Kernel-based learning of orthogonal functions

Anna Scampicchio [*] **Gianluigi Pillonetto** [*] **Mauro Bisiacco** [*]

[*] *Department of Information Engineering, University of Padova, Italy*
(e-mails: {anna.scampicchio,giapi,bisiacco}@dei.unipd.it.)

Abstract: The paper deals with the reconstruction of functions from sparse and noisy data in suitable intersections of Hilbert spaces that account for orthogonality constraints. Such problem is becoming more and more relevant in several areas like imaging, dictionary learning, compressed sensing. We propose a new approach where it is interpreted as a particular kernel-based multi-task learning problem, with regularization formulated in a reproducing kernel Hilbert space. Special penalty terms are then designed to induce orthogonality. We show that the problem can be given a Bayesian interpretation. This then permits to overcome nonconvexity through a novel Markov chain Monte Carlo scheme able to recover the posterior of the unknown functions and also to understand from data if the orthogonal constraints really hold.

Keywords: orthogonal functions; regularization; kernels; Reproducing kernel Hilbert spaces; stochastic simulation; Markov chain Monte Carlo

1. INTRODUCTION

Many machine learning problems can be formulated as estimation of an unknown function from sparse and noisy data (Hastie et al., 2001). The problem can be solved using regularization theory, e.g. by adopting kernel-based methods (Schölkopf and Smola, 2001) that link Tikhonov regularization with reproducing kernel Hilbert spaces (RKHSs) (Aronszajn, 1950; Cucker and Smale, 2001). A more complex situation emerges when a collection of maps, known to share some common features, has to be reconstructed. In such scenario, data from a function can be useful also to reconstruct the other ones and the so-called multitask learning problem arises (Caruana, 1997; Thrun and Pratt, 1997; Bakker and Heskes, 2003). One can e.g. exploit vector-valued RKHSs that were developed in (Micchelli and Pontil, 2005) and lead to multitask regularized kernel methods (Evgeniou et al., 2005). Advantages of these approaches are described e.g. in (Pillonetto et al., 2010; Zhou and Zhao, 2016; Maurer et al., 2016; Liu et al., 2019; Zhang et al., 2019) in different scientific fields like biomedicine and imaging. Learning rates for some multitask algorithms have been also recently derived in (Xu et al., 2018).

A particular joint estimation problem involves functions known to be mutually orthogonal. It is becoming more and more relevant in many contexts including imaging, compressed sensing, dictionary learning and conformal mapping (Aharon et al., 2006; Tang et al., 2001; Ozolins et al., 2013; Lai and Osher, 2014; Dong et al., 2016). Non-convexity is the big issue and to overcome it many optimization algorithms have bene proposed in the literature (Edelman et al., 1998; Absil et al., 2007, 2008; Wen and Yin, 2013; Lai and Osher, 2014). However, a common limitation of all these approaches is that they are only guaranteeing to find a local minimum. Some interesting advances can be found in (Yuan et al., 2019) but only when one or two constraints are active.

The main novelty in this work is that we interpret estimation under orthogonality constraints as a particular kernel-based multitask learning problem. Beyond orthogonality, the unknown functions are assumed just to be smooth. To simplify the exposition, we also assume that direct and noisy samples of the maps are available but our approach can be also easily extended to the case where only linear transformations can be observed. Our novel algorithm allows also to assess if orthogonality assumptions really hold. This means that it can detect from data whether some of the constraints are not active and remove them from the estimation process.

Our new technique overcomes non-convexity adopting stochastic simulation in place of deterministic optimization. First, the problem is formulated by introducing RKHS norms and other special regularizers that induce orthogonality constraints. Then, we prove that such formulation admits a stochastic interpretation where the constraints are interpreted as particular Bayesian priors containing also hyperparameters that regulate both function smoothness and the interaction among the tasks (possibly also establishing whether orthogonality constraints are really present). This permits to use Markov chain Monte Carlo approaches (Raftery and Lewis, 1996) and, in particular, we define a Gibbs sampling scheme able to reconstruct in sampled form the posterior of all the functions. Numerical results involving simulated data are then used to illustrate the potential of our new modeling and computational framework.

[⋆] This paper has been supported by project PRIN 2015 2015PJ28EP, PI and National Coordinator Gianluigi Pillonetto.

The paper is organized as follows. Section 2 states the orthogonal functions learning problem, whose solution is reported in Section 3. In particular, after a review of function estimation in the RKHS framework with its Bayesian interpretation, we state the problem both in the deterministic and in the probabilistic setup. The latter will be crucial to build a suitable Markov Chain Monte Carlo scheme that solves the problem by overcoming its nonconvexity. Section 4 collects the numerical experiments that show the effectiveness of the proposed approach. Conclusions are drawn in Section 5.

2. PROBLEM STATEMENT

We consider unknown tasks (functions) denoted by $f_i : \mathcal{X} \to \mathbb{R}$ for $i = 1, \dots, r$. Each f_i is assumed to belong to a Hilbert space $\mathcal{H}$ with inner-product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. Also, the $\{f_i\}_{i=1}^r$ are known to form an orthonormal set. One can also think of a single unknown multi-task function that embeds all the f_i. It is denoted by $f : \mathcal{X} \times \{1, 2, \dots, r\}$ and belongs to the Hilbert space $\mathcal{S}$. If $f, g \in \mathcal{S}$, then the inner-product is given by

$$\langle f, g \rangle_{\mathcal{S}} = \sum_{i=1}^{r} \langle f_i, g_i \rangle_{\mathcal{H}}.$$

Each $f \in \mathcal{S}$ is associated with an $r \times r$ Gram matrix $\mathbb{K}_f$ whose (i, j)-entry is

$$[\mathbb{K}_f]_{ij} = \langle f_i, f_j \rangle_{\mathcal{H}}.$$

In view of the stated assumptions, it holds that

$$\mathbb{K}_f = I_r$$

where I_r is the $r \times r$ identity matrix. We assume that direct and noisy samples of any f_i are available. Suppose to collect n_i input/output pairs $\{x_{ki}, y_{ki}\}_{k=1}^{n_i}$ for each $i = 1, ..., r$. Denote with $\mathscr{X}_i$ and $\mathscr{Y}_i$ the sets of inputs and outputs for the $i-$th task. For each f_i, the measurements model is

$$y_{ki} = f_i(x_{ki}) + e_{ki}, \qquad k = 1, ..., n_i, \tag{1}$$

where e_{ki} is modeled as Gaussian white noise of variance σ_{ki}^2. From this, the aim is to estimate f from data.

3. KERNEL-BASED ORTHOGONAL FUNCTION ESTIMATION

3.1 Single-task case: review of RKHSs

Before studying the multi-task case, we focus on estimating a single $f_i(\cdot)$ from $\mathscr{X}_i$ and $\mathscr{Y}_i$. To overcome ill-posedness, we resort to the classic nonparametric regularized approach and state the problem as

$$\hat{f}_i = \arg \min_{f_i \in \mathscr{H}} \sum_{k=1}^{n_i} \frac{(y_{ki} - f_i(x_{ki}))^2}{\sigma_{ki}^2} + \gamma_i \|f_i\|_{\mathscr{H}}^2. \tag{2}$$

Uniqueness of the solution and small sensitivity with respect to data perturbation is provided by choosing a Reproducing Kernel Hilbert Space (RKHS) $\mathscr{H}$ as hypothesis space. This particular structure tightens up completeness with respect to the norm induced by the inner product $\langle \cdot, \cdot \rangle_{\mathscr{H}}$ and well-defined function pointwise evaluation. Moreover, from Moore-Aronszajn Theorem we know that each RKHS is in one-to-one correspondence with a positive semi-definite kernel operator $\mathscr{K}(\cdot, \cdot)$ such that

- $\mathscr{K}(x, \cdot) \in \mathscr{H}$ for all x in the domain, i.e. all kernel sections belong to the space;
- $\langle \mathscr{K}(x, \cdot), f \rangle_{\mathscr{H}} = f(x)$, which is the so-called reproducing property.

From this it results that each function in $\mathscr{H}$ inherits the properties encoded by the kernel, e.g. regularity and smoothness.

At this point, the real power of RKHS framework emerges: indeed this result, together with the Representer Theorem, yields that the solution of (2) is a linear combination of kernel sections $\{\mathscr{K}(x_{ki}, \cdot)\}_{k=1}^{n_i}$. This means that the starting infinite-dimensional problem admits a finite-dimensional representation. Thanks to this strong result, problem (2) can be reformulated in this way: introducing $\Sigma_i = \text{diag}(\sigma_{1i}^2 \cdots \sigma_{n_i i}^2)$, kernel matrix $K_{\mathscr{H}} \in \mathbb{R}^{n_i \times n_i}$ such that $[K_{\mathscr{H}}]_{a,b} = \mathscr{K}(x_a, x_b)$ for $a, b = 1, ..., n_i$ and collecting all outputs in vector Y_i, we get

$$\begin{aligned} \hat{c}_i &= \arg \min_{c_i} (Y_i - K_{\mathscr{H}} c_i)^\top \Sigma_i^{-1} (Y_i - K_{\mathscr{H}} c_i) + \gamma_i c_i^\top K_{\mathscr{H}} c_i \\ &= (K_{\mathscr{H}} \Sigma_i^{-1} K_{\mathscr{H}} + \gamma_i K_{\mathscr{H}})^{-1} K_{\mathscr{H}} \Sigma_i^{-1} Y_i \\ &= (K_{\mathscr{H}} + \gamma_i \Sigma_i)^{-1} Y_i. \end{aligned} \tag{3}$$

Interestingly, this result admits a Bayesian interpretation. In fact, giving c_i a prior distribution $c_i \sim \mathcal{N}(0, (\gamma_i K_{\mathscr{H}})^{-1})$ and considering the measurements model $Y_i = K_{\mathscr{H}} c_i + e_i$ with $e_i \sim \mathcal{N}(0, \Sigma_i)$ and known Σ_i, the posterior distribution $c_i | Y_i$ is Gaussian as well, and its mean is indeed the solution (3). Moreover, we can notice that the objective in (3) is the negative logarithm of the joint distribution $p(Y_i, c_i) = p(Y_i | c_i) p(c_i)$. Exploring such distribution is equivalent to exploring the posterior of c_i, since $p(c_i | Y_i) \propto p(Y_i, c_i)$.

3.2 Multi-task case: deterministic viewpoint

The general problem of multi-task orthogonal function estimation is stated as follows.

$$\hat{f} = \arg \min_{f} \sum_{i=1}^{r} \left[\sum_{k=1}^{n_i} \frac{(y_{ki} - f_i(x_{ki}))^2}{\sigma_{ki}^2} + \gamma_i \|f_i\|_{\mathscr{H}}^2 \right] + \\ + \sum_{i=1}^{r} \sum_{j>i} \gamma_{ij} |\langle f_i, f_j \rangle_{\mathcal{H}}|^2. \tag{4}$$

Adherence to observed data is balanced by two regularization terms ruled by $\mathscr{H}$ and $\mathcal{H}$ respectively. The first one is tuned by γ_i and ensures well-posedness of the function estimation problem; in addition, it sets the degree of smoothness for each $f_i(\cdot)$. The other term encodes the orthogonality constraint, and the inter-task interactions are ruled by γ_{ij} for $i = 1, ..., r$ and $j > i$: in particular, high values for γ_{ij} aim at setting $|\langle f_i, f_j \rangle_{\mathcal{H}}|$ as close to zero as possible. For problem (4) to be well-defined, we assume that $\mathcal{H}$ and $\mathscr{H}$ have non-empty intersection.

To simplify notation, we consider the same set $\mathscr{X}$ of input locations for each function f_i, with $\mathscr{X} = \cup_{i=1}^r \mathscr{X}_i$ of cardinality n. This can be done without loss of generality: for example, if for some i and input location x_{ki} the measurement y_{ki} is not available, then it suffices to set the corresponding weight σ_{ki}^2 to infinity.

At this point, we take a finite-dimensional approximation of (4) drawing inspiration from the Representer Theorem:

$$\begin{aligned} \arg \min_{\substack{c_i,\\ i=1\dots r}} & \sum_{i=1}^{r} (Y_i - K_{\mathscr{H}} c_i)^\top \Sigma_i^{-1} (Y_i - K_{\mathscr{H}} c_i) + \\ & + \gamma_i c_i^\top K_{\mathscr{H}} c_i + \sum_{i,\, j>i} \gamma_{ij} c_i^\top K_{\mathcal{H}} c_j c_j^\top K_{\mathcal{H}} c_i . \end{aligned} \tag{5}$$

Regularizing each f_i over both $\mathcal{H}$ and $\mathscr{H}$ is now encoded by the corresponding kernel matrices $K_{\mathcal{H}}$ and $K_{\mathscr{H}}$. Assuming that the input locations set $\mathscr{X}$ is unique for all f_i implies that kernel matrices are the same for each task.

Remark 1. If $\mathcal{H}$ is a RKHS, then $K_{\mathcal{H}}$ is the associated kernel matrix. Conversely, if $\mathcal{H}$ is just a Hilbert space as e.g. $\mathcal{L}^2$, matrix $K_{\mathcal{H}}$ has to be computed offline according to the kernel sections $\{K_{\mathscr{H}}(x_k, \cdot)\}_{k=1}^{n}$, which are known to be dense in the intersection of $\mathcal{H}$ and $\mathscr{H}$.

3.3 Multi-task case: Bayesian interpretation

To solve the nonconvex optimization problem stated in (5), we leverage on its Bayesian interpretation along the lines of Section 3.1. To this aim, we have to build a suitable Bayesian model such that, taking the negative logarithm of the joint density of all variables, we obtain the objective described by (5).
Together with (1), we assume to have the following fictitious measurements model

$$z_{ij} = \sqrt{c_i^\top K_{\mathcal{H}} c_j c_j^\top K_{\mathcal{H}} c_i} + \epsilon_{ij} \tag{6}$$

for each $i = 1, ..., r$ and $j > i$. We model ϵ_{ij} as Gaussian white noise of variance γ_{ij}^{-1}. Moreover, we assume that all hyperparameters γ_i, γ_{ij} and σ_{ki}^2 for $i = 1, ..., r$, $j > i$, $k = 1, ..., n$ are mutually independent. The resulting Bayesian network is presented in Figure 1.

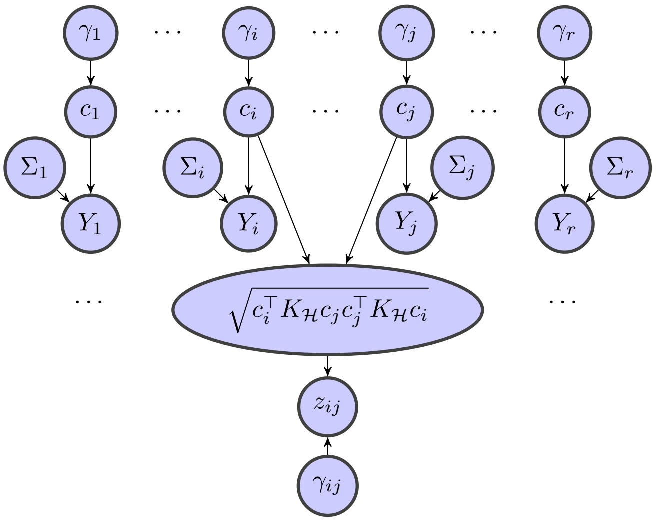

Fig. 1. Bayesian network associated to problem (5)

The joint density associated with such model is the following. For ease of notation, let $Y = \{Y_i\}_{i=1}^r$, $Z = \{z_{ij}\}_{i=1,j>i}^r$, $c = \{c_i\}_{i=1}^r$, $\Sigma = \{\Sigma_i\}_{i=1}^r$, $\gamma = \{\gamma_i\}_{i=1}^r$ and $\Gamma = \{\gamma_{ij}\}_{i=1,j>i}^r$: then, the network architecture yields

$$\begin{aligned} & p(Y, Z, c, \Sigma, \gamma, \Gamma) \\ =& p(Y, Z | c, \Sigma, \gamma, \Gamma) p(c | \Sigma, \gamma, \Gamma) p(\Sigma) p(\gamma) p(\Gamma) \\ =& p(Y | c, \Sigma) p(Z | c, \Gamma) p(c | \gamma) p(\Sigma) p(\gamma) p(\Gamma) \\ =& \left(\prod_{i=1}^{r} p(Y_i | c_i, \Sigma_i) p(c_i | \gamma_i) p(\Sigma_i) p(\gamma_i) \right) \times \\ & \times \left(\prod_{\substack{i=1 \\ j>i}}^{r} p(z_{ij} | c_i, c_j, \gamma_{ij}) p(\gamma_{ij}) \right). \end{aligned}$$

In particular, focusing on the factors $p(Y|c, \Sigma)$, $p(Z|c, \Gamma)$ and $p(c|\gamma)$, we have that

$$\begin{aligned} & p(Y_i | c_i, \Sigma_i) \propto e^{-(Y_i - K_{\mathcal{H}} c_i)^\top \Sigma_i^{-1} (Y_i - K_{\mathcal{H}} c_i)} \\ & p(c_i | \gamma_i) \propto e^{-\gamma_i c_i^\top K_{\mathscr{H}} c_i} \\ & p(z_{ij} | c_i, c_j, \gamma_{ij}) \propto e^{-\gamma_{ij} (z_{ij} - \sqrt{c_i^\top K_{\mathcal{H}} c_j c_j^\top K_{\mathcal{H}} c_i})^2} . \end{aligned}$$

Now, the key assumption consists in having all observations $z_{ij} = 0$ for each $i = 1, ..., r$ and $j > i$. Therefore, by inspection, we get that each c_i has the following prior distribution

$$\begin{aligned} & c_i | \gamma_i, c_{j>i}, \{\gamma_{ij}\}_{j>i} \sim \mathcal{N}\Big(0, P_i^{-1}\Big), \text{ with} \\ & P_i = \left(\mu_i K_{\mathscr{H}} + \sum_{j>i} \gamma_{ij} c_i^\top K_{\mathcal{H}} c_j c_j^\top K_{\mathcal{H}} c_i \right). \end{aligned}$$

This, together with the measurements model (1), implies that the posterior for each c_i is Gaussian as well. However, the direct computation of the posterior mean $\hat{c}_i$ is not feasible, due to the dependence from the hyperparameters that have to be estimated from data as well. To overcome this problem, we resort to the Markov Chain Monte Carlo paradigm. The rationale is the following: first we simulate the posterior $p(c, \gamma, \Gamma, \Sigma | Y, Z = 0) = \pi(c, \gamma, \Gamma, \Sigma)$, building a Markov chain whose invariant distribution is $\pi(c, \gamma, \Gamma, \Sigma)$; then we use N values of c_i sampled from such chain to approximate the posterior mean according to the Monte Carlo paradigm. The first step is conveniently implemented via Gibbs sampling. The strategy consists in sequentially updating the full conditionals, which in our case are

- $\pi(c_i | c_{j>i}, \gamma_i, \{\gamma_{ij}\}_{j>i}, \Sigma_i)$ for each $i = 1, ..., r$;
- $\pi(\gamma_i | c_i)$ for $i = 1, ..., r$;
- $\pi(\sigma_{ki}^2 | c_i)$ for $i = 1, ..., r$ and $k = 1, ..., n$;
- $\pi(\gamma_{ij} | c_i, c_j)$ for $i = 1, ..., r$ and $j > i$.

We retrieve the closed form expression for each of the full conditionals resorting to the properties of conjugate distributions. In particular, since Gaussian prior and likelihood yield a Gaussian posterior, we have

$$\begin{aligned} & c_i | Y_i, z_{i,j} = 0, c_{j>i}, \gamma_i, \gamma_{ij}, \Sigma_i \ \sim \mathcal{N}(\hat{c}_i, \hat{P}_i) \\ & \begin{cases} \hat{c}_i = P_i K_{\mathscr{H}} (K_{\mathscr{H}} P_i K_{\mathscr{H}} + \Sigma_i^{-1})^{-1} Y_i \\ \hat{P}_i = (K_{\mathscr{H}} \Sigma_i^{-1} K_{\mathscr{H}} + P_i^{-1})^{-1}. \end{cases} \end{aligned} \tag{7}$$

We assume that all γ_i, γ_{ij} and σ_{ki}^2 for $i = 1, ..., r$, $j > i$, and $k = 1, ..., n$ are endowed with an uninformative prior distribution over the positive real axis. Hence, denoting with $Gamma(a, b)$ a Gamma random variable with mean a/b, we get

$$\gamma_i | c_i \sim Gamma\left(\frac{n}{2}, \frac{c_i^\top K_{\mathscr{H}} c_i}{2} \right), \tag{8}$$

$$\sigma_{ki}^{-2}|c_i \sim Gamma\left(\frac{1}{2}, \frac{(y_{ki} - [K_{\mathscr{H}} c_i]_k)^2}{2}\right), \qquad (9)$$

$$\gamma_{ij}|c_i, c_j \sim Gamma\left(\frac{1}{2}, \frac{c_i^\top K_{\mathcal{H}} c_j c_j^\top K_{\mathcal{H}} c_i}{2}\right). \qquad (10)$$

In this Section we focus on the most general scenario in which all $r(r-1)/2$ values of γ_{ij} are assumed distinct. One could instead want to consider a single value $\lambda = \gamma_{ij}$ for all $i = 1, ..., r$ and $j > i$. In this situation, (10) becomes

$$\lambda|c \sim Gamma\left(\frac{r(r-1)}{4}, \frac{\sum_{i=1,j>i} c_i^\top K_{\mathcal{H}} c_j c_j^\top K_{\mathcal{H}} c_i}{2}\right).$$

The overall Gibbs sampling scheme for the general scenario is summarized in Algorithm 1.

Algorithm 1 The input is the number r, data sets $\mathscr{X}$ and $\mathscr{Y}$ (also expressed in vectors $\{Y_i\}_{i=1}^r$) and the number M of MCMC iterations. The output is a stochastic simulator of the posterior $\pi(c, \Sigma, \gamma, \Gamma)$.

```
for $m = 1...M$ do
    for $i = 1...r$ do
        if m=1 then
            set $c_i(1) = (K_{\mathscr{H}} + \gamma_i \Sigma_i)^{-1} Y_i$;
        else
            sample $c_i(m)$ from (7);
        end if
        sample $\gamma_i(m)$ from (8);
        sample $\sigma_{ki}^{-2}(m)$ from (9), $k = 1, ..., n$;
        build $\Sigma_i(m)$;
    end for
    for $i = 1...r$ and $j > i$ do
        sample $\gamma_{ij}(m)$ as (10)
    end for
end for
```

Standard Gibbs sampling theory ensures that the sequential updating procedure of Algorithm 1 yields a Markov chain whose invariant distribution is the posterior of interest $\pi(c, \gamma, \Sigma, \Gamma)$. Moreover, since the full conditionals are well defined, it results that the chain is irreducible: this is a sufficient condition for the Law of Large Numbers to hold, thus legitimating the use of Monte Carlo integration. Indeed, at the end of Algorithm 1, we are ready to estimate the posterior mean $\hat{c}_i$ for each i by selecting the last $N < M$ samples of the Markov chain and compute

$$c_i^\star = \frac{1}{N} \sum_{m=M-N}^{M} \hat{c}_i(m) \quad \text{for each } i = 1, ..., r \qquad (11)$$

according to the Monte Carlo procedure. Notice that M and N have to be chosen large enough for the Law of Large Numbers to give sensible results and to discard the first "burn-in" $M - N$ samples of the Markov Chain.

The stochastic simulation scheme above presented provides also another important information: since hyperparameters γ_{ij} are estimated from data as well, their values indicate whether the orthogonality constraints really hold. This allows us to eventually discard the ones that are not active, e.g. whose γ_{ij} takes a value that is lower than a certain threshold.

4. NUMERICAL EXPERIMENTS

The functions of interest in our tests are

$$f_i(x) = \sin(2\pi x i), \qquad i = 1, 2, 3. \qquad (12)$$

Each f_i is defined over $\mathcal{X} = [0, 1]$ and assumed to belong to the Hilbert space $\mathcal{H} = \mathcal{L}^2$. The RKHS $\mathscr{H}$ enforcing smoothness is the Sobolev space induced by the spline kernel

$$\mathscr{K}(x_a, x_b) = \min(x_a, x_b) \quad \text{with } x_a, x_b \in [0, 1]. \qquad (13)$$

Such space is known to be contained in $\mathcal{L}^2$, so the problem is well posed. In this scenario, regularizer $K_{\mathscr{H}}$ entering (5) is the kernel matrix associated to (13); on the other hand, $K_{\mathcal{H}}$ is defined by the $\mathcal{L}^2$-inner product of kernel sections in $\mathscr{H}$, which are known to be ramp and constant functions. In particular, the (a, b) element of matrix $K_{\mathcal{H}}$ is, for $a, b = 1, ..., n$,

$$[K_{\mathcal{H}}]_{a,b} = \int_0^1 \mathscr{K}(x_a, x)\mathscr{K}(x_b, x)dx$$
$$= \frac{x_a^3}{3} + x_a^2(x_b - x_a) + \frac{x_a}{2}(x_b - x_a)^2 + x_a x_b(1 - x_b).$$

Inputs are uniformly sampled from $\mathcal{X}$ and are collected in $\mathscr{X} = \{x_1, x_2, ..., x_{100}\}$; assume without loss of generality that $x_{k_1} < x_{k_2}$ if $k_1 < k_2$.

We consider two scenarios in which a triple $g = (g_1, g_2, g_3)$ defined by means of (12) has to be estimated. In particular, we will perform our tests on

E_1) $g = (g_1, g_2, g_3)$, with $g_1 = f_1$, $g_2 = f_2$ and $g_3 = f_3$;
E_2) $g = (g_1, g_2, g_3)$, with $g_i = f_1$ for all $i = 1, 2, 3$.

These two scenarios describe two opposite situations: the first involves a triple of orthogonal functions over $\mathcal{L}^2$, while the latter is a degenerate case aimed at testing the flexibility of our approach.

We assume that all output measures are available. Measurements noise is defined in terms of components g_j, $j = 1, 2, 3$: in particular, we have that the nominal noise variance is $\sigma_{nom}^2 = 0.2^2$ and that the measurements of g_1 and g_2 corresponding to 30 adjacent samples in $\mathscr{X}$ can be corrupted by very high noise. Defining σ_{kj}^2 the noise variance affecting the measure of g_j at the input location x_k, the situation is modeled in this way:

$$\sigma_{kj}^2 = \begin{cases} \sim \mathcal{U}(0, 100) & \text{if } i = 1, 2 \text{ and } k \in [\bar{k}, \bar{k} + 30] \\ 0.2^2 = \sigma_{nom}^2 & \text{otherwise,} \end{cases}$$

where $\bar{k} \in \{1, ..., 80\}$ is randomly chosen at each experiment and sets the outliers location in $\mathscr{X}$. In all tests we run our MCMC Algorithm 1 setting the noise variance constant and equal to σ_{nom}^2. Moreover, we aggregate all hyperparameters γ_{ij} in a single parameter λ.

For each experiment E_i we perform a Monte Carlo study of 100 runs. The performance will be studied from two points of view. The first one will show whether the orthogonality constraint improves the estimation in terms of fit, while the latter will assess whether the estimate of λ is coherent with the true orthogonality level of the components of g. As regards the fit measure, we will consider for each component g_j

$$Fit = 100\%\left(1 - \frac{\|g_j - \hat{g}_j\|_2}{\|g_j\|_2}\right).$$

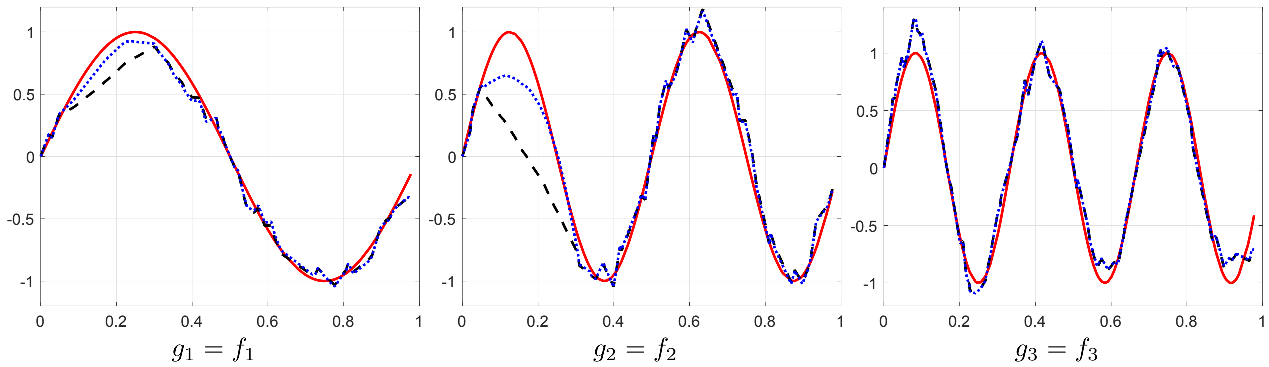

Fig. 2. Single-run of E_1. LEGEND: Solid Red=true function; Dashed Black=single task estimate without orthogonality constraint; Dotted Blue = multi-task estimate with orthogonality constraint. In this run, the estimated value of λ is 1.9×10^4 and the outliers window involves the interval $x_8, ...x_{38}$.

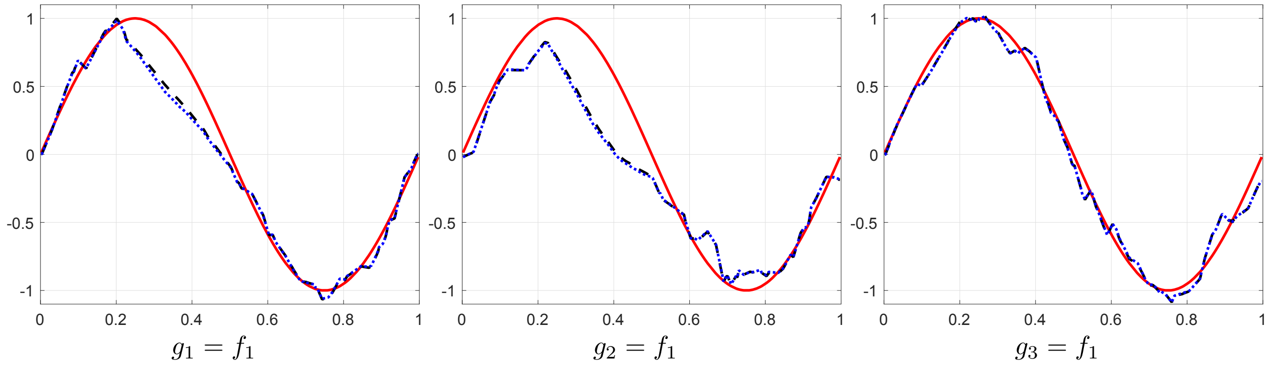

Fig. 3. Single-run of E_2. LEGEND: see Figure 2. The estimated value of λ is 2.4 and the outliers are located in the interval $x_{17}, ...x_{47}$.

$\mathbf{E_i}$	$\mathbf{g_i}$	**Single Task**	**Orthogonal Multitask**
E_1	$g_1 = f_1$	74.7	76.8
	$g_2 = f_2$	49.6	73.6
	$g_3 = f_3$	84.0	84.1
E_2	$g_1 = f_1$	74.9	73.8
	$g_2 = f_1$	75.1	73.9
	$g_3 = f_1$	89.1	88.9

Table 1. Mean fits for each experiment E_i in the 100-runs Monte Carlo experiment. Notice that at each run a different set of outliers is drawn.

In practice, the $\mathcal{L}^2$ norm will be numerically computed by taking the Euclidean norm on the vectors containing the pointwise function evaluations over a grid of 1000 equispaced samples of $\mathcal{X}$.

Figures 2 and 3 plot the performance for a single run for E_1 and E_2 respectively, and Table 1 gathers the mean fits in the overall Monte Carlo study. The solution of the orthogonal multitask estimation is compared with the decoupled single-task solution obtained by setting $\lambda = 0$.

Let us discuss the two approaches in terms of fit. As regards experiment E_1, which involves the estimation of three orthogonal functions, a general improvement given by the orthogonality constraint can be noted. The most significative one concerns component $g_2 = f_2$: there, the outliers strongly affect the goodness of the function estimate and the additional information about orthogonality significantly improves the performance. On the other hand, function $f_1 = g_1$ is sufficiently slow to be less encumbered by data unreliability.

In experiment E_2 we test the flexibility of our approach when the true functions are not orthogonal (in particular, they are exactly the same). We can see that the fits do not change significatively w.r.t. the unconstrained case: this means that the estimated λ has a low value, thus practically disactivating the orthogonality constraint. In fact, the estimated λ in the experiment of Figure 3 is equal to 2.4, while the one of Figure 2 is 1.9×10^4.

Hence, our approach is able to effectively estimate hyper-parameter λ that tunes the strength of the orthogonality constraint. In particular, it allows us to detect the true orthogonality level of the functions to be estimated.

5. CONCLUSIONS

This paper deals with multitask learning of functions whose joint information is described by their orthogonality relation. Such property enters as a suitable regularization term in the classic nonparametric function estimation problem over Reproducing Kernel Hilbert Spaces. Then, the probabilistic interpretation of such problem has been exploited to set up a suitable Markov Chain Monte Carlo scheme that overcomes the nonconvexity of the original formulation and provides an estimate of all the involved parameters. The proposed approach yields very promising results that will be further investigated in future works.

REFERENCES

Absil, P.A., Mahony, R., and Sepulchre, R. (2007). *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, USA.

Absil, P., Mahony, R., and Sepulchre, R. (2008). *Optimization algorithms on matrix manifolds*. Princeton University Press.

Aharon, M., Elad, M., and Bruckstein, A. (2006). K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11), 4311–4322.

Aronszajn, N. (1950). Theory of reproducing kernels. *Trans. of the American Mathematical Society*, 68, 337–404.

Bakker, B. and Heskes, T. (2003). Task clustering and gating for Bayesian multitask learning. *J. Mach. Learn. Res.*, 4, 83–99.

Caruana, R. (1997). Multitask learning. *Mach. Learn.*, 28(1), 41–75.

Cucker, F. and Smale, S. (2001). On the mathematical foundations of learning. *Bulletin of the American mathematical society*, 39, 1–49.

Dong, B., Liu, R., and Wang, H.W. (2016). Trust-but-verify: Verifying result correctness of outsourced frequent itemset mining in data-mining-as-a-service paradigm. *IEEE Transactions on Services Computing*, 9(1), 18–32.

Edelman, A., Arias, T., and Smith, S. (1998). The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20, 303–353.

Evgeniou, T., Micchelli, C., and Pontil, M. (2005). Learning multiple tasks with kernel methods. *J. Mach. Learn. Res.*, 6, 615–637.

Hastie, T.J., Tibshirani, R.J., and Friedman, J. (2001). *The Elements of Statistical Learning. Data Mining, Inference and Prediction.* Springer, Canada.

Lai, R. and Osher, S. (2014). A splitting method for orthogonality constrained problems. *Journal of Scientific Computing*, 58(2), 431–449.

Liu, Z., Huang, B., Cui, Y., Xu, Y., Zhang, B., Zhu, L., Wang, Y., Jin, L., and Wu, D. (2019). Multi-task deep learning with dynamic programming for embryo early development stage classification from time-lapse videos. *IEEE Access*, 7, 122153–122163.

Maurer, A., Pontil, M., and Romera-Paredes, B. (2016). The benefit of multitask representation learning. *J. Mach. Learn. Res.*, 17(1), 2853–2884.

Micchelli, C. and Pontil, M. (2005). On learning vector-valued functions. *Neural Comput.*, 17(1), 177–204.

Ozolins, V., Lai, R., Caflisch, R., and Osher, S. (2013). Compressed modes for variational problems in mathematics and physics. In *Proc Natl Acad Sci U.S.A.*, volume 110, 18368–73.

Pillonetto, G., Dinuzzo, F., and De Nicolao, G. (2010). Bayesian on-line multi-task learning of Gaussian processes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(2), 193–205.

Raftery, A. and Lewis, S. (1996). *Markov Chain Monte Carlo in Practice*, chapter Implementing MCMC. London: Chapman and Hall.

Schölkopf, B. and Smola, A.J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* (Adaptive Computation and Machine Learning). MIT Press.

Tang, B., Sapiro, G., and Caselles, V. (2001). Color image enhancement via chromaticity diffusion. *IEEE Transactions on Image Processing*, 10(5), 701–707.

Thrun, S. and Pratt, L. (1997). *Learning to learn.* Kluwer.

Wen, Z. and Yin, W. (2013). A feasible method for optimization with orthogonality constraints. *Math. Program.*, 142, 397–434.

Xu, Y., Li, X., Chen, D., and Li, H. (2018). Learning rates of regularized regression with multiple gaussian kernels for multi-task learning. *IEEE Transactions on Neural Networks and Learning Systems*, 29(11), 5408–5418.

Yuan, H., Gu, X., Lai, R., and Wen, Z. (2019). Global optimization with orthogonality constraints via stochastic diffusion on manifold. *J. Sci. Comput.*, 80(2), 1139–1170.

Zhang, J., Zhang, Y., Ji, D., and Liu, M. (2019). Multi-task and multi-view training for end-to-end relation extraction. *Neurocomputing*, 364, 245 – 253.

Zhou, Q. and Zhao, Q. (2016). Flexible clustered multi-task learning by learning representative tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2), 266–278.