Joint learning of RKHS and kernel sum-of-squares functions: a representer theorem and its convex dual

Louis Allain[1,2]* **Sébastien Da Veiga**[2]

[1]Safran AI Research, Magny-Les-Hameaux, France [2]Univ Rennes, Ensai, CNRS, CREST - UMR 9194, F-35000 Rennes, France

Date published: 2026-09-06 Last modified: 2026-09-06

Abstract

Kernel sum-of-squares (kSoS) models represent a non-negative function as a quadratic form in an reproducing kernel Hilbert space (RKHS), guaranteeing non-negativity everywhere. It has emerged as a promising kernel approach to model non-negative phenomena, leveraging a representer theorem that makes the problem tractable. This result only holds for kSoS functions, and with the same RKHS. However, real-world problems may require to learn general-valued functions *jointly* with kSoS functions, and may also benefit from different kernels for each kSoS function for better predictive accuracy. In this work, we thus consider such wider class of statistical learning problems, where we are interested in jointly learning p real valued functions and q non-negative functions defined on different RKHS. Building on representer theorems from traditional kernel methods and kSoS, we introduce a new generalized representer theorem for that learning problem. The induced finite-dimensional problem is a semi-definite program (SDP), easily solvable using off-the-shelf solvers. To scale better than SDP solvers, we also establish a convex dual formulation, which writes as an optimization problem with only $\mathcal{O}(n)$ variables and no positive semi-definite constraints. Finally, for completeness and to highlight the potential of the kSoS framework, we derive the explicit dual formulations for three kSoS problems previously introduced in the literature and provide two new problems leveraging the generalized representer theorem. All our experiments are reproducible with our accompanying Python code.

Keywords: kernel sum-of-squares, statistical learning

Contents

*Corresponding author: louis.allain@safrangroup.com

1 Introduction

Non-negative functions often appear in physics and statistics applications. When modelling real physical systems for example, prediction of quantities such as concentration, prices or distances, must satisfy non-negativity everywhere at the risk of violating physical laws. In statistics, common estimation problems include modelling a variance or a probability density function, and the final estimate can be highly sensitive to local non-negativity violation. Even a minor negative estimate can have great consequences on the real world decision taken based upon that statistical model. Furthermore, non-negative functions can even appear through the reformulation of an initial complex problem. This is for instance the case in global optimization or in optimal transport, but also when the statistical task involves the estimation of monotonic or convex functions. In machine learning and statistics, the ability to estimate a non-negative function, being non-negative everywhere, is thus critical.

The most common way to account for non-negativity in a statistical problem is through the use of a so-called link function. Given such a link function $\psi : \mathbb{R} \longrightarrow \mathcal{Y}$, the unknown function is modelled as $\psi(f(x))$ and we learn the function f freely. Common examples of link functions are the exponential function $\psi(z) = e^z$ and the popular ReLU function $\psi(z) = \max(0, z)$ or its smooth variant $\psi(z) = \ln(1 + e^z)$, the Softplus function. Using a link function is simple yet has strong expressive power: indeed the estimation of f can be done using a parametric model such as a neural network, or kernel methods, both universal approximators. But the choice of the link function here is crucial. Indeed, as noted by Marteau-Ferey et al. (2020), nice properties of the loss function such as convexity may not be preserved for some link functions, and they may also complexify potential additional constraints that are tractable for f but not for $\psi(f(x))$.

Other prominent works in the literature focus on linear models with non-negative coefficients and basis functions. Here, the function of interest is modelled as $f(x) = \sum_i w_i h_i(x)$ for some non-negative basis functions h_i (e.g. B-splines or specific kernel functions), and such a linear model yields a non-negative function as long as the coefficients w_i are positive. Despite the seeming expressiveness that comes from the choice of the basis, especially with kernel methods, restricting the coefficient to be positive however reduces the search space to the conical hull, which ultimately limits the expressiveness of the final estimated function f.

Finally, a more principled way to handle non-negativity is to directly model the function of interest as a sum-of-squares: $f(x) = \sum_i h_i(x)^2$. This structurally guarantees non-negativity without requiring a complex, possibly non-convex link function. For optimization problems, this sum-of-squares representation was introduced by Parrilo (2000) and Lasserre (2001), in order to transform a global and non-convex optimization problem into a sequence of semi-definite programs. For a modern

introduction, we refer the interested reader to Bach et al. (2024). Equivalently, remark that a sum-of-squares function can be written as the quadratic form $f(x) = h(x)^\top \mathbf{Q} h(x)$, where $\mathbf{Q}$ is a real symmetric, positive semi-definite matrix. Using the spectral decomposition of $\mathbf{Q} = \sum_{i \in I} \mu_i u_i u_i^\top$, with eigenvalues μ_i, eigenvectors u_i and I the rank of $\mathbf{Q}$, we can retrieve the sum-of-squares form of f as

$$f(x) = h(x)^\top \sum_{i \in I} \mu_i u_i u_i^\top h(x) = \sum_{i \in I} \mu_i (h(x)^\top u_i)(u_i^\top h(x)) = \sum_{i \in I} (\sqrt{\mu_i} u_i^\top h(x))^2.$$

Even though such sum-of-squares representations allow to model and estimate non-negative functions everywhere, once again they may lack expressive power due to the restricted choice of the basis functions h_i, where mostly polynomial basis are used in the literature (Lasserre 2009; Jaini et al. 2019; Mula and Nouy 2024).

To address this need, Marteau-Ferey et al. (2020) recently introduced a new statistical learning problem based on the kernel sum-of-squares (kSoS) framework to estimate non-negative functions, bridging together the power of sum-of-squares representations and the expressive power of kernel methods. The latter gained popularity because they often offer convex optimization problems, perform well on small datasets, and provide great flexibility through the choice of the kernel function, allowing users to tackle problems across highly diverse data types. In their work, Marteau-Ferey et al. (2020) interestingly show that kSoS models satisfy appreciated properties in learning problems: convexity of the loss function, universal approximation of non-negative functions (with universal kernels), finite-dimensional representation through a representer theorem as in standard kernel problems. In addition, kSoS models are also differentiable and integrable in closed form, which is useful for problems involving constrained outputs such as density estimation. Altogether, this makes kernel sum-of-squares a powerful framework for non-negative function estimation. As an illustration, it has since been used in various fields such as global optimization (Rudi et al. 2025), optimal transport (Vacher et al. 2024), uncertainty quantification (Allain et al. 2025, 2026) and statistical learning of convex functions (Muzellec et al. 2022), to name a few.

However, the initial kSoS proposal of Marteau-Ferey et al. (2020), although influential, can only handle the estimation of one unknown kSoS function, or several ones but with the same RKHS. This means that statistical problems which involve additional unknown unconstrained functions, or several kSoS functions which should require different hyperparameters, can not be directly dealt with in this framework. If the former can be accommodated with through a sequential learning procedure, this is not satisfactory from a statistics perspective and almost surely leads to sub-optimal estimations. Less importantly from a theoretical point of view but fundamental in practice, in the original work of Marteau-Ferey et al. (2020) the loss function is required to be bounded below, which limits the range of achievable learning problems, and the dual formulations corresponding to the statistical tasks illustrated in their experiments were omitted for conciseness.

In this paper, we thus aim at extending the representer theorem and dual formulation of Marteau-Ferey et al. (2020), by allowing joint estimation of unconstrained functions and non-negative ones, but also give explicit and detailed dual formulations for several learning problems, both for reference and availability to the community. Our detailed contributions are as follows:

- First, we introduce a generalized representer theorem for statistical problems which involve both unconstrained and non-negative functions. This encompasses for example heteroscedastic regression and non-crossing multi-quantile regression, but also estimation of conformal scores as in Allain et al. (2025) even though we do not illustrate this problem here.
- Second, we show that the resulting optimization problem admits a convex dual formulation if the loss function is convex. This removes the positive semi-definite constraints from the primal form and considerably reduces the complexity of the optimization problem, which drops from $\mathcal{O}(n^3)$ to $\mathcal{O}(n)$ variables for the dual formulation. By strong duality it recovers the primal solution exactly, while scaling much more easily with respect to the dataset size.

Besides, all our results are given in a general potentially rank-deficient setting, which allows to consider standard low-rank approximations that can be smoothly incorporated in both primal and dual formulations. This pushes the applicability of kSoS problems to even larger scales.

- Third, we derive the explicit dual formulas for the kSoS examples studied by Marteau-Ferey et al. (2020), namely density estimation, heteroscedastic regression with known mean function and non-crossing multi-quantile regression with known median function. For each of them we provide numerical illustrations of what can be expected from kSoS models in practice.

- Fourth, we introduce two new case studies in which we jointly learn a real-valued function and kSoS functions: heteroscedastic regression where both mean and variance functions are estimated, and non-crossing multi-quantile regression where the median function is learned jointly along with the quantile functions. Thanks to our general representer theorem, we show that these two problems admit a finite-dimensional representation, and we also derive their dual formulation explicitly. In order to showcase the potential of such framework, we finally conduct several illustrative numerical experiments.

Throughout this paper, we will thus evaluate the initial kSoS approach or our extension on several problems. However, it is important to clarify the primary objective of the empirical illustrations presented in this work. Our goal is not to position kSoS methods as the state-of-the-art solution that outperforms established competitors across a competitive benchmark. Instead, these examples are meant to highlight the elegance, structural flexibility, and highly efficient dual optimization problems of the kSoS framework. By demonstrating its natural adaptability to diverse statistical problems, our intent here is to advocate for its broader adoption and inspire further exploration of its potential within the community.

The paper is organized as follows. In Section 2, we begin by introducing the kSoS framework developed by Marteau-Ferey et al. (2020). We follow in Section 3 by generalizing this framework to jointly learn RKHS and kSoS functions, and show that a finite-dimensional representation still holds through a new representer theorem. We then prove that an associated dual formulation can still be derived, and that kernel low-rank approximation can be used to handle large scale estimation problems. Then, Section 4 is dedicated to several practical learning problems. For completeness, in Section 4.1 we first derive explicitly the dual formulations for three kSoS problems studied before, namely density estimation, heteroscedastic regression and non-crossing multi-quantile regression. But in Section 4.2, we introduce two new statistical problems in which we learn concurrently an unconstrained function in a RKHS together with kSoS functions. These two problems are heteroscedastic regression with unknown mean and variance functions, and non-crossing multi-quantile regression problem with unknown median and quantile functions.

2 Kernel sum-of-squares

Let us begin by introducing the kSoS framework developed by Marteau-Ferey et al. (2020). Inspired by previous work on sum-of-squares, they proposed to model a non-negative function using a quadratic form which involves functions in a reproducing kernel Hilbert space (RKHS). Before providing the definition, we first introduce the necessary notations. We will write a RKHS $\mathscr{H}$ with associated feature map $\phi : \mathscr{X} \longrightarrow \mathscr{H}$ and kernel k. We denote by $\mathcal{S}(\mathscr{H})$ the set of bounded Hermitian linear operators from $\mathscr{H}$ to $\mathscr{H}$ and by $\mathcal{S}_+(\mathscr{H})$ those that are positive semi-definite (PSD). We also write $\mathbb{S}^r := \mathbb{S}(\mathbb{R}^{r\times r})$ the set of real and symmetric matrices of size r, and $\mathbb{S}^r_+ := \mathbb{S}_+(\mathbb{R}^{r\times r})$ the set of real, symmetric and PSD matrices of size r. A kernel sum-of-squares function is defined for each $x \in \mathscr{X}$ as the quadratic form

$$f_{\mathscr{A}}(x) = \langle \phi(x), \mathscr{A}\phi(x)\rangle_{\mathscr{H}}, \quad \mathscr{A} \in \mathcal{S}_+(\mathscr{H}). \tag{1}$$

Because $\mathscr{A}$ is a positive semi-definite operator, it admits a spectral decompositon given by $\mathscr{A} = \sum_{i\geq 0} \mu_i (u_i \otimes u_i)$ where $\mu_i > 0$ are the eigenvalues and $u_i \in \mathscr{H}$ the eigenfunctions of $\mathscr{A}$. Using this decomposition, the function $f_{\mathscr{A}}$ writes as

$$f_{\mathscr{A}}(x) = \sum_{i\geq 0} \mu_i \langle \phi(x), (u_i \otimes u_i)\phi(x) \rangle_{\mathscr{H}} = \sum_{i\geq 0} \mu_i \langle \phi(x), u_i \langle u_i, \phi(x) \rangle_{\mathscr{H}} \rangle_{\mathscr{H}} = \sum_{i\geq 0} (\sqrt{\mu_i} \langle u_i, \phi(x) \rangle_{\mathscr{H}})^2.$$

This simple derivation shows that the quadratic form in Equation 1 actually corresponds to a sum-of-squares function, and is thus non-negative everywhere. Interestingly, such a kSoS function is entirely characterized by its associated positive semi-definite operator $\mathscr{A}$.

With this definition, we are now ready to discuss statistical learning tasks where we want to estimate a non-negative function $f_{\mathscr{A}}$, which is equivalent to estimate its associated operator $\mathscr{A}$. Marteau-Ferey et al. (2020) introduced a regularized learning problem, where the unknown optimization variable is a kSoS function as defined in Equation 1. Considering a training sample of features $x_1, \dots, x_n$, a loss function L defined on the point-wise function evaluations at the training points and a regularization function Ω on the unknown operator, their problem writes as:

$$\inf_{\mathscr{A} \in \mathcal{S}_+(\mathscr{H})} L(f_{\mathscr{A}}(x_1), \dots, f_{\mathscr{A}}(x_n)) + \Omega(\mathscr{A}). \tag{2}$$

We can observe the similarity with standard kernel problems, where the traditional optimization variable $f \in \mathscr{H}$ is replaced with $\mathscr{A} \in \mathcal{S}_+(\mathscr{H})$. As noted by Marteau-Ferey et al. (2020), the existence and uniqueness of solutions in Problem 2 heavily depend on the shape of the regularization function. In their original paper, they mainly focus on the elastic-net regularization given by

$$\Omega(\mathscr{A}) = \lambda_1 \|\mathscr{A}\|_\star + \lambda_2 \|\mathscr{A}\|_F^2, \tag{3}$$

but their theoretical result actually holds for a wider class of regularization functions, which we will introduce and discuss later.

The main result in Marteau-Ferey et al. (2020) is the following representer theorem, which states that Problem 2 admits a solution and that this solution is finite-dimensional.

Theorem 2.1. *Let L be a lower semi-continuous and bounded below function and Ω defined as in Equation 3. The learning problem in Equation 2 admits a solution $\mathscr{A}^\star$ writing as:*

$$\mathscr{A}^\star = \sum_{i,j=1}^{n} \mathbf{B}_{ij} \phi(x_i)\phi(x_j)^\top \text{ for some matrix } \mathbf{B} \in \mathbb{S}_+^n,$$

with $f_{\mathscr{A}^\star}(x) = \sum_{i,j=1}^{n} \mathbf{B}_{ij} k(x, x_i) k(x, x_j)$.

Crucially, the estimation of the infinite-dimensional operator $\mathscr{A}$ is replaced with the estimation of a PSD matrix $\mathbf{B}$ of size $n \times n$. As an illustration of this representer theorem, Marteau-Ferey et al. (2020) present two learning problems, heteroscedastic regression and non-crossing multi-quantiles regression. The former is inherently a non-negative function estimation task since the goal is to learn the variance function, while the latter comes from a smart reparameterization trick: quantile functions are modelled as non-negative increments around the median function to ensure non-crossing, and these increments are precisely the non-negative functions which are learned. In addition, their original work was also generalized later for global optimization (Rudi et al. 2025) or to learn convex functions (Muzellec et al. 2022).

Unfortunately, although powerful and ground-breaking for non-negative function estimation, this theorem does not apply to all practical scenarios. First, some statistical problems necessitate to learn, in addition of a positive function, an unconstrained general-valued function. For instance in heteroscedastic regression, we may want to learn the variance as a non-negative function and the mean as a general real-valued function. Similarly, in the non-crossing multi-quantile setting

of Marteau-Ferey et al. (2020), the median function may be learned simultaneously but without non-negativity constraints. This practical situation also arises when learning a conformal score function as was done in the work of Allain et al. (2025). Second, other statistical problems require to learn multiple kSoS functions at once: this has been done in Theorem 6 from Marteau-Ferey et al. (2020), which encompasses their example on multi-quantile regression. But this extension only considers multiple kSoS functions defined with operators acting on the same RKHS. Even if in practice we may use the same kernel for all functions, it is frequent to model them with different kernel lengthscales for flexibility, which ultimately yields different RKHS: the original setting introduced by Marteau-Ferey et al. (2020) thus does not apply here. To handle those two practical situations, there is the need to extend Theorem 2.1.

Interestingly, preliminary attempts have been proposed in previous work: in the context of conformal score learning, Allain et al. (2025) proved a representer theorem for one RKHS function and one kSoS function for a very specific loss function, while Vacher et al. (2024) also considered a problem with two RKHS functions and one kSoS function for a loss related to optimal transport. In parallel, Allain et al. (2026) proved a representer theorem for a general loss function and multiple kSoS functions, each with its own kernel, but without additional unconstrained functions. In the next section, we thus unify all settings into a single framework, showing that it is possible to learn p real-valued functions, each in their own RKHS, alongside q kSoS functions with operators defined on different RHKS as well.

3 Generalized representer theorem

Our main result is Theorem 3.1 below, which follows the generalization of Theorem 2.1 to multiple kSoS functions proposed by Allain et al. (2026). Let us first start with some notations.

We consider a collection of p RKHS $\mathscr{H}_l^g$, $l = 1, \dots, p$ and q RKHS $\mathscr{H}_s^f$, $s = 1, \dots, q$. To each RKHS $\mathscr{H}_l^g$ and $\mathscr{H}_s^f$ is associated a unique kernel k_l^g and k_s^f with feature map ϕ_l^g and ϕ_s^f, respectively. For our learning problem, these function spaces allow us to introduce our key objects of interest:

- The p *unknown* real-valued functions $g_l : \mathcal{X} \to \mathbb{R}$ in a RKHS $\mathscr{H}_l^g$ for $l = 1, \dots, p$
- The q *unknown* kSoS functions $f_{\mathscr{A}_s} : \mathcal{X} \to \mathbb{R}^+$ parameterized by a positive semi-definite operator $\mathscr{A}_s \in \mathcal{S}_+(\mathscr{H}_s^f)$ defined on a RKHS $\mathscr{H}_s^f$ with $f_{\mathscr{A}_s}(x) = \langle \phi_s^f(x), \mathscr{A}_s \phi_s^f(x) \rangle_{\mathscr{H}_s^f}$ for $s = 1, \dots, q$
- The collection of RKHS functions $g := (g_1, \dots, g_p) \in \mathscr{K}^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g) := \mathscr{H}_1^g \times \cdots \times \mathscr{H}_p^g$
- The collection of operators $\mathscr{A} := (\mathscr{A}_1, \dots, \mathscr{A}_q) \in \mathscr{K}^f(\mathscr{H}_1^f, \dots, \mathscr{H}_q^f) := \mathcal{S}_+(\mathscr{H}_1^f) \times \cdots \times \mathcal{S}_+(\mathscr{H}_q^f)$
- The collection of RKHS function evaluations $g(x) := (g_1(x), \dots, g_p(x)) \in \mathbb{R}^p, \ \forall x \in \mathcal{X}$
- The collection of kSoS function evaluations $f_{\mathscr{A}}(x) := f_{\mathscr{A}_1, \dots, \mathscr{A}_q}(x) = \left(f_{\mathscr{A}_1}(x), \dots, f_{\mathscr{A}_q}(x) \right) \in \mathbb{R}_+^q, \ \forall x \in \mathcal{X}$

Finally, for a training sample $x_1, \dots, x_n$ of features, we denote $\mathbf{K}_l^g$ the kernel matrix with elements $[\mathbf{K}_l^g]_{ij} = k_l^g(x_i, x_j)$ and $k_l^g(x)$ the column vector defined by

$$k_l^g(x) = (k_l^g(x, x_1), \dots, k_l^g(x, x_n))^\top$$

for any $x \in \mathcal{X}$. Similarly, we introduce $\mathbf{K}_s^f$, $k_s^f(x)$ and further consider a decomposition $\mathbf{K}_s^f = \mathbf{V}_s^\top \mathbf{V}_s$ with $\mathbf{V}_s \in \mathbb{R}^{r_s \times n}$ where r_s is the rank of $\mathbf{K}_s^f$, and the empirical feature map

$$\Phi_s(x) = (\mathbf{V}_s \mathbf{V}_s^\top)^{-1} \mathbf{V}_s k_s^f(x). \tag{4}$$

With these notations, we now introduce a new statistical learning problem for both RKHS and kSoS functions which extends Problem 2. We consider a training sample of features $x_1, \dots, x_n$, a new loss function L which also incorporates point-wise evaluations of the RKHS functions, and a regularization broken down into two parts: the usual kSoS regularizer of Marteau-Ferey et al. (2020) plus a regularizer for each RKHS function. Our learning problem writes as:

$$\inf_{\substack{g \in \mathcal{K}^g(\mathcal{H}_1^g, \dots, \mathcal{H}_p^g) \\ \mathscr{A} \in \mathcal{K}^f(\mathcal{H}_1^f, \dots, \mathcal{H}_q^f)}} L\left(g(x_1), \dots, g(x_n), f_{\mathscr{A}}(x_1), \dots, f_{\mathscr{A}}(x_n)\right) + \sum_{l=1}^{p} R_l\left(\|g_l\|_{\mathcal{H}_l^g}\right) + \sum_{s=1}^{q} \Omega_s(\mathscr{A}_s). \tag{5}$$

The kSoS regularizers Ω_s are assumed to satisfy the same broad assumption given by Marteau-Ferey et al. (2020), which we recall in Assumption 3.1 below. For the RKHS functions, we consider the usual RKHS-norm regularization $R_l\left(\|g_l\|_{\mathcal{H}_l^g}\right)$ (Schölkopf and Smola 2002) with the R_l functions satisfying Assumption 3.2.

Assumption 3.1. *For each $s = 1, \dots, q$, Ω_s writes as*

$$\Omega_s(\mathscr{A}_s) = \begin{cases} \mathrm{Tr}\left(h_s(\mathscr{A}_s)\right) = \sum_k h_s(\sigma_k) & \text{if } \mathscr{A}_s = U\,\mathrm{Diag}(\sigma)\,U^\top,\ \sum_k h_s(\sigma_k) < +\infty, \\ +\infty & \text{otherwise,} \end{cases}$$

where $h_s : \mathbb{R} \to \mathbb{R}_+$ is lower semi-continuous, non-decreasing on $\mathbb{R}_+$ with $h_s(0) = 0$, and such that $h_s(\sigma) \to +\infty$ as $|\sigma| \to +\infty$.

Assumption 3.2. *For each $l = 1, \dots, p$, $R_l : [0, \infty) \to \mathbb{R}_+$ is an increasing, lower semi-continuous and coercive function.*

Our generalized representer theorem states that Problem 5 admits a solution and that this solution is finite-dimensional. Compared to Theorem 2.1, observe that we do not require the loss to be bounded below, but only the complete objective function to be coercive, which broadens the applicability of the theorem. Obviously our theorem encompasses Theorem 2.1, because lower semi-continuous functions which are bounded below are automatically coercive.

Theorem 3.1. *Let L be a a lower semi-continuous loss function and regularizers R_l satisfying Assumption 3.2 and Ω_s satisfying Assumption 3.1. Then, under the assumption that the objective function is coercive, Problem 5 admits a solution $(g^\star, \mathscr{A}^\star)$ with the finite-dimensional representation*

$$g_l^\star(x) = \sum_{i=1}^{n} \gamma_{li}\, k_l^g(x, x_i), \qquad \mathscr{A}_s^\star = \sum_{i,j=1}^{n} [\mathbf{B}_s]_{ij}\, \phi_s^f(x_i)\phi_s^f(x_j)^\top \tag{6}$$

such that $f_{\mathscr{A}_s^\star}$ can be written as

$$f_{\mathscr{A}_s^\star}(x) = \sum_{i,j=1}^{n} [\mathbf{B}_s]_{ij}\, k_s^f(x_i, x)\, k_s^f(x_j, x) \tag{7}$$

for p vectors $\gamma_l \in \mathbb{R}^n$, $l = 1, \dots, p$ and q matrices $\mathbf{B}_s \in \mathbb{S}_+^n$, $s = 1, \dots, q$.

The proof is given in Section 6.2 and follows the main steps of representer theorems in the literature. We first show that the objective function does not increase after projection in the finite-dimensional subspace spanned by the data. Then, we show that if the minimum exists it must have bounded norm, and then conclude by showing that such a minimum exists. Remark that our proof differs from the scheme proposed by both Allain et al. (2025) and Vacher et al. (2024): in their work they first show that a minimum of the infinite-dimensional problem exists, and then use a block-wise argument for the representer theorem (fix all functions but one, then previous representer theorems apply, and iterate over the functions). We prefer here the projection argument because it delivers

existence and the representer form simultaneously.

The outcome of this representer theorem is a primal formulation with p vectors of size n and q PSD matrices of size $n \times n$ to learn. Clearly this primal formulation is easy to implement and solve using off-the-shelf solvers such as SCS (O'Donoghue 2021; O'Donoghue et al. 2023). But of practical interest is also the following equivalent finite-dimensional reparameterization based on the empirical feature map, which was proposed in Marteau-Ferey et al. (2020). Indeed, this alternative parameterization was shown to be more computationally efficient by Allain et al. (2025) while still relying on easy-to-use semi-definite program (SDP) solvers, but it also paves the way for kernel low-rank approximation techniques which we will discuss later. More precisely, define

$$\tilde{f}_{\mathbf{A}_s}(x) = \Phi_s(x)^\top \mathbf{A}_s \, \Phi_s(x) \tag{8}$$

for $s = 1, \dots, q$, where the matrix $\mathbf{A}_s$ is of size r_s, the rank of $\mathbf{K}_s^f$, since the empirical feature map $\Phi_s(x)$ is of size r_s. With these notations, the following proposition shows that we obtain the same solution if we optimize the PSD matrices $\mathbf{A}_1, \dots, \mathbf{A}_q$ instead of $\mathbf{B}_1, \dots, \mathbf{B}_q$.

Proposition 3.1. *Under the assumptions of Theorem 3.1, the following problem has at least one solution, which is unique if $g_l \mapsto R_l(\|g_l\|_{\mathscr{H}_l^g})$, Ω_s are strongly convex for all $1 \leq l \leq p$ and $1 \leq s \leq q$ and L is convex:*

$$\begin{aligned} \inf_{\substack{\gamma_1, \dots, \gamma_p \in \mathbb{R}^n \\ \mathbf{A}_1 \in \mathbb{S}_+^{r_1}, \dots, \mathbf{A}_q \in \mathbb{S}_+^{r_q}}} & L\left((k_l^g(x_i)^\top \gamma_l)_{1 \leq i \leq n,\, 1 \leq l \leq p},\, (\tilde{f}_{\mathbf{A}_s}(x_i))_{1 \leq i \leq n,\, 1 \leq s \leq q} \right) \\ & + \sum_{l=1}^{p} R_l\left(\sqrt{\gamma_l^\top \mathbf{K}_l^g \gamma_l} \right) + \sum_{s=1}^{q} \Omega_s(\mathbf{A}_s). \end{aligned} \tag{9}$$

Moreover, for any given solution $\mathbf{A}_1^\star \in \mathbb{S}_+^{r_1}, \dots, \mathbf{A}_q^\star \in \mathbb{S}_+^{r_q}$ of Problem 9, the functions $(\tilde{f}_{\mathbf{A}_s^\star})_{1 \leq s \leq q}$ are minimizers of Problem 5.

Proof. The proof is exactly the same as in Allain et al. (2026, Proposition A.6), since the new parameterization does not involve functions g_l. □

In case of rank-deficient kernels, where $r_s < n$, this reparameterization allows to limit the size of the optimization variables. As mentioned before, it was also shown to be more efficient from a numerical perspective in the extensive experiments of Allain et al. (2025). In practice we thus heavily recommend to solve Problem 9.

However, even with this reformulation, the finite-dimensional problem is still a semi-definite program, with unknown variables of size $n \times n$ or $r_s \times r_s$. In order to reduce the number of optimization variables and break free from the PSD constraints, we now propose to adapt the dual formulation proposed by Marteau-Ferey et al. (2020) for Theorem 2.1 to our generalized representer theorem, which holds when the loss function L is convex. We show that Fenchel duality (Borwein and Lewis 2006 Theorem 3.3.5) applies to Problem 9, when considering specific regularizers. Specifically, for RKHS functions we consider the ridge regularization

$$R_l(u) = \lambda_l^g u^2, \tag{10}$$

which satisfies Assumption 3.2. For kSoS functions, we consider the same regularizer as Marteau-Ferey et al. (2020), the elastic-net regularizer $\Omega_s(\mathscr{A}) = \lambda_{s1}^f \|\mathscr{A}\|_\star + \lambda_{s2}^f \|\mathscr{A}\|_F^2$, which satisfies Assumption 3.1 with functions $h_s(z) = \lambda_{s1}^f |z| + \lambda_{s2}^f z^2$.

Theorem 3.2. *Assume L is a lower semi-continuous proper convex function and R_l and Ω_s are as defined above with $\lambda_l^g, \lambda_{s2}^f > 0$ for all $l = 1, \dots, p$ and $s = 1, \dots, q$. Assume further that there exist*

$\gamma_l^0 \in \mathbb{R}^n$, $\mathbf{A}_s^0 \in \mathbb{S}_+^{r_s}$ such that L is continuous in $(k_l^g(x_i)^\top \gamma_l^0)_{1\leq i\leq n,\, 1\leq l\leq p}$ and $(\tilde{f}_{\mathbf{A}_s^0}(x_i))_{1\leq i\leq n,\, 1\leq s\leq q}$. Then Problem 9 has the following dual formulation:

$$\sup_{\substack{\alpha_1,\dots,\alpha_q\in\mathbb{R}^n \\ \beta_1,\dots,\beta_p\in\mathbb{R}^n}} -L^\star((-\beta_l)_{1\leq l\leq p},(-\alpha_s)_{1\leq s\leq q}) - \sum_{l=1}^{p}\frac{1}{4\lambda_l^g}\beta_l^\top \mathbf{K}_l^g \beta_l \\ - \sum_{s=1}^{q}\frac{1}{4\lambda_{s2}^f}\left\|\left[\mathbf{V}_s \mathrm{Diag}(\alpha_s)\mathbf{V}_s^\top - \lambda_{s1}^f \mathbf{I}_{r_s}\right]_+\right\|_F^2 \tag{11}$$

where $L^\star$ is the Fenchel conjugate of L and $[\mathbf{M}]_+$ denotes the positive part of a symmetric matrix $\mathbf{M} \in \mathbb{S}^r$, i.e. $[\mathbf{M}]_+ = \mathbf{Q}_+\mathbf{Q}^\top$ if $\mathbf{M}$ has the eigendecomposition $\mathbf{M} = \mathbf{Q}\Lambda\mathbf{Q}^\top$ and Λ_+ is a diagonal matrix with elements $[\Lambda_+]_{ii} = \max([\Lambda]_{ii}, 0)$ for $i = 1,\dots,r$. Moreover, if $\alpha_1^\star,\dots,\alpha_q^\star,\beta_1^\star,\dots,\beta_p^\star \in \mathbb{R}^n$ is a solution of Problem 11, a solution of Problem 5 can be retrieved as

$$g_l^\star(x) = \frac{1}{2\lambda_l^g} k_l^g(x)^\top \beta_l^\star, \tag{12}$$

$$f_{\mathscr{A}_s^\star}(x) = \tilde{f}_{\mathbf{A}_s^\star}(x),\ \mathbf{A}_s^\star = \frac{1}{2\lambda_{s2}^f}\left[\mathbf{V}_s \mathrm{Diag}(\alpha_s)\mathbf{V}_s^\top - \lambda_{s1}^f \mathbf{I}_{r_s}\right]_+ . \tag{13}$$

The proof can be found in Section 6.3 and relies on Theorem 3.3.5 from Borwein and Lewis (2006), for which we also derive the Fenchel conjugate of the regularization functions. Theorem 3.2 allows to transform the primal problem with q positive semi-definite constraints and $pn + qn^2$ variables, into an optimization problem with only $(p+q)n$ variables and no positive semi-definite constraint. If the dual formulation is differentiable, as in all the examples we study here, highly efficient off-the-shelf solvers can be used such as `L-BFGS` (Liu and Nocedal 1989) or accelerated gradient descent (`AGD`) methods (Barré et al. 2022; Truong and Nguyen 2021). They only require the gradient of the dual objective function: the gradient of the regularizer conjugates is available in closed-form, and the gradient of the conjugate loss function is tractable in many instances, see Section 6.4 for examples. When the loss function conjugate is not smooth or when it involves simple non-linear constraints, note that `AGD` methods can be adapted, such as proximal or projected variants.

This dual formulation, thanks to its linear scaling in the optimization variables, makes it possible to consider larger sample size problems than the primal, up to a few thousand samples in our experience. To scale even further, interestingly our statistical learning problem can also benefit from kernel low-rank approximation, significantly speeding up computation when the data exhibit a low-rank structure. Indeed, recall that Equation 4 gives the empirical feature map for a rank-deficient kernel matrix $\mathbf{K}_s^f$, while it simplifies to $\Phi_s(x) = \mathbf{V}_s^{-\top} k_s^f(x)$ when the kernel matrix is full rank, that is when $r_s = n$. The idea is that even in cases where the kernel matrix is inherently full rank, we can replace it with a low-rank approximation to accelerate computations by using Equation 4. To compute such low-rank kernel approximation, we can rely on the numerous methods available in the kernel literature, such as Nyström (Williams and Seeger 2000), Random Fourier Features (Rahimi and Recht 2007) or randomized SVD (Halko et al. 2011). Taking advantage of such low-rank approximation has major practical benefits. First, for the primal formulation given in Problem 9, as previously observed this reduces the complexity of the SDP problem from q PSD matrices all of size $n \times n$ to q matrices each of small size $r_s \times r_s$ for $1 \leq s \leq q$. Second, for the dual formulation, although the number of optimization variables remains the same, computing the positive part of a matrix actually now applies to matrices of size $r_s \times r_s$ for $1 \leq s \leq q$ instead of q matrices of size $n \times n$, thereby reducing the complexity of each dual iteration from $\mathcal{O}(n^3)$ to $\mathcal{O}(\max_{1\leq s\leq q}(r_s)^3)$, since the positive part involves an eigenvalue decomposition. This leads to significant computational savings in practice, with almost no approximation error if the low-rank approximations are carefully designed, as we illustrate in our

numerical experiments below. This strategy also proved successful in Vacher et al. (2024) and Allain et al. (2026). As future work, we plan to quantify the error incurred by this type of approximation in the kSoS framework, similarly to the bounds which have been obtained in the kernel literature, see e.g. Yang et al. (2012).

To summarize, in this section we introduced a new statistical learning problem involving both RKHS and kSoS functions, defined on potentially different functions spaces, extending Theorem 2.1. Following a reparametrization, we also showed that, for specific regularizers and a convex loss function, our primal problem admits a convex dual formulation, which reduces the number of optimization variables and removes any positive semi-definite constraints. Our results also easily accommodate for kernel low-rank approximations, which can help both the primal and dual formulation to scale to much larger datasets. In the next section, we finally give several illustrations of kSoS statistical learning problems.

4 Illustration of kSoS on statistical learning problems

In this section, we illustrate the relevance of kernel sum-of-squares statistical learning through several practical applications. First, in Section 4.1, we revisit three examples already introduced by Marteau-Ferey et al. (2020) that rely solely on unknown kSoS functions: heteroscedastic regression with a known mean, non-crossing multi-quantile regression with a known median, and density estimation. To ensure completeness and facilitate practical implementation, we explicitly derive the dual formulations for these examples, expanding upon the original paper of Marteau-Ferey et al. (2020). While the first two regression problems are treated sequentially, Section 4.2 extends this framework to joint learning scenarios. Specifically, we propose to jointly estimate the mean in the heteroscedastic regression model and the median in the multi-quantile regression model.

In all experiments, we exclusively use the Matérn 5/2 kernel, parameterized by lengthscales $\theta^{(\cdot)}$. Hyperparameter tuning is systematically performed using a 5-fold cross-validation scheme: this applies to kSoS-only and joint learning examples, but also to the pre-computed mean (via kernel ridge regression) and median (via kernel quantile regression) functions discussed in Section 4.1. We refer the reader to Section Section 7 for all the details and results of the cross-validation procedure on each task. In addition, all optimization problems are solved with a low-rank approximation for the kernel matrices $\mathbf{K}_s^f$. Specifically, we define $\mathbf{V}_s = \Sigma_s^{1/2}\mathbf{U}_s^\top$ where $\Sigma_s \in \mathbb{R}^{r_s \times r_s}$ is diagonal and $\mathbf{U}_s \in \mathbb{R}^{n \times r_s}$ such that $\mathbf{U}_s^\top \mathbf{U}_s = \mathbf{I}_{r_s}$ is semi-unitary obtained through the economy eigendecomposition $\mathbf{K}_s^f = \mathbf{U}_s \Sigma_s \mathbf{U}_s^\top$, and the rank is chosen in order to guarantee a reconstruction error less than 99.5%. In practice, this eigendecomposition is obtained through a singular value decomposition.

4.1 kSoS-only learning problems

In this section we focus ourselves on kSoS-only problems and give the detailed dual formulations for the three original examples considered by Marteau-Ferey et al. (2020).

4.1.1 Heteroscedastic regression with known mean

Assume that we have i.i.d. data $(x_i, y_i)_{i=1,\dots,n}$ from the heteroscedastic Gaussian noise model

$$y_i \sim \mathcal{N}(\mu(x_i), v(x_i))$$

where $\mu(\cdot)$ and $v(\cdot)$ are the mean and variance function, respectively. Up to constants, the negative log-likelihood writes:

$$L(\mu, v) = \sum_{i=1}^{n} \frac{(y_i - \mu(x_i))^2}{v(x_i)} + \sum_{i=1}^{n} \log(v(x_i)).$$

We place ourselves in the following setting: μ is considered to be a known function, potentially learned separately, and we model the variance function $v = 1/f_{\mathscr{A}}$ as the inverse of a kernel SoS function defined by a PSD operator $\mathscr{A} \in \mathcal{S}_+\left(\mathscr{H}^f\right)$ with kernel k^f and lengthscale θ^f. With this parameterization, the negative log-likelihood becomes convex in $\mathscr{A}$. Proposition 4.1 below gives the primal and dual formuations associated to learning the function $f_{\mathscr{A}} = 1/v$.

Proposition 4.1. *Assume that Ω is as in Equation 3 with $\lambda_2^f > 0$. For some known mean function $\mu(\cdot)$, unknown variance function $v(\cdot) > 0$ and denoting $t = (t_1, \dots, t_n)$ with $t_i = (y_i - \mu(x_i))^2$ the vector of squared residuals, the maximum likelihood estimator for heteroscedastic regression writes*

$$\inf_{\mathscr{A} \in \mathcal{S}_+(\mathscr{H}^f)} \sum_{i=1}^n t_i f_{\mathscr{A}}(x_i) - \sum_{i=1}^n \log(f_{\mathscr{A}}(x_i)) + \Omega(\mathscr{A})$$

with finite-dimensional equivalent:

$$\inf_{\mathbf{A} \in \mathbb{S}_+^r} \sum_{i=1}^n t_i \tilde{f}_{\mathbf{A}}(x_i) - \sum_{i=1}^n \log(\tilde{f}_{\mathbf{A}}(x_i)) + \Omega(\mathbf{A}).$$

The associated dual formulation is

$$\sup_{\substack{\alpha \in \mathbb{R}^n \\ \alpha + t > 0}} \quad n + \sum_{i=1}^n \log(\alpha_i + t_i) - \frac{1}{4\lambda_2^f} \|[\mathbf{V}\mathrm{Diag}(\alpha)\mathbf{V}^\top - \lambda_1^f \mathbf{I}_r]_+\|_F^2. \tag{14}$$

The proof can be found in Section 6.4. The primal formulation is derived directly from Theorem 2.1, or from Theorem 3.1 with $p = 0$ and $q = 1$. On the other hand, the dual formulation is obtained by computing the Fenchel conjugate of the convex and bounded below loss function $L(u_1, \dots, u_n) = \sum_{i=1}^n t_i u_i - \sum_{i=1}^n \log(u_i)$.

Let us then investigate this learning problem on the following synthetic dataset inspired by Gramacy and Lee (2009):

$$x_i \sim \mathcal{U}(-1, 1), \quad y_i = \mu(x_i) + \sigma(x_i)\epsilon_i,$$

where $\mu(x) = \omega(x)\mathbb{I}_{x \leq c} + (x - 9/10)\mathbb{I}_{x > c}$, $\omega(x) = \sin(\pi(2x + 1/5)) + 0.2\cos(4\pi(2x + 1/5))$, $c = 0.86$, $\sigma(x) = \sqrt{0.1 + 2x^2}$ and $\epsilon_i \sim \mathcal{N}(0, 1)$. Figure 1 shows the best model selected by cross-validation on a training sample of size $n = 1000$ (full lines) versus the true mean (dashed red line) and the true confidence bands at level 95% (dashed orange line). More precisely, we first solve a kernel ridge regression problem for the mean function (full red line), and then solve Equation 14 to estimate the variance function, from which we construct the estimated 95% confidence bands (full orange lines). We observe that the predicted intervals are able to recover the shape of the true heteroscedastic noise: narrow where the variance is small, in the center, and wide near the boundaries of the input domain, where the true variance is larger. For future reference, we also display the root mean squared error (RMSE) for both mean and variance functions computed on a test set of size 300.

4.1.2 Non-crossing multi-quantile regression with known median

Given an i.i.d. sample $(x_i, y_i)_{i=1,\dots,n}$ from a joint distribution P_{XY}, our objective is to estimate multiple quantiles of the conditional distribution $P(Y|X = x)$, i.e. estimate functions $q_\tau(x)$ where $P(Y > q_\tau(X)|X = x) = \tau$ for several $\tau \in (0, 1)$. For notational convenience, let us denote $\tau_{-1}, \dots, \tau_{-q_-} \in (0, 0.5)$ the quantile levels to be estimated that are less than 0.5 in decreasing order, $\tau_1, \dots, \tau_{q_+} \in (0.5, 1)$ those which are greater than 0.5 in increasing order, and $q_{0.5}$ the conditional median which is assumed to be known, once again possibly learned beforehand.

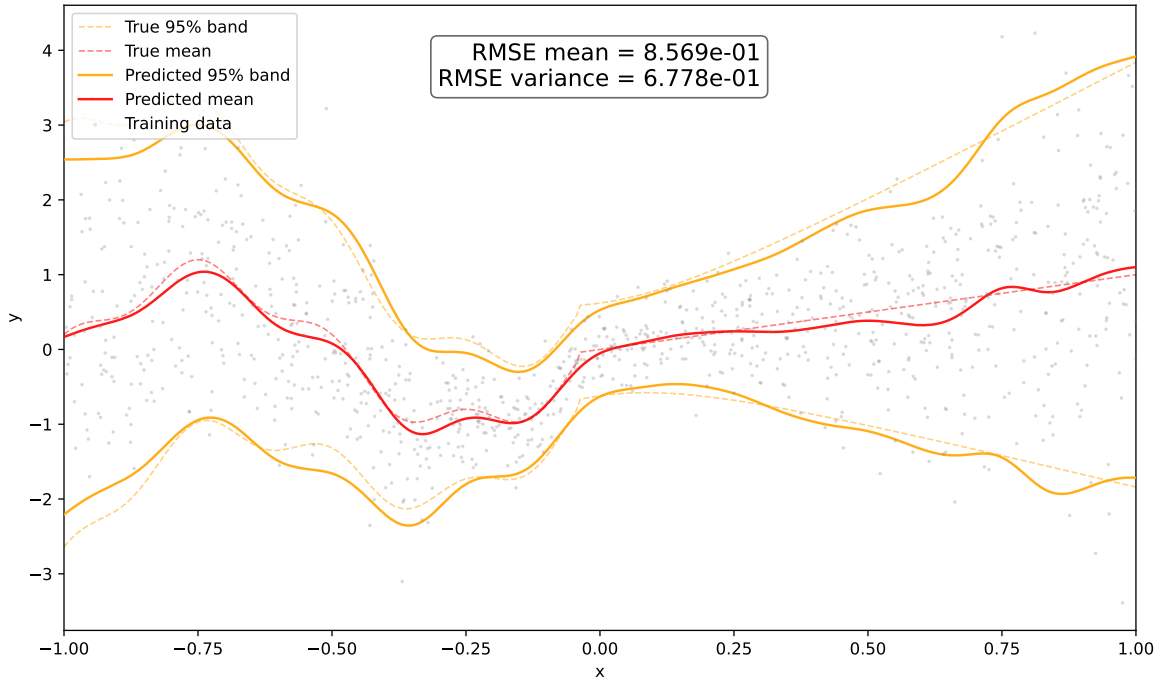

Figure 1: Heteroscedastic regression with known mean, trained with cross-validation on $n = 1000$ training samples (grey). Mean (red) and 95% confidence intervals (orange), true (dashed line) versus estimation (full line).

We focus here on the pinball loss $\rho_\tau(u) = \max(\tau u, (\tau-1)u)$: the statistical learning problem associated to multi-quantile regression thus writes

$$\inf_{\{q_{\tau_{-j}}\}_{j=1}^{q_-}, \{q_{\tau_k}\}_{k=1}^{q_+}} \sum_{j=1}^{q_-} \sum_{i=1}^{n} \rho_{\tau_{-j}}(y_i - q_{\tau_{-j}}(x_i)) + \sum_{k=1}^{q_+} \sum_{i=1}^{n} \rho_{\tau_k}(y_i - q_{\tau_k}(x_i)).$$

In order to enforce non-crossing among quantiles, i.e. to guarantee that $\forall x$, $q_\tau(x) \leq q_{\tau'}(x)$ if $\tau \leq \tau'$, we follow here the proposal of Marteau-Ferey et al. (2020) and model the quantiles as

$$q_{\tau_{-j}}(x) = q_{0.5}(x) - \sum_{j'=1}^{j} f_{\mathscr{A}_{-j'}}(x)$$
$$q_{\tau_k}(x) = q_{0.5}(x) + \sum_{k'=1}^{k} f_{\mathscr{A}_{k'}}(x)$$

for kSoS functions $f_{\mathscr{A}_{-j}}$ for $j = 1, \ldots, q_-$ and $f_{\mathscr{A}_k}$ for $k = 1, \ldots, q_+$. Thanks to these expressions writing as cumulative sums of non-negative functions, non-crossing is guaranteed everywhere by design. The following proposition gives the primal and dual formulations associated to this learning problem, with a known conditional median function.

Proposition 4.2. *Assume that Ω_{-j} and Ω_k are elastic-net regularizations as in Equation 3 with $\lambda^f_{-j2}, \lambda^f_{k2} > 0$. Denoting $t = (t_1, \ldots, t_n)$ with $t_i = y_i - q_{0.5}(x_i)$ the vector of residuals around the median,*

the non-crossing multi-quantile regression problem with known median writes

$$\inf_{\substack{\{\mathscr{A}_{-j}\}_{j=1}^{q_-} \in \prod_{j=1}^{q_-} \mathcal{S}_+\left(\mathscr{H}_{-j}^f\right) \\ \{\mathscr{A}_k\}_{k=1}^{q_+} \in \prod_{k=1}^{q_+} \mathcal{S}_+\left(\mathscr{H}_k^f\right)}} \sum_{j=1}^{q_-} \sum_{i=1}^{n} \rho_{\tau_{-j}}\left(t_i + \sum_{j'=1}^{j} f_{\mathscr{A}_{-j'}}(x_i)\right) + \sum_{j=1}^{q_-} \Omega_{-j}(\mathscr{A}_{-j}) + \sum_{k=1}^{q_+} \sum_{i=1}^{n} \rho_{\tau_k}\left(t_i - \sum_{k'=1}^{k} f_{\mathscr{A}_{k'}}(x_i)\right) + \sum_{k=1}^{q_+} \Omega_k(\mathscr{A}_k),$$

with finite-dimensional equivalent:

$$\inf_{\substack{\{\mathbf{A}_{-j}\}_{j=1}^{q_-} \in \prod_{j=1}^{q_-} \mathbb{S}_+^{r_{-j}} \\ \{\mathbf{A}_k\}_{k=1}^{q_+} \in \prod_{k=1}^{q_+} \mathbb{S}_+^{r_k}}} \sum_{j=1}^{q_-} \sum_{i=1}^{n} \rho_{\tau_{-j}}\left(t_i + \sum_{j'=1}^{j} \tilde{f}_{\mathbf{A}_{-j'}}(x_i)\right) + \sum_{j=1}^{q_-} \Omega_{-j}(\mathbf{A}_{-j}) + \sum_{k=1}^{q_+} \sum_{i=1}^{n} \rho_{\tau_k}\left(t_i - \sum_{k'=1}^{k} \tilde{f}_{\mathbf{A}_{k'}}(x_i)\right) + \sum_{k=1}^{q_+} \Omega_k(\mathbf{A}_k).$$

The associated dual formulation is

$$\begin{aligned} \sup_{\substack{\alpha_{-j} \in [\tau_{-j}-1, \tau_{-j}],\ j=1,\dots,q_- \\ \alpha_k \in [\tau_k - 1, \tau_k],\ k=1,\dots,q_+}} \quad & t^\top \left(\sum_{j=1}^{q_-} \alpha_{-j} + \sum_{k=1}^{q_+} \alpha_k\right) \\ & - \sum_{j=1}^{q_-} \frac{1}{4\lambda_{-j2}^f} \left\| \left[\mathbf{V}_{-j} \mathrm{Diag}\left(-\sum_{l=j}^{q_-} \alpha_{-l}\right) \mathbf{V}_{-j}^\top - \lambda_{-j1}^f \mathbf{I}_{r_{-j}} \right]_+ \right\|_F^2 \\ & - \sum_{k=1}^{q_+} \frac{1}{4\lambda_{k2}^f} \left\| \left[\mathbf{V}_k \mathrm{Diag}\left(\sum_{l=k}^{q_+} \alpha_l\right) \mathbf{V}_k^\top - \lambda_{k1}^f \mathbf{I}_{r_k} \right]_+ \right\|_F^2. \end{aligned} \tag{15}$$

The proof can be found in Section 6.4 and makes use of our generalized Theorem 3.1 with $p = 0$ and $q = q_- + q_+$ to derive the primal formulation. Remark that since we consider different RKHS for each kSoS function, we apply Theorem 3.1, even though the less general version of Allain et al. (2026, Theorem A.2) would have sufficed here. The dual formulation is obtained by deriving the Fenchel conjugate of the loss function, which writes as the composition of the pinball loss with a linear map. Consider now the following data-generating process inspired by Allain et al. (2026):

$$x_i \sim \mathcal{U}[0, 4\pi], \quad y_i = \sin(x_i) + \epsilon_i \left[\sigma_-(x_i)\, \mathbf{1}_{\{\epsilon_i < 0\}} + \sigma_+(x_i)\, \mathbf{1}_{\{\epsilon_i \geq 0\}} \right],$$

where $\sigma_-(x) = 0.5$, $\sigma_+(x) = 0.4(\sin(x) + 1) + 0.1$ and $\epsilon_i \sim \mathcal{N}(0, 1)$.

Figure 2 displays the best model selected by cross-validation when the training is performed sequentially, on the same $n = 1000$ training samples. First, we train a kernel quantile regression problem to target the true conditional median (dashed teal). Then, we solve Equation 15 for all other quantiles (dashed purple to yellow). In addition to the predictions on the entire feature space in the center, we also provide two zoomed-in views on the left and right to better investigate the estimated quantile functions. In the top row, we show the resulting estimations when we use operators defined on the same RKHS (thus with same lengthscales θ^f) for all quantiles. For comparison, in the bottom row, we allow different spaces for the quantiles below the median and those above, which we refer to as the "free" setting. First, observe that for both configurations, we have non-crossing quantiles, as expected. However, some quantiles are poorly estimated in the same lengthscale setting, such as the 0.4 level:

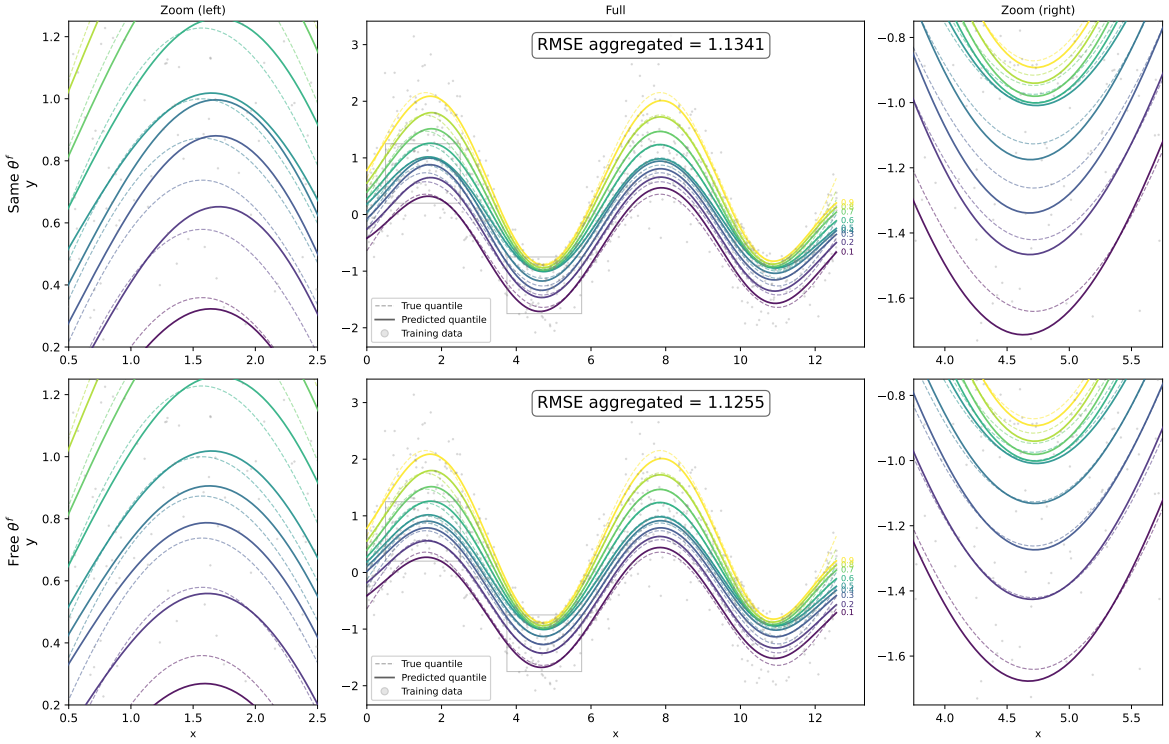

Figure 2: Non-crossing multi-quantiles regression with known median, trained with cross-validation on $n = 1000$ training samples (grey). True quantiles (dashed lines) versus estimation (full lines) with same lengthscales (top row) and free ones (bottom row). Left and right panels are zoomed-in views for the boxed areas in the center.

on the left zoomed-in view it is clear that it collapses on the median. This in turn highly degrades the quality of the lower level quantiles 0.3 and 0.2. In the bottom row, where we allow different lengthscales for lower and upper quantiles, this gives them more freedom to adapt to the data: this time we see that the 0.4 level quantile does not collide with the median anymore, and that the 0.3 and 0.2 level quantiles are in turn much better estimated. In the right zoomed-in view, we can also note that this correction, allowed by selecting different lengthscales across quantile levels, makes it possible to better reproduce the waves of the true signal. Indeed, all the quantiles below the median are better estimated than with the same lengthscale at the top. In conclusion, this illustration shows the benefits of allowing different spaces for kSoS functions: in the small data regime we can expect a more accurate estimation of the targets, which is also supported by the smaller RMSE displayed in the figure (aggregated across all quantiles and computed on a test set of size 300).

4.1.3 Density estimation

Finally, our goal in density estimation is to recover the unknown probability density function $p(x)$ from an i.i.d. sample $(x_i)_{i=1,\dots,n}$ drawn from $p(x)$. To achieve this, the model is trained by minimizing the empirical negative log-likelihood loss, which is written as:

$$L(p) = -\frac{1}{n}\sum_{i=1}^{n} \log(p(x_i)).$$

Marteau-Ferey et al. (2020) proposed to parameterize $p(x) = f_{\mathscr{A}}(x)\nu(x)$ as the product of a kSoS function defined by a PSD operator $\mathscr{A} \in \mathcal{S}_+\left(\mathscr{H}^f\right)$ with kernel k^f and lengthscale θ^f, and where ν is a reference probability density function. They also impose the additional constraint $\int_{\mathcal{X}} f_{\mathscr{A}}(x)\nu(x)dx = 1$. Unfortunately, the corresponding infinite-dimensional penalized kSoS problem does not fall in the setting of their representer theorem or ours. Indeed, their proof relies on the fact that, after projection on the data span, the regularizers must not increase and the evaluations at the data points x_i are invariant. But if the learning problem is constrained, the projection must be feasible for the proof to still be valid: this is not the case for the integral constraint above. Therefore, following Marteau-Ferey et al. (2020) we do not apply the representer theorem and simply consider the sub-optimal finite-dimensional approximation. While being sub-optimal, it allows to strictly enforce an integral equal to 1. Proposition 4.3 gives the finite-dimensional approximation and its associated dual formulation. Note that since $\nu(x_i)$ is a constant in the optimization problem, we drop ν from the objective function.

Proposition 4.3. *For some reference measure ν, the finite-dimensional maximum likelihood estimator for density estimation writes:*

$$\begin{aligned} \inf_{\mathbf{A}\in\mathbb{S}_+^r} \quad & -\sum_{i=1}^{n} \log(\tilde{f}_{\mathbf{A}}(x_i)) + \Omega(\mathbf{A}) \\ \text{s.t.} \quad & \int_{\mathcal{X}} \tilde{f}_{\mathbf{A}}(x)\nu(x)dx = 1. \end{aligned}$$

The associated dual formulation with Ω from Equation 3 is

$$\sup_{\substack{\theta\in\mathbb{R},\,\alpha\in\mathbb{R}^n \\ \alpha\geq 0}} \quad -\theta + n + \sum_{i=1}^{n} \log(\alpha_i) - \frac{1}{4\lambda_2^f}\|[\mathbf{V}\mathrm{Diag}(\alpha)\mathbf{V}^\top - \lambda_1^f \mathbf{I}_r - \theta\mathbf{W}_\nu]_+\|_F^2 \tag{16}$$

where $\mathbf{W}_\nu = (\mathbf{V}\mathbf{V}^\top)^{-1}\mathbf{V}\mathbf{M}_\nu\mathbf{V}^\top(\mathbf{V}\mathbf{V}^\top)^{-1}$ with

$$[\mathbf{M}_\nu]_{i,j} = \int_{\mathcal{X}} k^f(x, x_i)k^f(x, x_j)\nu(x)dx. \tag{17}$$

The proof for the dual formulation is given in Section 6.4, where it is obtained by considering the Lagrangian formulation with Lagrange multiplier θ to include the integral constraint into the dual.

For illustration purposes, we generate data following a mixture of two banana-shaped distributions:

$$p(x_1, x_2) = \frac{1}{2}\,\text{Banana}(x_1, x_2; -0.1, -0.4, 0.5) + \frac{1}{2}\,\text{Banana}(x_1, x_2; 0.3, 0.3, -0.5)$$

where $\text{Banana}(x_1, x_2; \mu, o, d)$ denotes the distribution with density proportional to

$$\exp\left(-\frac{(x_1-\mu)^2}{2\sigma_1^2}\right)\exp\left(-\frac{\left(x_2 - o - d\,(x_1-\mu)^2\right)^2}{2\sigma_2^2}\right)$$

with $\sigma_1 = 0.5$ and $\sigma_2 = 0.2$, which is inspired by the banana-shape distribution introduced in Haario et al. (1999).

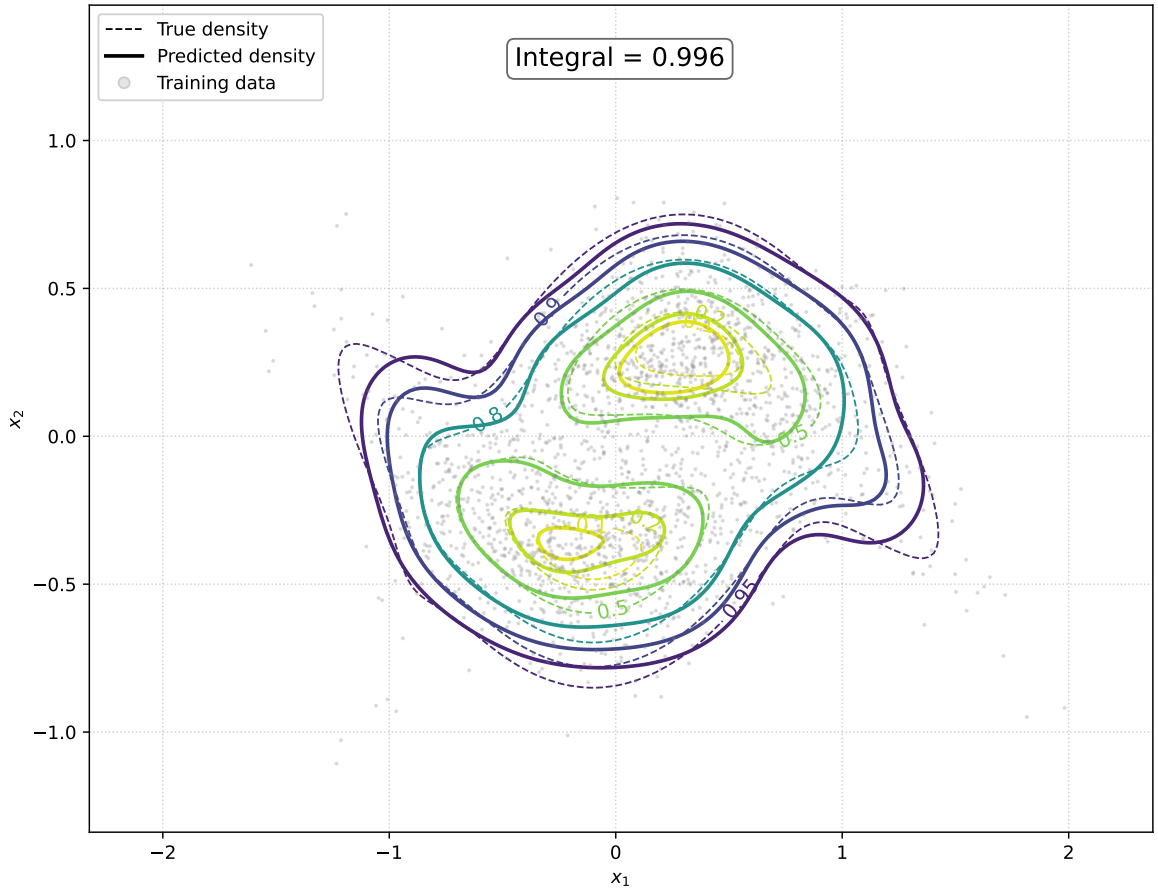

Figure 3: Density estimation, trained with cross-validation on $n = 2000$ training samples (grey). Isoquantile contour levels for true density (dashed lines) versus estimation (full lines).

Figure 3 shows the best model selected by cross-validation, where we specifically use an anisotropic kernel and take the reference density $\nu(x)$ equal to a standard Gaussian with $\mathbf{M}_\nu$ estimated by Monte-Carlo. The isoquantile contour levels of the true density are displayed in dashed lines, while the estimated ones (full lines) are obtained by solving Equation 16 on a dataset of size $n = 2000$. We can observe that they are remarkably well estimated: not only the small ones, from levels 0.1 in yellow to 0.5 in green, that constitute the two islands in the center, but also the high levels

such as 0.8 in seal and 0.9 in blue. Only the right and left 0.95 contour lines of the distribution are harder to learn. Overall, we see that the true density is accurately reconstructed and we can check numerically that the integral is indeed very close to 1 as displayed in the figure, up to numerical approximations. These approximations come from two sources: first, we estimate the displayed integral with Monte-Carlo on the density obtained after solving the problem and second, inside the optimization algorithm we estimate the matrix $\mathbf{M}_\nu$ in Equation 17 with another Monte-Carlo sample (although this approximation error may be removed since it can be computed in closed-form for specific choices of kernels and reference density). To summarize, these results show that, while being sub-optimal in theory, the finite-dimensional approximation works very well in practice.

4.2 Joint learning problems

Now, we introduce two new regression problems which leverage our generalized representer theorem for learning jointly RKHS and kSoS functions. These are natural generalizations of the regression problems considered in Section 4.1.

4.2.1 Heteroscedastic regression

We place ourselves in the same learning setting as in Section 4.1.1. However, this time the mean function μ is supposed to be unknown and must be estimated jointly with the kSoS function $f_{\mathscr{A}} = 1/\nu$. Since the loss function is not convex in $(\mu, \mathscr{A})$, we use the parameterization $g(x) = \mu(x)/\nu(x) = \mu(x) f_{\mathscr{A}}(x)$ for a function g in a RKHS $\mathscr{H}^g$ with kernel k^g and lengthscale θ^g, such that the loss function is now jointly convex in $(g, \mathscr{A})$. The primal and dual formulations for this joint learning problem are given in the following proposition.

Proposition 4.4. *Assume that R is as in Equation 10 and Ω is as in Equation 3 with $\lambda^g, \lambda_1^f, \lambda_2^f > 0$. The maximum likelihood estimator for heteroscedastic regression writes*

$$\inf_{g \in \mathscr{H}^g,\ \mathscr{A} \in \mathcal{S}_+(\mathscr{H}^f)} \quad \sum_{i=1}^n y_i^2 f_{\mathscr{A}}(x_i) + \sum_{i=1}^n \frac{g(x_i)^2}{f_{\mathscr{A}}(x_i)} - 2\sum_{i=1}^n y_i g(x_i) - \sum_{i=1}^n \log(f_{\mathscr{A}}(x_i)) + R(\|g\|_{\mathscr{H}^g}) + \Omega(\mathscr{A}),$$

with finite-dimensional equivalent:

$$\inf_{\gamma \in \mathbb{R}^n,\ \mathbf{A} \in \mathbb{S}_+^r} \quad \sum_{i=1}^n y_i^2 \tilde{f}_{\mathbf{A}}(x_i) + \sum_{i=1}^n \frac{(k^g(x_i)^\top \gamma)^2}{\tilde{f}_{\mathbf{A}}(x_i)} - 2\sum_{i=1}^n y_i k^g(x_i)^\top \gamma - \sum_{i=1}^n \log(\tilde{f}_{\mathbf{A}}(x_i)) + R(\sqrt{\gamma^\top \mathbf{K}^g \gamma}) + \Omega(\mathbf{A}).$$

The associated dual formulation is:

$$\sup_{\substack{\alpha,\ \beta \in \mathbb{R}^n \\ \beta \odot y + \alpha - \beta^{\odot 2}/4 > 0}} \quad n + \sum_{i=1}^n \log\left(\beta_i y_i + \alpha_i - \beta_i^2/4\right) - \frac{1}{4\lambda^g} \beta^\top K^g \beta - \frac{1}{4\lambda_2^f} \|[\mathbf{V}\mathrm{Diag}(\alpha)\mathbf{V}^\top - \lambda_1^f \mathbf{I}_r]_+\|_F^2. \tag{18}$$

The proof is given in Section 6.4. Theorem 3.1 directly applies with $p = 1$ and $q = 1$ to obtain the primal formulation, and the dual formulation is once again obtained by computing the Fenchel conjugate of the loss function.

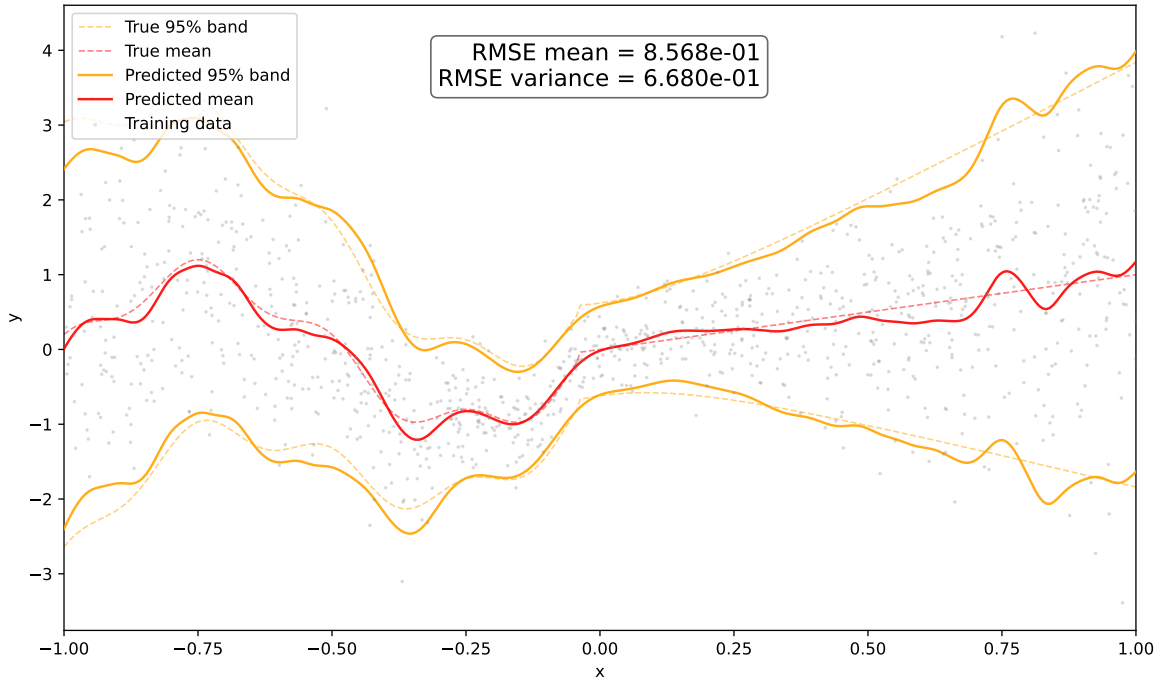

Figure 4: Heteroscedastic regression trained with cross-validation on $n = 1000$ training samples (grey). Mean (red) and 95% confidence intervals (orange), true (dashed line) versus estimation (full line).

Figure 4 shows the best model after cross-validation. We solve the joint learning problem defined in Equation 18 on the exact same dataset as in Section 4.1.1, to estimate the mean and variance functions. Interestingly, when compared to the sequential estimation in Figure 1, the joint mean and confidence intervals are closer to the target, especially in the center. In the joint setting, the estimation becomes slightly rough on the right, but this is largely compensated by a better prediction accuracy in the center and in the left of the figure. Evaluating the RMSE for both functions on a test set of size 300, these improvements translate both in smaller mean and variance RMSE compared to the sequential estimation in Section 4.1.1. This result shows that jointly learning the mean and variance functions may be beneficial in practice.

4.2.2 Non-crossing multi-quantile regression

We consider here the same formalism as in the quantile regression example in Section 4.1.2. However, this time the conditional median function $q_{0.5}$ is supposed to be unknown, and we parameterize it with a function g in a RKHS $\mathscr{H}^g$ with kernel k^g and lengthscale θ^g. The primal and dual formulations are given in the following proposition.

Proposition 4.5. *Assume that R is as in Equation 10 and Ω_{-j} and Ω_k as in Equation 3 with*

$\lambda^g, \lambda^f_{-j2}, \lambda^f_{k2} > 0$. *The non-crossing multi-quantile regression problem writes*

$$\begin{aligned}\inf_{\substack{g\in\mathscr{H}^g\\ \{\mathscr{A}_{-j}\}_{j=1}^{q_-}\in\prod_{j=1}^{q_-}\mathscr{S}_+\left(\mathscr{H}^f_{-j}\right)\\ \{\mathscr{A}_k\}_{k=1}^{q_+}\in\prod_{k=1}^{q_+}\mathscr{S}_+\left(\mathscr{H}^f_k\right)}} & \sum_{i=1}^n \rho_{\tau_{0.5}}(y_i - g(x_i)) + R(\|g\|_{\mathscr{H}^g}) \\ & + \sum_{j=1}^{q_-}\sum_{i=1}^n \rho_{\tau_{-j}}\left(y_i - g(x_i) + \sum_{j'=1}^{j} f_{\mathscr{A}_{-j'}}(x_i)\right) + \sum_{j=1}^{q_-}\Omega_{-j}(\mathscr{A}_{-j}) \\ & + \sum_{k=1}^{q_+}\sum_{i=1}^n \rho_{\tau_k}\left(y_i - g(x_i) - \sum_{k'=1}^{k} f_{\mathscr{A}_{k'}}(x)\right) + \sum_{k=1}^{q_+}\Omega_k(\mathscr{A}_k),\end{aligned}$$

with finite-dimensional equivalent:

$$\begin{aligned}\inf_{\substack{\gamma\in\mathbb{R}^n\\ \{\mathbf{A}_{-j}\}_{j=1}^{q_-}\in\prod_{j=1}^{q_-}\mathbb{S}_+^{r_{-j}}\\ \{\mathbf{A}_k\}_{k=1}^{q_+}\in\prod_{k=1}^{q_+}\mathbb{S}_+^{r_k}}} & \sum_{i=1}^n \rho_{\tau_{0.5}}(y_i - k^g(x_i)^\top\gamma) + R(\sqrt{\gamma^\top\mathbf{K}^g\gamma}) \\ & + \sum_{j=1}^{q_-}\sum_{i=1}^n \rho_{\tau_{-j}}\left(y_i - k^g(x_i)^\top\gamma + \sum_{j'=1}^{j}\tilde{f}_{\mathbf{A}_{-j'}}(x_i)\right) + \sum_{j=1}^{q_-}\Omega_{-j}(\mathbf{A}_{-j}) \\ & + \sum_{k=1}^{q_+}\sum_{i=1}^n \rho_{\tau_k}\left(y_i - k^g(x_i)^\top\gamma - \sum_{k'=1}^{k}\tilde{f}_{\mathbf{A}_{k'}}(x_i)\right) + \sum_{k=1}^{q_+}\Omega_k(\mathbf{A}_k).\end{aligned}$$

The associated dual formulation is:

$$\begin{aligned}\sup_{\substack{\beta\in[-0.5,0.5]^n\\ \alpha_{-j}\in[\tau_{-j}-1,\tau_{-j}]^n,\ j=1,\dots,q_-\\ \alpha_k\in[\tau_k-1,\tau_k]^n,\ k=1,\dots,q_+}} & y^\top\left(\beta + \sum_{j=1}^{q_-}\alpha_{-j} + \sum_{k=1}^{q_+}\alpha_k\right) \\ & - \frac{1}{4\lambda^g}\left(\beta + \sum_{j=1}^{q_-}\alpha_{-j} + \sum_{k=1}^{q_+}\alpha_k\right)^\top K^g\left(\beta + \sum_{j=1}^{q_-}\alpha_{-j} + \sum_{k=1}^{q_+}\alpha_k\right) \\ & - \sum_{j=1}^{q_-}\frac{1}{4\lambda^f_{-j2}}\left\|\left[\mathbf{V}_{-j}\mathrm{Diag}\left(-\sum_{l=j}^{q_-}\alpha_{-l}\right)\mathbf{V}_{-j}^\top - \lambda^f_{-j1}\mathbf{I}_{r_{-j}}\right]_+\right\|_F^2 \\ & - \sum_{k=1}^{q_+}\frac{1}{4\lambda^f_{k2}}\left\|\left[\mathbf{V}_k\mathrm{Diag}\left(\sum_{l=k}^{q_+}\alpha_l\right)\mathbf{V}_k^\top - \lambda^f_{k1}\mathbf{I}_{r_k}\right]_+\right\|_F^2.\end{aligned} \tag{19}$$

The proof can be found in Section 6.4. Our Theorem 3.1 directly applies with $p = 1$ and $q = q_- + q_+$ to recover the primal formulation, and the dual formulation is obtained by computing the Fenchel conjugate of the loss function.

Figure 5 shows the best models selected by cross-validation for quantile levels from 0.1 to 0.9. In the top row, we show estimated quantiles in a sequential manner with free lengthscales for upper and lower quantiles, as done in Section 4.1.2. In the bottom row, the quantiles are jointly estimated with the conditional median function by solving Equation 19. In the top row, we observe a good estimation overall: however, as shown in the left and right zoomed-in views, some quantiles are

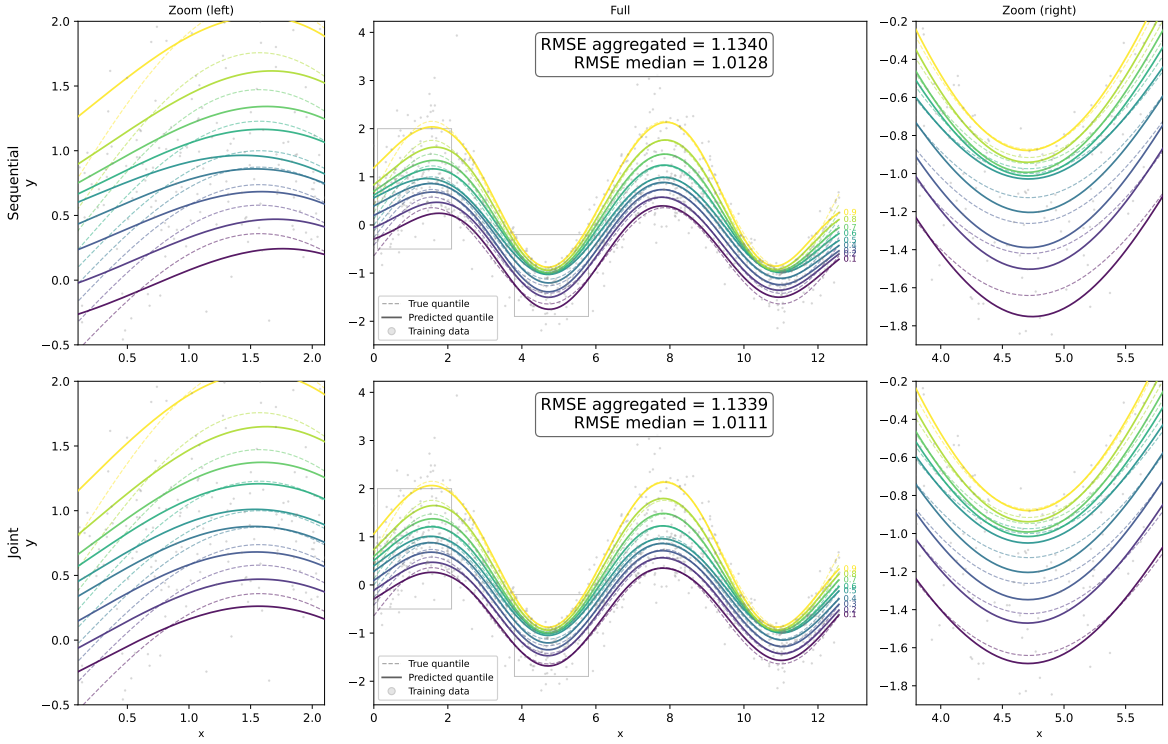

Figure 5: Non-crossing multi-quantiles regression trained with cross-validation on $n = 1000$ training samples (grey). True quantiles (dashed lines) versus estimation (full lines) with free lengthscales and sequential (top row) or joint (bottom row) learning problem. Left and right panels are zoomed-in views for the boxed areas in the center.

not estimated accurately in some areas. Looking at the bottom row, we see that jointly learning the median allows quantile levels to better adapt to the data and compensante some errors that were impossible to resolve because of the frozen median. The two zoomed-in views and the smaller RMSE clearly show that the joint learning problem better estimates both the median and the quantiles: we thus reach the same conclusion as in the heteroscedastic regression problem.

5 Conclusion

Non-negative functions arise frequently in practice, from physical systems to statistics or learning in general. To capture them effectively, Marteau-Ferey et al. (2020) introduced the kernel sum-of-squares framework, combining the rigorous non-negativity of sum-of-squares models with the expressiveness of kernel methods. Their framework originally assumed kSoS-only problems and shared RKHS for all functions. In this work, we extend this seminal work by introducing a generalized learning framework that jointly estimates unconstrained RKHS functions and kSoS functions, each with distinct spaces. Our main theoretical contribution is a generalized representer theorem that guarantees a finite-dimensional solution for a common class of regularization functions. Furthermore, for specific penalties often used in applications, such as ridge for RKHS functions and elastic-net for kSoS functions, we derive a convex dual formulation. Significantly, this dual eliminates positive semi-definite constraints from the optimization problems while reducing the number of optimization variables, allowing the framework to scale to large datasets. Finally, we state all our results in the rank-deficient setting, enabling further scaling with kernel low-rank approximation techniques. We illustrated this framework on multiple examples: in addition to deriving the missing dual formulations for previous kSoS-only examples, we demonstrated that the flexibility in the function space choice may be beneficial in practice, as well as the joint learning approach.

The kernel sum-of-squares framework is thus a highly promising tool for modelling non-negative functions. By extending it to accommodate broader learning problems and providing accessible dual formulations, we hope to facilitate its adoption in practical applications. From a theoretical perspective, future work will focus on establishing convergence rates and deriving bounds for low-rank kSoS approximations. On the practical side, we are working on the development of a user-friendly Python package dedicated to kSoS to further promote them.

References

Allain, Louis, Sébastien Da Veiga, and Brian Staber. 2025. "Scalable and Adaptive Prediction Bands with Kernel Sum-of-Squares." In *Advances in Neural Information Processing Systems*, edited by D. Belgrave, C. Zhang, H. Lin, et al., vol. 38. Curran Associates, Inc.https://proceedings.neurips.cc/paper_files/paper/2025/file/bff09ce4b210b185a265c9bcd58048bb-Paper-Conference.pdf .

Allain, Louis, Sébastien Da Veiga, and Brian Staber. 2026. *Asymmetric Conformal Prediction with Penalized Kernel Sum-of-Squares.* https://arxiv.org/abs/2601.22834.

Bach, Francis, Elisabetta Cornacchia, Luca Pesce, and Giovanni Piccioli. 2024. *Theory and Applications of the Sum-of-Squares Technique.* https://arxiv.org/abs/2306.16255.

Barré, Mathieu, Adrien Taylor, and Francis Bach. 2022. "A Note on Approximate Accelerated Forward-Backward Methods with Absolute and Relative Errors, and Possibly Strongly Convex Objectives." *Open Journal of Mathematical Optimization* 3: 1–15. https://doi.org/10.5802/ojmo.12.

Borwein, Jonathan, and Adrian Lewis. 2006. *Convex Analysis and Nonlinear Optimization: Theory*

and Examples. Springer. https://doi.org/https://doi.org/10.1007/978-0-387-31256-9.

Gramacy, Robert B., and Herbert K. H. Lee. 2009. "Adaptive Design and Analysis of Supercomputer Experiments." *Technometrics* 51 (2): 130–45. https://doi.org/10.1198/tech.2009.0015.

Haario, Heikki, Eero Saksman, and Johanna Tamminen. 1999. "Adaptive Proposal Distribution for Random Walk Metropolis Algorithm." *Computational Statistics* 14 (3): 375–95.

Halko, Nathan, Per-Gunnar Martinsson, and Joel A Tropp. 2011. "Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions." *SIAM Review* 53 (2): 217–88.

Jaini, Priyank, Kira A Selby, and Yaoliang Yu. 2019. "Sum-of-Squares Polynomial Flow." *International Conference on Machine Learning*, 3009–18.

Lasserre, Jean B. 2001. "Global Optimization with Polynomials and the Problem of Moments." *SIAM Journal on Optimization* 11 (3): 796–817. https://doi.org/10.1137/S1052623400366802.

Lasserre, Jean-Bernard. 2009. *Moments, Positive Polynomials and Their Applications*. Imperial College Press.

Liu, Dong C, and Jorge Nocedal. 1989. "On the Limited Memory BFGS Method for Large Scale Optimization." *Mathematical Programming* 45 (1): 503–28.

Marteau-Ferey, Ulysse, Francis Bach, and Alessandro Rudi. 2020. "Non-Parametric Models for Non-Negative Functions." In *Advances in Neural Information Processing Systems*, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, vol. 33. Curran Associates, Inc.https://proceedings.neurips.cc/paper_files/paper/2020/file/968b15768f3d19770471e9436d97913c-Paper.pdf .

Mohri, M., A. Rostamizadeh, and A. Talwalkar. 2012. *Foundations of Machine Learning*. Adaptive Computation and Machine Learning Series. MIT Press. https://books.google.fr/books?id=maz6AQAAQBAJ.

Mula, Olga, and Anthony Nouy. 2024. "Moment-SoS Methods for Optimal Transport Problems: O. Mula, a. Nouy." *Numerische Mathematik* 156 (4): 1541–78.

Muzellec, Boris, Francis Bach, and Alessandro Rudi. 2022. *Learning PSD-Valued Functions Using Kernel Sums-of-Squares*. https://arxiv.org/abs/2111.11306.

O'Donoghue, Brendan. 2021. "Operator Splitting for a Homogeneous Embedding of the Linear Complementarity Problem." *SIAM Journal on Optimization* 31: 1999–2023.

O'Donoghue, Brendan, Eric Chu, Neal Parikh, and Stephen Boyd. 2023. *SCS: Splitting Conic Solver, Version 3.2.7*.

Parrilo, Pablo. 2000. "Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization." *PhD Thesis*, August. https://www.mit.edu/~parrilo/pubs/files/thesis.pdf.

Rahimi, Ali, and Benjamin Recht. 2007. "Random Features for Large-Scale Kernel Machines." In

Advances in Neural Information Processing Systems, edited by J. Platt, D. Koller, Y. Singer, and S. Roweis, vol. 20. Curran Associates, Inc.

Rockafellar, Ralph Tyrell. 1970. *Convex Analysis*. Princeton University Press. https://doi.org/doi:10.1515/9781400873173.

Rudi, Alessandro, Ulysse Marteau-Ferey, and Francis Bach. 2025. "Finding Global Minima via Kernel Approximations." *Mathematical Programming* 209 (1): 703–84. https://doi.org/10.1007/s10107-024-02081-4.

Schölkopf, Bernhard, and Alexander J. Smola. 2002. *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning. MIT Press. http://www.worldcat.org/oclc/48970254.

Steinwart, Ingo, and Andreas Christmann. 2008. *Support Vector Machines*. Information Science and Statistics. Springer.

Truong, Truong T., and Huy T. Nguyen. 2021. "Backtracking Gradient Descent Method and Some Applications in Large Scale Optimisation. Part 2: Algorithms and Experiments." *Applied Mathematics & Optimization* 84: 2557–86. https://doi.org/10.1007/s00245-020-09718-8.

Vacher, Adrien, Boris Muzellec, Francis Bach, François-Xavier Vialard, and Alessandro Rudi. 2024. "Optimal Estimation of Smooth Transport Maps with Kernel SoS." *SIAM Journal on Mathematics of Data Science* 6 (2): 311–42. https://doi.org/10.1137/22M1528847.

Williams, Christopher, and Matthias Seeger. 2000. "Using the Nyström Method to Speed up Kernel Machines." In *Advances in Neural Information Processing Systems*, edited by T. Leen, T. Dietterich, and V. Tresp, vol. 13. MIT Press.

Yang, Tianbao, Yu-Feng Li, Mehrdad Mahdavi, Rong Jin, and Zhi-Hua Zhou. 2012. "Nyström Method Vs Random Fourier Features: A Theoretical and Empirical Comparison." *Advances in Neural Information Processing Systems* 25.

6 Proofs

6.1 Notations

We first recall the notations already introduced in Section 3 which are necessary for the proofs. For our learning problem, we consider a collection of p RKHS $\mathscr{H}_l^g$, $l = 1, \dots, p$ and q RKHS $\mathscr{H}_s^f$, $s = 1, \dots, q$. To each RKHS $\mathscr{H}_l^g$ and $\mathscr{H}_s^f$ is associated a unique kernel k_l^g and k_s^f with feature map ϕ_l^g and ϕ_s^f, respectively. For a sample $x_1, \dots, x_n$ of features, we denote $\mathbf{K}_l^g$ the kernel matrix with elements $[\mathbf{K}_l^g]_{ij} = k_l^g(x_i, x_j)$ and $k_l^g(x)$ the column vector defined by

$$k_l^g(x) = (k_l^g(x, x_1), \dots, k_l^g(x, x_n))^\top$$

for any $X \in \mathscr{X}$. Similarly, we introduce $\mathbf{K}_s^f$, $k_s^f(x)$ and further consider a decomposition $\mathbf{K}_s^f = \mathbf{V}_s^\top \mathbf{V}_s$ with $\mathbf{V}_s \in \mathbb{R}^{r_s \times n}$ where r_s is the rank of $\mathbf{K}_s^f$, and the empirical feature map $\Phi_s(x) = (\mathbf{V}_s \mathbf{V}_s^\top)^{-1} \mathbf{V}_s k_s^f(x)$. If $r_s = n$, such decomposition can be obtained with a Cholesky decomposition, while if $r_s < n$ we can consider $\mathbf{V}_s = \Sigma_s^{1/2} \mathbf{U}_s^\top$ where $\Sigma_s \in \mathbb{R}^{r_s \times r_s}$ is diagonal and $\mathbf{U}_s \in \mathbb{R}^{n \times r_s}$ such that $\mathbf{U}_s^\top \mathbf{U}_s = \mathbf{I}_{r_s}$ is semi-unitary obtained through the economy eigendecomposition $\mathbf{K}_s^f = \mathbf{U}_s \Sigma_s \mathbf{U}_s^\top$.

Next, for a Hilbert space $\mathscr{H}$ we write $\mathcal{S}(\mathscr{H})$ the set of bounded Hermitian linear operators from $\mathscr{H}$ to $\mathscr{H}$ and $\mathcal{S}_+(\mathscr{H})$ those that are positive semi-definite. We also write $\mathbb{S}^r := \mathbb{S}(\mathbb{R}^{r\times r})$ the set of real and symmetric matrices of size r, and $\mathbb{S}^r_+ := \mathbb{S}_+(\mathbb{R}^{r\times r})$ the set of real, symmetric and positive semi-definite matrices of size r.

For convenience, we denote

- The p *unknown* real-valued functions $g_l : \mathcal{X} \to \mathbb{R}$ in a RKHS $\mathscr{H}_l^g$ for $l = 1, \dots, p$
- The q *unknown* kernel sum-of-squares (kSoS) functions $f_{\mathscr{A}_s} : \mathcal{X} \to \mathbb{R}^+$ parameterized by a positive semi-definite operator $\mathscr{A}_s \in \mathcal{S}_+(\mathscr{H}_s^f)$ defined on a RKHS $\mathscr{H}_s^f$ with $f_{\mathscr{A}_s}(x) = \langle \phi_s^f(x), \mathscr{A}_s \phi_s^f(x) \rangle_{\mathscr{H}_s^f}$ for $s = 1, \dots, q$
- The collection of RKHS functions $g := (g_1, \dots, g_p) \in \mathscr{K}^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g) := \mathscr{H}_1^g \times \cdots \times \mathscr{H}_p^g$
- The collection of operators $\mathscr{A} := (\mathscr{A}_1, \dots, \mathscr{A}_q) \in \mathscr{K}^f(\mathscr{H}_1^f, \dots, \mathscr{H}_q^f) := \mathcal{S}_+(\mathscr{H}_1^f) \times \cdots \times \mathcal{S}_+(\mathscr{H}_q^f)$
- The collection of RKHS function evaluations $g(x) := (g_1(x), \dots, g_p(x)) \in \mathbb{R}^p, \ \forall x \in \mathcal{X}$
- The collection of kSoS function evaluations $f_{\mathscr{A}}(x) := f_{\mathscr{A}_1,\dots,\mathscr{A}_q}(x) = \left(f_{\mathscr{A}_1}(x), \dots, f_{\mathscr{A}_q}(x)\right) \in \mathbb{R}^q_+, \ \forall x \in \mathcal{X}$

Finally, we consider specific regularizers $R_l\left(\|g_l\|_{\mathscr{H}_l^g}\right)$ for RKHS functions and $\Omega_s(\mathscr{A}_s)$ for kSoS functions which satisfy Assumption 3.2 and Assumption 3.1, respectively.

6.2 Representer theorem

In this section, to prove Theorem 3.1, we closely follow the proof of Marteau-Ferey et al. (2020, sec. B.3) and Allain et al. (2026, Theorem A.2) and it is built in two parts. First, Proposition 6.1 below shows that the solution of the infinite-dimensional problem of Equation 5 lies in a space of finite-dimension. Then, we conclude by showing that this solution can be written as in Equation 6: this directly follows from (Allain et al. 2026 Lemma A.5) and classical representation results for functions in an RKHS, see Steinwart and Christmann (2008, Theorem 5.5), Schölkopf and Smola (2002, Theorem 4.2) or Mohri et al. (2012, Theorem 6.11). Before stating Proposition 6.1, we need additional notations for the spaces associated with the RKHS functions and the kSoS ones:

- For every Hilbert space $\mathscr{H}_l^g$, we write $\mathscr{H}_{nl}^g$ for the finite-dimensional subspace of $\mathscr{H}_l^g$ generated by $\{(\phi_l^g(x_i))_{1\le i\le n}\}$, and define P_{nl} the orthogonal projection onto $\mathscr{H}_{nl}^g$ such that

$$P_{nl} \in \mathcal{S}(\mathscr{H}_l^g), \quad P_{nl}^2 = P_{nl}, \quad \text{range}(P_{nl}) = \mathscr{H}_{nl}^g.$$

  For $g = \left(g_1, \dots, g_p\right) \in \mathscr{K}^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g)$, we define its projection $g_\parallel := (P_{n1}g_1, \dots, P_{np}g_p)$ onto $\mathscr{K}_n^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g) := \left\{g_\parallel : \ g \in \mathscr{K}^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g)\right\}$ with $g = g_\parallel + g_\perp$. Observe that we have the inclusion $\mathscr{K}_n^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g) \subset \mathscr{K}^g(\mathscr{H}_1^g, \dots, \mathscr{H}_p^g)$.

- Similarly for every Hilbert space $\mathscr{H}_s^f$, we write $\mathscr{H}_{ns}^f$ for the finite-dimensional subspace of $\mathscr{H}_s^f$ generated by $\{(\phi_s^f(x_i))_{1\le i\le n}\}$, and define Π_{ns} the orthogonal projection onto $\mathscr{H}_{ns}^f$ such that

$$\Pi_{ns} \in \mathcal{S}(\mathscr{H}_s^f), \quad \Pi_{ns}^2 = \Pi_{ns}, \quad \text{range}(\Pi_{ns}) = \mathscr{H}_{ns}^f.$$

  We further denote

$$\mathscr{K}_n^f(\mathscr{H}_1^f, \dots, \mathscr{H}_q^f) := \left\{\left(\Pi_{n1}\mathscr{A}_1\Pi_{n1}, \dots, \Pi_{nq}\mathscr{A}_q\Pi_{nq}\right) : \ \left(\mathscr{A}_1, \dots, \mathscr{A}_q\right) \in \mathscr{K}^f\left(\mathscr{H}_1^f, \dots, \mathscr{H}_q^f\right)\right\}$$

and since $\Pi_{ns}\mathscr{A}_s\Pi_{ns} \in \mathcal{S}_+(\mathscr{H}_s^f)$, $\forall s = 1,\dots,q$, we have the inclusion $\mathscr{K}_n^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f) \subset \mathscr{K}^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)$.

Finally, we denote the objective function

$$J(g,\mathscr{A}) := L\left(g(x_1),\dots,g(x_n), f_{\mathscr{A}}(x_1),\dots,f_{\mathscr{A}}(x_n)\right) + \sum_{l=1}^{p} R_l\left(\|g_l\|_{\mathscr{H}_l^g}\right) + \sum_{s=1}^{q} \Omega_s(\mathscr{A}_s).$$

Proposition 6.1. *Let L be a a lower semi-continuous loss function and R_l, Ω_s regularizers that satisfy Assumption 3.2 and Assumption 3.1 respectively. Under the assumption that $J(g,\mathscr{A})$ is coercive, the problem*

$$\inf_{\substack{g\in\mathscr{K}^g(\mathscr{H}_1^g,\dots,\mathscr{H}_p^g)\\ \mathscr{A}\in\mathscr{K}^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)}} L\left(g(x_1),\dots,g(x_n), f_{\mathscr{A}}(x_1),\dots,f_{\mathscr{A}}(x_n)\right) + \sum_{l=1}^{p} R_l\left(\|g_l\|_{\mathscr{H}_l^g}\right) + \sum_{s=1}^{q} \Omega_s(\mathscr{A}_s)$$

admits a solution $(g^\star,\mathscr{A}^\star) \in \mathscr{K}_n^g(\mathscr{H}_1^g,\dots,\mathscr{H}_p^g) \times \mathscr{K}_n^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)$.

Proof. The proof follows the same three main steps as in Allain et al. (2026, Proposition A.4), but that we carefully adapt to handle the additional RKHS functions:

- We first show that the objective function does not increase when restricting to the finite-dimensional space, meaning that

$$\inf_{\substack{g\in\mathscr{K}^g(\mathscr{H}_1^g,\dots,\mathscr{H}_p^g)\\ \mathscr{A}\in\mathscr{K}^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)}} J(g,\mathscr{A}) = \inf_{\substack{g\in\mathscr{K}_n^g(\mathscr{H}_1^g,\dots,\mathscr{H}_p^g)\\ \mathscr{A}\in\mathscr{K}_n^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)}} J(g,\mathscr{A}).$$

- Second, we show that if the minimum exists, then it has bounded norm.
- Finally, we show that such a minimum exists.

First step.

As for the kSoS part of the learning problem, Allain et al. (2026) show that the objective function does not increase when projecting the operator $\mathscr{A}$ to $\mathscr{K}_n^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)$ under Assumption 3.1. In addition, following the classical representer theorem proof (Steinwart and Christmann 2008 Theorem 5.5), (Schölkopf and Smola 2002 Theorem 4.2) or (Mohri et al. 2012 Theorem 6.11), we have $g(x_i) = g_{\|}(x_i)$, $\forall i = 1,\dots,n$ and $\|g_l\|_{\mathscr{H}_l^g} = \sqrt{\|g_{l\|}\|^2_{\mathscr{H}_l^g} + \|g_{l\perp}\|^2_{\mathscr{H}_l^g}} \geq \|g_{l\|}\|_{\mathscr{H}_l^g}, \forall l = 1,\dots,p$. The first step then follows since all R_l are increasing functions by Assumption 3.2.

Second step.

The key idea is to show that we can replace $\mathscr{K}_n^g(\mathscr{H}_1^g,\dots,\mathscr{H}_p^g) \times \mathscr{K}_n^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)$ with

$$\mathscr{K}_{R_0} := \left\{(\tilde{g},\tilde{\mathscr{A}}) \in \mathscr{K}^g(\mathscr{H}_{n1}^g,\dots,\mathscr{H}_{np}^g) \times \mathscr{K}^f(\mathscr{H}_{n1}^f,\dots,\mathscr{H}_{nq}^f) : \ \max\left(\max_{1\leq l\leq p}\|\tilde{g}_l\|_{\mathscr{H}_{nl}^g}, \max_{1\leq s\leq q}\|\tilde{\mathscr{A}}_s\|_F\right) \leq R_0\right\}$$

for some constant $R_0 \geq 0$. First, define for $1 \leq l \leq p$ the injections $U_{nl} : \mathscr{H}_{nl}^g \to \mathscr{H}_l^g$ with $U_{nl}U_{nl}^\star = P_{nl}$ and $U_{nl}^\star U_{nl} = \mathbf{I}_{\mathscr{H}_{nl}^g}$, and for $1 \leq s \leq p$ the injections $V_{ns} : \mathscr{H}_{ns}^f \to \mathscr{H}_s^f$ with $V_{ns}V_{ns}^\star = \Pi_{ns}$ and $V_{ns}^\star V_{ns} = \mathbf{I}_{\mathscr{H}_{ns}^f}$. By definition we thus have:

$$\inf_{\substack{g\in\mathscr{K}_n^g(\mathscr{H}_1^g,\dots,\mathscr{H}_p^g)\\ \mathscr{A}\in\mathscr{K}_n^f(\mathscr{H}_1^f,\dots,\mathscr{H}_q^f)}} J(g,\mathscr{A}) = \inf_{\substack{\tilde{g}\in\mathscr{K}^g(\mathscr{H}_{n1}^g,\dots,\mathscr{H}_{np}^g)\\ \tilde{\mathscr{A}}\in\mathscr{K}^f(\mathscr{H}_{n1}^f,\dots,\mathscr{H}_{nq}^f)}} J((U_{nl}\tilde{g}_l)_{1\leq l\leq p}, (V_{ns}\tilde{\mathscr{A}}_s V_{ns}^\star)_{1\leq s\leq q}).$$

Now, in the objective function J, the regularization parts are invariant up to the injections: since all U_{nl} are linear isometries the norm of $\tilde{g}_l$ is unchanged, and this is also the case for the kSoS regularizers Ω_s if they satisfy Assumption 3.1, see Allain et al. (2026, Lemma A.3). This means that for any $(\tilde{g}, \tilde{\mathscr{A}}) \in \mathscr{K}^g(\mathscr{H}^g_{n1}, \dots, \mathscr{H}^g_{np}) \times \mathscr{K}^f(\mathscr{H}^f_{n1}, \dots, \mathscr{H}^f_{nq})$,

$$J((U_{nl}\tilde{g}_l)_{1\le l\le p}, (V_{ns}\tilde{\mathscr{A}}_s V^\star_{ns})_{1\le s\le q}) = L((U_{nl}\tilde{g}_l(x_i))_{1\le i\le n,\, 1\le l\le p}, (f_{V_{ns}\tilde{\mathscr{A}}_s V^\star_{ns}}(x_i))_{1\le i\le n,\, 1\le s\le q}) + \sum_{l=1}^{p} R_l(\|\tilde{g}_l\|_{\mathscr{H}^g_l}) + \sum_{s=1}^{q} \Omega_s(\tilde{\mathscr{A}}_s).$$

Let $\tilde{g}^0_l$, $\tilde{\mathscr{A}}^0_s$ be such that $J_0 = J((U_{nl}\tilde{g}^0_l)_{1\le l\le p}, (V_{ns}\tilde{\mathscr{A}}^0_s V^\star_{ns})_{1\le s\le q}) < +\infty$. Because J is coercive, there exists $R_0 \ge 0$ such that $\max(\max_{1\le l\le p}\|\tilde{g}_l\|_{\mathscr{H}^g_{nl}}, \max_{1\le s\le q}\|\tilde{\mathscr{A}}_s\|_F) \ge R_0$ implies

$$J((U_{nl}\tilde{g}_l)_{1\le l\le p}, (V_{ns}\tilde{\mathscr{A}}_s V^\star_{ns})_{1\le s\le q}) \ge J_0,$$

which shows that

$$\inf_{\substack{\tilde{g}\in\mathscr{K}^g(\mathscr{H}^g_{n1},\dots,\mathscr{H}^g_{np}) \\ \tilde{\mathscr{A}}\in\mathscr{K}^f(\mathscr{H}^f_{n1},\dots,\mathscr{H}^f_{nq})}} J((U_{nl}\tilde{g}_l)_{1\le l\le p}, (V_{ns}\tilde{\mathscr{A}}_s V^\star_{ns})_{1\le s\le q}) = \inf_{(\tilde{g},\tilde{\mathscr{A}})\in\mathscr{K}_{R_0}} J((U_{nl}\tilde{g}_l)_{1\le l\le p}, (V_{ns}\tilde{\mathscr{A}}_s V^\star_{ns})_{1\le s\le q}).$$

Third step.

We conclude by showing that this last minimization problem has a solution: indeed, the objective function is lower semi-continuous, and hence reaches its minimum on any non-empty compact set. Since the set $\mathscr{K}_{R_0}$ is compact because $\mathscr{K}^g(\mathscr{H}^g_{n1}, \dots, \mathscr{H}^g_{np}) \times \mathscr{K}^f(\mathscr{H}^f_{n1}, \dots, \mathscr{H}^f_{nq})$ is finite, and non-empty because it contains $\tilde{g}^0_l$, $\tilde{\mathscr{A}}^0_s$, the result directly follows. □

Remark 6.1. If the loss function L is bounded below and the individual regularizers R_l, Ω_s satisfy Assumption 3.2 and Assumption 3.1 respectively, the objective J is trivially coercive. In Marteau-Ferey et al. (2020), they consider a bounded below loss function together with Assumption 3.1 to derive their representer theorem. Here, we use a more general assumption that allows to consider a larger class of learning problems, such as heteroscedastic regression with unknown mean which involves an unbounded loss function, see Section 6.4.

6.3 Dual formulation

To prove Theorem 3.2, we follow Marteau-Ferey et al. (2020) and use Theorem 3.3.5 from Borwein and Lewis (2006). With the notations of Borwein and Lewis (2006), Problem 9 can be written as the primal formulation:

$$p^\star = \inf_{x\in E} f(x) + g(Ax)$$

where

- $x = (\gamma_1, \dots, \gamma_p, \mathbf{A}_1, \dots, \mathbf{A}_q) \in E = (\mathbb{R}^n)^p \times \mathbb{S}^{r_1} \times \dots \times \mathbb{S}^{r_q}$
- $f(\gamma_1, \dots, \gamma_p, \mathbf{A}_1, \dots, \mathbf{A}_q) = \sum_{l=1}^p \lambda^g_l \gamma_l^\top \mathbf{K}^g_l \gamma_l + \sum_{s=1}^q \Omega^+_s(\mathbf{A}_s)$ where

$$\Omega^+_s(\mathbf{A}) = \begin{cases} \Omega_s(\mathbf{A}) & \text{if } \mathbf{A} \succeq 0 \\ +\infty & \text{otherwise} \end{cases}$$

- For $(y, z) \in Y = (\mathbb{R}^n)^p \times (\mathbb{R}^n)^q$, $g(y, z) = L(y, z)$

- $A : E \mapsto Y$ is the linear map defined by

$$A(\gamma_1, \dots, \gamma_p, \mathbf{A}_1, \dots, \mathbf{A}_q) = ((k_l^g(x_i)^\top \gamma_l)_{1 \leq i \leq n,\, 1 \leq l \leq p},\ (\tilde{f}_{\mathbf{A}_s}(x_i))_{1 \leq i \leq n,\, 1 \leq s \leq q})$$

The associated dual problem is

$$d^\star = \sup_{\phi \in Y} \ -f^\star(A^\star \phi) - g^\star(-\phi) \tag{20}$$

and if $A\text{dom}\, f \cap \text{cont}\, g \neq \emptyset$, $p^\star = d^\star$ as long as f and g are convex. This is the case here: (a) f is convex as the sum of convex functions, (b) g is convex since we have assumed that the loss function L is convex and (c) we have also assumed that there exist $\gamma_l^0 \in \mathbb{R}^n$, $\mathbf{A}_s^0 \in \mathbb{S}_+^{r_s}$ such that L is continuous in $(k_l^g(x_i)^\top \gamma_l^0)_{1 \leq i \leq n,\, 1 \leq l \leq p}$. In addition, by our previous results, the primal problem is coercive and proper, yielding a finite minimum $p^\star$: since strong duality holds, $d^\star$ is thus finite and Theorem 3.3.5 from Borwein and Lewis (2006) guarantees that the dual supremum is attained.

We now need to derive $f^\star$ and $A^\star$ (recall that $g^\star = L^\star$ by definition). First, we denote the components of $\phi \in Y$ as $\phi = (\beta_1, \dots, \beta_p, \alpha_1, \dots, \alpha_q)$, and by separability it holds that $A^\star \phi = ((\mathbf{K}_l^g \beta_l)_{1 \leq l \leq p}, (\mathbf{V}_s \text{Diag}(\alpha_s) \mathbf{V}_s^\top)_{1 \leq s \leq q})$. The first components of $A^\star$ readily come from the adjoint of $k_l^g(x_i)^\top \gamma_l$, while the second ones were derived in Marteau-Ferey et al. (2020). For $f^\star$, let us write

$$f^\star(\gamma_1^\star, \dots, \gamma_p^\star, \mathbf{A}_1^\star, \dots, \mathbf{A}_q^\star) = \sum_{l=1}^p R_l^\star(\gamma_l^\star) + \sum_{s=1}^q (\Omega_s^+)^\star(\mathbf{A}_s^\star).$$

For $R_l(\gamma_l) = \lambda_l^g \gamma_l^\top \mathbf{K}_l^g \gamma_l$, it is straightforward to show that $R_l^\star(\gamma_l^\star) = \frac{1}{4\lambda_l^g} (\gamma_l^\star)^\top (\mathbf{K}_l^g)^\dagger \gamma_l^\star$ where $\mathbf{M}^\dagger$ denotes the Moore-Penrose pseudo inverse of $\mathbf{M}$. On the other hand, for $\Omega_s(\mathbf{A}) = \lambda_{s1}^f \|\mathbf{A}\|_\star + \lambda_{s2}^f \|\mathbf{A}\|_F^2$, we have

$$(\Omega_s^+)^\star(\mathbf{A}_s^\star) = \frac{1}{4\lambda_{s2}^f} \left\| \left[\mathbf{A}_s^\star - \lambda_{s1}^f \mathbf{I}_{r_s} \right]_+ \right\|_F^2, \tag{21}$$

see Marteau-Ferey et al. (2020). This implies that

$$\begin{aligned}
f^\star(A^\star \phi) &= f^\star((\mathbf{K}_l^g \beta_l)_{1 \leq l \leq p}, (\mathbf{V}_s \text{Diag}(\alpha_s) \mathbf{V}_s^\top)_{1 \leq s \leq q}) \\
&= \sum_{l=1}^p \frac{1}{4\lambda_l^g} \beta_l^\top \mathbf{K}_l^g (\mathbf{K}_l^g)^\dagger \mathbf{K}_l^g \beta_l + \sum_{s=1}^q \frac{1}{4\lambda_{s2}^f} \left\| \left[\mathbf{V}_s \text{Diag}(\alpha_s) \mathbf{V}_s^\top - \lambda_{s1}^f \mathbf{I}_{r_s} \right]_+ \right\|_F^2 \\
&= \sum_{l=1}^p \frac{1}{4\lambda_l^g} \beta_l^\top \mathbf{K}_l^g \beta_l + \sum_{s=1}^q \frac{1}{4\lambda_{s2}^f} \left\| \left[\mathbf{V}_s \text{Diag}(\alpha_s) \mathbf{V}_s^\top - \lambda_{s1}^f \mathbf{I}_{r_s} \right]_+ \right\|_F^2
\end{aligned}$$

and the dual formulation in Theorem 3.2 follows by replacing ϕ with $(\beta_1, \dots, \beta_p, \alpha_1, \dots, \alpha_q)$ in Equation 20.

Finally, to recover the primal solutions, we again follow Marteau-Ferey et al. (2020) and Borwein and Lewis (2006, Exercise 4.2.17): if f and g are closed, then the primal solution $\bar{x}$ is given by $\bar{x} = \nabla f^\star(A^\star \bar{\phi})$ for $\bar{\phi}$ an optimal dual solution. Since L is a proper convex function which is lower semi-continuous, then it is closed. Besides, $\lambda_l^g \gamma_l^\top \mathbf{K}_l^g \gamma_l$, $\Omega_s^+(\mathbf{A}_s^\star)$ are also closed convex functions. By separability, we obtain Equations 12 and 13 by computing $\nabla f^\star$, where the latter is given in Marteau-Ferey et al. (2020):

$$\nabla((\Omega_s^+)^\star)(\mathbf{A}_s^\star) = \frac{1}{2\lambda_{s2}^f} \left[\mathbf{A}_s^\star - \lambda_{s1}^f \mathbf{I}_{r_s} \right]_+ \tag{22}$$

and is trivial for $\lambda_l^g \gamma_l^\top \mathbf{K}_l^g \gamma_l$.

We can also easily extend the previous dual formulation to accommodate for potential additional linear terms in the objective function, as elaborated in the following corollary.

Corollary 6.1. *Consider a new instance of Problem 9 where we add to the objective function an additional term* $\sum_{s=1}^q \mathrm{Tr}(\mathbf{A}_s \mathbf{W}_s)$ *for some matrices* $\mathbf{W}_s \in \mathbb{R}^{r_s \times r_s}$. *Then, this new problem also satisfies Proposition 3.1 and under the same assumptions of Theorem 3.2, it has the following dual formulation:*

$$\sup_{\substack{\alpha_1,\dots,\alpha_q \in \mathbb{R}^n \\ \beta_1,\dots,\beta_p \in \mathbb{R}^n}} -L^\star((-\beta_l)_{1\le l\le p}, (-\alpha_s)_{1\le s\le q}) - \sum_{l=1}^p \frac{1}{4\lambda_l^g} \beta_l^\top \mathbf{K}_l^g \beta_l - \sum_{s=1}^q \frac{1}{4\lambda_{s2}^f} \|[\mathbf{V}_s \mathrm{Diag}(\alpha_s) \mathbf{V}_s^\top - \lambda_{s1}^f \mathbf{I}_{r_s} - \mathbf{W}_s]_+\|_F^2.$$

Proof. The proof is the same as for Theorem 3.2, except that here we have

$$\begin{aligned} f(\gamma_1, \dots, \gamma_p, \mathbf{A}_1, \dots, \mathbf{A}_q) &= \sum_{l=1}^p \lambda_l^g \gamma_l^\top \mathbf{K}_l^g \gamma_l + \sum_{s=1}^q \left(\Omega_s^+(\mathbf{A}_s) + \mathrm{Tr}(\mathbf{A}_s \mathbf{W}_s)\right) \\ &= \sum_{l=1}^p \lambda_l^g \gamma_l^\top \mathbf{K}_l^g \gamma_l + \sum_{s=1}^q \widetilde{\Omega}_s(\mathbf{A}_s), \end{aligned}$$

which satisfies all the required assumptions as before. The only difference is in the computation of $f^\star$. The first part is unchanged, and for the second one we have

$$\begin{aligned} \widetilde{\Omega}_s^\star(\mathbf{A}_s^\star) &= \sup_{\mathbf{A}_s \in \mathbb{S}_+^{r_s}} \mathrm{Tr}(\mathbf{A}_s \mathbf{A}_s^\star) - \widetilde{\Omega}_s(\mathbf{A}_s) \\ &= \sup_{\mathbf{A}_s \in \mathbb{S}_+^{r_s}} \mathrm{Tr}(\mathbf{A}_s \mathbf{A}_s^\star) - \Omega_s^+(\mathbf{A}_s) - \mathrm{Tr}(\mathbf{A}_s \mathbf{W}_s) \\ &= \sup_{\mathbf{A}_s \in \mathbb{S}_+^{r_s}} \mathrm{Tr}(\mathbf{A}_s(\mathbf{A}_s^\star - \mathbf{W}_s)) - \Omega_s^+(\mathbf{A}_s) \\ &= (\Omega_s^+)^\star(\mathbf{A}_s^\star - \mathbf{W}_s). \end{aligned}$$

□

6.4 Specific learning problems

Proof of Proposition 4.1 (heteroscedastic regression with known mean).

In this setting, we have $p = 0$ and $q = 1$. Since $v(\cdot) > 0$, $t_i \neq 0$ and thus the loss function $L(u_1, \dots, u_n) = \sum_{i=1}^n t_i u_i - \sum_{i=1}^n \log(u_i)$ is bounded below. Since it is trivially lower semi-continuous, it satisfies the assumptions of Theorem 3.1 (see Remark 6.1), which yields the finite-dimensional representation. In addition, Ω is strongly convex and Proposition 3.1 applies. Now, L is continuous in $(\tilde{f}_{\mathbf{A}^0}(x_i))_{1\le i\le n}$ for e.g. $\mathbf{A}^0 = \mathbf{I}_r$ and is convex, such that Theorem 3.2 holds and the only remaining part is to compute $L^\star$. Since $L(u_1, \dots, u_n) = \sum_{i=1}^n t_i u_i - \sum_{i=1}^n \log(u_i) = \sum_{i=1}^n l_i(u_i)$ with $l_i(u) = t_i u - \log(u)$ is separable, we have $L^\star(-\alpha) = \sum_{i=1}^n l_i^\star(-\alpha_i)$. But $l_i^\star(-\alpha_i) = [t_i u - \log(u)]^\star(-\alpha_i) = [-\log(u)]^\star(-\alpha_i - t_i) = -(1 + \log(\alpha_i + t_i))$ for $\alpha_i + t_i > 0$, yielding $-L^\star(-\alpha) = \sum_{i=1}^n (1 + \log(\alpha_i + t_i)) = n + \sum_{i=1}^n \log(\alpha_i + t_i)$ and the dual formulation follows by plugging $-L^\star(-\alpha)$ in Equation 11.

Proof of Proposition 4.4 (heteroscedastic regression with unknown mean).

We have $p = 1$ and $q = 1$, but contrary to the proof of Proposition 4.1 this time the loss function, although lower semi-continuous, is not bounded below. However, the objective $J(g, \mathscr{A})$ is coercive. Indeed, $f_{\mathscr{A}}(x_i) = \langle \phi^f(x_i), \mathscr{A}\phi^f(x_i)\rangle_{\mathscr{H}^f} \leq \|\mathscr{A}\|_\star \|\phi^f(x_i)\|^2_{\mathscr{H}^f} = \|\mathscr{A}\|_\star k^f(x_i, x_i)$. Let $C = \max_i k^f(x_i, x_i) < +\infty$. We can then lower bound the objective:

$$\begin{aligned} J(g, \mathscr{A}) &\geq 0 - \sum_{i=1}^n \log(C\|\mathscr{A}\|_\star) + R(\|g\|_{\mathscr{H}^g}) + \Omega(\mathscr{A}) \\ &= R(\|g\|_{\mathscr{H}^g}) + \Omega(\mathscr{A}) - n\log(\|\mathscr{A}\|_\star) - n\log(C) \\ &= R(\|g\|_{\mathscr{H}^g}) + \lambda_1^f \|\mathscr{A}\|_\star + \lambda_2^f \|\mathscr{A}\|_F^2 - n\log(\|\mathscr{A}\|_\star) - n\log(C). \end{aligned}$$

Since $\lambda_1^f > 0$, $\|\mathscr{A}\|_\star$ dominates $\log(\|\mathscr{A}\|_\star)$ which implies that when $\|\mathscr{A}\|_F^2$ goes to infinity $J(g, \mathscr{A}) \to +\infty$ since $\|\mathscr{A}\|_F \leq \|\mathscr{A}\|_\star$ by monotonicity of the Schatten norms. Furthermore, because $\|g\| \to \infty$, $R(\|g\|) \to \infty$ by assumption, the objective $J(g, \mathscr{A}) \to +\infty$ as either $\|g\| \to \infty$ or $\|\mathscr{A}\|_F^2$, and therefore J is coercive and satisfies the assumptions of Theorem 3.1. On the other hand, since $\lambda_2^f > 0$, Ω is strictly convex and Proposition 3.1 holds. For the dual formulation in Theorem 3.2, L is continuous in $(k^g(x_i)^\top \gamma^0)_{1\leq i\leq n}$ and $(\tilde{f}_{\mathbf{A}^0}(x_i))_{1\leq i\leq n}$ for e.g. $\gamma^0 = 0$, $\mathbf{A}^0 = \mathbf{I}_r$. For $\mathbf{u}, \mathbf{v} \in \mathbf{R}^n$, denote $L(\mathbf{u}, \mathbf{v}) = L(u_1, \dots, u_n, v_1, \dots, v_n) = \sum_{i=1}^n (y_i^2 v_i + \frac{u_i^2}{v_i} - 2y_i u_i - \log(v_i)) = \sum_{i=1}^n l_i(u_i, v_i)$ with $l_i(u, v) = y_i^2 v + \frac{u^2}{v} - 2y_i u - \log(v)$. By separability, $L^\star(-\beta, -\alpha) = \sum_{i=1}^n l_i^\star(-\beta_i, -\alpha_i)$. In addition,

$$\begin{aligned} l_i^\star(u^\star, v^\star) &= \sup_{u\in\mathbb{R},\, v>0} \quad uu^\star + vv^\star - y_i^2 v - \frac{u^2}{v} + 2y_i u + \log(v) \\ &= \sup_{u\in\mathbb{R},\, v>0} \quad h(u) + (v^\star - y_i^2)v + \log(v) \end{aligned}$$

with $h(u) = (u^\star + 2y_i)u - \frac{u^2}{v}$, which attains its maximum at $u = (u^\star + 2y_i)v/2$, such that

$$\begin{aligned} l_i^\star(u^\star, v^\star) &= \sup_{v>0} \quad \frac{(u^\star + 2y_i)^2 v}{4} + (v^\star - y_i^2)v + \log(v) \\ &= \sup_{v>0} \quad \xi v + \log(v) \\ &= [-\log(v)]^\star(\xi) = -(1 + \log(-\xi)) \end{aligned}$$

for $\xi = \frac{(u^\star+2y_i)^2}{4} + (v^\star - y_i^2) = (u^\star)^2/4 + y_i u^\star + v^\star < 0$. This yields $-L^\star(-\beta, -\alpha) = \sum_{i=1}^n (1 + \log\left(\beta_i y_i + \alpha_i - \beta_i^2/4\right))$ which, when plugged in Equation 11, gives the result.

Proof of Proposition 4.2 (non-crossing multi-quantile regression with known median).

Here, we have $p = 0$ and $q = q_- + q_+$. The loss function is lower semi-continuous and bounded below because the pinball loss is non-negative, such that Theorem 3.1 holds. This is the same for Proposition 3.1 because Ω is strictly convex. Besides, the loss function is continuous in $(f_{\mathbf{A}^0_{-j}}(x_i), f_{\mathbf{A}^0_k}(x_i))_{1\leq i\leq n,\, 1\leq j\leq q_-,\, 1\leq k\leq q_+}$ for e.g. $\mathbf{A}^0_{-j} = \mathbf{I}_{r_{-j}}$, $\mathbf{A}^0_k = \mathbf{I}_{r_k}$ and Theorem 3.2 applies as well. Now, denoting $u_{i,j} = \tilde{f}_{\mathbf{A}_{-j}}(x_i)$ and $v_{i,k} = \tilde{f}_{\mathbf{A}_k}(x_i)$, the loss function writes as

$$\begin{aligned} &L((u_{i,j})_{1\leq i\leq n,\, 1\leq j\leq q_-}, (v_{i,k})_{1\leq i\leq n,\, 1\leq k\leq q_+}) \\ &= \sum_{j=1}^{q_-}\sum_{i=1}^{n} \rho_{\tau_{-j}}\left(t_i + \sum_{j'=1}^{j} u_{i,j}\right) + \sum_{k=1}^{q_+}\sum_{i=1}^{n} \rho_{\tau_k}\left(t_i - \sum_{k'=1}^{k} v_{i,k}\right) \\ &= \sum_{j=1}^{q_-}\sum_{i=1}^{n} \rho_{\tau_{-j}}\left(t_i - \tilde{u}_{i,j}\right) + \sum_{k=1}^{q_+}\sum_{i=1}^{n} \rho_{\tau_k}\left(t_i - \tilde{v}_{i,k}\right) \\ &= \tilde{L}((\tilde{u}_{i,j})_{1\leq i\leq n,\, 1\leq j\leq q_-}, (\tilde{v}_{i,k})_{1\leq i\leq n,\, 1\leq k\leq q_+}) \\ &= (\tilde{L}\circ T)((u_{i,j})_{1\leq i\leq n,\, 1\leq j\leq q_-}, (v_{i,k})_{1\leq i\leq n,\, 1\leq k\leq q_+}) \end{aligned}$$

where $T : (\mathbb{R}^n)^{q_-} \times (\mathbb{R}^n)^{q_+} \mapsto (\mathbb{R}^n)^{q_-} \times (\mathbb{R}^n)^{q_+}$ is the linear map defined as

$$T((u_{i,j})_{1\leq i\leq n,\, 1\leq j\leq q_-}, (v_{i,k})_{1\leq i\leq n,\, 1\leq k\leq q_+}) = \left(\left(-\sum_{j'=1}^{j} u_{i,j}\right)_{1\leq i\leq n,\, 1\leq j\leq q_-}, \left(\sum_{k'=1}^{k} v_{i,k}\right)_{1\leq i\leq n,\, 1\leq k\leq q_+}\right).$$

Denoting $\underline{\alpha} = ((\alpha_{-j})_{1\leq j\leq q_-}, (\alpha_k)_{1\leq k\leq q_+})$ and $\underline{\tilde{\alpha}} = ((\tilde{\alpha}_{-j})_{1\leq j\leq q_-}, (\tilde{\alpha}_k)_{1\leq k\leq q_+})$, by Theorem 16.3 in Rockafellar (1970), we thus have

$$\begin{aligned} L^\star((\alpha_{-j})_{1\leq j\leq q_-}, (\alpha_k)_{1\leq k\leq q_+}) &= \inf_{\substack{\underline{\tilde{\alpha}} \\ T^\star\underline{\tilde{\alpha}}=\underline{\alpha}}} \tilde{L}^\star((\tilde{\alpha}_{-j})_{1\leq j\leq q_-}, (\tilde{\alpha}_k)_{1\leq k\leq q_+}) \\ &= \tilde{L}^\star(T^{-\top}\underline{\alpha}) \end{aligned}$$

since $T^\star = T^\top$ is invertible. Plugging this formula in Theorem 3.2 gives the dual formulation

$$\begin{aligned} \sup_{\underline{\alpha}\in(\mathbb{R}^n)^{q_-}\times(\mathbb{R}^n)^{q_+}} \quad & -\tilde{L}^\star(-T^{-\top}\underline{\alpha}) \\ & - \sum_{j=1}^{q_-} \frac{1}{4\lambda^f_{-j2}} \left\| \left[\mathbf{V}_{-j} \mathrm{Diag}\left(\alpha_{-j}\right) \mathbf{V}_{-j}^\top - \lambda^f_{-j1} \mathbf{I}_{r_{-j}} \right]_+ \right\|_F^2 \\ & - \sum_{k=1}^{q_+} \frac{1}{4\lambda^f_{k2}} \left\| \left[\mathbf{V}_k \mathrm{Diag}\left(\alpha_k\right) \mathbf{V}_k^\top - \lambda^f_{k1} \mathbf{I}_{r_k} \right]_+ \right\|_F^2 \end{aligned}$$

and we use the change of variable $T^{-\top}\underline{\alpha} \leftrightarrow \underline{\alpha}$ to get

$$\begin{aligned} \sup_{\underline{\alpha}\in(\mathbb{R}^n)^{q_-}\times(\mathbb{R}^n)^{q_+}} \quad & -\tilde{L}^\star(-\underline{\alpha}) \\ & - \sum_{j=1}^{q_-} \frac{1}{4\lambda^f_{-j2}} \left\| \left[\mathbf{V}_{-j} \mathrm{Diag}\left(- \sum_{l=j}^{q_-} \alpha_{-l} \right) \mathbf{V}_{-j}^\top - \lambda^f_{-j1} \mathbf{I}_{r_{-j}} \right]_+ \right\|_F^2 \\ & - \sum_{k=1}^{q_+} \frac{1}{4\lambda^f_{k2}} \left\| \left[\mathbf{V}_k \mathrm{Diag}\left(\sum_{l=k}^{q_+} \alpha_l \right) \mathbf{V}_k^\top - \lambda^f_{k1} \mathbf{I}_{r_k} \right]_+ \right\|_F^2 \end{aligned}$$

by definition of $T^\top$. Besides, $\tilde{L}$ is separable, and the Fenchel conjugate $\rho_\tau^\star(u^\star)$ of the pinball loss $\rho_\tau(y-u)$ is

$$\rho_\tau^\star(u^\star) = \begin{cases} yu^\star & \text{if } -\tau \leq u^\star \leq 1-\tau \\ +\infty & \text{otherwise,} \end{cases}$$

which yields

$$\tilde{L}^\star(\underline{\alpha}) = \sum_{j=1}^{q_-}\sum_{i=1}^{n} t_i\alpha_{i,j} + \sum_{k=1}^{q_+}\sum_{i=1}^{n} t_i\alpha_{i,k} = t^\top\left(\sum_{j=1}^{q_-}\alpha_{-j} + \sum_{k=1}^{q_+}\alpha_k\right)$$

if $\alpha_{-j} \in [-\tau_{-j}, 1-\tau_{-j}]$, $\alpha_k \in [-\tau_k, 1-\tau_k]$ for $j = 1, \ldots, q_-$, $k = 1, \ldots, q_+$ and $+\infty$ otherwise. The final dual formulation then directly follows.

Proof of Proposition 4.5 (non-crossing multi-quantile regression with unknown median).

The proof is similar to the one of Proposition 4.2, except that here we have $p = 1$ and $q = q_- + q_+$, but Theorem 3.1, Proposition 3.1 and Theorem 3.2 still apply with the exact same reasoning. The

main difference is that, denoting $w_i = k^g(x_i)^\top \gamma$, the loss function now writes

$$
\begin{aligned}
&L((w_i)_{1\le i\le n}, (u_{i,j})_{1\le i\le n,\, 1\le j\le q_-}, (v_{i,k})_{1\le i\le n,\, 1\le k\le q_+}) \\
&= \sum_{i=1}^{n} \rho_{0.5}\left(y_i - w_i\right) + \sum_{j=1}^{q_-}\sum_{i=1}^{n} \rho_{\tau_{-j}}\left(y_i - w_i + \sum_{j'=1}^{j} u_{i,j}\right) + \sum_{k=1}^{q_+}\sum_{i=1}^{n} \rho_{\tau_k}\left(y_i - w_i - \sum_{k'=1}^{k} v_{i,k}\right) \\
&= \sum_{i=1}^{n} \rho_{0.5}\left(y_i - \tilde{w}_i\right) + \sum_{j=1}^{q_-}\sum_{i=1}^{n} \rho_{\tau_{-j}}\left(y_i - \tilde{u}_{i,j}\right) + \sum_{k=1}^{q_+}\sum_{i=1}^{n} \rho_{\tau_k}\left(y_i - \tilde{v}_{i,k}\right) \\
&= \tilde{L}((\tilde{w}_i)_{1\le i\le n}, (\tilde{u}_{i,j})_{1\le i\le n,\, 1\le j\le q_-}, (\tilde{v}_{i,k})_{1\le i\le n,\, 1\le k\le q_+}) \\
&= (\tilde{L}\circ T)((w_i)_{1\le i\le n}, (u_{i,j})_{1\le i\le n,\, 1\le j\le q_-}, (v_{i,k})_{1\le i\le n,\, 1\le k\le q_+})
\end{aligned}
$$

where $T : \mathbb{R}^n \times (\mathbb{R}^n)^{q_-} \times (\mathbb{R}^n)^{q_+} \mapsto \mathbb{R}^n \times (\mathbb{R}^n)^{q_-} \times (\mathbb{R}^n)^{q_+}$ is the linear map defined as

$$
\begin{aligned}
&T((w_i)_{1\le i\le n}, (u_{i,j})_{1\le i\le n,\, 1\le j\le q_-}, (v_{i,k})_{1\le i\le n,\, 1\le k\le q_+}) \\
&= \left((w_i)_{1\le i\le n}, \left(w_i - \sum_{j'=1}^{j} u_{i,j}\right)_{1\le i\le n,\, 1\le j\le q_-}, \left(w_i + \sum_{k'=1}^{k} v_{i,k}\right)_{1\le i\le n,\, 1\le k\le q_+}\right)
\end{aligned}
$$

such that

$$
\begin{aligned}
L^\star(\beta, (\alpha_{-j})_{1\le j\le q_-}, (\alpha_k)_{1\le k\le q_+}) = &\inf_{\substack{\tilde{\beta},\, \underline{\tilde{\alpha}} \\ T^\star(\tilde{\beta},\underline{\tilde{\alpha}}) = (\beta,\underline{\alpha})}} \tilde{L}^\star(\tilde{\beta}, (\tilde{\alpha}_{-j})_{1\le j\le q_-}, (\tilde{\alpha}_k)_{1\le k\le q_+}) \\
&= \tilde{L}^\star(T^{-\top}(\beta, \underline{\alpha})).
\end{aligned}
$$

The dual formulation is thus

$$
\begin{aligned}
\sup_{\beta\in\mathbb{R}^n,\, \underline{\alpha}\in(\mathbb{R}^n)^{q_-}\times(\mathbb{R}^n)^{q_+}} \quad & -\tilde{L}^\star(-T^{-\top}(\beta, \underline{\alpha})) \\
& - \frac{1}{4\lambda^g}\beta^\top K^g \beta \\
& - \sum_{j=1}^{q_-} \frac{1}{4\lambda^f_{-j2}} \left\| \left[\mathbf{V}_{-j}\mathrm{Diag}\left(\alpha_{-j}\right)\mathbf{V}_{-j}^\top - \lambda^f_{-j1}\mathbf{I}_{r_{-j}}\right]_+ \right\|_F^2 \\
& - \sum_{k=1}^{q_+} \frac{1}{4\lambda^f_{k2}} \left\| \left[\mathbf{V}_{k}\mathrm{Diag}\left(\alpha_{k}\right)\mathbf{V}_{k}^\top - \lambda^f_{k1}\mathbf{I}_{r_{k}}\right]_+ \right\|_F^2
\end{aligned}
$$

which, after the change of variable $T^{-\top}(\beta, \underline{\alpha}) \leftrightarrow (\beta, \underline{\alpha})$, is equivalent to

$$
\begin{aligned}
\sup_{\beta\in\mathbb{R}^n,\, \underline{\alpha}\in(\mathbb{R}^n)^{q_-}\times(\mathbb{R}^n)^{q_+}} \quad & -\tilde{L}^\star(-\beta, -\underline{\alpha}) \\
& - \frac{1}{4\lambda^g}\left(\beta + \sum_{j=1}^{q_-}\alpha_{-j} + \sum_{k=1}^{q_+}\alpha_k\right)^\top K^g \left(\beta + \sum_{j=1}^{q_-}\alpha_{-j} + \sum_{k=1}^{q_+}\alpha_k\right) \\
& - \sum_{j=1}^{q_-} \frac{1}{4\lambda^f_{-j2}} \left\| \left[\mathbf{V}_{-j}\mathrm{Diag}\left(-\sum_{l=j}^{q_-}\alpha_{-l}\right)\mathbf{V}_{-j}^\top - \lambda^f_{-j1}\mathbf{I}_{r_{-j}}\right]_+ \right\|_F^2 \\
& - \sum_{k=1}^{q_+} \frac{1}{4\lambda^f_{k2}} \left\| \left[\mathbf{V}_{k}\mathrm{Diag}\left(\sum_{l=k}^{q_+}\alpha_{l}\right)\mathbf{V}_{k}^\top - \lambda^f_{k1}\mathbf{I}_{r_{k}}\right]_+ \right\|_F^2
\end{aligned}
$$

by definition of $T^\top$. The Fenchel conjugate of $\tilde{L}$ is the same as in the proof of Proposition 4.2 with an additional pinball loss term for $\tau = 0.5$ and where t_i is replaced with y_i, which yields the final dual formulation.

Proof of Proposition 4.3 (density estimation).

We start by considering the Lagrangian formulation of the finite-dimensional constrained problem, which gives the dual

$$\begin{aligned} d &= \sup_{\theta\in\mathbb{R}} \inf_{\mathbf{A}\in\mathbb{S}_+^r} \; -\sum_{i=1}^n \log(\tilde{f}_{\mathbf{A}}(x_i)) + \Omega(\mathbf{A}) + \theta(\mathrm{Tr}(\mathbf{A}\mathbf{W}_\nu) - 1) \\ &= \sup_{\theta\in\mathbb{R}} \; -\theta + \left(\inf_{\mathbf{A}\in\mathbb{S}_+^r} \; -\sum_{i=1}^n \log(\tilde{f}_{\mathbf{A}}(x_i)) + \Omega(\mathbf{A}) + \theta\,\mathrm{Tr}(\mathbf{A}\mathbf{W}_\nu) \right) \end{aligned} \tag{23}$$

where we have used the fact that $\int_{\mathcal{X}} \tilde{f}_{\mathbf{A}}(x)\nu(x)dx = \mathrm{Tr}(\mathbf{A}\mathbf{W}_\nu)$, see Marteau-Ferey et al. (2020). Since the primal is feasible and finite for e.g. $\mathbf{A} = \mathbf{I}_r / \mathrm{Tr}(\mathbf{W}_\nu) \succeq 0$, Slater's condition holds and we have strong duality. On the other hand, for fixed θ, the inner problem

$$\inf_{\mathbf{A}\in\mathbb{S}_+^r} \; -\sum_{i=1}^n \log(\tilde{f}_{\mathbf{A}}(x_i)) + \Omega(\mathbf{A}) + \theta\,\mathrm{Tr}(\mathbf{A}\mathbf{W}_\nu)$$

satisfies all the assumptions of Corollary 6.1 with $p = 0$, $q = 1$, $\mathbf{W}_1 = \theta\mathbf{W}_\nu$ and $L(u_1, \dots, u_n) = -\sum_{i=1}^n \log(u_i)$ with Fenchel conjugate $L^\star(u_1^\star, \dots, u_n^\star) = \sum_{i=1}^n \left(-(1 + \log(-u_i^\star))\right)$ for $u_i^\star < 0$. The final dual formulation trivially follows by plugging the dual formulation of the inner problem

$$\sup_{\alpha\in\mathbb{R}^n} \; -L^\star(-\alpha) - \frac{1}{4\lambda_2^f} \|[\mathbf{V}\mathrm{Diag}(\alpha)\mathbf{V}^\top - \lambda_1^f \mathbf{I}_r - \theta\mathbf{W}_\nu]_+\|_F^2$$

from Corollary 6.1 in Equation 23 with $L^\star$ given above.

7 Details about numerical experiments

For all our numerical experiments, we detail below the results of the cross-validation procedure that leads to the best set of hyperparameters which we used to generate each figure in Section 4. In particular, all cross-validations are made with 5 folds and the features on the training samples are scaled to zero mean and unit variance beforehand such that the candidate lengthscales that are tested are always comparable across all tasks. For readability we only provide the best cross-validation combinations in each instance. Finally, except for density estimation where we use the `L-BFGS` solver, the `AGD` solver is chosen for solving all dual formulations (this choice was made to select the most efficient solver in each case).

7.1 Heteroscedastic regression with known mean

For lengthscales we use the list of candidates $[0.4, 0.5, 0.6, 0.7, 0.8]$ and for regularization penalties $[0.01, 0.1, 1]$. We first perform cross-validation on the kernel ridge regression part to estimate the hyperparameters $(\theta^\mu, \lambda^\mu)$ for the mean function alone, with a least-squares score: results are displayed in Table 1(a). Then, we run another cross-validation to estimate the hyperparameters $(\theta^f, \lambda_1^f, \lambda_2^f)$ of the variance function, with a score given by the negative log-likelihood. The results are given in Table 1(b).

Table 1: CV results for sequential heteroscedastic regression: mean function (left) and variance function (right).

(a)

θ^{μ}	λ^{μ}	CV score
0.5	**0.1**	**154.5067**
0.6	0.1	155.0722
0.4	1	155.0790
0.8	0.01	155.0830
0.4	0.1	155.1892
0.6	1	155.5632

(b)

θ^{f}	λ_1^{f}	λ_2^{f}	CV score
0.6	**0.01**	**0.01**	**33.4455**
0.5	0.01	0.1	33.9745
0.6	1	0.1	34.0711
0.5	1	0.01	34.1408
0.7	0.1	0.1	34.1881
0.6	1	0.01	34.1991

7.2 Non-crossing multi-quantile regression with known median

For sequential non-crossing multi-quantile regression, we first perform cross-validation on the kernel quantile regression part for the median, to estimate the lengthscale $\theta^{q_{0.5}}$, while $\lambda^{q_{0.5}}$ is fixed to 1. The lengthscale candidates are $[0.7, 0.9, 1.1, 1.3]$. The score function used here is the median pinball loss. The results are displayed in Table 2.

Table 2: CV results for median regression.

$\theta^{q_{0.5}}$	CV score
0.7	**77.2404**
0.9	78.7977
1.1	83.1513
1.3	89.9806

Then, we perform two cross-validations to choose the lengthscales in each of the settings we investigate, with a score function given by the sum of all the quantile pinball losses and the same grid of lengthscale candidates. In the same lengthscale setting, we estimate a common lengthscale θ^{f} for all quantile functions: results are given in Table 3(a). In the free lengthscale setting, we lessen this contraint by allowing two different lengthscales: θ^{f}_{low}, common to all quantile functions below the median, and θ^{f}_{upp} common to those above. We give the results in Table 3(b). In all instances we fix $(\lambda_1^{f}, \lambda_2^{f})$ to 1.

Table 3: CV results for quantiles in sequential multi-quantile regression: same lengthscales (left) and free lengthscales (right).

(a)

θ^{f}	CV score
0.7	**269.0395**
0.9	269.7175
1.1	274.9520
1.3	275.5860

(b)

θ^{f}_{low}	θ^{f}_{upp}	CV score
1.3	**0.7**	**268.9982**
0.7	0.7	269.0395
1.1	0.7	269.0574
0.9	0.7	269.1660

7.3 Density estimation

For the density estimation problem, we use the usual L^2 norm criterion equal to $\int_{\mathcal{X}}(p(x)-\hat{p}(x))^2\,\mathrm{d}x$ as the score function, which writes in the cross-validation setting as $\int_{\mathcal{X}}\hat{p}(x)^2\,\mathrm{d}x-\frac{2}{n}\sum_{k=1}^{K}\sum_{i\in I_K}\hat{p}_{-I_K}(x_i)$ where I_k is the set of indices in the k-th fold and $\hat{p}_{-I_K}$ is the estimate trained without the samples in I_k. Our density estimation problem is a 2 dimensional example, we then use an anisotropic kernel with different lengthscales $(\theta^f_{x_1},\theta^f_{x_2})$ on each dimension. Cross-validation estimates the hyperparameters $(\theta^f_{x_1},\theta^f_{x_2},\lambda^f_1,\lambda^f_2)$, with lengthscale candidates $[0.3, 0.4, 0.5, 0.6, 0.7, 0.8]$ and regularization penalties $[0.01, 0.1, 1.0]$. Results are displayed in Table 4.

Table 4: CV results for density estimation.

$\theta^f_{x_1}$	$\theta^f_{x_2}$	λ^f_1	λ^f_2	CV score
0.7	**0.8**	**1.0**	**1.0**	**-0.4779**
0.7	0.8	0.1	1.0	-0.4778
0.7	0.8	0.01	1.0	-0.4778
0.6	0.8	1.0	1.0	-0.4773
0.6	0.8	0.1	1.0	-0.4770
0.8	0.8	1.0	1.0	-0.4769

7.4 Heteroscedastic regression

For the joint learning problem in heteroscedastic regression with unknown mean, the cross-validation procedure now estimates all hyperparameters $(\theta^g,\theta^f,\lambda^g,\lambda^f_1,\lambda^f_2)$ jointly. We use the same candidate grids as in Section 7.1, results are given in Table 5.

Table 5: CV results for joint heteroscedastic regression.

θ^g	θ^f	λ^g	λ^f_1	λ^f_2	CV score
0.4	**0.6**	**0.01**	**0.01**	**0.01**	**37.8797**
0.4	0.6	0.01	1	0.01	37.9787
0.4	0.6	0.01	0.1	0.01	38.0116
0.5	0.7	0.01	1	0.1	38.1037
0.4	0.7	0.01	1	0.1	38.1186
0.4	0.7	0.01	1	0.01	38.1552

7.5 Non-crossing multi-quantile regression

When considering the sequential problem, as in the top row of Figure 5, the estimation is done sequentially following exactly the free setting in Section 7.2. The results of the cross-validations for the median and quantiles are given in Table 6 (observe that the numbers are not the same as in Section 7.2 because the random seed generating the data is not the same).

Table 6: CV results for sequential multi-quantile regression: median function (left) and quantile functions (right).

(a)

$\theta^{q_{0.5}}$	CV score
0.7	**81.8059**
0.9	82.8395
1.1	85.5509
1.3	91.2074

(b)

θ^f_{low}	θ^f_{upp}	CV score
0.9	**0.7**	**282.3186**
0.7	0.7	282.4091
1.3	0.7	282.6543
1.1	0.7	282.9056

For the joint learning problem, cross-validation estimates all hyperparameters $(\theta^g, \theta^f_{\text{low}}, \theta^f_{\text{upp}})$ jointly. The parameters $(\lambda^g, \lambda^f_1, \lambda^f_2)$ are all fixed to 1: results are displayed in Table 7.

Table 7: CV results for joint multi-quantile regression.

θ^g	θ^f_{low}	θ^f_{upp}	CV score
0.9	**1.3**	**0.7**	**282.6605**
0.9	0.9	0.7	282.7645
0.7	1.3	0.7	282.7816
0.9	1.1	0.7	282.8602
0.9	0.7	0.7	282.8913
0.7	1.3	0.9	282.9010