LOCALLY ADAPTIVE KERNEL ESTIMATION USING SPARSE FUNCTIONAL PROGRAMMING

Maria Peifer, Luiz F. O. Chamon, Santiago Paternain, and Alejandro Ribeiro

Electrical and Systems Engineering, University of Pennsylvania
e-mail: {mariaop,luizf,spater,aribeiro}@seas.upenn.edu

ABSTRACT

Reproducing Kernel Hilbert Space (RKHS)-based methods are widely used in signal processing and machine learning applications. Yet, they suffer from a parameter selection issue: selecting the RKHS in which to operate (or even the kernel parameter) is often a significant challenge. Moreover, since the RKHS determines properties such as shape and smoothness of the learned function, its choice affects the effectiveness of these techniques. Likewise, due to the homogeneous smoothness of functions promoted by these solutions, they are poor estimators for functions with varying degrees of smoothness. To overcome these limitations, we propose to locally adapt the RKHS (more specifically, its smoothness parameter) over which we seek to perform function estimation by using a sparse functional program. Under this formulation, we must solve an infinite dimensional, non-convex optimization problem. This problem, however, has zero duality gap and can therefore be solved exactly and efficiently in the dual domain using, for instance, gradient ascent techniques.

***Index Terms*—** Kernel regression, bandwidth adaptation, sparsity, sparse functional optimization.

1. INTRODUCTION

Reproducing kernel Hilbert spaces (RKHSs) have been at the core of successful non-parametric techniques used in signal processing, statistics, and machine learning [1–6]. Their attractiveness is due both to the richness of these functional spaces, which offers advantages over traditional parametric methods, and the straightforward nature of learning algorithms, mainly due to variational results known as the "representer theorem" [2, 4, 7]. Indeed, functions in RKHSs can be written as a (possibly infinite) linear combination of reproducing kernels evaluated over the function domain [4]. When learning under a smoothness prior, this linear combination is finite and depends only on the sampling points [8, 9]. In other words, the original non-parametric functional learning problem can be formulated as a finite dimensional optimization program.

Despite their success, RKHS-based methods suffer from an inherent issue: it is usually unclear over which RKHS to operate (i.e., the kernel and possibly its parameters). Although practice sometimes dictates the functional space, as is the case for bandlimited functions, it is more often than not unknown. And since the RKHS determines shape and smoothness properties of the estimated function, its choice is application-specific and ultimately affects the learning performance [7, 10–14]. In its simplest form, this issue arises as the problem of choosing a kernel parameter, e.g., the bandwidth of a Gaussian kernel (or radial basis function, RBF). Typically, this is addressed by grid search methods and cross-validation [12, 15] or using some application-specific heuristic (e.g., by maximizing the margin in support-vectors machines, SVMs [13, 14]). A more general approach attempts to learn the kernel either as a conic combination of a set of kernels [11, 16, 17] or using spectral representations of positive-definite functions [18–20]. These methods, however, quickly become impractical as they often must search over fine grids, use a large number of proposal kernels, and may require additional data.

Another limitation of these techniques (and RKHS-based methods in general) is that they learn functions that are homogeneously smooth and are therefore poor estimator of functions with heterogeneous degrees of smoothness, i.e., functions that are smooth in some regions and vary rapidly in others. In fact, RKHS methods cannot achieve the minimax error rate over function classes that allow these varying smoothness levels [21]. A common alternative in these scenarios, is to use different RKHSs for different regions of the domain, using plug-in rules [22], binary optimization [23], hypothesis testing [24], or gradient descent and alternating minimization [25–27]. Nevertheless, no optimality guarantees are provided for these methods due to the non-convexity of these locally adapted smoothness formulations.

In this work, we overcome the aforementioned limitations by learning a function that lies in the sum space of an uncountable number of RKHSs. In other words, for each point of the dataset we simultaneously determine its kernel, taken from a parameterized uncountable family, and its coefficient [as in (PIII)]. To solve this non-convex optimization problem, we write each RKHS function as a linear combination of kernels from a continuous dictionary [see (5)], where the weights of the linear combination are given by functions in L_2. We then minimize the support of these functional coefficients, obtaining sparse solutions that locally "select" the appropriate kernels. The resulting problem [see (PIV)] yields an optimization program that is *still* non-convex and *now* infinite dimensional. However, it allows us to leverage recent results on the strong duality of sparse functional programs [28]. In particular, we show that the optimization problem of interest can be solved exactly in the dual domain (Theorem 1). Hence, allowing to solve it efficiently using classical gradient ascent methods. Numerical examples are used to illustrate the effectiveness of this method at locally identifying the correct kernel parameters in different applications (Section 4).

2. PROBLEM FORMULATION

Classical RKHS function estimation methods seek to find a function in a predetermined RKHS that best fits the data. Explicitly, given a set of sample points $\{(\boldsymbol{x}_i, y_i)\}_{i=1,\dots,n}$, where $\boldsymbol{x}_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, and an RKHS $\mathcal{H}$, we wish to find a function $f \in \mathcal{H}$ that minimizes the estimation mean-square error (MSE) while promoting some smooth-

ness of f. Formally, we wish to solve the optimization problem

$$\underset{f\in\mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2 + \frac{\rho}{2}\|f\|_{\mathcal{H}}^2, \tag{PI}$$

where the $\|f\|_{\mathcal{H}}^2$ is the RKHS norm and $\rho > 0$ is a regularization parameter that controls the function smoothness to avoid overfitting. Despite its functional form, this problem is equivalent to a finite dimensional program. Indeed, there exist a solution of (PI) that can be represented using only kernels centered at the sample points [8, 9], i.e.,

$$f(\cdot) = \sum_{j=1}^{n} a_j k(\boldsymbol{x}_j, \cdot), \tag{1}$$

where $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is the reproducing kernel of $\mathcal{H}$ and $a_j \in \mathbb{R}$ are linear coefficients that determine f. An equivalent problem to (PI) can be formulated by collecting these coefficients into the $n\times 1$ vector $\boldsymbol{a} = [a_j]$, so that the RKHS norm can be written as

$$\|f\|_{\mathcal{H}}^2 = \boldsymbol{a}^\top \boldsymbol{K}\boldsymbol{a}, \tag{2}$$

where $\boldsymbol{K}$ is an $n \times n$ matrix whose (i, j)-th entry is $[\boldsymbol{K}]_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. For clarity, we also use an alternative form in which the MSE is constrained by a constant $\epsilon > 0$:

$$\begin{aligned} \underset{\boldsymbol{a}\in\mathbb{R}^n}{\text{minimize}} \quad & \boldsymbol{a}^\top \boldsymbol{K}\boldsymbol{a} \\ \text{subject to} \quad & \sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2 \le \epsilon \\ & f(\boldsymbol{x}_i) = \sum_{j=1}^{n} a_j k(\boldsymbol{x}_j, \boldsymbol{x}_i). \end{aligned} \tag{PII}$$

Since ϵ is arbitrary, (PII) is equivalent to (PI) in the sense that the latter can be used to solved the former [29].

A notable drawback of the classical formulation in (PI)–(PII), is that the RKHS $\mathcal{H}$ (or equivalently, its reproducing kernel k) must be known *a priori*. Moreover, since all centers use the same kernel [see (1)], these methods cannot efficiently represent functions with various degrees of smoothness. Take, for instance, the case of the RBF kernel with different bandwidths. These kernels are nested [30], so that the space associated with thinner kernels (smaller bandwidth) contains more functions than the space with thicker kernels (larger bandwidth). In particular, fast-varying functions can only be represented in the former space. However, accurately estimating functions in this richer RKHS requires considerably more samples. Hence, for a function with heterogeneous degrees of smoothness, it adds unnecessary complexity to the representation of the parts of the signal that vary slowly.

To overcome these limitations of classical kernel methods, we propose to locally adapt the RKHS at each center. Due to space constraints, we focus here on the issue of adapting the kernel parameter (e.g., the bandwidth of the RBF), but note that the multiple kernels extension is straightforward. Explicitly, we consider representations of the functions of the form

$$f(\cdot) = \sum_{j=1}^{n} a_j k_{\omega_j}(\boldsymbol{x}_j, \cdot), \tag{3}$$

where the ω_j are parameters that determine the RKHS at each center, such as the bandwidth of an RBF or the exponent of a polynomial kernel. Observe that the resulting f in (3) lies in the sum space of the RKHS $\mathcal{H}_j$ defined by the kernels k_{ω_j}, i.e., $f \in \mathcal{H} = \uplus\mathcal{H}_j$. This space is an RKHS with reproducing kernel $k(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j=1}^{n} k_{\omega_j}(\boldsymbol{x}, \boldsymbol{y})$ and norm

$$\|g\|_{\mathcal{H}}^2 = \min\left\{\sum_{j=1}^{n}\|g_j\|_{\mathcal{H}_j}^2 : g = \sum_{j=1}^{n} g_j,\, g_j \in \mathcal{H}_j\right\} \tag{4}$$

for all $g \in \mathcal{H}$ [31, Part I, Section 6].

Evaluating the infinite dimensional minimization in (4) is intricate except in particular cases. For instance, when the intersection of the RKHSs is trivial, i.e., $\bigcap_j \mathcal{H}_j = \{0\}$, the representation of g is unique and (4) reduces to $\|g\|_{\mathcal{H}} = \sum_j \|g_j\|_{\mathcal{H}_j}$. To avoid these difficulties associated with computing the norm in a generic sum space, we substitute the functional norm by $\|\boldsymbol{a}\|_2^2$, where $\|\cdot\|_2$ is the Euclidean norm. Notice from (2) that this akin to assuming $k(\boldsymbol{x}, \boldsymbol{y}) = \delta_K(\|\boldsymbol{x} - \boldsymbol{y}\|_2)$, where δ_K is the Kronecker's delta. Hence, though this regularizer favors smooth solutions across all RKHSs $\mathcal{H}_j$, it is not directly related to the norm in those Hilbert spaces. With these modifications, we can pose the problem of interest as

$$\begin{aligned} \underset{\boldsymbol{a}\in\mathbb{R}^n,\, \omega_j\ge 0}{\text{minimize}} \quad & \|\boldsymbol{a}\|_2^2 \\ \text{subject to} \quad & \sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2 \le \epsilon \\ & f(\boldsymbol{x}_i) = \sum_{j=1}^{n} a_j k_{\omega_j}(\boldsymbol{x}_j, \boldsymbol{x}_i) \end{aligned} \tag{PIII}$$

Solving (PIII) provides a direct way to estimate the function f while adapting the RKHS in which it lies, thus overcoming the aforementioned limitations of traditional kernel methods. Nevertheless, the nonlinearity of the equality constraint in (a_j, ω_j) makes (PIII) a non-convex optimization problem, thus posing a serious challenge to its solution. In the following section, we present an alternative formulation of (PIII) that, although non-convex, has zero duality gap (as shown in Theorem 1) and hence can be solved exactly in the dual domain. To do so, we introduce a functional version of the representation (3).

3. A SPARSE FUNCTIONAL SOLUTION

In the previous section, we argued that solved (PIII) addresses common issues with classical RKHS-based methods, such as choosing kernel parameters and estimating functions with heterogeneous degrees of smoothness. This optimization problem, however, is non-convex due to the nonlinear equality constraint. It is therefore not straightforward to find solutions of (PIII). In the sequel, we overcome this issue by deriving an alternative formulation that is both non-convex and infinite dimensional. Although this would seem to make the optimization problem harder, we show in Theorem 1 that the resulting problem has zero duality gap and cam hence be solved efficiently in the dual domain using gradient ascent or any other convex optimization method.

The first step in this reformulation consists of deriving a continuous representation of f as opposed to the discrete one in (3). Explicitly, we construct for each data point $\boldsymbol{x}_j$ an overcomplete, continuous dictionary containing all reproducing kernels $k_\omega(\boldsymbol{x}_j, \cdot)$ for $\omega \in [\omega_{\text{min}}, \omega_{\text{max}}]$. Then, instead of a real-valued coefficient a_j, each center is associated to a functional coefficient $\alpha_j : \Omega \to \mathbb{R}$. Thus, our solutions f are written as

$$f(\cdot) = \sum_{j=1}^{n}\int_{\omega_{\text{min}}}^{\omega_{\text{max}}} \alpha_j(\omega) k_\omega(\boldsymbol{x}_j, \cdot) d\omega. \tag{5}$$

Observe that by allowing the α_j to be distributions, the representation in (5) encompasses all functions functions from (3): suffices to take $\bar{\alpha}_j(\omega) = a_j\delta(\omega - \omega_j)$, where δ is the Dirac delta. Hence, using the representation in (5) would not reduce the feasible set of (PIII). In fact, this continuous representation spans a much larger set of functions.

The second step of this reformulation is therefore to pose an optimization problem whose solutions are (at least approximately) Dirac deltas as in $\bar{\alpha}$. In doing so, we could use this problem to obtain solutions of (PIII). To this end, notice that $\bar{\alpha}$ has the smallest possible support: indeed, the measure of the support of a Dirac delta distribution is zero [**?**]. Thus, we can "select" kernel parameters by promoting sparser α_j in (5). Hence, we reformulate (PIII) as the functional program

$$\begin{aligned} \underset{\alpha_j \in L_2}{\text{minimize}} \quad & \sum_{j=1}^{n} \left[\frac{1}{2} \left\| \alpha_j \right\|_{L_2}^2 + \gamma \left\| \alpha_j \right\|_{L_0} \right] \\ \text{subject to} \quad & \sum_{i=1}^{n} \left[y_i - \hat{y}_i \right]^2 \leq \epsilon \\ & \hat{y}_i = \sum_{j=1}^{n} \int_{\omega_{\min}}^{\omega_{\max}} \alpha_j(\omega) k_\omega(\boldsymbol{x}_j, \boldsymbol{x}_i) d\omega, \end{aligned} \tag{PIV}$$

where $\gamma > 0$ is a parameter that controls the sparsity of the solution and we define the "L_0-norm" to be the Lebesgue measure of the support of a function, i.e.,

$$\left\| f \right\|_{L_0} = \int_{\omega_{\min}}^{\omega_{\max}} \mathbb{I}[f(\omega) \neq 0] d\omega, \tag{6}$$

for the indicator function $\mathbb{I}[\omega \in \mathcal{E}] = 1$ for $\omega \in \mathcal{E}$ and zero otherwise.

The objective of (PIV) is composed of two terms: the first is a shrinkage term, i.e., the functional counterpart of the Euclidean norm in the objective of (PIII); the second, promotes sparsity of the functional coefficients α_j by minimizing their support. Notice that for this problem to be well-posed we constraint $\alpha_j \in L_2$. Otherwise, the cost function is ill-defined. Thus, the solutions of (PIV) cannot contain Dirac deltas and will instead present accumulations of mass around the optimal value of ω_j (see, e.g., Fig. 1). Moreover, recall that the main role of sparsity in (PIV) is in estimating the kernel parameter by forcing the α_j to accumulate around a specific parameter value. Still, a side-effect of this sparsification is that (PIV) will also promote parsimonious solutions by removing unnecessary centers, i.e., setting $\alpha_j \equiv 0$.

Still, the reformulation in (PIV) remains non-convex and is now also infinite dimensional, since the optimization variables are functions in L_2. Nevertheless, it is possible to compute its dual function in closed, which provides a lower bound for the optimization problem in (PIV). To do so, start by formulating the Lagrangian

$$\begin{aligned} \mathcal{L}(\alpha_j, \hat{y}_i, \lambda_i, \mu) = & \left[\gamma \int_\Omega \mathbb{I}[\alpha_j(\omega) \neq 0] d\omega + \frac{1}{2} \left\| \alpha_j \right\|_{L_2}^2 \right] \\ & + \sum_{i=1}^{n} \lambda_i \left[\hat{y}_i - \sum_{j=1}^{n} \int_\Omega \alpha_j(\omega) k_\omega(\boldsymbol{x}_j, \boldsymbol{x}_i) d\omega \right] \\ & + \mu \left[\sum_{i=1}^{n} \left[y_i - \hat{y}_i \right]^2 - \epsilon \right]. \end{aligned} \tag{7}$$

Then, define the dual function

$$g(\lambda_i, \mu) = \min_{\alpha_j \in L_2, \hat{y}_i} \mathcal{L}(\alpha_j, \hat{y}_i, \lambda_i, \mu). \tag{8}$$

Observe from (7) that the minimization in (8) separates across the primal variables as

$$\begin{aligned} g(\lambda_i, \mu) = & \min_{\alpha_j} \sum_{j=1}^{n} \int_\Omega F(\omega, \alpha_j(\omega)) d\omega \\ & + \min_{\hat{\boldsymbol{y}}} \left[\mu \left\| \boldsymbol{y} - \hat{\boldsymbol{y}} \right\|_2^2 + \boldsymbol{\lambda}^T \hat{\boldsymbol{y}} \right] - \mu\epsilon.. \end{aligned}$$

where we defined

$$\begin{aligned} F(\omega, \alpha_j(\omega)) = & \gamma \, \mathbb{I}(\alpha_j(\omega) \neq 0) + \frac{1}{2} |\alpha_j(\omega)|^2 \\ & - \sum_i \lambda_i \alpha_j(\omega) k_\omega(\boldsymbol{x}_j, \boldsymbol{x}_i), \end{aligned}$$

Using the result in [32, Thm. 3A], we can exchange the minimization with the integral and compute a closed form solution for the minimizers of (8). Explicitly, we obtain

$$\hat{\boldsymbol{y}}^\star(\omega, \boldsymbol{\lambda}) = \boldsymbol{y} + \frac{\boldsymbol{\lambda}}{2\mu}, \tag{9}$$

$$\alpha_j^\star(\omega, \boldsymbol{\lambda}) = [\boldsymbol{K}_\omega \boldsymbol{\lambda}]_j \times \mathbb{I}\left[|[\boldsymbol{K}_\omega \boldsymbol{\lambda}]_j| \geq \sqrt{2\gamma} \right], \tag{10}$$

where $[\boldsymbol{K}_\omega]_{ij} = k_\omega(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a kernel matrix and $[\boldsymbol{\lambda}]_j = \lambda_j$ is an $n \times 1$ vector that collects the dual variables. By substituting the minimizers in (9) and (10) into (7), we obtain an expression for the dual function:

$$\begin{aligned} g(\boldsymbol{\lambda}, \mu) = & \gamma \sum_{j=1}^{n} \int_\Omega \mathbb{I}\left[|[\boldsymbol{K}_\omega \boldsymbol{\lambda}]_j| \geq \sqrt{2\gamma} \right] d\omega - \frac{1}{2} \boldsymbol{\lambda}^\top \boldsymbol{Q} \boldsymbol{\lambda} \\ & - \frac{\boldsymbol{\lambda}^\top \boldsymbol{\lambda}}{2\mu} + \boldsymbol{\lambda}^\top \boldsymbol{y} - \mu\epsilon \end{aligned} \tag{11}$$

where $\boldsymbol{Q} = \int_\Omega \boldsymbol{K}_\omega \boldsymbol{M} \boldsymbol{K}_\omega d\omega$, with $\boldsymbol{M} = \operatorname{diag}\left(\mathbb{I}[|[\boldsymbol{K}_\omega \boldsymbol{\lambda}]_j| \geq \sqrt{2\gamma}] \right)$.

Recall that, the dual function is a lower bound on the value of (PIV). Hence, the best lower bound is given by the value of the dual problem

$$\underset{\boldsymbol{\lambda} \in \mathbb{R}^n, \mu \geq 0}{\text{maximize}} \quad g(\boldsymbol{\lambda}, \mu). \tag{DIV}$$

Notice that (DIV) is finite dimensional. Moreover, since the dual problem is always a convex program, it can be solved efficiently by using dual ascent. Still, the question of whether it is worth solving, i.e., how suboptimal are the α_j obtained from (DIV). The following theorem shows that they are actually optimal solutions of (PIV).

Theorem 1 ([28])**.** *If the function $\omega \mapsto k_\omega(\boldsymbol{x}_i, \boldsymbol{x}_j)$ has no point masses for all fixed $\boldsymbol{x}_i, \boldsymbol{x}_j$, then strong duality holds for* (PIV)*. Hence, if P is the optimal value of* (PIV) *and D is the optimal value of* (DIV)*, then $P = D$.*

Theorem 1 implies that (PIV) can be solved exactly in the dual domain. Indeed, we can obtain a pair of optimal dual variables $\boldsymbol{\lambda}^\star, \mu^\star$ by using supergradient ascent. In particular, the supergradients of the dual function g in (11) with respect to $\boldsymbol{\lambda}$ and μ are given by evaluating the constraint slacks at the minimizers in (9) and (10). Notice, however, that evaluating these supergradient involves computing the integral in (11). We propose to do so by using Monte Carlo approximations, leading to a stochastic gradient ascent (SGA) algorithm. Since Monte Carlo is an unbiased estimator of integrals, SGA converges almost surely to a pair of optimal dual variables $\boldsymbol{\lambda}^\star, \mu^\star$ under diminishing step size [33]. Theorem 1 then allows us to obtain a solution $\alpha_j^\star$ for (PIV) as $\alpha_j^\star(\omega) = \alpha_j^\star(\omega, \boldsymbol{\lambda}^\star)$.

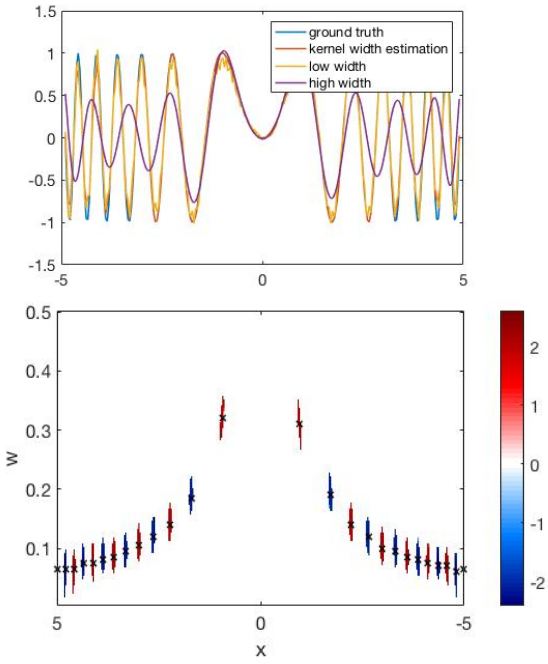

Fig. 1: (Top) Functions estimated by our method and fixed kernel method. (Bottom) Kernel widths and coefficients estimated by our method for each kernel center.

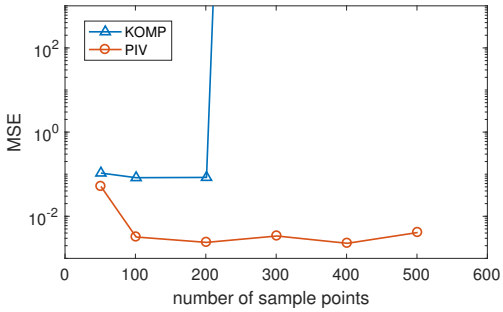

Fig. 2: Comparison of the MSE obtained by our method and KOMP as a function of sample size using 26 kernels.

4. SIMULATIONS

We start the numerical experiments by estimating the function $y_i = \sin(\frac{\pi}{2}x_i^2)+\xi_i, i = 0, \ldots, 100$, where the x_i are evenly spaced points in $[-5, 5]$ and the $\{\xi_i\}$ are zero-mean i.i.d. Gaussian random variables with variance $\sigma^2 = 10^{-3}$. In Figure 1, we show the target function and the estimation arising from (PIV) and fixed width kernel methods. As it can be observed in Figure 1, the wide kernel can only fit the lower frequency signal. On the other hand, the thin kernel fits the high frequency signal but cannot accurately model the low frequency signal with the number of training samples that are available. In contrast, (PIV) locally adapt the kernel parameters to place thin kernels where the signal varies rapidly and thicker kernels where the signal is smoother. The estimated $\boldsymbol{\alpha}(\omega)$ is represented underneath the original function. The decreasing smoothness of the signal away from zero as well as the symmetry of the signal are captured. We compare our method to kernel orthogonal matching pursuit (KOMP) [6], a commonly used backward greedy selection method that iteratively

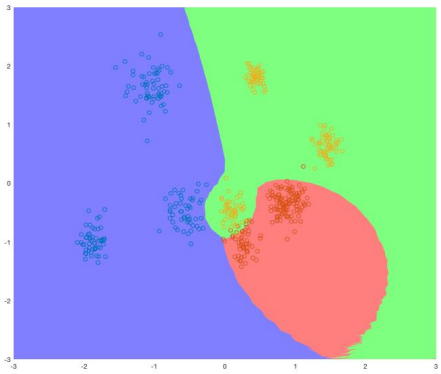

Fig. 3: Test data and estimated classes. First class: green, Second class: red, third class blue.

removes kernel centers until a desired sparsity is obtained. In particular, we set the sparsity to allow 26 kernels for both methods. The number of kernels was based on the number of peaks present in the function $\boldsymbol{\alpha}(\omega)$ (cf,. Figure 1). In Figure 2 we observe that KOMP's performance is dependent on the initial sample size, whereas our algorithm always finds a sparse solution with similar MSE.

We also use (PIV) in a multiclass classification problem using a one-against-all strategy as described in [34]. The data is generated from a Gaussian mixture model as in [35] with different standard deviations, creating two features for each data point. The labels y_i were drawn randomly from the label set. The dataset consists of 500 training samples and 500 test samples. The kernel centers are localized and correspond to the centers of the clusters in the data (see Figure 3) and that the procedure locally adapts the kernel width to the dataset. The classification achieves 97% accuracy using at most 25 kernels per class, a reduction of 95% compared to non-sparse methods. Figure 3 shows the estimated classes and the testing points. Our algorithm is able to model the fast variation in the center using thin kernels and keep the complexity of the model low in the areas that don't have a lot of change.

5. CONCLUSION

In this work we proposed a method to overcome two of the main limitations of kernel-based methods. These being, the kernel parameter selection and the inability of representing efficiently functions with various degrees of smoothness. To this end, we write each function belonging to a RKHS as a linear combination of kernels from a continuous dictionary, where the weights of said combination are given by functions in L_2. This reformulation yields an infinite dimensional, non-convex optimization problem with zero duality gap. Hence, we can solve it exactly and efficiently in the dual domain by running gradient ascent. The effectiveness of the method was illustrated with numerical experiments, where we aim to estimate a function with various degrees of smoothness and in a multi-class classification problem. In particular, we observed its ability to estimate the kernel width and remove uninformative kernels.

6. REFERENCES

[1] Roman Rosipal and Leonard J Trejo, "Kernel partial least squares regression in reproducing kernel hilbert space," *Jour-*

nal of machine learning research, vol. 2, no. Dec, pp. 97–123, 2001.

[2] Christopher M. Bishop, *Pattern recognition and machine learning*, Springer, 2006.

[3] Ming Yuan, T Tony Cai, et al., "A reproducing kernel hilbert space approach to functional linear regression," *The Annals of Statistics*, vol. 38, no. 6, pp. 3412–3444, 2010.

[4] Alain Berlinet and Christine Thomas-Agnan, *Reproducing kernel Hilbert spaces in probability and statistics*, Springer Science & Business Media, 2011.

[5] J. Arenas-Garcia, K.B. Petersen, G. Camps-Valls, and L.K. Hansen, "Kernel multivariate analysis framework for supervised subspace learning: A tutorial on linear and kernel multivariate methods," *IEEE Signal Process. Mag.*, vol. 30[4], pp. 16–29, 2013.

[6] Alec Koppel, Garrett Warnell, Ethan Stump, and Alejandro Ribeiro, "Parsimonious online learning with kernels via sparse projections in function space," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4671–4675.

[7] Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola, "Kernel methods in machine learning," *The annals of statistics*, pp. 1171–1220, 2008.

[8] George Kimeldorf and Grace Wahba, "Some results on tchebycheffian spline functions," *Journal of mathematical analysis and applications*, vol. 33, no. 1, pp. 82–95, 1971.

[9] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola, "A generalized representer theorem," in *International conference on computational learning theory*. Springer, 2001, pp. 416–426.

[10] Nabil Benoudjit and Michel Verleysen, "On the kernel widths in radial-basis function networks," *Neural Processing Letters*, vol. 18, no. 2, pp. 139–154, 2003.

[11] Gert RG Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I Jordan, "Learning the kernel matrix with semidefinite programming," *Journal of Machine learning research*, vol. 5, no. Jan, pp. 27–72, 2004.

[12] James Bergstra and Yoshua Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.

[13] Cheng-Hsuan Li, Hsin-Hua Ho, Yu-Lung Liu, Chin-Teng Lin, Bor-Chen Kuo, and Jin-Shiuh Taur, "An automatic method for selecting the parameter of the normalized kernel function to support vector machines," *J. Inf. Sci. Eng.*, vol. 28, pp. 1–15, 2012.

[14] B. Kuo, H. Ho, C. Li, C. Hung, and J. Taur, "A kernel-based feature selection method for svm with rbf kernel for hyperspectral image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 1, pp. 317–326, 2014.

[15] Max Kuhn and Kjell Johnson, *Applied Predictive Modeling*, Springer, 2016.

[16] Charles A Micchelli and Massimiliano Pontil, "Learning the kernel function via regularization," *Journal of machine learning research*, vol. 6, no. Jul, pp. 1099–1125, 2005.

[17] Mehmet Gönen and Ethem Alpaydın, "Multiple kernel learning algorithms," *Journal of machine learning research*, vol. 12, no. Jul, pp. 2211–2268, 2011.

[18] C.S. Ong, Alexander J. Smola, and Robert C. Williamson, "Learning the kernel with hyperkernels," *Journal of Machine Learning Research*, vol. 6, pp. 1043–1071, 2005.

[19] Andrew Gordon Wilson and Ryan Prescott Adams, "Gaussian process kernels for pattern discovery and extrapolation," in *International Conference on Machine Learning*, 2013, pp. III–1067–III–1075.

[20] Zichao Yang, Andrew Wilson, Alex Smola, and Le Song, "A la carte – Learning fast kernels," in *International Conference on Artificial Intelligence and Statistics*, 2015, pp. 1098–1106.

[21] D.L. Donoho and I.M. Johnstone, "Minimax estimation via wavelet shrinkage," *The Annals of Statistics*, vol. 26[3], pp. 879–921, 1998.

[22] M. Brockmann, T. Gasser, and E. Herrmann, "Locally adaptive bandwidth choice for kernel regression estimators," *Journal of the American Statistical Association*, vol. 88[24], pp. 1302–1309, 1993.

[23] X. Liu, L. Wang, J. Zhang, and J. Yin, "Sample-adaptive multiple kernel learning," in *AAAI Conference on Artificial Intelligence*, 1993, pp. 1975–1981.

[24] A. K. Ghosh, "Kernel discriminant analysis using case-specific smoothing parameters," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, no. 5, pp. 1413–1418, 2008.

[25] Jin Yuan, Liefeng Bo, Kesheng Wang, and Tao Yu, "Adaptive spherical gaussian kernel in sparse bayesian learning framework for nonlinear regression," *Expert Systems with Applications*, vol. 36, no. 2, Part 2, pp. 3982–3989, 2009.

[26] H. Fan, Q. Song, and S.B. Shrestha, "Kernel online learning with adaptive kernel width," *Neurocomputing*, vol. 175[A], pp. 233–242, 2016.

[27] Badong Chen, Junli Liang, Nanning Zheng, and José C. Príncipe, "Kernel least mean square with adaptive kernel size," *Neurocomputing*, vol. 191, no. 5, pp. 95–106, 2016.

[28] L.F.O. Chamon, Y.C. Eldar, and A. Ribeiro, "Strong duality of sparse functional optimization," in *Int. Conf. on Acoust., Speech and Signal Process.*, 2018, `http://bit.ly/2zVHJLy`.

[29] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, 2004.

[30] Akira Tanaka, Hideyuki Imai, Mineichi Kudo, and Masaaki Miyakoshi, "Theoretical analyses on a class of nested rkhs's," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2072–2075.

[31] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American Mathematical Society*, vol. 68, no. 3, pp. 337–404, 1950.

[32] R. T. Rockafellar, *Integral functionals, normal integrands and measurable selections*, Springer, 1976.

[33] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," 2016, arXiv:1606.04838.

[34] J. Weston and C. Watkins, "Support vector machines for multi-class pattern recognition," in *European Symposium on Artificial Neural Networks*, 1999, pp. 219–224.

[35] Ji Zhu and Trevor Hastie, "Kernel logistic regression and the import vector machine," *Journal of Computational and Graphical Statistics*, vol. 14, no. 1, pp. 185–205, 2005.