$

Moreover, $S_t^\top$ can be computed recursively by the Sherman-Morrison formula (which is also presented in [2]):

$$S_t^{-1} = S_{t-1}^{-1} - \frac{S_{t-1}^{-1}\Phi_\theta(x_t)(S_{t-1}^{-1}\Phi_\theta(x_t))^T}{1+\Phi_\theta(x_t)^T S_{t-1}^{-1}\Phi_\theta(x_t)}, \quad S_0^{-1} = \lambda^{-1} I_d. \tag{9}$$

In the sequel, we will assume that the labels y_t are uniformly bounded: $|y_t| \le Y$. The regret [2]

$$R_T(w) = \frac{1}{2}\sum_{t=1}^{T}(\langle x_t, w_t\rangle - y_t)^2 - \frac{1}{2}\sum_{t=1}^{T}(\langle x_t, w\rangle - y_t)^2$$

of the VAW algorithm satisfies the bound

$$R_T(w) \le \frac{\lambda}{2}\|w\|_2^2 + \frac{mY^2}{2}\ln\left(1+\frac{\rho^2 T}{\lambda m}\right), \tag{10}$$

if $\|\Phi_\theta(x_t)\|_2 \le \rho$: see [2, Theorem 11.8], [11, Theorem 7.34]. In our case $\rho = \sqrt{2m}$. Thus,

$$R_T(w) \le \frac{\lambda}{2}\|w\|_2^2 + \frac{mY^2}{2}\ln\left(1+\frac{2T}{\lambda}\right). \tag{11}$$

Note that for random features of the form (**??**) the bound ρ remains the same.

3 Main results

Consider N translation invariant kernels $k_j(x,y) = \kappa_j(x-y)$, $j = 1,\dots,N$. Let $\mathcal{H}_j$ be the correspondent RKHS's. Put $\mathcal{H} = \mathcal{H}_1 + \dots + \mathcal{H}_N := \{f_1 + \dots + f_N : f_j \in \mathcal{H}_j,\ j = 1 \dots, N\}$. It is known that $\mathcal{H}$ with the norm

$$\|f\|_{\mathcal{H}}^2 = \min\left\{\sum_{j=1}^N \|f_j\|_{\mathcal{H}_j}^2 : f = \sum_{j=1}^N f_j\right\}$$

is an RKHS with the kernel $k = k_1 + \dots + k_N$ [24, Proposition 12.27]. Denote by $B_R(\mathcal{H}) = \{f \in \mathcal{H} : \|f\|_{\mathcal{H}} \le R\}$ the R-ball in an RKHS $\mathcal{H}$.

Lemma 2. *For $f \in \mathcal{H} = \mathcal{H}_1 + \dots + \mathcal{H}_N$ take $f_j = \int_\Theta \gamma_j(\theta)\phi_j(x;\theta)d\theta \in \mathcal{H}_j$ such that*

$$f = \sum_{j=1}^N f_j, \qquad \|f\|^2_{\mathcal{H}} = \sum_{j=1}^N \|f_j\|^2_{\mathcal{H}_j}.$$

For each kernel k_j define

$$\widehat{w}_j = \frac{1}{m}\left(\frac{\gamma_j(\theta_{j1})}{p_j(\theta_{j1})}, \dots, \frac{\gamma_j(\theta_{jm})}{p_j(\theta_{jm})}\right),$$

as in Lemma 1. Here $\theta_{jk} \sim p_j$, $k = 1, \dots, m$ are i.i.d. random variables for each $j = 1, \dots, N$. Let $|y| \le Y$. Then

$$\mathsf{E}_\theta \left(\sum_{j=1}^N \langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle - y\right)^2 \le 2\frac{N}{m}\|f\|^2_{\mathcal{H}} + (f(x) - y)^2, \tag{12}$$

where $\Phi_{\theta_j}(x) = (\phi(x, \theta_{jk}))_{k=1}^m$.

Proof. Since the estimate $\langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle$ of $f_j(x)$ is unbiased: see (6), we have

$$\begin{aligned}
&\mathsf{E}_\theta \left(\sum_{j=1}^N \langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle - y\right)^2 - \left(\sum_{j=1}^N f_j(x) - y\right)^2 \\
&= \mathsf{E}_\theta \left(\sum_{j=1}^N \langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle\right)^2 - \left(\sum_{j=1}^N f_j(x)\right)^2 \\
&= \mathsf{E}_\theta \left(\sum_{j=1}^N \langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle - \sum_{j=1}^N f_j(x)\right)^2 \qquad (13)
\end{aligned}$$

Using the inequality $(\sum_{i=1}^N a_i)^2 \le N \sum_{i=1}^N a_i^2$, by Lemma 1 we get

$$\begin{aligned}
\mathsf{E}_\theta \left(\sum_{j=1}^N \langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle - f_j(x)\right)^2 &\le N \sum_{j=1}^N \mathsf{E}_\theta \left(\langle \widehat{w}_j, \Phi_{\theta_j}(x)\rangle - f_j(x)\right)^2 \\
&\le 2\frac{N}{m}\sum_{j=1}^N \|f_j\|^2_{\mathcal{H}_j} = 2\frac{N}{m}\|f\|^2_{\mathcal{H}}. \qquad (14)
\end{aligned}$$

The inequalities (13), (14) imply (12). □

Let us first apply the VAW algorithm to the sequence $(\Phi_\theta(x_t), y_t)$, where

$$\Phi_\theta(x) = (\Phi_{\theta_1}(x), \dots, \Phi_{\theta_N}(x)), \quad \Phi_{\theta_j}(x) = (\phi_j(x, \theta_{jk}))_{k=1}^m. \tag{15}$$

That is, we concatenate random feature vectors Φ_{θ_j}, related to each kernel k_j, into a Nm-dimensional vector.

Theorem 1. *Let $w_t = (w_{t,1}, \dots, w_{t,Nm}) \in \mathbb{R}^{Nm}$ be generated by the VAW algorithms applied to the sequence $(\Phi_\theta(x_t), y_t)$. Then*

$$\frac{1}{2}\mathsf{E}_\theta \sum_{t=1}^{T} (\langle w_t, \Phi_\theta(x_t)\rangle - y_t)^2 \leq \frac{1}{2}\sum_{t=1}^{T}(f(x_t) - y_t)^2 + \left(\frac{\lambda}{2} + NT\right)\frac{\|f\|^2_{\mathcal{H}}}{m} + \frac{NmY^2}{2}\ln\left(1 + \frac{2T}{\lambda}\right) \tag{16}$$

for any f in the RKHS $\mathcal{H}$, generated by the kernel $k = k_1 + \dots + k_N$. For $T \to +\infty$,

$$\frac{1}{2}\mathsf{E}_\theta \sum_{t=1}^{T} (\langle w_t, \Phi_\theta(x_t)\rangle - y_t)^2 \leq \frac{1}{2}\inf_{f \in B_R(\mathcal{H})}\sum_{t=1}^{T}(f(x_t) - y_t)^2 + O\left(N(R^2 + Y^2 \ln T)\sqrt{T}\right), \tag{17}$$

if $m \propto \sqrt{T}$.

Proof. Denote by $R_T^{\mathrm{VAW}}(w_1, \dots, w_N)$ the regret of the VAW algorithm w.r.t. the fixed vector $(w_1, \dots, w_N) \in (\mathbb{R}^m)^N$. For $f \in \mathcal{H}$ take f_j, $\widehat{w}_j$ as in Lemma 2. Then

$$\frac{1}{2}\sum_{t=1}^{T}(\langle w_t, \Phi_\theta(x_t)\rangle - y_t)^2 = R_T^{\mathrm{VAW}}(\widehat{w}_1, \dots, \widehat{w}_N) + \frac{1}{2}\sum_{t=1}^{T}\left(\sum_{j=1}^{N}\langle \widehat{w}_j, \Phi_{\theta_j}(x_t)\rangle - y_t\right)^2.$$

By (11) and Lemma 1,

$$\begin{aligned}\mathsf{E}_\theta R_T^{\mathrm{VAW}}(\widehat{w}_1, \dots, \widehat{w}_N) &\leq \frac{\lambda}{2}\sum_{j=1}^{N}\mathsf{E}_\theta\|\widehat{w}_j\|_2^2 + \frac{NmY^2}{2}\ln\left(1 + \frac{2T}{\lambda}\right)\\ &\leq \frac{\lambda}{2}\frac{\|f\|^2_{\mathcal{H}}}{m} + \frac{NmY^2}{2}\ln\left(1 + \frac{2T}{\lambda}\right).\end{aligned} \tag{18}$$

By Lemma 2,

$$\frac{1}{2}\sum_{t=1}^{T}\mathsf{E}_\theta\left(\sum_{j=1}^{N}\langle \widehat{w}_j, \Phi_{\theta_j}(x_t)\rangle - y_t\right)^2 \leq T\frac{N}{m}\|f\|^2_{\mathcal{H}} + \frac{1}{2}\sum_{t=1}^{T}(f(x_t) - y_t)^2,$$

Combining (18) with the last inequalities yields (16). The relation (17) follows immediately. $\square$

Assume that $m \geq d$. Then a simple analysis shows that the time and space complexities of the proposed algorithm are $O(N^2m^2)$ per iteration: see (8), (9). To reduce these complexities we consider the following two-level procedure:

- generate N m-dimensional vectors of random features, related to each kernel k_i, and apply VAW algorithm to each sequence $(\Phi_{\theta_j}(x_t), y_t)$,

- regarding the predictions of these algorithms as expert opinions, combine them by a meta-algorithm.

Our main suggestion is to use VAW also as a meta-algorithm. Assuming that $m \geq \max\{d, N\}$, the overall time and space complexities in this case are $O(Nm^2)$ per iteration. This estimate reflects the complexities of the expert algorithms, as the meta-algorithm's contribution is negligible. The loss estimates are given in Theorem 2. Note that the justification of these estimates do not require the boundedness of the expert outputs $\langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle$.

Theorem 2. *Let $w_{t,j} \in \mathbb{R}^m$ be generated by the VAW algorithms applied to $(\Phi_{\theta_j}(x_t), y_t)$, and $\alpha_t \in \mathbb{R}^N$ be generated by the VAW algorithm applied to (z_t, y_t), where z_t is the vector of expert predictions:*

$$z_t = (\langle w_{t,1}, \Phi_{\theta_1}(x_t)\rangle, \dots, \langle w_{t,N}, \Phi_{\theta_N}(x_t)\rangle).$$

Then

$$\begin{aligned}
&\frac{1}{2}\mathsf{E}_\theta \sum_{t=1}^T (\langle \alpha_t, z_t\rangle - y_t)^2 \leq \frac{1}{2}\sum_{t=1}^T (y_t - f_j(x_t))^2 + \frac{\lambda}{2} \\
&+ \left(\frac{\lambda}{2} + T\right)\frac{\|f_j\|^2_{\mathcal{H}_j}}{m} + \frac{mY^2}{2}\ln\left(1 + \frac{2T}{\lambda}\right) \\
&+ \frac{NY^2}{2}\ln\left(1 + \frac{Y^2}{\lambda}\left(2T(T+1) + 2mT\ln\left(1 + \frac{2T}{\lambda}\right)\right)\right),
\end{aligned} \tag{19}$$

for any $f_j \in \mathcal{H}_j$, $j = 1, \dots, N$. For $T \to +\infty$

$$\begin{aligned}
\frac{1}{2}\mathsf{E}_\theta \sum_{t=1}^T (\langle \alpha_t, z_t\rangle - y_t)^2 &\leq \frac{1}{2}\min_{1\leq j\leq N}\inf_{f_j \in B_R(\mathcal{H}_j)}\sum_{t=1}^T (y_t - f_j(x_t))^2 \\
&+ O\left((R^2 + Y^2\ln T)\sqrt{T}\right), \quad \text{if} \quad m \propto \sqrt{T}.
\end{aligned} \tag{20}$$

Proof. Take f_j, $\widehat{w}_j$ as in Lemma 2. Denote by $R_T^{\text{VAW}}(\delta)$ the regret of the VAW algorithm applied to the sequence (z_t, y_t), and by $R_T^{\text{VAW}}(\widehat{w}_j)$ the regret of the VAW algorithm applied to $(\Phi_{\theta_j}(x_t), y_t)$. For any $\delta_i \geq 0$, $\sum_{i=1}^N \delta_i = 1$ we have

$$\begin{aligned}
\frac{1}{2}\sum_{t=1}^T (\langle \alpha_t, z_t\rangle - y_t)^2 &= R_T^{\text{VAW}}(\delta) + \frac{1}{2}\sum_{t=1}^T (\langle \delta, z_t\rangle - y_t)^2 \\
&\leq R_T^{\text{VAW}}(\delta) + \frac{1}{2}\sum_{t=1}^T\sum_{j=1}^N \delta_j (z_{t,j} - y_t)^2 \\
&= R_T^{\text{VAW}}(\delta) + \sum_{j=1}^N \delta_j R_T^{\text{VAW}}(\widehat{w}_j) \\
&+ \frac{1}{2}\sum_{t=1}^T\sum_{j=1}^N \delta_j (\langle \widehat{w}_j, \Phi_{\theta_j}(x_t)\rangle - y_t)^2
\end{aligned} \tag{21}$$

Let us estimate the first term. From the general bound (10) for the regret of the VAW algorithm it follows that

$$R_T^{\mathrm{VAW}}(\delta) \le \frac{\lambda}{2}\|\delta\|_2^2 + \frac{NY^2}{2}\ln\left(1+\frac{Z_T^2 T}{\lambda N}\right), \tag{22}$$

if $\sum_{j=1}^N z_{t,j}^2 \le \sum_{j=1}^N \langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle^2 \le Z_T^2$. Due to the logarithmic scaling of Z_T, its rough estimate would be enough. We have

$$z_{t,j}^2 \le 2(\langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 + 2y_t^2.$$

By the bound (11),

$$\begin{aligned}
&\frac{1}{2}(\langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 \le \frac{1}{2}\sum_{t=1}^T (\langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 \\
&= \frac{1}{2}\sum_{t=1}^T (\langle w_j, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 + R_T^{\mathrm{VAW}}(w_j) \\
&\le \frac{1}{2}\sum_{t=1}^T (\langle w_j, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 + \frac{\lambda\|w_j\|^2}{2} + \frac{mY^2}{2}\ln\left(1+\frac{2T}{\lambda}\right)
\end{aligned}$$

for any $w_j \in \mathbb{R}^m$. Put $w_j = 0$ in the right-hand side of the last formula:

$$\frac{1}{2}(\langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 \le \frac{1}{2}TY^2 + \frac{mY^2}{2}\ln\left(1+\frac{2T}{\lambda}\right).$$

Thus,

$$\sum_{j=1}^N z_{t,j}^2 \le Z_T^2 := 2(T+1)NY^2 + 2mNY^2\ln\left(1+\frac{2T}{\lambda}\right). \tag{23}$$

From (22), (23) we get

$$\begin{aligned}
R_T^{\mathrm{VAW}}(\delta) &\le \frac{\lambda}{2}\|\delta\|_2^2 \\
&\quad + \frac{NY^2}{2}\ln\left(1+\frac{Y^2}{\lambda}\left(2T(T+1) + 2mT\ln\left(1+\frac{2T}{\lambda}\right)\right)\right).
\end{aligned} \tag{24}$$

The estimate of the expectation of the second term in (21) follows from (11) and Lemma 1:

$$\sum_{j=1}^n \delta_j \mathop{\mathsf{E}}_{\theta} R_T^{\mathrm{VAW}}(\widehat{w}_j) \le \frac{\lambda}{2m}\sum_{j=1}^n \delta_j\|f_j\|_{\mathcal{H}_j}^2 + \frac{mY^2}{2}\ln\left(1+\frac{2T}{\lambda}\right), \tag{25}$$

Finally, estimate the expectation of the last term in (21) by Lemma 2 (applied with $N=1$):

$$\frac{1}{2}\sum_{t=1}^{T}\sum_{j=1}^{N}\delta_j \mathsf{E}_\theta(\langle \widehat{w}_j, \Phi_{\theta_j}(x_t)\rangle - y_t)^2$$
$$\leq \sum_{j=1}^{N}\delta_j\left(\frac{T}{m}\|f_j\|^2_{\mathcal{H}_j} + \frac{1}{2}\sum_{t=1}^{T}(f_j(x_t)-y_t)^2\right). \tag{26}$$

To get (19) consider the vectors of the standard basis $\delta = e_j$ of $\mathbb{R}^N$, and combine (21), (24), (25) with (26). The relation (20) follows directly. $\square$

Now assume that the upper bound Y for y_t is known. Then the last term in (19) can be improved by changing expert predictions from z_t to

$$\overline{z}_t = \min(Y, \max(z_t, -Y)), \quad z_{t,j} = \langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle. \tag{27}$$

where the max and min operations are applied component-wise. Let $\overline{R}_T^{\text{VAW}}(\delta)$ be the regret of the VAW algorithm applied to the sequence $(\overline{z}_t, y_t)$. Then

$$\sum_{t=1}^{T}(\langle \alpha_t, \overline{z}_t\rangle - y_t)^2 = \overline{R}_T^{\text{VAW}}(\delta) + \sum_{t=1}^{T}(\langle \delta, \overline{z}_t\rangle - y_t)^2$$
$$\leq \overline{R}_T^{\text{VAW}}(\delta) + \sum_{t=1}^{T}\sum_{j=1}^{N}\delta_j(z_{t,j}-y_t)^2$$

for any $\delta_i \geq 0$, $\sum_{i=1}^N \delta_i = 1$, since $(\overline{z}_{t,j}-y_t)^2 \leq (z_{t,j}-y_t)^2$. In (22) we can put $Z_T = Y$:

$$\overline{R}_T^{\text{VAW}}(\delta) \leq \frac{\lambda}{2} + \frac{NY^2}{2}\ln\left(1+\frac{Y^2T}{\lambda}\right),$$

and use this bound instead of (24).

Under the same assumption, the bounds (19) can be further improved by using another algorithms for combining expert opinions, instead of VAW. Recall that a loss function $\ell : [-Y,Y]^2 \to \mathbb{R}$ is called η-exponentially concave if the function $F(z) = e^{-\eta\ell(y,z)}$ is concave for all $y \in [-Y,Y]$. In particular, the loss the function $\ell(y,z) = (y-z)^2$ is η-exp-concave for $\eta \leq 1/(8Y^2)$ (see [2, Section 3.3]). Applying exponentially weighted average (EWA) forecaster: $\alpha_{1,j} = 1/N$,

$$\alpha_{t,j} = \frac{\alpha_{t-1,j}\exp(-\eta(\overline{z}_{t,j}-y_t)^2)}{\sum_{k=1}^N \alpha_{t-1,k}\exp(-\eta(\overline{z}_{t,k}-y_t)^2)}, \quad t = 2,\dots,T \tag{28}$$

with $\eta = 1/(8Y^2)$, we get the estimate

$$\overline{R}_{T,j}^{\text{EWA}} := \frac{1}{2}\sum_{t=1}^{T}(\langle \alpha_t, \overline{z}_t\rangle - y_t)^2 - \frac{1}{2}\sum_{t=1}^{T}(\overline{z}_{t,j}-y_t)^2 \leq 4Y^2\ln N, \tag{29}$$

see [2, Proposition 3.1]. The related improved bounds are given in the next theorem.

Theorem 3. *Assume that the constant Y is known. Let $w_{t,j} \in \mathbb{R}^m$ be generated by the VAW algorithms applied to the sequence $(\Phi_{\theta_j}(x_t), y_t)$, and $\alpha_t \in \mathbb{R}^N$ be generated by the EWA forecaster applied to the sequence $(\overline{z}_t, y_t)$, where $\overline{z}_t$ is the vector of truncated expert predictions (27). Then*

$$\frac{1}{2}\mathsf{E}_\theta \sum_{t=1}^T (\langle \alpha_t, \overline{z}_t\rangle - y_t)^2 \le \frac{1}{2}\sum_{t=1}^T (f_j(x_t) - y_t)^2 + 4Y^2 \ln N + \left(\frac{\lambda}{2} + T\right)\frac{\|f_j\|^2_{\mathcal{H}_j}}{m} + \frac{mY^2}{2}\ln\left(1 + \frac{2T}{\lambda}\right) \tag{30}$$

for any $f_j \in \mathcal{F}_j$, $j = 1, \dots, N$. The estimate (20) of Theorem 2 remains true.

Proof. Using (29), we get

$$\begin{aligned}\frac{1}{2}\sum_{t=1}^T (\langle \alpha_t, \overline{z}_t\rangle - y_t)^2 &= \overline{R}^{\mathrm{EWA}}_{T,j} + \frac{1}{2}\sum_{t=1}^T (\overline{z}_{t,j} - y_t)^2 \\ &\le 4Y^2 \ln N + \frac{1}{2}\sum_{t=1}^T (\langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle - y_t)^2 \\ &\le 4Y^2 \ln N + R^{\mathrm{VAW}}_T(\widehat{w}_j) + \frac{1}{2}\sum_{t=1}^T (\langle \widehat{w}_j, \Phi_{\theta_j}(x_t)\rangle - y_t)^2.\end{aligned}$$

The assertion follows from this inequality combined with the estimates (25), (26) applied to $\delta = e_j$. □

Let us call the algorithms, analyzed in Theorems 2 and 3 by VAW^2 (double VAW) and VAW-EWA respectively. Under the mentioned assumption $m \ge \max\{d, N\}$ their time and space complexities are the same: $O(Nm^2)$, and are determined by the complexities of the expert VAW algorithms.

Although the estimate (30) is slightly better than (19), the estimate (20) for large T in Theorem 3 is not improved. It is not clear if the improvement, obtained by applying the EWA forecaster to the truncated expert opinions $\overline{z}_t$ instead of applying VAW algorithm to original expert opinions z_t, is essential. The numerical experiments, presented in Section 4, show that the linear combinations of expert predictions, used in VAW^2, can produce better results than the convex combinations of the VAW-EWA or similar meta-algorithms.

Using a similar notation, the basic algorithms used in [14, 16] can be called OGD-OGD and OGD-EWA, since they use the online gradient descent (OGD) for expert strategies, and either OGD or EWA-type meta-algorithms. Their time and space per iteration complexities are lower: $O(Ndm)$. However the related regret bounds are not applicable, since the quadratic loss function does not satisfy the global Lipschitz condition, and coefficients $\widehat{w}$ in Lemma 1 are not bounded.

4 Computer experiments

In the organization of computer experiments we followed [5] and the related code[1]. Code to reproduce our results is available at[2], along with instructions for running the experiments. All algorithms were run using $N = 76$ kernels: 51 Gaussian and 25 Laplacian: see (3), (4). Their parameters were set as follows:

$$\sigma^2 \in \{10^{2i/25-2}\}_{i=0}^{50} \quad \text{for Gaussian kernels,}$$
$$\sigma \in \{10^{i/6-2}\}_{i=0}^{24} \quad \text{for Laplacian kernels.}$$

Following [5] we used random features of the form $(\cos\langle\theta_i, x\rangle, \sin\langle\theta_i, x\rangle)$, $\theta_i \sim q$, $i = 1, \dots, m$. This is a well-known slight variation of the approach described above (see [19] for a discussion). It is related to the kernel representation

$$\int_\Theta p(\theta)\langle\phi(x;\theta)\phi(y;\theta)\rangle\, d\theta = \int_{\mathbb{R}^d} \cos\langle\theta, x-y\rangle q(\theta) d\theta = \kappa(x-y) = k(x,y).$$

The number m of random features was set to 50. The mean squared losses (MSE) $\frac{1}{T}\sum_{t=1}^{T}(\widehat{f}_t(x_t) - y_t)^2$ of various algorithms $\widehat{f}_t$ were averaged over 5 experiments (see). We chose $\lambda = 1$ for the VAW algorithm in all cases.

Furthermore, we used the same datasets as in [5]. They are briefly described in Table 1 and are available from the UCI Machine Learning Repository[3]. In addition we generated an artificial data by AR(4) model:

$$x_t = \nu_0 x_{t-4} + \nu_1 x_{t-3} + \nu_2 x_{t-2} + \nu_3 x_{t-1} + \epsilon_t, \quad y_t = x_{t+1}, \tag{31}$$

$t = 1, \dots, 5000$, where $\nu_0 = 0.5$, $\nu_1 = -0.3, \nu_2 = 0.2, \nu_3 = 0.1$, $\epsilon_t \sim \mathcal{N}(0,1)$, $x_k = 0$, $k = -3, \dots, 0$.

As in the mentioned code of [5], for all datasets the features and labels were normalized as follows:

$$y_i := \frac{y_i - \underline{y}}{\overline{y} - \underline{y}}, \quad \underline{y} = \min_{j=1,\dots,n} y_j, \quad \overline{y} = \max_{j=1,\dots,n} y_j, \tag{32}$$
$$x_i := x_i / \max_{j=1,\dots,n} \|x_j\|_2. \tag{33}$$

We compared the VAW^2 algorithm, analyzed in Theorem 2, and the VAW-EWA algorithm, analyzed in Theorem 3, with several other algorithms:

- Raker [16]: this is the algorithm of OGD-EWA type in our notation. It combines OGD the predictions of expert strategies by the EWA-type meta-algorithm.
- OMKL-GF [5]: a data-driven kernel selection scheme where a bipartite feedback graph is constructed at every time instant.
- VAW-Aggr: the predictions of VAW expert strategies are combined by the Vovk VAW-Aggregating algorithm [2, Section 3.5]. The quadratic

[1] https://github.com/pouyamghari/Graph-Aided-Online-Multi-Kernel-Learning
[2] https://github.com/O-Gurt/VAW2
[3] https://archive.ics.uci.edu/

Name	Size	Data description	Label
Airfoil	(1503, 5)	airfoils at various wind tunnel speeds and angles of attack	scaled sound pressure
Bias	(7750, 21)	temperature measurements and predictions together with auxiliary geographic variables	next-day minimum air temperature
Concrete	(1030, 8)	concrete specifications such as the amount of cement or water	compressive strength
Naval	(11934, 15)	features of a naval vessel, characterized by a gas turbine propulsion plant	lever position

Таблица 1. Summary of real-world datasets used for evaluation.

loss is η-mixable with $\eta = 2$ [2, Section 3.6]. Thus, using the VAW-Aggregating meta-algorithm with $\eta = 2$, it is possible to achieve the regret estimate slightly better then for the EWA meta-algorithm [2, Proposition 3.2].

- VAW-ML-Prod, VAW-ML-Poly, VAW-BOA: the predictions of VAW expert strategies are combined by second-order online algorithms, which use both the cumulative loss (first-order statistic) and the variance of losses (second-order statistic) to adapt their learning rates dynamically [3, 25]. These algorithms are implemented within the Opera library[4], which we employed.

We do not consider the VAW algorithm from Theorem 1 due to its high computational and space complexities.

The results of experiments are collected in Table 2. We do not describe here the parameters of Raker and OMKL-GF algorithms. The results of [5] were reproduced by running their publicly available code with the parameters they specified. While [5] averaged results over 20 experiments, we used 5. So, the results presented here are slightly different. Note that theoretically all these algorithms, except VAW^2, require knowledge of the interval containing the labels, and should be used with the truncated expert predictions. Since here we consider $y_t \in [0, 1]$, instead of $y_t \in [-Y, Y]$, the truncation was performed accordingly:

$$\overline{z}_t = \min(1, \max(z_t, 0)), \quad z_{t,j} = \langle w_{t,j}, \Phi_{\theta_j}(x_t)\rangle.$$

For the VAW^2 algorithm we present the results both for original and truncated expert predictions: VAW^2(trunc). However, these options give almost the same results. The lowest MSE values are shown in bold. VAW^2 algorithm shows the best result across all datasets.

[4] https://github.com/Dralliag/opera-python

	AR(4)	Airfoil	Bias	Concrete	Naval
Raker	23.24	28.64	12.70	35.29	11.32
OMKL-GF	20.47	24.37	7.05	34.24	4.60
VAW^2	16.56	22.80	**4.09**	**10.96**	**0.29**
VAW^2(trunc)	16.51	**22.78**	**4.09**	10.97	**0.29**
VAW-Aggr	16.40	26.74	5.02	13.57	0.45
VAW-EWA	16.49	27.61	5.41	15.08	0.62
VAW-BOA	**16.34**	26.42	4.98	13.88	0.52
VAW-ML-Poly	**16.34**	26.10	4.96	13.33	0.37
VAW-ML-Prod	**16.34**	26.27	4.97	13.64	0.48

Таблица 2. MSE (scaled up by 10^3) of MKL algorithms with 76 kernels.

Figure 1 illustrates the MSE of these algorithms over the iterations. We excluded VAW^2(trunc), VAW-BOA, AW-ML-Poly to improve the clarity. VAW^2 consistently achieves the lowest MSE trajectory across considered real world datasets, indicating strong performance throughout learning.

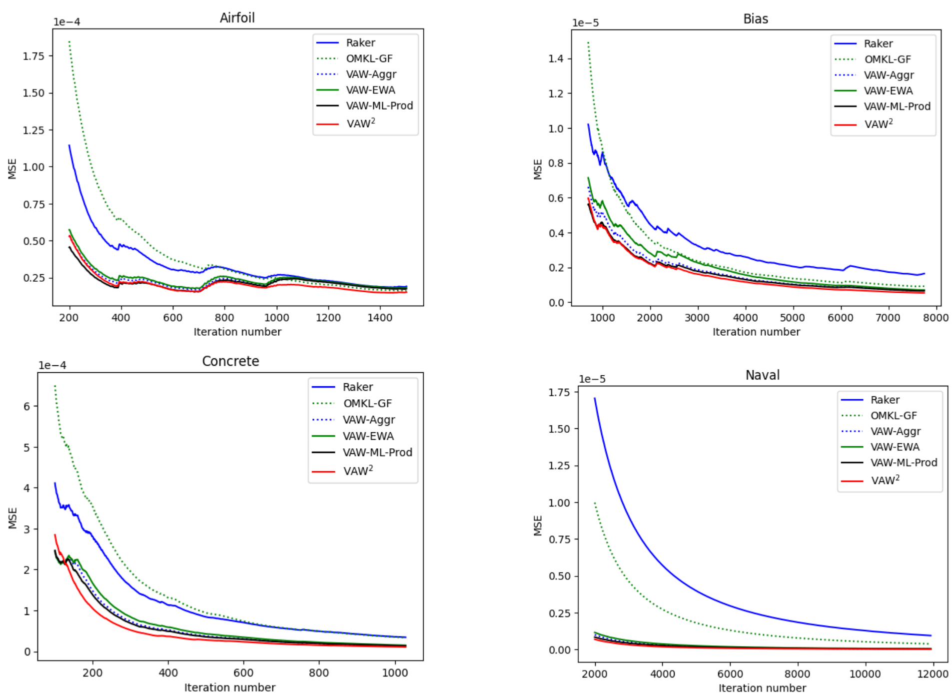

Fig. 1. MSE performance of MKL algorithms.

To further understand the behavior of the suggested algorithms, in Figure 2 we plot the terminal expert weight vectors α_T, assigned by VAW^2, VAW-EWA and VAW-ML-Prod algorithms. We see that ML-prod exhibits sparsity, concentrating its weighting on a small number of kernels. EWA distributes weight more broadly, while VAW distinguishes itself by the essential use of negative weights.

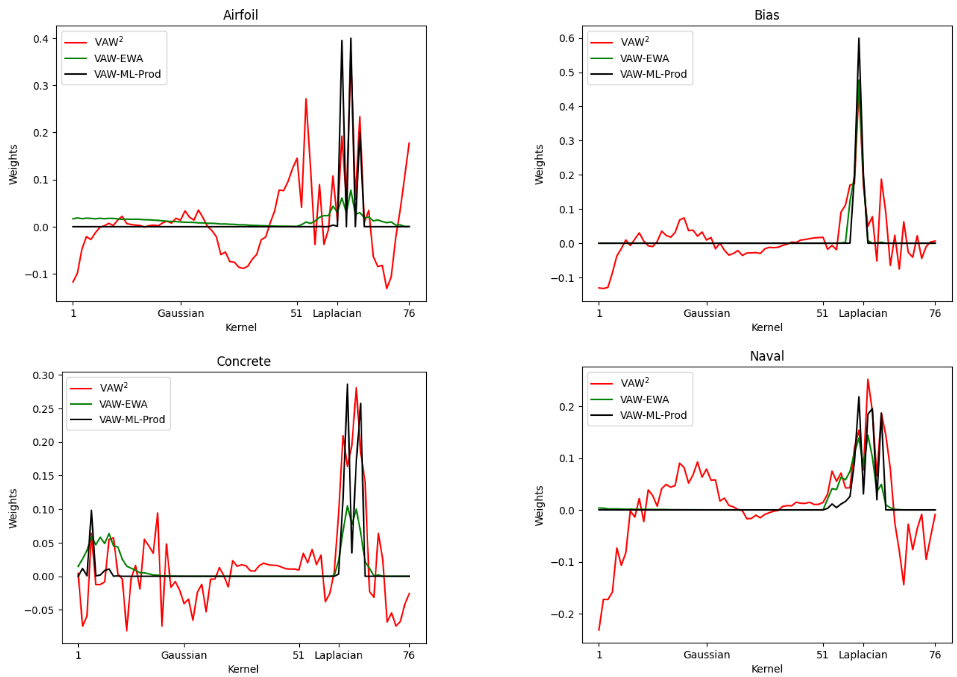

FIG. 2. Terminal weights of VAW^2, VAW-EWA and VAW-ML-Prod algorithms.

5 Conclusion

We introduced VAW^2, a novel online multi-kernel learning algorithm for least squares regression in RKHS. By leveraging VAW at both the expert level (for kernel-specific predictions) and the meta level (for dynamic kernel combination), VAW^2 achieves a balance between computational efficiency and theoretical guarantees. A key feature of VAW^2 is its computational efficiency compared to direct application of the VAW algorithm to concatenated feature vectors, making it scalable for practical applications. We derived a regret bound of $O(T^{1/2} \ln T)$ in expectation with respect to artificial randomness, when the number of random features scales as $T^{1/2}$. The framework accommodates both VAW and EWA meta-algorithms, with truncation strategies further enhancing robustness when label bounds are known. Computational experiments showed encouraging results on some benchmark datasets.

Future work could extend this analysis to derive dynamic regret bounds for non-stationary environments, incorporate mechanisms for online kernel dictionary adaptation, and refine loss bounds under specific data assumptions.

It is interesting to perform more extensive benchmarking of the VAW^2 algorithm across diverse datasets and application domains.

References

[1] K.S. Azoury, M.K. Warmuth, *Relative loss bounds for on-line density estimation with the exponential family of distribution*, Machine Learning, **43**:3, (2001), 211–246.

[2] N. Cesa-Bianchi, G. Lugosi, *Prediction, learning, and games*, Cambridge University Press, Cambrodge, 2006.

[3] P. Gaillard, G. Stoltz, T. Van Erven, *A second-order bound with excess losses*, In: Proceedings of The 27th Conference on Learning Theory. PMLR. 2014, 176–196.

[4] P. Gaillard, S. Gerchinovitz, M. Huard, G. Stoltz, *Uniform regret bounds over $\mathbb{R}^d$ for the sequential linear regression problem with the square loss*, In: Proceedings of the 30th International Conference on Algorithmic Learning Theory. PMLR. 2019, 404ñïS432.

[5] P.M. Ghari, Y. Shen, *Graph-aided online multi-kernel learning*, Journal of Machine Learning Research, **24**:21, (2023), 1–44.

[6] M. Gönen, E. Alpaydın, *Multiple kernel learning algorithms*, Journal of Machine Learning Research, **12**:64, (2011), 2211–2268.

[7] T. Hofmann, B. Schölkopf, A.J. Smola, *Kernel methods in machine learning*, The Annals of Statistics, **36**:3, (2008), 1171 – 1220.

[8] S.C.H. Hoi, D. Sahoo, J. Lu, P. Zhao, *Online learning: A comprehensive survey*, Neurocomputing, **459**:12, (2021), 249–289.

[9] J. Kivinen, A.J. Smola, R.C. Williamson, *Online learning with kernels*, IEEE transactions on signal processing, **52**:8, (2004), 2165–2176.

[10] J. Lu, S.C.H. Hoi, J. Wang, P. Zhao, Z.-Y. Liu, *Large scale online kernel learning*, Journal of Machine Learning Research, **17**:47, (2016), 1–43.

[11] F. Orabona, *A modern introduction to online learning*, arXiv:1912.13213v6 [cs.LG], 2023.

[12] A. Rahimi, B. Recht, *Random features for large-scale kernel machines*, In: Advances in Neural Information Processing Systems 20 (NIPS 2007). **20**, 2007, 1177–1184.

[13] A. Rahimi, B. Recht, *Uniform approximation of functions with random bases*, In: 2008 46th Annual Allerton Conference on Communication, Control, and Computing. 2008, 555-561.

[14] D. Sahoo, S.C.H. Hoi, B. Li, *Large scale online multiple kernel regression with application to time-series prediction*, In: ACM Transactions on Knowledge Discovery from Data (TKDD) **13**:1 (2019), pp. 1–33.

[15] B. Schölkopf, A.J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT Press, 2002.

[16] Y. Shen, T. Chen, G.B. Giannakis, *Random feature-based online multi-kernel learning in environments with unknown dynamics*, Journal of Machine Learning Research, **20**:22, (2019), 1–36.

[17] K. Slavakis, P. Bouboulis, S. Theodoridis, *Online learning in reproducing kernel Hilbert spaces*, In: Academic Press Library in Signal Processing, Elsevier **1** 2014, 883ñïS987.

[18] B.K. Sriperumbudur, K. Fukumizu, G.R.G. Lanckriet, *Universality, characteristic kernels and RKHS embedding of measures*, Journal of Machine Learning Research, **12**:70, (2011), 2389–2410.

[19] D.J. Sutherland, J. Schneider, *On the error of random Fourier features*, arXiv:1506.02785 [cs.LG], 2015.

[20] S. Van Vaerenbergh, I. Santamaría, *Online regression with kernels*, In: Regularization, Optimization, Kernels, and Support Vector Machines. Machine Learning & Pattern Recognition Series. Chapman and Hall/CRC, New York, 2014. Chap. 21, 477–501.

[21] V. Vovk, *Competitive on-line statistics*, International Statistical Review, **69**:2, (2001), 213–248.
[22] V. Vovk, *On-line regression competitive with reproducing kernel Hilbert spaces*, In: International Conference on Theory and Applications of Models of Computation. Springer. 2006, 452ñŚ463.
[23] Z. Wang, K. Crammer, S. Vucetic, *Breaking the curse of kernelization: Budgeted stochastic gradient descent for large-scale svm training*, Journal of Machine Learning Research, **13**:1, (2012), 3103–3131.
[24] M.J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*, Cambridge University Press, Cambridge, 2019.
[25] O. Wintenberger, *Optimal learning with Bernstein online aggregation*, Machine Learning, **106**, (2017), 119–141.

Dmitry Borisovich Rokhlin
Institute of Mathematics, Mechanics and Computer Sciences and Regional Scientific and Educational Mathematical Center, Southern Federal University
Email address: dbrohlin@sfedu.ru

Olga Vladimirovna Gurtovaya
Institute of Mathematics, Mechanics and Computer Sciences, Southern Federal University
Email address: imedashvili@sfedu.ru