Infinite Task Learning in RKHSs

Romain Brault[1†] **Alex Lambert**[2†] **Zoltán Szabó**[3] **Maxime Sangnier**[4] **Florence d'Alché-Buc**[2]

[1]CentraleSupélec; [2]Télécom ParisTech, IP Paris; [3]École Polytechnique, IP Paris; [4]Sorbonne Université.

Abstract

Machine learning has witnessed tremendous success in solving tasks depending on a single hyperparameter. When considering simultaneously a finite number of tasks, multi-task learning enables one to account for the similarities of the tasks via appropriate regularizers. A step further consists of learning a continuum of tasks for various loss functions. A promising approach, called *Parametric Task Learning*, has paved the way in the continuum setting for affine models and piecewise-linear loss functions. In this work, we introduce a novel approach called *Infinite Task Learning*: its goal is to learn a function whose output is a function over the hyperparameter space. We leverage tools from operator-valued kernels and the associated Vector-Valued Reproducing Kernel Hilbert Space that provide an explicit control over the role of the hyperparameters, and also allows us to consider new type of constraints. We provide generalization guarantees to the suggested scheme and illustrate its efficiency in cost-sensitive classification, quantile regression and density level set estimation.

1 INTRODUCTION

Several fundamental problems in machine learning and statistics can be phrased as the minimization of a loss function described by a hyperparameter. The hyperparameter might capture numerous aspects of the problem: (i) the tolerance w.r.t. outliers as the ϵ-insensitivity in Support Vector Regression (Vapnik et al., 1997), (ii) importance of smoothness or sparsity such as the weight of the ℓ_2-norm in Tikhonov regularization (Tikhonov et al., 1977), ℓ_1-norm in LASSO (Tibshirani, 1996), or more general structured-sparsity inducing norms (Bach et al., 2012), (iii) Density Level-Set Estimation (DLSE), see for example one-class support vector machines One-Class Support Vector Machine (OCSVM, Schölkopf et al., 2000), (iv) confidence as examplified by Quantile Regression (QR, Koenker et al., 1978), or (v) importance of different decisions as implemented by Cost-Sensitive Classification (CSC, Zadrozny et al., 2001).

In various cases including QR, CSC or DLSE, one is interested in solving the parameterized task for several hyperparameter values. Multi-Task Learning (Evgeniou and Pontil, 2004) provides a principled way of benefiting from the relationship between similar tasks while preserving local properties of the algorithms: ν-property in DLSE (Glazer et al., 2013) or quantile property in QR (Takeuchi, Le, et al., 2006).

A natural extension from the traditional multi-task setting is to provide a prediction tool being able to deal with *any* value of the hyperparameter. In their seminal work, (Takeuchi, Hongo, et al., 2013) extended multi-task learning by considering an infinite number of parametrized tasks in a framework called Parametric Task Learning (PTL). Assuming that the loss is piecewise affine in the hyperparameter, the authors are able to get the whole solution path through parametric programming, relying on techniques developed by Hastie et al. (2004).[1]

In this paper, we relax the affine model assumption on the tasks as well as the piecewise-linear assumption on the loss, and take a different angle. We propose Infinite Task Learning (ITL) within the framework of function-valued function learning to handle a continuum number of parameterized tasks. For that purpose we leverage tools from operator-valued kernels and the associated Vector-Valued Reproducing Kernel Hilbert Space (vv-RKHS, Pedrick, 1957). The idea is that

Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89.

†Both authors contributed equally to this work.

[1]Alternative optimization techniques to deal with countable or continuous hyperparameter spaces could include semi-infinite (Stein, 2012) or bi-level programming (Wen et al., 1991).

the output is a function on the hyperparameters — modelled as scalar-valued Reproducing Kernel Hilbert Space (RKHS)—, which provides an explicit control over the role of the hyperparameters, and also enables us to consider new type of constraints. In the studied framework each task is described by a (scalar-valued) RKHS over the input space which is capable of dealing with nonlinearities. The resulting ITL formulation relying on vv-RKHS specifically encompasses existing multi-task approaches including joint quantile regression (Sangnier et al., 2016) or multi-task variants of density level set estimation (Glazer et al., 2013) by encoding a continuum of tasks.

Our **contributions** can be summarized as follows:

- We propose ITL, a novel vv-RKHS-based scheme to learn a continuum of tasks parametrized by a hyperparameter and design new regularizers.
- We prove excess risk bounds on ITL and illustrate its efficiency in quantile regression, cost-sensitive classification, and density level set estimation.

The paper is structured as follows. The ITL problem is defined in Section 2. In Section 3 we detail how the resulting learning problem can be tackled in vv-RKHSs. Excess risk bounds is the focus of Section 4. Numerical results are presented in Section 5. Conclusions are drawn in Section 6. Details of proofs are given in the supplement.

2 FROM PARAMETERIZED TO INFINITE TASK LEARNING

After introducing a few notations, we gradually define our goal by moving from single parameterized tasks (Section 2.1) to ITL (Section 2.3) through multi-task learning (Section 2.2).

Notations: $\mathbb{1}_S$ is the indicator function of set S. We use the $\sum_{i,j=1}^{n,m}$ shorthand for $\sum_{i=1}^{n}\sum_{j=1}^{m}$. $|x|_+ = \max(x, 0)$ denotes positive part. $\mathcal{F}(\mathcal{X}; \mathcal{Y})$ stands for the set of $\mathcal{X} \to \mathcal{Y}$ functions. Let $\mathcal{Z}$ be Hilbert space and $\mathcal{L}(\mathcal{Z})$ be the space of $\mathcal{Z} \to \mathcal{Z}$ bounded linear operators. Let $K : \mathcal{X} \times \mathcal{X} \to \mathcal{L}(\mathcal{Z})$ be an operator-valued kernel, i. e. $\sum_{i,j=1}^{n}\langle z_i, K(x_i, x_j)z_j\rangle_{\mathcal{Z}} \geqslant 0$ for all $n \in \mathbb{N}^*$ and $x_1, \ldots, x_n \in \mathcal{X}$ and $z_1, \ldots, z_n \in \mathcal{Z}$ and $K(x, z) = K(z, x)^*$ for all x, $z \in \mathcal{X}$. K gives rise to the Vector-Valued Reproducing Kernel Hilbert Space $\mathcal{H}_K = \overline{\operatorname{span}}\,\{ K(\cdot, x)z \mid\ x \in \mathcal{X},\ z \in \mathcal{Z} \} \subset \mathcal{F}(\mathcal{X}; \mathcal{Z})$, where $\overline{\operatorname{span}}\,\{\cdot\}$ denotes the closure of the linear span of its argument. For futher details on vv-RKHS the reader is referred to (Carmeli et al., 2010).

2.1 Learning Parameterized Tasks

A *supervised parametrized task* is defined as follows. Let $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ be a random variable with joint distribution $\mathbf{P}_{X,Y}$ which is assumed to be fixed but unknown; we also assume that $\mathcal{Y} \subset \mathbb{R}$. We have access to n independent identically distributed (i. i. d.) observations called training samples: $\mathcal{S} := ((x_i, y_i))_{i=1}^n \sim \mathbf{P}_{X,Y}^{\otimes n}$. Let Θ be the domain of hyperparameters, and $v_\theta \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be a loss function associated to $\theta \in \Theta$. Let $\mathcal{H} \subset \mathcal{F}(\mathcal{X}; \mathcal{Y})$ denote our hypothesis class; throughout the paper $\mathcal{H}$ is assumed to be a Hilbert space with inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$. For a given θ, the goal is to estimate the minimizer of the expected risk

$$R^\theta(h) := \mathbf{E}_{X,Y}[v_\theta(Y, h(X))] \tag{1}$$

over $\mathcal{H}$, using the training sample $\mathcal{S}$. This task can be addressed by solving the regularized empirical risk minimization problem

$$\min_{h\in\mathcal{H}} R_{\mathcal{S}}^\theta(h) + \Omega(h), \tag{2}$$

where $R_{\mathcal{S}}^\theta(h) := \frac{1}{n}\sum_{i=1}^n v_\theta(y_i, h(x_i))$ is the empirical risk and $\Omega : \mathcal{H} \to \mathbb{R}$ is a regularizer. Below we give three examples.

Quantile Regression: In this setting $\theta \in (0, 1)$. For a given hyperparameter θ, in Quantile Regression the goal is to predict the θ-quantile of the real-valued output conditional distribution $\mathbf{P}_{Y|X}$. The task can be tackled using the pinball loss (Koenker et al., 1978) defined in Eq. (3) and illustrated in Fig. S.3.

$$\begin{aligned} v_\theta(y, h(x)) &= |\theta - \mathbb{1}_{\mathbb{R}_-}(y - h(x))||y - h(x)|, \\ \Omega(h) &= \tfrac{\lambda}{2}\|h\|_{\mathcal{H}}^2, \quad \lambda > 0. \end{aligned} \tag{3}$$

Cost-Sensitive Classification: Our next example considers binary classification ($\mathcal{Y} = \{-1, 1\}$) where a (possibly) different cost is associated with each class, as it is often the case in medical diagnosis. The sign of $h \in \mathcal{H}$ yields the estimated class and in cost-sensitive classification one takes

$$\begin{aligned} v_\theta(y, h(x)) &= \left|\tfrac{1}{2}(\theta + 1) - \mathbb{1}_{\{-1\}}(y)\right||1 - yh(x)|_+, \\ \Omega(h) &= \tfrac{\lambda}{2}\|h\|_{\mathcal{H}}^2, \quad \lambda > 0. \end{aligned} \tag{4}$$

The $\theta \in [-1, 1]$ hyperparameter captures the trade-off between the importance of correctly classifying the samples having -1 and $+1$ labels. When θ is close to -1, the obtained h focuses on classifying well class -1, and vice-versa. Typically, it is desirable for a physician to choose *a posteriori* the value of the hyperparameter at which he wants to predict. Since this cost can rarely be considered to be fixed, this motivates to learn one model giving access to all hyperparameter values.

Density Level-Set Estimation: Examples of parameterized tasks can also be found in the unsupervised setting. For instance in outlier detection, the goal is to separate outliers from inliers. A classical technique to tackle this task is OCSVM (Schölkopf et al., 2000). OCSVM has a free parameter $\theta \in (0, 1]$, which can be proven to be an upper bound on the fraction of outliers. When using a Gaussian kernel with a bandwidth tending towards zero, OCSVM consistently estimates density level sets (Vert et al., 2006). This unsupervised learning problem can be empirically described by the minimization of a regularized empirical risk $R_S^\theta(h, t) + \Omega(h)$, solved *jointly* over $h \in \mathcal{H}$ *and* $t \in \mathbb{R}$ with

$$v_\theta(t, h(x)) = -t + \frac{1}{\theta}|t - h(x)|_+, \quad \Omega(h) = \tfrac{1}{2}\|h\|_{\mathcal{H}}^2.$$

2.2 Solving a Finite Number of Tasks as Multi-Task Learning

In all the aforementioned problems, one is rarely interested in the choice of a single hyperparameter value (θ) and associated risk $\left(R_S^\theta\right)$, but rather in the joint solution of multiple tasks. The naive approach of solving the different tasks independently can easily lead to inconsistencies. A principled way of solving many parameterized tasks has been cast as a MTL problem (Evgeniou, Micchelli, et al., 2005) which takes into account the similarities between tasks and helps providing consistent solutions. For example it is possible to encode the similarities of the different tasks in MTL through an explicit constraint function (Ciliberto et al., 2017). In the current work, the similarity between tasks is designed in an implicit way through the use of a kernel on the hyperparameters. Moreover, in contrast to MTL, in our case the input space and the training samples are the same for each task; a task is specified by a value of the hyperparameter. This setting is sometimes refered to as multi-output learning (Álvarez et al., 2012).

Formally, assume that we have p tasks described by parameters $(\theta_j)_{j=1}^p$. The idea of multi-task learning is to minimize the sum of the local loss functions $R_S^{\theta_j}$, i. e.

$$\arg\min_h \sum_{j=1}^p R_S^{\theta_j}(h_j) + \Omega(h),$$

where the individual tasks are modelled by the real-valued h_j functions, the overall $\mathbb{R}^p$-valued model is the vector-valued function $h\colon x \mapsto (h_1(x), \ldots, h_p(x))$, and Ω is a regularization term encoding similarities between tasks.

It is instructive to consider two concrete examples:

- In joint quantile regression one can use the regularizer to encourage that the predicted conditional quantile estimates for two similar quantile values are similar. This idea forms the basis of the approach proposed by Sangnier et al. (2016) who formulates the joint quantile regression problem in a vector-valued Reproducing Kernel Hilbert Space with an appropriate decomposable kernel that encodes the links between the tasks. The obtained solution shows less quantile curve crossings compared to estimators not exploiting the dependencies of the tasks as well as an improved accuracy.
- A multi-task version of DLSE has recently been presented by Glazer et al. (2013) with the goal of obtaining nested density level sets as θ grows. Similarly to joint quantile regression, it is crucial to take into account the similarities of the tasks in the joint model to efficiently solve this problem.

2.3 Towards Infinite Task Learning

In the following, we propose a novel framework called Infinite Task Learning in which we learn a function-valued function $h \in \mathcal{F}(\mathcal{X}; \mathcal{F}(\Theta; \mathcal{Y}))$. Our goal is to be able to handle new tasks after the learning phase and thus, not to be limited to given predefined values of the hyperparameter. Regarding this goal, our framework generalizes the Parametric Task Learning approach introduced by Takeuchi, Hongo, et al. (2013), by allowing a wider class of models and relaxing the hypothesis of piece-wise linearity of the loss function. Moreover a nice byproduct of this vv-RKHS based approach is that one can benefit from the functional point of view, design new regularizers and impose various constraints on the whole continuum of tasks, e. g.,

- The continuity of the $\theta \mapsto h(x)(\theta)$ function is a natural desirable property: for a given input x, the predictions on similar tasks should also be similar.
- Another example is to impose a shape constraint in QR: the conditional quantile should be increasing w. r. t. the hyperparameter θ. This requirement can be imposed through the functional view of the problem.
- In DLSE, to get nested level sets, one would want that for all $x \in \mathcal{X}$, the decision function $\theta \mapsto \mathbb{1}_{\mathbb{R}_+}(h(x)(\theta) - t(\theta))$ changes its sign only once.

To keep the presentation simple, in the sequel we are going to focus on ITL in the supervised setting; unsupervised tasks can be handled similarly.

Assume that h belongs to some space $\mathcal{H} \subseteq \mathcal{F}(\mathcal{X}; \mathcal{F}(\Theta; \mathcal{Y}))$ and introduce an integrated loss function

$$V(y, h(x)) := \int_\Theta v(\theta, y, h(x)(\theta)) d\mu(\theta), \tag{5}$$

where the local loss $v\colon \Theta \times \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ denotes v_θ seen as a function of three variables including the hyperparameter and μ is a probability measure on Θ which encodes the importance of the prediction at different hyperparameter values. Without prior information and for compact Θ, one may consider μ to be uniform. The true risk reads then

$$R(h) := \mathbf{E}_{X,Y}\left[V(Y, h(X))\right]. \tag{6}$$

Intuitively, minimizing the expectation of the integral over θ in a rich enough space corresponds to searching for a pointwise minimizer $x \mapsto h^*(x)(\theta)$ of the parametrized tasks introduced in Eq. (1) with, for instance, the implicit space constraint that $\theta \mapsto h^*(x)(\theta)$ is a continuous function for each input x. We show in Proposition S.7.1 that this is precisely the case in QR.

Interestingly, the empirical counterpart of the true risk minimization can now be considered with a much richer family of penalty terms:

$$\min_{h\in\mathcal{H}} R_{\mathcal{S}}(h) + \Omega(h), \quad R_{\mathcal{S}}(h) := \frac{1}{n}\sum\nolimits_{i=1}^{n} V(y_i, h(x_i)). \tag{7}$$

Here, $\Omega(h)$ can be a weighted sum of various penalties

- imposed directly on $(\theta, x) \mapsto h(x)(\theta)$, or
- integrated constraints on either $\theta \mapsto h(x)(\theta)$ or $x \mapsto h(x)(\theta)$ such as

$$\int_{\mathcal{X}} \Omega_1(h(x)(\cdot))\mathrm{d}\mathbf{P}(x) \text{ or } \int_{\Theta} \Omega_2(h(\cdot)(\theta))\mathrm{d}\mu(\theta)$$

  which allow the property enforced by Ω_1 or Ω_2 to hold pointwise on $\mathcal{X}$ or Θ respectively.

It is worthwhile to see a concrete example before turning to the numerical solution (Section 3): in quantile regression, the monotonicity assumption of the $\theta \mapsto h(x)(\theta)$ function can be encoded by choosing Ω_1 as

$$\Omega_1(f) = \lambda_{nc}\int_{\Theta} |-(\partial f)(\theta)|_+ \mathrm{d}\mu(\theta).$$

Many different models ($\mathcal{H}$) could be applied to solve this problem. In our work we consider Reproducing Kernel Hilbert Spaces as they offer a simple and principled way to define regularizers by the appropriate choice of kernels and exhibit a significant flexibility.

3 SOLVING THE PROBLEM IN RKHSs

This section is dedicated to solving the ITL problem defined in Eq. (7). In Section 3.1 we focus on the objective ($\widetilde{V}$). The applied vv-RKHS model family is detailed in Section 3.2 with various penalty examples followed by representer theorems, giving rise to computational tractability.

3.1 Sampled Empirical Risk

In practice solving Eq. (7) can be rather challenging due to the integral over θ. One might consider different numerical integration techniques to handle this issue. We focus here on Quasi Monte Carlo (QMC) methods[2] as they allow (i) efficient optimization over vv-RKHSs which we will use for modelling $\mathcal{H}$ (Proposition 3.1), and (ii) enable us to derive generalization guarantees (Proposition 4.1). Indeed, let

$$\widetilde{V}(y, h(x)) := \sum\nolimits_{j=1}^{m} w_j v(\theta_j, y, h(x)(\theta_j)) \tag{8}$$

be the QMC approximation of Eq. (5). Let $w_j = m^{-1}F^{-1}(\theta_j)$, and $(\theta_j)_{j=1}^m$ be a sequence with values in $[0,1]^d$ such as the Sobol or Halton sequence where μ is assumed to be absolutely continuous w. r. t. the Lebesgue measure and F is the associated cdf. Using this notation and the training samples $\mathcal{S} = ((x_i, y_i))_{i=1}^n$, the empirical risk takes the form

$$\widetilde{R}_{\mathcal{S}}(h) := \frac{1}{n}\sum\nolimits_{i=1}^{n} \widetilde{V}(y_i, h(x_i)) \tag{9}$$

and the problem to solve is

$$\min_{h\in\mathcal{H}} \widetilde{R}_{\mathcal{S}}(h) + \Omega(h). \tag{10}$$

3.2 Hypothesis class ($\mathcal{H}$)

Recall that $\mathcal{H} \subseteq \mathcal{F}(\mathcal{X}; \mathcal{F}(\Theta; \mathcal{Y}))$, in other words $h(x)$ is a $\Theta \mapsto \mathcal{Y}$ function for all $x \in \mathcal{X}$. In this work we assume that the $\Theta \mapsto \mathcal{Y}$ mapping can be described by an RKHS $\mathcal{H}_{k_\Theta}$ associated to a $k_\Theta\colon \Theta \times \Theta \to \mathbb{R}$ scalar-valued kernel defined on the hyperparameters. Let $k_{\mathcal{X}}\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a scalar-valued kernel on the input space. The $x \mapsto$ (hyperparameter $\mapsto$ output) relation, i.e. $h\colon \mathcal{X} \to \mathcal{H}_{k_\Theta}$ is then modelled by the Vector-Valued Reproducing Kernel Hilbert Space $\mathcal{H}_K = \overline{\operatorname{span}}\left\{ K(\cdot, x)f \mid x \in \mathcal{X}, f \in \mathcal{H}_{k_\Theta} \right\}$, where the operator-valued kernel K is defined as $K(x, z) = k_{\mathcal{X}}(x, z)I$, and $I = I_{\mathcal{H}_{k_\Theta}}$ is the identity operator on $\mathcal{H}_{k_\Theta}$.

This so-called decomposable Operator-Valued Kernel has several benefits and gives rise to a function space with a well-known structure. One can consider elements $h \in \mathcal{H}_K$ as mappings from $\mathcal{X}$ to $\mathcal{H}_{k_\Theta}$, and also as functions from $(\mathcal{X} \times \Theta)$ to $\mathbb{R}$. It is indeed known that there is an isometry between $\mathcal{H}_K$ and $\mathcal{H}_{k_{\mathcal{X}}} \otimes \mathcal{H}_{k_\Theta}$, the RKHS associated to the product kernel $k_{\mathcal{X}} \otimes k_\Theta$. The equivalence between these views allows a great flexibility and enables one to follow a functional point of view (to analyse statistical aspects) or to leverage the

[2]See Section S.10.1 of the supplement for a discussion on other integration techniques.

tensor product point of view (to design new kind of penalization schemes). Below we detail various regularizers before focusing on the representer theorems.

- **Ridge Penalty**: For QR and CSC, a natural regularization is the squared vv-RKHS norm

$$\Omega^{\mathrm{RIDGE}}(h) = \frac{\lambda}{2}\|h\|^2_{\mathcal{H}_K}, \quad \lambda > 0. \tag{11}$$

  This choice is amenable to excess risk analysis (see Proposition 4.1). It can be also seen as the counterpart of the classical (multi-task regularization term introduced by Sangnier et al. (2016), compatible with an infinite number of tasks. $\|\cdot\|^2_{\mathcal{H}_K}$ acts by constraining the solution to a ball of a finite radius within the vv-RKHS, whose shape is controlled by both $k_{\mathcal{X}}$ and k_Θ.

- **$L^{2,1}$-penalty**: For DLSE, it is more adequate to apply an $L^{2,1}$-RKHS mixed regularizer:

$$\Omega^{\mathrm{DLSE}}(h) = \frac{1}{2}\int_\Theta \|h(\cdot)(\theta)\|^2_{\mathcal{H}_{k_{\mathcal{X}}}} d\mu(\theta) \tag{12}$$

  which is an example of a Θ-integrated penalty. This Ω choice allows the preservation of the θ-property (see Fig. 2), i. e. that the proportion of the outliers is θ.

- **Shape Constraints**: Taking the example of QR it is advantageous to ensure the monotonicity of the estimated quantile function Let $\partial_\Theta h$ denotes the derivative of $h(x)(\theta)$ with respect to θ. Then one should solve

$$\begin{aligned} &\operatorname*{arg\,min}_{h\in\mathcal{H}_K} \widetilde{R}_S(h) + \Omega^{\mathrm{RIDGE}}(h) \\ &\text{s. t.} \quad \forall (x,\theta) \in \mathcal{X}\times\Theta, (\partial_\Theta h)(x)(\theta) \geqslant 0. \end{aligned}$$

  However, the functional constraint prevents a tractable optimization scheme. To mitigate this bottleneck, we penalize if the derivative of h w. r. t. θ is negative:

$$\Omega_{\mathrm{nc}}(h) := \lambda_{\mathrm{nc}} \int_{\mathcal{X}}\int_\Theta |-(\partial_\Theta h)(x)(\theta)|_+ d\mu(\theta) d\mathbf{P}(x). \tag{13}$$

  When $\mathbf{P} := \mathbf{P}_X$ this penalization can rely on the same anchors and weights as the ones used to approximate the integrated loss function:

$$\widetilde{\Omega}_{\mathrm{nc}}(h) = \lambda_{\mathrm{nc}} \sum\nolimits_{i,j=1}^{n,m} w_j |-(\partial_{\mathcal{X}} h)(x_i)(\theta_j)|_+. \tag{14}$$

  Thus, one can modify the overall regularizer in QR to be

$$\Omega(h) := \Omega^{\mathrm{RIDGE}}(h) + \widetilde{\Omega}_{\mathrm{nc}}(h). \tag{15}$$

3.3 Representer Theorems

Apart from the flexibility of regularizer design, the other advantage of applying vv-RKHS as hypothesis class is that it gives rise to finite-dimensional representation of the ITL solution under mild conditions. The representer theorem Proposition 3.1 applies to CSC when $\lambda_{nc} = 0$ and to QR when $\lambda_{nc} > 0$.

Proposition 3.1 (Representer). *Assume that for $\forall\theta \in \Theta, v_\theta$ is a proper lower semicontinuous convex function with respect to its second argument. Then*

$$\operatorname*{arg\,min}_{h\in\mathcal{H}_K} \widetilde{R}_S(h) + \Omega(h), \quad \lambda > 0$$

with $\Omega(h)$ defined as in Eq. (15), has a unique solution h^*, and $\exists$ $(\alpha_{ij})_{i,j=1}^{n,m}, (\beta_{ij})_{i,j=1}^{n,m} \in \mathbb{R}^{2nm}$ such that $\forall x \in \mathcal{X}$

$$h^*(x) = \sum_{i=1}^n k_{\mathcal{X}}(x,x_i)\left(\sum_{j=1}^m \alpha_{ij}k_\Theta(\cdot,\theta_j) + \beta_{ij}(\partial_2 k_\Theta)(\cdot,\theta_j)\right).$$

Sketch of the proof. *First, we prove that the function to minimize is coercive, convex, lower semicontinuous, hence it has a unique minimum. Then $\mathcal{H}_K$ is decomposed into two orthogonal subspaces and we use the reproducing property to get the finite representation.*

For DLSE, we similarly get a representer theorem with the following modelling choice. Let $k_b : \Theta\times\Theta \to \mathbb{R}$ be a scalar-valued kernel (possibly different from k_θ), $\mathcal{H}_{k_b}$ the associated RKHS and $t \in \mathcal{H}_{k_b}$. Assume also that $\Theta \subseteq [\epsilon, 1]$ where $\epsilon > 0$.[3] Then, learning a continuum of level sets boils down to the minimization problem

$$\operatorname*{arg\,min}_{h\in\mathcal{H}_K, t\in\mathcal{H}_{k_b}} \widetilde{R}_S(h,t) + \widetilde{\Omega}(h,t), \quad \lambda > 0, \tag{16}$$

where $\widetilde{\Omega}(h,t) = \frac{1}{2}\sum_{j=1}^m w_j \|h(\cdot)(\theta_j)\|^2_{\mathcal{H}_{k_{\mathcal{X}}}} + \frac{\lambda}{2}\|t\|^2_{\mathcal{H}_{k_b}}$, $\widetilde{R}_S(h,t) = \frac{1}{n}\sum_{i,j=1}^{n,m} \frac{w_j}{\theta_j}\left(|t(\theta_j) - h(x_i)(\theta_j)|_+ - t(\theta_j)\right)$.

Proposition 3.2 (Representer). *Assume that k_Θ is bounded:* $\sup_{\theta\in\Theta} k_\Theta(\theta,\theta) < +\infty$. *Then the minimization problem described in Eq. (16) has a unique solution (h^*, t^*) and there exist $(\alpha_{ij})_{i,j=1}^{n,m} \in \mathbb{R}^{n\times m}$ and $(\beta_j)_{j=1}^m \in \mathbb{R}^m$ such that for $\forall(x,\theta) \in \mathcal{X}\times[\epsilon, 1]$,*

$$\begin{aligned} h^*(x)(\theta) &= \sum\nolimits_{i,j=1}^{n,m} \alpha_{ij} k_{\mathcal{X}}(x,x_i) k_\Theta(\theta,\theta_j), \\ t^*(\theta) &= \sum\nolimits_{j=1}^{m} \beta_j k_b(\theta,\theta_j). \end{aligned}$$

Sketch of the proof. *First we show that the infimum exists, and that it must be attained in some subspace of $\mathcal{H}_K\times\mathcal{H}_{k_b}$ over which the objective function is coercive. By the reproducing property, we get the claimed finite decomposition.*

Remarks:

[3] We choose $\Theta \subseteq [\epsilon, 1]$, $\epsilon > 0$ rather than $\Theta \subseteq [0, 1]$ because the loss might not be integrable on [0, 1].

Table 1: Quantile Regression on 20 UCI datasets. Reported: 100×value of the pinball loss, 100×crossing loss (smaller is better). p.-val.: outcome of the Mann-Whitney-Wilcoxon test of JQR vs. ∞-QR and Independent vs. ∞-QR. Boldface: significant values w. r. t. ∞-QR.

DATASET	JQR				IND-QR				∞-QR	
	(PINBALL	P.-VAL.)	(CROSS	P.-VAL.)	(PINBALL	P.-VAL.)	(CROSS	P.-VAL.)	PINBALL	CROSS
CobarOre	159 ± 24	$9 \cdot 10^{-01}$	0.1 ± 0.4	$6 \cdot 10^{-01}$	150 ± 21	$2 \cdot 10^{-01}$	0.3 ± 0.8	$7 \cdot 10^{-01}$	165 ± 36	2.0 ± 6.0
engel	175 ± 555	$6 \cdot 10^{-01}$	0.0 ± 0.2	$1 \cdot 10^{+00}$	63 ± 53	$8 \cdot 10^{-01}$	4.0 ± 12.8	$8 \cdot 10^{-01}$	47 ± 6	0.0 ± 0.1
BostonHousing	49 ± 4	$8 \cdot 10^{-01}$	0.7 ± 0.7	$2 \cdot 10^{-01}$	49 ± 4	$8 \cdot 10^{-01}$	$\mathbf{1.3 \pm 1.2}$	$1 \cdot 10^{-05}$	49 ± 4	0.3 ± 0.5
caution	88 ± 17	$6 \cdot 10^{-01}$	0.1 ± 0.2	$6 \cdot 10^{-01}$	89 ± 19	$4 \cdot 10^{-01}$	$\mathbf{0.3 \pm 0.4}$	$2 \cdot 10^{-04}$	85 ± 16	0.0 ± 0.1
ftcollinssnow	154 ± 16	$8 \cdot 10^{-01}$	0.0 ± 0.0	$6 \cdot 10^{-01}$	155 ± 13	$9 \cdot 10^{-01}$	0.2 ± 0.9	$8 \cdot 10^{-01}$	156 ± 17	0.1 ± 0.6
highway	103 ± 19	$4 \cdot 10^{-01}$	0.8 ± 1.4	$2 \cdot 10^{-02}$	99 ± 20	$9 \cdot 10^{-01}$	$\mathbf{6.2 \pm 4.1}$	$1 \cdot 10^{-07}$	105 ± 36	0.1 ± 0.4
heights	127 ± 3	$1 \cdot 10^{+00}$	0.0 ± 0.0	$1 \cdot 10^{+00}$	127 ± 3	$9 \cdot 10^{-01}$	0.0 ± 0.0	$1 \cdot 10^{+00}$	127 ± 3	0.0 ± 0.0
sniffer	43 ± 6	$8 \cdot 10^{-01}$	0.1 ± 0.3	$2 \cdot 10^{-01}$	44 ± 5	$7 \cdot 10^{-01}$	$\mathbf{1.4 \pm 1.2}$	$6 \cdot 10^{-07}$	44 ± 7	0.1 ± 0.1
snowgeese	55 ± 20	$7 \cdot 10^{-01}$	0.3 ± 0.8	$3 \cdot 10^{-01}$	53 ± 18	$6 \cdot 10^{-01}$	0.4 ± 1.0	$5 \cdot 10^{-02}$	57 ± 20	0.2 ± 0.6
ufc	81 ± 5	$6 \cdot 10^{-01}$	$\mathbf{0.0 \pm 0.0}$	$4 \cdot 10^{-04}$	82 ± 5	$7 \cdot 10^{-01}$	$\mathbf{1.0 \pm 1.4}$	$2 \cdot 10^{-04}$	82 ± 4	0.1 ± 0.3
BigMac2003	80 ± 21	$7 \cdot 10^{-01}$	$\mathbf{1.4 \pm 2.1}$	$4 \cdot 10^{-04}$	74 ± 24	$9 \cdot 10^{-02}$	$\mathbf{0.9 \pm 1.1}$	$7 \cdot 10^{-05}$	84 ± 24	0.2 ± 0.4
UN3	98 ± 9	$8 \cdot 10^{-01}$	0.0 ± 0.0	$1 \cdot 10^{-01}$	99 ± 9	$1 \cdot 10^{+00}$	$\mathbf{1.2 \pm 1.0}$	$1 \cdot 10^{-05}$	99 ± 10	0.1 ± 0.4
birthwt	141 ± 13	$1 \cdot 10^{+00}$	0.0 ± 0.0	$6 \cdot 10^{-01}$	140 ± 12	$9 \cdot 10^{-01}$	0.1 ± 0.2	$7 \cdot 10^{-02}$	141 ± 12	0.0 ± 0.0
crabs	$\mathbf{11 \pm 1}$	$4 \cdot 10^{-05}$	0.0 ± 0.0	$8 \cdot 10^{-01}$	$\mathbf{11 \pm 1}$	$2 \cdot 10^{-04}$	$\mathbf{0.0 \pm 0.0}$	$2 \cdot 10^{-05}$	13 ± 3	0.0 ± 0.0
GAGurine	61 ± 7	$4 \cdot 10^{-01}$	0.0 ± 0.1	$3 \cdot 10^{-03}$	62 ± 7	$5 \cdot 10^{-01}$	$\mathbf{0.1 \pm 0.2}$	$4 \cdot 10^{-04}$	62 ± 7	0.0 ± 0.0
geyser	105 ± 7	$9 \cdot 10^{-01}$	0.1 ± 0.3	$9 \cdot 10^{-01}$	105 ± 6	$9 \cdot 10^{-01}$	0.2 ± 0.3	$6 \cdot 10^{-01}$	104 ± 6	0.1 ± 0.2
gilgais	51 ± 6	$5 \cdot 10^{-01}$	0.1 ± 0.1	$1 \cdot 10^{-01}$	49 ± 6	$6 \cdot 10^{-01}$	$\mathbf{1.1 \pm 0.7}$	$2 \cdot 10^{-05}$	49 ± 7	0.3 ± 0.3
topo	69 ± 18	$1 \cdot 10^{+00}$	0.1 ± 0.5	$1 \cdot 10^{+00}$	71 ± 20	$1 \cdot 10^{+00}$	$\mathbf{1.7 \pm 1.4}$	$3 \cdot 10^{-07}$	70 ± 17	0.0 ± 0.0
mcycle	66 ± 9	$9 \cdot 10^{-01}$	0.2 ± 0.3	$7 \cdot 10^{-03}$	66 ± 8	$9 \cdot 10^{-01}$	$\mathbf{0.3 \pm 0.3}$	$7 \cdot 10^{-06}$	65 ± 9	0.0 ± 0.1
cpus	$\mathbf{7 \pm 4}$	$2 \cdot 10^{-04}$	$\mathbf{0.7 \pm 1.0}$	$5 \cdot 10^{-04}$	$\mathbf{7 \pm 5}$	$3 \cdot 10^{-04}$	$\mathbf{1.2 \pm 0.8}$	$6 \cdot 10^{-08}$	16 ± 10	0.0 ± 0.0

- Models with bias: it can be advantageous to add a bias to the model, which is here a function of the hyperparameter θ: $h(x)(\theta) = f(x)(\theta) + b(\theta)$, $f \in \mathcal{H}_K$, $b \in \mathcal{H}_{k_b}$, where $k_b : \Theta \times \Theta \to \mathbb{R}$ is a scalar-valued kernel. This can be the case for example if the kernel on the hyperparameters is the constant kernel, i. e. $k_\Theta(\theta, \theta') = 1$ ($\forall \theta, \theta' \in \Theta$), hence the model $f(x)(\theta)$ would not depend on θ. An analogous statement to Proposition 3.1 still holds for the biased model if one adds a regularization $\lambda_b \|b\|^2_{\mathcal{H}_{k_b}}$, $\lambda_b > 0$ to the risk.
- Relation to JQR: In ∞-QR, by choosing k_Θ to be the Gaussian kernel, $k_b(x, z) = \mathbb{1}_{\{x\}}(z)$, $\mu = \frac{1}{m}\sum_{j=1}^m \delta_{\theta_j}$, where δ_θ is the Dirac measure concentrated on θ, one gets back Sangnier et al. (2016)'s Joint Quantile Regression (JQR) framework as a special case of our approach. In contrast to the JQR, however, in ∞-QR one can predict the quantile value at any $\theta \in (0, 1)$, even outside the $(\theta_j)_{j=1}^m$ used for learning.
- Relation to q-OCSVM: In DLSE, by choosing $k_\Theta(\theta, \theta') = 1$ (for all $\theta, \theta' \in \Theta$) to be the constant kernel, $k_b(\theta, \theta') = \mathbb{1}_{\{\theta\}}(\theta')$, $\mu = \frac{1}{m}\sum_{j=1}^m \delta_{\theta_j}$, our approach specializes to q-OCSVM (Glazer et al., 2013).
- Relation to Kadri et al. (2016): Note that Operator-Valued Kernels for functional outputs have also been used in (Kadri et al., 2016), under the form of integral operators acting on L^2 spaces. Both kernels give rise to the same space of functions; the benefit of our approach being to provide an *exact* finite representation of the solution (see Proposition 3.1).
- Efficiency of the decomposable kernel: this kernel choice allows to rewrite the expansions in Propositions 3.1 and 3.2 as a Kronecker products and the complexity of the prediction of n' points for m' quantile becomes $\mathcal{O}(m'mn + n'nm)$ instead of $\mathcal{O}(m'mn'n)$.

4 Excess Risk Bounds

Below we provide a generalization error analysis to the solution of Eq. (10) for QR and CSC (with Ridge regularization and without shape constraints) by stability argument (Bousquet et al., 2002), extending the work of Audiffren et al. (2013) to Infinite-Task Learning. The proposition (finite sample bounds are given in Corollary S.9.6) instantiates the guarantee for the QMC scheme.

Proposition 4.1 (Generalization)**.** *Let $h^* \in \mathcal{H}_K$ be the solution of Eq. (10) for the* QR *or* CSC *problem with* QMC *approximation. Under mild conditions on the kernels $k_\mathcal{X}, k_\Theta$ and $\mathbf{P}_{X,Y}$, stated in the supplement, one has*

$$R(h^*) \leqslant \widetilde{R}_S(h^*) + \mathcal{O}_{\mathbf{P}_{X,Y}}\left(\frac{1}{\sqrt{\lambda n}}\right) + \mathcal{O}\left(\frac{\log(m)}{\sqrt{\lambda} m}\right). \quad (17)$$

Sketch of the proof. *The error resulting from sam-*

pling $\mathbf{P}_{X,Y}$ *and the inexact integration is respectively bounded by* β*-stability (Kadri et al., 2016) and* QMC *results.*[4]

$(\mathbf{n}, \mathbf{m})$ Trade-off: The proposition reveals the interplay between the two approximations, n (the number of training samples) and m (the number of locations taken in the integral approximation), and allows to identify the regime in $\lambda = \lambda(n, m)$ driving the excess risk to zero. Indeed by choosing $m = \sqrt{n}$ and discarding logarithmic factors for simplicity, $\lambda \gg n^{-1}$ is sufficient. The mild assumptions imposed are: boundedness on both kernels and the random variable Y, as well as some smoothness of the kernels.

5 Numerical Examples

In this section we provide numerical examples illustrating the efficiency of the proposed ITL approach.[5] We used the following datasets in our experiments:

- Quantile Regression: we used (i) a sine synthetic benchmark (Sangnier et al., 2016): a sine curve at 1Hz modulated by a sine envelope at 1/3Hz and mean 1, distorted with a Gaussian noise of mean 0 and a linearly decreasing standard deviation from 1.2 at $x = 0$ to 0.2 at $x = 1.5$. (ii) 20 standard regression datasets from UCI. The number of samples varied between 38 (CobarOre) and 1375 (Height). The observations were standardised to have unit variance and zero mean for each attribute.
- Density Level-Set Estimation: The Wilt database from the UCI repository with 4839 samples and 5 attributes, and the Spambase UCI dataset with 4601 samples and 57 attributes served as benchmarks.

Additional experiments related to the CSC problem are provided in Section S.10.5.

Note on Optimization: There are several ways to solve the non-smooth optimization problems associated to the QR, DLSE and CSC tasks. One could proceed for example by duality—as it was done in JQR Sangnier et al. (2016)—, or apply sub-gradient descent techniques (which often converge quite slowly). In order to allow unified treatment and efficient solution in our experiments we used the L-BFGS-B (Zhu et al., 1997) optimization scheme which is widely popular in large-scale learning, with non-smooth extensions (Keskar et al., 2017; Skajaa, 2010). The technique requires only evaluation of objective function along with its gradient, which can be computed automatically using reverse mode automatic differentiation (as in Abadi et al. (2016)). To benefit from from the available fast smooth implementations (Fei et al., 2014; Jones et al., 2001), we applied an infimal convolution (see Section S.10.3 of the supplementary material) on the non-differentiable terms of the objective. Under the assumtion that $m = \mathcal{O}(\sqrt{n})$ (see Proposition 4.1), the complexity per L-BFGS-B iteration is $\mathcal{O}(n^2\sqrt{n})$. An experiment showing the impact of increasing m on a synthetic dataset is provided in Fig. S.4.

QR: The efficiency of the non-crossing penalty is illustrated in Fig. 1 on the synthetic sine wave dataset described in Section 5 where $n = 40$ and $m = 20$ points have been generated. Many crossings are visible on the right plot, while they are almost not noticible on the left plot, using the non-crossing penalty. Concerning our real-world examples, to study the efficiency of the proposed scheme in quantile regression the following experimental protocol was applied. Each dataset (Section 5) was splitted randomly into a training set (70%) and a test set (30%). We optimized the hyperparameters by minimizing a 5-folds cross validation with a Bayesian optimizer[6] (For further details see Section S.10.4). Once the hyperparameters were obtained, a new regressor was learned on the whole training set using the optimized hyperparameters. We report the value of the pinball loss and the crossing loss on the test set for three methods: our technique is called ∞-QR, we refer to Sangnier et al. (2016)'s approach as JQR, and independent learning (abbreviated as IND-QR) represents a further baseline.

We repeated 20 simulations (different random training-test splits); the results are also compared using a Mann-Whitney-Wilcoxon test. A summary is provided in Table 1. Notice that while JQR is tailored to predict finite many quantiles, our ∞-QR method estimates the *whole quantile function* hence solves a more challenging task. Despite the more difficult problem solved, as Table 1 suggest that the performance in terms of pinball loss of ∞-QR is comparable to that of the state-of-the-art JQR on all the twenty studied benchmarks, except for the 'crabs' and 'cpus' datasets (p.-val. $< 0.25\%$). In addition, when considering the non-crossing penalty one can observe that ∞-QR outperforms the IND-QR baseline on eleven datasets (p.-val. $< 0.25\%$) and JQR on two datasets. This illustrates the efficiency of the constraint based on the continuum scheme.

DLSE: To assess the quality of the estimated model by ∞-OCSVM, we illustrate the θ-property

[4]The QMC approximation may involve the Sobol sequence with discrepancy $m^{-1}\log(m)^s$ $(s = \dim(\Theta))$.

[5]The code is available at `https://bitbucket.org/RomainBrault/itl`.

[6]We used a Gaussian Process model and minimized the Expected improvement. The optimizer was initialized using 27 samples from a Sobol sequence and ran for 50 iterations.

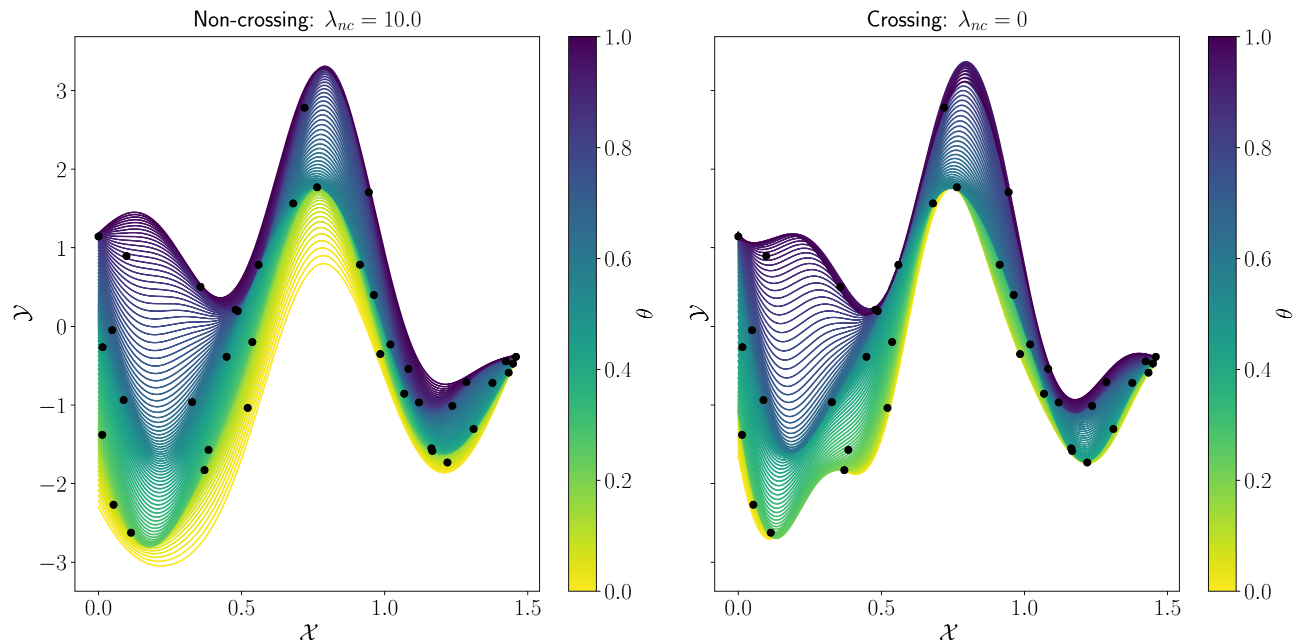

Figure 1: Impact of crossing penalty on toy data. Left plot: strong non-crossing penalty ($\lambda_{nc} = 10$). Right plot: no non-crossing penalty ($\lambda_{nc} = 0$). The plots show 100 quantiles of the continuum learned, linearly spaced between 0 (blue) and 1 (red). Notice that the non-crossing penalty does not provide crossings to occur in the regions where there is no points to enforce the penalty (e. g. $x \in [0.13, 0.35]$). This phenomenon is alleviated by the regularity of the model.

(Schölkopf et al., 2000): the proportion of inliers has to be approximately $1 - \theta$ ($\forall \theta \in (0, 1)$). For the studied datasets (Wilt, Spambase) we used the raw inputs without applying any preprocessing. Our input kernel was the exponentiated χ^2 kernel $k_{\mathcal{X}}(x, z) := \exp\left(-\gamma_{\mathcal{X}} \sum_{k=1}^{d} (x_k - z_k)^2/(x_k + z_k)\right)$ with bandwidth $\gamma_{\mathcal{X}} = 0.25$. A Gauss-Legendre quadrature rule provided the integral approximation in Eq. (8), with $m = 100$ samples. We chose the Gaussian kernel for k_Θ; its bandwidth parameter γ_Θ was the $0.2-$quantile of the pairwise Euclidean distances between the θ_j's obtained via the quadrature rule. The margin (bias) kernel was $k_b = k_\Theta$. As it can be seen in Fig. 2, the θ-property holds for the estimate which illustrates the efficiency of the proposed continuum approach for density level-set estimation.

6 Conclusion

In this work we proposed Infinite Task Learning, a novel nonparametric framework aiming at jointly solving parametrized tasks for a continuum of hyperparameters. We provided excess risk guarantees for the studied ITL scheme, and demonstrated its practical efficiency and flexibility in various tasks including cost-sensitive classification, quantile regression and density level set estimation.

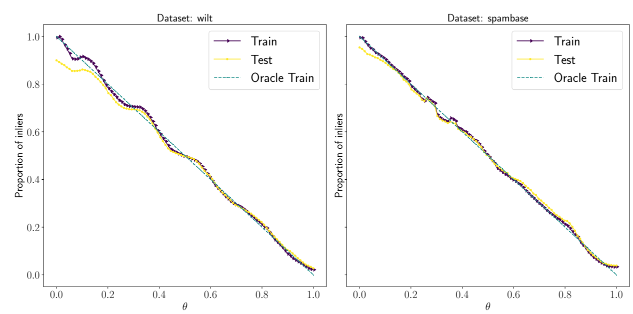

Figure 2: Density Level-Set Estimation: the θ-property is approximately satisfied.

Acknowledgments

The authors thank Arthur Tenenhaus for some insightful discussions. This work was supported by the Labex *DigiCosme* (project ANR-11-LABEX-0045-DIGICOSME) and the industrial chair *Machine Learning for Big Data* at Télécom ParisTech.

References

Abadi, M. et al. (2016). “Tensorflow: Large-scale machine learning on heterogeneous distributed systems.” In: *USENIX Symposium on Operating Sys-*

tems Design and Implementation (OSDI), pp. 265–283 (cit. on p. 7).

Álvarez, M. A., L. Rosasco, and N. D. Lawrence (2012). "Kernels for vector-valued functions: a review." In: *Foundations and Trends in Machine Learning* 4.3, pp. 195–266 (cit. on p. 3).

Audiffren, J. and H. Kadri (2013). "Stability of Multi-Task Kernel Regression Algorithms." In: *Asian Conference on Machine Learning (ACML)*. Vol. 29. PMLR, pp. 1–16 (cit. on p. 6).

Bach, F. et al. (2012). "Optimization with sparsity-inducing penalties." In: *Foundations and Trends in Machine Learning* 4.1, pp. 1–106 (cit. on p. 1).

Bousquet, O. and A. Elisseeff (2002). "Stability and generalization." In: *Journal of Machine Learning Research* 2, pp. 499–526 (cit. on p. 6).

Carmeli, C. et al. (2010). "Vector valued reproducing kernel Hilbert spaces and universality." In: *Analysis and Applications* 8 (1), pp. 19–61 (cit. on p. 2).

Ciliberto, C. et al. (2017). "Consistent multitask learning with nonlinear output relations." In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 1986–1996 (cit. on p. 3).

Evgeniou, T., C. A. Micchelli, and M. Pontil (2005). "Learning Multiple Tasks with kernel methods." In: *JMLR* 6, pp. 615–637 (cit. on p. 3).

Evgeniou, T. and M. Pontil (2004). "Regularized multi–task learning." In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 109–117 (cit. on p. 1).

Fei, Y. et al. (2014). "Parallel L-BFGS-B algorithm on GPU." In: *Computers & Graphics* 40, pp. 1–9 (cit. on p. 7).

Glazer, A., M. Lindenbaum, and S. Markovitch (2013). "q-OCSVM: A q-quantile estimator for high-dimensional distributions." In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 503–511 (cit. on pp. 1–3, 6).

Hastie, T. et al. (2004). "The entire regularization path for the support vector machine." In: *Journal of Machine Learning Research* 5.Oct, pp. 1391–1415 (cit. on p. 1).

Jones, E., T. Oliphant, P. Peterson, et al. (2001). *SciPy: Open source scientific tools for Python* (cit. on p. 7).

Kadri, H. et al. (2016). "Operator-valued Kernels for Learning from Functional Response Data." In: *Journal of Machine Learning Research* 17, pp. 1–54 (cit. on pp. 6, 7).

Keskar, N. and A. Wächter (2017). "A limited-memory quasi-Newton algorithm for bound-constrained non-smooth optimization." In: *Optimization Methods and Software*, pp. 1–22 (cit. on p. 7).

Koenker, R. and G. Bassett Jr (1978). "Regression quantiles." In: *Econometrica: journal of the Econometric Society*, pp. 33–50 (cit. on pp. 1, 2).

Pedrick, G. (1957). "Theory of reproducing kernels for Hilbert spaces of vector-valued functions." PhD thesis. University of Kansas (cit. on p. 1).

Sangnier, M., O. Fercoq, and F. d'Alché-Buc (2016). "Joint quantile regression in vector-valued RKHSs." In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 3693–3701 (cit. on pp. 2, 3, 5–7).

Schölkopf, B. et al. (2000). "New support vector algorithms." In: *Neural computation* 12.5, pp. 1207–1245 (cit. on pp. 1, 3, 8).

Skajaa, A. (2010). "Limited memory BFGS for non-smooth optimization." In: *Master's thesis* (cit. on p. 7).

Stein, O. (2012). "How to solve a semi-infinite optimization problem." In: *European Journal of Operational Research* 223.2, pp. 312–320 (cit. on p. 1).

Takeuchi, I., T. Hongo, et al. (2013). "Parametric task learning." In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 1358–1366 (cit. on pp. 1, 3).

Takeuchi, I., Q. V. Le, et al. (2006). "Nonparametric quantile estimation." In: *Journal of Machine Learning Research* 7, pp. 1231–1264 (cit. on p. 1).

Tibshirani, R. (1996). "Regression shrinkage and selection via the Lasso." In: *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288 (cit. on p. 1).

Tikhonov, A. N. and V. Y. Arsenin (1977). *Solution of Ill-posed Problems*. Winston & Sons (cit. on p. 1).

Vapnik, V., S. E. Golowich, and A. J. Smola (1997). "Support vector method for function approximation, regression estimation and signal processing." In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 281–287 (cit. on p. 1).

Vert, R. and J.-P. Vert (2006). "Consistency and convergence rates of one-class SVMs and related algorithms." In: *Journal of Machine Learning Research* 7, pp. 817–854 (cit. on p. 3).

Wen, U.-P. and S.-T. Hsu (1991). "Linear bi-level programming problemsa review." In: *Journal of the Operational Research Society* 42.2, pp. 125–133 (cit. on p. 1).

Zadrozny, B. and C. Elkan (2001). "Learning and making decisions when costs and probabilities are both unknown." In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 204–213 (cit. on p. 1).

Zhu, C. et al. (1997). "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization." In: *ACM Transactions on Mathematical Software (TOMS)* 23.4, pp. 550–560 (cit. on p. 7).

SUPPLEMENTARY MATERIAL

Acronyms

CSC	Cost-Sensitive Classification
DLSE	Density Level-Set Estimation
e. g.	exempli gratia
∞-CSC	Infinite Cost-Sensitive Classification
i. e.	id est
i. i. d.	independent identically distributed
ITL	Infinite Task Learning
L-BFGS-B	Limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm for Bound constraind optimization
MC	Monte-Carlo
MTL	Multi-Task Learning
OCSVM	One-Class Support Vector Machine
OVK	Operator-Valued Kernel
PTL	Parametric Task Learning
QMC	Quasi Monte Carlo
QR	Quantile Regression
RKHS	Reproducing Kernel Hilbert Space
r. v.	random variable
vv-RKHS	Vector-Valued Reproducing Kernel Hilbert Space
w. r. t.	with respect to

Below we provide the proofs of the results stated in the main part of the paper.

S.7 Quantile Regression

Let us recall the expression of the pinball loss (see Fig. S.3):

$$v_\theta : (y, y') \in \mathbb{R}^2 \mapsto \max\left(\theta(y - y'), (\theta - 1)(y - y')\right) \in \mathbb{R}. \tag{18}$$

Proposition S.7.1. *Let X, Y be two random variables (sr. v.s) respectively taking values in $\mathcal{X}$ and $\mathbb{R}$, and $q\colon \mathcal{X} \to \mathcal{F}([0,1], \mathbb{R})$ the associated conditional quantile function. Let μ be a positive measure on $[0,1]$ such that $\int_0^1 \mathbf{E}\left[v_\theta\left(Y, q(X)(\theta)\right)\right] d\mu(\theta) < \infty$. Then for $\forall h \in \mathcal{F}\left(\mathcal{X}; \mathcal{F}\left([0,\,1]\,;\,\mathbb{R}\right)\right)$*

$$R(h) - R(q) \geqslant 0,$$

where R is the risk defined in Eq. (6).

Proof. The proof is based on the one given in (Li et al., 2007) for a single quantile. Let $f \in \mathcal{F}\left(\mathcal{X}; \mathcal{F}\left([0,\,1]\,;\,\mathbb{R}\right)\right)$, $\theta \in (0,1)$ and $(x, y) \in \mathcal{X} \times \mathbb{R}$. Let also

$$s = \begin{cases} 1 \text{ if } y \leqslant f(x)(\theta) \\ 0 \text{ otherwise} \end{cases}, \qquad t = \begin{cases} 1 \text{ if } y \leqslant q(x)(\theta) \\ 0 \text{ otherwise} \end{cases}.$$

It holds that

$$\begin{aligned}\nu_\theta(y, h(x)(\theta)) - \nu_\theta(y, q(x)(\theta)) &= \theta(1-s)(y - h(x)(\theta)) + (\theta-1)s(y-h(x)(\theta)) \\ &\quad - \theta(1-t)(y-q(x)(\theta)) - (\theta-1)t(y-q(x)(\theta)) \\ &= \theta(1-t)(q(x)(\theta) - h(x)(\theta)) + \theta((1-t)-(1-s))h(x)(\theta) \\ &\quad + (\theta-1)t(q(x)(\theta - h(x)(\theta))) + (\theta-1)(t-s)h(x)(\theta) + (t-s)y \\ &= (\theta-t)(q(x)(\theta) - h(x)(\theta)) + (t-s)(y-h(x)(\theta)).\end{aligned}$$

Then, notice that

$$\mathbf{E}[(\theta-t)(q(X)(\theta)-h(X)(\theta))] = \mathbf{E}[\mathbf{E}[(\theta-t)(q(X)(\theta)-h(X)(\theta))]|X] = \mathbf{E}[\mathbf{E}[(\theta-t)|X](q(X)(\theta)-h(X)(\theta))]$$

and since q is the true quantile function,

$$\mathbf{E}[t|X] = \mathbf{E}[\mathbf{1}_{\{Y \leqslant q(X)(\theta)\}}|X] = \mathbf{P}[Y \leqslant q(X)(\theta)|X] = \theta,$$

so

$$\mathbf{E}[(\theta-t)(q(X)(\theta)-h(X)(\theta))] = 0.$$

Moreover, $(t-s)$ is negative when $q(x)(\theta) \leqslant y \leqslant h(x)(\theta)$, positive when $h(x)(\theta) \leqslant y \leqslant q(x)(\theta)$ and 0 otherwise, thus the quantity $(t-s)(y-h(x)(\theta))$ is always positive. As a consequence,

$$R(h) - R(q) = \int_{[0,1]} \mathbf{E}[\nu_\theta(Y, h(X)(\theta)) - \nu_\theta(Y, q(X)(\theta))]d\mu(\theta) \geqslant 0$$

which concludes the proof. □

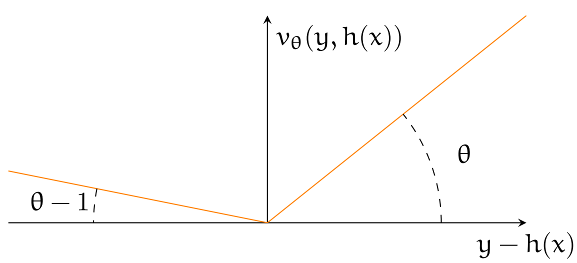

Figure S.3: Pinball loss for $\theta = 0.8$.

The Proposition S.7.1 allows us to derive conditions under which the minimization of the risk above yields the true quantile function. Under the assumption that (i) q is continuous (as seen as a function of two variables), (ii) $\mathrm{Supp}(\mu) = [0,1]$, then the minimization of the integrated pinball loss performed in the space of continuous functions yields the true quantile function on the support of $\mathbf{P}_{X,Y}$.

S.8 Representer Propositions

Proof of Proposition 3.1. First notice that

$$J : h \in \mathcal{H}_K \mapsto \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} w_j \nu(\theta_j, y_i, h(x_i)(\theta_j)) + \frac{\lambda}{2}\|h\|^2_{\mathcal{H}_K} \in \mathbb{R} \tag{19}$$

is a proper lower semicontinuous strictly convex function (Bauschke et al., 2011, Corollary 9.4), hence J admits a unique minimizer $h^* \in \mathcal{H}_K$ (Bauschke et al., 2011, Corollary 11.17). Let

$$\mathcal{U} = \operatorname{span}\left\{ (K(\cdot, x_i)k_\Theta(\cdot, \theta_j))_{i,j=1}^{n,m} \;\middle|\; \forall x_i \in \mathcal{X}, \forall \theta_j \in \Theta \right\} \subset \mathcal{H}_K. \tag{20}$$

Then $\mathcal{U}$ is a finite-dimensional subspace of $\mathcal{H}_K$, thus closed in $\mathcal{H}_K$, and it holds that $\mathcal{U} \oplus \mathcal{U}^\perp = \mathcal{H}_K$, so h^* can be decomposed as $h^* = h^*_{\mathcal{U}} + h^*_{\mathcal{U}^\perp}$ with $h^*_{\mathcal{U}} \in \mathcal{U}$ and $h^*_{\mathcal{U}^\perp} \in \mathcal{U}^\perp$. Moreover, for all $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant m$,

$$h^*_{\mathcal{U}^\perp}(x_i)(\theta_j) = \langle h^*_{\mathcal{U}^\perp}(x_i), k_\Theta(\cdot, \theta_j)\rangle_{\mathcal{H}_{k_\Theta}} = \langle h^*_{\mathcal{U}^\perp}, K(\cdot, x_i)k_\Theta(\cdot, \theta_j)\rangle_{\mathcal{H}_K} = 0,$$

so $J(h^*) = J(h^*_{\mathcal{U}}) + \lambda \left\|h^*_{\mathcal{U}^\perp}\right\|^2_{\mathcal{H}_K}$. However h^* is the minimizer of J, therefore $h^*_{\mathcal{U}^\perp} = 0$ and there exist $(\alpha_{ij})_{i,j=1}^{n,m}$ such that $\forall x, \theta \in \mathcal{X} \times \Theta$, $h^*(x)(\theta) = \sum_{i,j=1}^{n,m} \alpha_{ij} k_{\mathcal{X}}(x, x_i) k_\Theta(\theta, \theta_j)$.

Derivative shapes constraints: Reminder: for a function h of one variable, we denote ∂h the derivative of h. For a function $k(\theta, \theta')$ of two variables we denote $\partial_1 k$ the derivative of k with respect to θ and $\partial_2 k$ the derivative of k with respect to θ'. From Zhou (2008), notice that if $f \in \mathcal{H}_k$, where $\mathcal{H}_k$ is a scalar-valued RKHS on a compact subset Θ of $\mathbb{R}^d$, and $k \in \mathcal{C}^2(\Theta \times \Theta)$ (in the sense of Ziemer (2012)) then $\partial f \in \mathcal{H}_k$. Hence if one add a new term of the form:

$$\lambda_{\text{nc}} \sum_{i=1}^{n} \sum_{j=1}^{m} \Omega_{\text{nc}}\left((\partial \left[h(x_i)\right]) (\theta_j) \right) = \lambda_{\text{nc}} \sum_{i=1}^{n} \sum_{j=1}^{m} \Omega_{\text{nc}}\left((\partial h(x_i))(\theta_j) \right)$$

where g is a strictly monotonically increasing function and $\lambda_{\text{nc}} > 0$, a new representer theorem can be obtained by constructing the new set

$$\mathcal{U} = \operatorname{span} \left\{ (K(\cdot, x_i)k_\Theta(\cdot, \theta_j))_{i,j=1}^{n,m} \;\middle|\; \forall x_i \in \mathcal{X}, \forall \theta_j \in \Theta \right\} \cup \left\{ (K(\cdot, x_i)(\partial_2 k_\Theta)(\cdot, \theta_j))_{i,j=1}^{n,m} \;\middle|\; \forall x_i \in \mathcal{X}, \forall \theta_j \in \Theta \right\} \subset \mathcal{H}_K.$$

The proof is the same as Proposition 3.1 with the new set $\mathcal{U}$ to obtain the expansion $h(x)(\theta) = \sum_{i=1}^n \sum_{j=1}^m \alpha_{ij} k_{\mathcal{X}}(x, x_i) k_\Theta(\theta, \theta_j) + \beta_{ij} k(x, x_i)(\partial_2 k_\Theta)(\theta, \theta_j)$. For the regularization notice that for a symmetric function $(\partial_1 k)(\theta, \theta') = (\partial_2 k)(\theta', \theta)$. Hence $\langle (\partial_1 k)(\cdot, \theta'), k(\cdot, \theta)\rangle_{\mathcal{H}_k} = \langle k(\cdot, \theta'), (\partial_2 k)(\cdot, \theta)\rangle_{\mathcal{H}_k}$ and $(\partial k_{\theta'})(\theta) = (\partial^* k_\theta)(\theta')$ and

$$\begin{aligned}
\|h\|^2_{\mathcal{H}_K} &= \langle h, h\rangle_{\mathcal{H}_K} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{i'=1}^{n} \sum_{j'=1}^{m} \alpha_{ij}\alpha_{i'j'} k_{\mathcal{X}}(x_i, x_{i'}) k_\Theta(\theta_j, \theta_{j'}) + \alpha_{ij}\beta_{i'j'} k_{\mathcal{X}}(x_i, x_{i'})(\partial_2 k_\Theta)(\theta_j, \theta_{j'}) \\
&\quad + \alpha_{i'j'}\beta_{ij} k_{\mathcal{X}}(x_i, x_{i'})(\partial_1 k_\Theta)(\theta_j, \theta_{j'}) + \beta_{ij}\beta_{i'j'} k_{\mathcal{X}}(x_i, x_{i'})(\partial_1 \partial_2 k_\Theta)(\theta_j, \theta_{j'})
\end{aligned}$$

Eventually $(\partial h(x))(\theta) = \sum_{i=1}^n \sum_{j=1}^m \alpha_{ij} k_{\mathcal{X}}(x, x_i)(\partial_1 k_\Theta)(\theta, \theta_j) + \beta_{ij} k(x, x_i)(\partial_1 \partial_2 k_\Theta)(\theta, \theta_j)$. □

To prove Proposition 3.2, the following lemmas are useful.

Lemma S.8.1. *(Carmeli et al., 2010) Let* $k_{\mathcal{X}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $k_\Theta : \Theta \times \Theta \to \mathbb{R}$ *be two scalar-valued kernels and* $K(\theta', \theta) = k_\Theta(\theta, \theta') I_{\mathcal{H}_{k_{\mathcal{X}}}}$. *Then* H_K *is isometric to* $\mathcal{H}_{k_{\mathcal{X}}} \otimes \mathcal{H}_{k_\Theta}$ *by means of the isometry* $W : f \otimes g \in \mathcal{H}_{k_{\mathcal{X}}} \otimes \mathcal{H}_{k_\Theta} \mapsto (\theta \mapsto g(\theta) f) \in \mathcal{H}_K$.

Remark 1. *Given* $k_{\mathcal{X}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $k_\Theta : \Theta \times \Theta \to \mathbb{R}$ *two scalar-valued kernels, we define* $K : (x, z) \in \mathcal{X} \times \mathcal{X} \mapsto k_{\mathcal{X}}(x, z) I_{\mathcal{H}_{k_\Theta}} \in \mathcal{L}(\mathcal{H}_{k_\Theta})$, $K' : (\theta, \theta') \in \Theta \times \Theta \mapsto k_\Theta(\theta, \theta') I_{\mathcal{H}_{k_{\mathcal{X}}}} \in \mathcal{L}(\mathcal{H}_{k_{\mathcal{X}}})$. *Lemma S.8.1 allows us to say that* $\mathcal{H}_K$ *and* $\mathcal{H}_{K'}$ *are isometric by means of the isometry*

$$W : h \in \mathcal{H}_{K'} \mapsto (x \mapsto (\theta \mapsto h(\theta)(x))) \in \mathcal{H}_K. \tag{21}$$

Lemma S.8.2. *Let* $k_{\mathcal{X}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $k_\Theta : \Theta \times \Theta \to \mathbb{R}$ *be two scalar-valued kernels and* $K : (\theta, \theta') \mapsto k_\Theta(\theta, \theta') I_{\mathcal{H}_{k_{\mathcal{X}}}}$. *For* $\theta \in \Theta$, *define* $K_\theta : f \in \mathcal{H}_{k_{\mathcal{X}}} \mapsto (\theta' \mapsto K(\theta', \theta) f) \in \mathcal{H}_K$. *It is easy to see that* K^*_θ *is the evaluation operator* $K^*_\theta : h \in \mathcal{H}_K \mapsto h(\theta) \in \mathcal{H}_{k_{\mathcal{X}}}$. *Then* $\forall m \in \mathbb{N}^*, \forall (\theta_j)_{j=1}^m \in \Theta^m$,

$$\left(+_{j=1}^m \operatorname{Im}\left(K_{\theta_j}\right) \right) \oplus \left(\cap_{j=1}^m \operatorname{Ker}\left(K^*_{\theta_j}\right) \right) = \mathcal{H}_K \tag{22}$$

Proof. The statement boils down to proving that $\mathcal{V} := \left(+_{j=1}^m \operatorname{Im}\left(K_{\theta_j}\right)\right)$ is closed in $\mathcal{H}_K$, since it is straightforward that $\mathcal{V}^\perp = \left(\cap_{j=1}^m \operatorname{Ker}\left(K^*_{\theta_j}\right)\right)$. Let $(e_j)_{j=1}^k$ be an orthonormal basis of span $\left\{ (k_\Theta(\cdot,\theta_j))_{j=1}^m \right\} \subset \mathcal{H}_{k_\Theta}$. Such basis can be obtained by applying the Gram-Schmidt orthonormalization method to $(k_\Theta(\cdot,\theta_j))_{j=1}^m$. Then, $V = \operatorname{span}\left\{ e_j \cdot f, 1 \leqslant j \leqslant k, f \in \mathcal{H}_{k_\mathcal{X}} \right\}$. Notice also that $1 \leqslant j, l \leqslant k, \forall f, g \in \mathcal{H}_{k_\mathcal{X}}$,

$$\langle e_j \cdot f, e_l \cdot g \rangle_{\mathcal{H}_K} = \langle e_j, e_l \rangle_{\mathcal{H}_{k_\Theta}} \cdot \langle f, g \rangle_{\mathcal{H}_{k_\mathcal{X}}} \tag{23}$$

Let $(h_n)_{n\in\mathbb{N}^*}$ be a sequence in $\mathcal{V}$ converging to some $h \in \mathcal{H}_K$. By definition, one can find sequences $(f_{1,n})_{n\in\mathbb{N}^*}, \dots, (f_{k,n})_{n\in\mathbb{N}^*} \in \mathcal{H}_{k_\mathcal{X}}$ such that $\forall n \in \mathbb{N}^*$, $h_n = \sum_{j=1}^k e_j \cdot f_{n,j}$. Let $p, q \in \mathbb{N}^*$. It holds that, using the orthonormal property of $(e_j)_{j=1}^k$ and Eq. (23), $\|h_p - h_q\|^2_{\mathcal{H}_K} = \left\|\sum_{j=1}^k e_j(f_{j,p} - f_{j,q})\right\|^2_{\mathcal{H}_K} = \sum_{j=1}^k \|f_{j,p} - f_{j,q}\|^2_{\mathcal{H}_{k_\mathcal{X}}}$. $(h_n)_{n\in\mathbb{N}^*}$ being convergent, it is a Cauchy sequence, thus so are the sequences $(f_{j,n})_{n\in\mathbb{N}^*}$. But $\mathcal{H}_{k_\mathcal{X}}$ is a complete space, so these sequences are convergent in $\mathcal{H}_{k_\mathcal{X}}$, and by denoting $f_j = \lim_{n\to\infty} f_{j,n}$, one gets $h = \sum_{j=1}^k e_k \cdot f_j$. Therefore $h \in \mathcal{V}$, $\mathcal{V}$ is closed and the orthogonal decomposition Eq. (22) holds. ☐

Lemma S.8.3. *Let* $k_\mathcal{X}, k_\Theta$ *be two scalar kernels and* $K : (\theta, \theta') \mapsto k_\Theta(\theta, \theta') I_{\mathcal{H}_{k_\mathcal{X}}}$. *Let also* $m \in \mathbb{N}^*$ *and* $(\theta_j)_{j=1}^m \in \Theta^m$, *and* $\mathcal{V} = \left(+_{j=1}^m \operatorname{Im}\left(K_{\theta_j}\right)\right)$. *Then* $I : \mathcal{V} \to \mathbb{R}$ *defined as* $I(h) = \sum_{j=1}^m \|h(\theta_j)\|^2_{\mathcal{H}_{k_\mathcal{X}}}$ *is coercive.*

Proof. Notice first that if there exists θ_j such that $k_\Theta(\theta_j, \theta_j) = 0$, then $\operatorname{Im}\left(K_{\theta_j}\right) = 0$, so without loss of generality, we assume that $k_\Theta(\theta_j, \theta_j) > 0$ $(1 \leqslant j \leqslant m)$. Notice that I is the quadratic form associated to the $L : \mathcal{H}_K \to \mathcal{H}_K$ linear mapping $L(h) = \sum_{j=1}^m K_{\theta_j} K^*_{\theta_j}$. Indeed, $\forall h \in \mathcal{V}$, $I(h) = \sum_{j=1}^m \langle K^*_{\theta_j} h, K^*_{\theta_j} h \rangle_{\mathcal{H}_{k_\mathcal{X}}} = \sum_{j=1}^m \langle h, K_{\theta_j} K^*_{\theta_j} h \rangle_{\mathcal{H}_K} = \langle h, Lh \rangle_{\mathcal{H}_K}$. Moreover, $\forall 1 \leqslant j \leqslant m$, $K_{\theta_j} K^*_{\theta_j}$ has the same eigenvalues as $K^*_{\theta_j} K_{\theta_j}$, and $\forall f \in \mathcal{H}_{k_\mathcal{X}}$, $K^*_{\theta_j} K_{\theta_j} f = k_\Theta(\theta_j, \theta_j) f$, so that the only possible eigenvalue is $k_\Theta(\theta_j, \theta_j)$. Let $h \in \mathcal{V}$, $h \neq 0$. Because of the Eq. (22), h cannot be simultaneously in all $\operatorname{Ker}\left(K^*_{\theta_j}\right)$, and there exists i_0 such that $I(h) \geqslant k_\Theta(\theta_{i_0}, \theta_{i_0}) \|h\|^2_{\mathcal{H}_K}$. Let $\gamma = \min\limits_{1\leqslant j\leqslant m} k_\Theta(\theta_j, \theta_j)$. By assumption $\gamma > 0$, and it holds that $\forall h \in \mathcal{V}$, $I(h) \geqslant \gamma \|h\|^2_{\mathcal{H}_K}$, which proves the coercivity of I. ☐

Proof of Proposition 3.2. Let $K : (x, z) \in \mathcal{X} \times \mathcal{X} \mapsto k_\mathcal{X}(x, z) I_{\mathcal{H}_{k_\Theta}} \in \mathcal{L}(\mathcal{H}_{k_\Theta})$, $K' : (\theta, \theta') \in \Theta \times \Theta \mapsto k_\Theta(\theta, \theta') I_{\mathcal{H}_{k_\mathcal{X}}} \in \mathcal{L}(\mathcal{H}_{k_\mathcal{X}})$, and define

$$J\colon \begin{cases} \mathcal{H}_K \times \mathcal{H}_{k_b} & \to \mathbb{R} \\ (h, t) & \mapsto \frac{1}{n} \displaystyle\sum_{i,j=1}^{n,m} \frac{w_j}{\theta_j} |t(\theta_j) - h(x_i)(\theta_j)|_+ + \sum_{j=1}^m w_j \left(\|h(\cdot)(\theta_j)\|^2_{\mathcal{H}_{k_\mathcal{X}}} - t(\theta_j) \right) + \frac{\lambda}{2} \|t\|^2_{\mathcal{H}_{k_b}}. \end{cases}$$

Let $\mathcal{V} = W\left(+_{j=1}^m \operatorname{Im}\left(K'_{\theta_j}\right)\right)$ where $W\colon \mathcal{H}_{K'} \to \mathcal{H}_K$ is defined in Eq. (21). Since W is an isometry, thanks to Eq. (22), it holds that $\mathcal{V} \oplus \mathcal{V}^\perp = \mathcal{H}_K$. Let $(h, t) \in \mathcal{H}_K \times \mathcal{H}_{k_b}$, there exists unique $h_{\mathcal{V}^\perp} \in \mathcal{V}^\perp$, $h_\mathcal{V} \in \mathcal{V}$ such that $h = h_\mathcal{V} + h_{\mathcal{V}^\perp}$. Notice that $J(h, t) = J(h_\mathcal{V} + h_{\mathcal{V}^\perp}, t) = J(h_\mathcal{V}, t)$ since $\forall 1 \leqslant j \leqslant m, \forall x \in \mathcal{X}$, $h_{\mathcal{V}^\perp}(x)(\theta_j) = W^{-1} h_{\mathcal{V}^\perp}(\theta_j)(x) = 0$. Moreover, J is bounded by below so that its infinimum is well-defined, and $\inf\limits_{(h,t)\in H_K\times H_{k_b}} J(h, t) = \inf\limits_{(h,t)\in V\times H_{k_b}} J(h, t)$. Finally, notice that J is coercive on $\mathcal{V} \times \mathcal{H}_{k_b}$ endowed with the sum of the norm (which makes it a Hilbert space): if $(h_n, t_n)_{n\in\mathbb{N}^*} \in \mathcal{V} \times \mathcal{H}_{k_b}$ is such that $\|h_n\|_{\mathcal{H}_K} + \|t_n\|_{\mathcal{H}_{k_b}} \underset{n\to\infty}{\to} +\infty$, then either $(\|h_n\|_{\mathcal{H}_K})_{n\in\mathbb{N}}$ or $(\|t_n\|_{\mathcal{H}_{k_b}})_{n\in\mathbb{N}}$ has to diverge :

- If $\|t_n\|_{\mathcal{H}_{k_b}} \underset{n\to\infty}{\to} +\infty$, since $t_n(\theta_j) = \langle t_n, k_b(\cdot, \theta_j) \rangle_{\mathcal{H}_{k_b}} \leqslant k_b(\theta_j, \theta_j) \|t_n\|_{\mathcal{H}_{k_b}} \leqslant \kappa_b \|t_n\|_{\mathcal{H}_{k_b}}$ $(\forall 1 \leqslant j \leqslant m)$, then $J(h_n, t_n) \geqslant \frac{\lambda}{2} \|t_n\|^2_{\mathcal{H}_{k_b}} - \sum_{j=1}^m w_j t(\theta_j) \underset{n\to\infty}{\to} +\infty$.
- If $\|h_n\|_{\mathcal{H}_K} \underset{n\to\infty}{\to} +\infty$, according to Lemma S.8.3, $J(h_n, t_n) \underset{n\to\infty}{\to} +\infty$ as long as all w_j are strictly positive.

Thus J is coercive, so that (Bauschke et al., 2011, Proposition 11.15) allows to conclude that J has a minimizer (h^*, t^*) on $\mathcal{V} \times \mathcal{H}_{k_b}$. Then, in the same fashion as Eq. (20), define $\mathcal{U}_1 = \operatorname{span}\left\{ (K(\cdot, x_i)k_\Theta(\cdot, \theta_j))_{i,j=1}^{n,m} \right\} \subset \mathcal{V}$ and $\mathcal{U}_2 = \operatorname{span}\left\{ (k_b(\cdot, \theta_j))_{j=1}^{m} \right\} \subset \mathcal{H}_{k_b}$, and use the reproducing property to show that $(h^*, t^*) \in \mathcal{U}_1 \times \mathcal{U}_2$, so that there there exist $(\alpha_{ij})_{i,j=1}^{n,m}$ and $(\beta_j)_{j=1}^{m}$ such that $\forall x, \theta \in \mathcal{X} \times \Theta$, $h^*(x)(\theta) = \sum_{i,j=1}^{n,m} \alpha_{ij} k_{\mathcal{X}}(x, x_i) k_\theta(\theta, \theta_j)$, $t^*(\theta) = \sum_{j=1}^{m} \beta_j k_b(\theta, \theta_j)$.

□

S.9 Generalization error in the context of stability

The analysis of the generalization error will be performed using the notion of uniform stability introduced in (Bousquet et al., 2002). For a derivation of generalization bounds in vv-RKHS, we refer to (**kadri2016operator**). In their framework, the goal is to minimize a risk which can be expressed as

$$R_{\mathcal{S},\lambda}(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h, x_i) + \lambda \|h\|_{\mathcal{H}_K}^2, \tag{24}$$

where $\mathcal{S} = ((x_1, y_1), \ldots, (x_n, y_n))$ are i. i. d. inputs and $\lambda > 0$. We almost recover their setting by using losses defined as

$$\ell \colon \begin{cases} \mathbb{R} \times \mathcal{H}_K \times \mathcal{X} & \to \mathbb{R} \\ (y, h, x) & \mapsto \widetilde{V}(y, f(x)), \end{cases}$$

where $\widetilde{V}$ is a loss associated to some local cost defined in Eq. (8). Then, they study the stability of the algorithm which, given a dataset $\mathcal{S}$, returns

$$h_{\mathcal{S}}^* = \operatorname*{arg\,min}_{h \in \mathcal{H}_K} R_{\mathcal{S},\lambda}(h). \tag{25}$$

There is a slight difference between their setting and ours, since they use losses defined for some y in the output space of the vv-RKHS, but this difference has no impact on the validity of the proofs in our case. The use of their theorem requires some assumption that are listed below. We recall the shape of the OVK we use : $K : (x, z) \in \mathcal{X} \times \mathcal{X} \mapsto k_{\mathcal{X}}(x, z) I_{\mathcal{H}_{k_\Theta}} \in \mathcal{L}(\mathcal{H}_{k_\Theta})$, where $k_{\mathcal{X}}$ and k_Θ are both bounded scalar-valued kernels, in other words there exist $(\kappa_{\mathcal{X}}, \kappa_\Theta) \in \mathbb{R}^2$ such that $\sup_{x \in X} k_{\mathcal{X}}(x, x) < \kappa_{\mathcal{X}}^2$ and $\sup_{\theta \in \Theta} k_\Theta(\theta, \theta) < \kappa_\Theta^2$.

Assumption 1. *$\exists \kappa > 0$ such that $\forall x \in \mathcal{X}$, $\|K(x, x)\|_{\mathcal{L}(\mathcal{H}_{k_\Theta})} \leqslant \kappa^2$.*

Assumption 2. *$\forall h_1, h_2 \in \mathcal{H}_{k_\Theta}$, the function $(x_1, x_2) \in \mathcal{X} \times \mathcal{X} \mapsto \langle K(x_1, x_2) h_1, h_2 \rangle_{\mathcal{H}_{k_\Theta}} \in \mathbb{R}$, is measurable.*

Remark 2. *Assumptions 1, 2 are satisfied for our choice of kernel.*

Assumption 3. *The application $(y, h, x) \mapsto \ell(y, h, x)$ is σ-admissible, i. e. convex with respect to f and Lipschitz continuous with respect to $f(x)$, with σ as its Lipschitz constant.*

Assumption 4. *$\exists \xi \geqslant 0$ such that $\forall (x, y) \in \mathcal{X} \times \mathcal{Y}$ and $\forall \mathcal{S}$ training set, $\ell(y, h_{\mathcal{S}}^*, x) \leqslant \xi$.*

Definition S.9.1. *Let $\mathcal{S} = ((x_i, y_i))_{i=1}^{n}$ be the training data. We call $\mathcal{S}^i$ the training data $\mathcal{S}^i = ((x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n))$, $1 \leqslant i \leqslant n$.*

Definition S.9.2. *A learning algorithm mapping a dataset $\mathcal{S}$ to a function $h_{\mathcal{S}}^*$ is said to be β-uniformly stable with respect to the loss function ℓ if $\forall n \geqslant 1$, $\forall 1 \leqslant i \leqslant n$, $\forall \mathcal{S}$ training set, $\|\ell(\cdot, h_{\mathcal{S}}^*, \cdot) - \ell(\cdot, h_{\mathcal{S}^i}^*, \cdot)\|_\infty \leqslant \beta$.*

Proposition S.9.1. *(Bousquet et al., 2002) Let $\mathcal{S} \mapsto h_{\mathcal{S}}^*$ be a learning algorithm with uniform stability β with respect to a loss ℓ satisfying Assumption 4. Then $\forall n \geqslant 1$, $\forall \delta \in (0, 1)$, with probability at least $1 - \delta$ on the drawing of the samples, it holds that*

$$R(h_{\mathcal{S}}^*) \leqslant R_{\mathcal{S}}(h_{\mathcal{S}}^*) + 2\beta + (4\beta + \xi) \sqrt{\frac{\log(1/\delta)}{n}}.$$

Proposition S.9.2. *(**kadri2016operator**) Under assumptions 1, 2, 3, a learning algorithm that maps a training set $\mathcal{S}$ to the function $h_{\mathcal{S}}^*$ defined in Eq. (25) is β-stable with $\beta = \frac{\sigma^2 \kappa^2}{2\lambda n}$.*

S.9.1 Quantile Regression

We recall that in this setting, $\nu(\theta, y, h(x)(\theta)) = \max(\theta(y - h(x)(\theta)), (1-\theta)(y - h(x)(\theta)))$ and the loss is

$$\ell\colon \begin{cases} \mathbb{R} \times \mathcal{H}_K \times \mathcal{X} & \to \mathbb{R} \\ (y, h, x) & \mapsto \frac{1}{m}\sum_{j=1}^{m} \max(\theta_j(y - h(x)(\theta_j)), (\theta_j - 1)(y - h(x)(\theta_j))). \end{cases} \tag{26}$$

Moreover, we will assume that $|Y|$ is bounded by $B \in \mathbb{R}$ as a r. v.. We will therefore verify the hypothesis for $y \in [-B, B]$ and not $y \in \mathbb{R}$.

Lemma S.9.3. *In the case of the* QR*, the loss ℓ is σ-admissible with $\sigma = 2\kappa_\Theta$.*

Proof. Let $h_1, h_2 \in \mathcal{H}_K$ and $\theta \in [0, 1]$. $\forall x, y \in \mathcal{X} \times \mathbb{R}$, it holds that

$$\nu(\theta, y, h_1(x)(\theta)) - \nu(\theta, y, h_2(x)(\theta)) = (\theta - t)(h_2(x)(\theta) - h_1(x)(\theta)) + (t - s)(y - h_1(x)(\theta)),$$

where $s = \mathbf{1}_{y \leqslant h_1(x)(\theta)}$ and $t = \mathbf{1}_{y \leqslant h_2(x)(\theta)}$. We consider all possible cases for t and s :

- $t = s = 0$: $|(t - s)(y - h_1(x)(\theta))| \leqslant |h_2(x)(\theta) - h_1(x)(\theta)|$
- $t = s = 1$: $|(t - s)(y - h_1(x)(\theta))| \leqslant |h_2(x)(\theta) - h_1(x)(\theta)|$
- $s = 1, t = 0$: $|(t - s)(y - h_1(x)(\theta))| = |h_1(x)(\theta) - y| \leqslant |h_1(x)(\theta) - h_2(x)(\theta)|$
- $s = 0, t = 1$: $|(t - s)(y - h_1(x)(\theta))| = |y - h_1(x)(\theta)| \leqslant |h_1(x)(\theta) - h_2(x)(\theta)|$ because of the conditions on t, s.

Thus $|\nu(\theta, y, h_1(x)(\theta)) - \nu(\theta, y, h_2(x)(\theta))| \leqslant (\theta + 1)|h_1(x)(\theta) - h_2(x)(\theta)| \leqslant (\theta + 1)\kappa_\Theta \|h_1(x) - h_2(x)\|_{\mathcal{H}_{k_\Theta}}$. By summing this expression over the $(\theta_j)_{j=1}^m$, we get that

$$|\ell(x, h_1, y) - \ell(x, h_2, y)| \leqslant \frac{1}{m}\sum_{j=1}^{m}(\theta_j + 1)\kappa_\Theta \|h_1(x) - h_2(x)\|_{\mathcal{H}_{k_\Theta}} \leqslant 2\kappa_\Theta \|h_1(x) - h_2(x)\|_{\mathcal{H}_{k_\Theta}}$$

and ℓ is σ-admissible with $\sigma = 2\kappa_\Theta$. □

Lemma S.9.4. *Let $\mathcal{S} = ((x_1, y_1), \dots, (x_n, y_n))$ be a training set and $\lambda > 0$. Then $\forall x, \theta \in \mathcal{X} \times (0, 1)$, it holds that $|h_{\mathcal{S}}^*(x)(\theta)| \leqslant \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}}$.*

Proof. Since $h_{\mathcal{S}}^*$ is the output of our algorithm and $0 \in \mathcal{H}_K$, it holds that

$$\lambda \|h_{\mathcal{S}}^*\|^2 \leqslant \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} \nu(\theta_j, y_i, 0) \leqslant \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} \max(\theta_j, 1 - \theta_j)|y_i| \leqslant B.$$

Thus $\|h_{\mathcal{S}}^*\| \leqslant \sqrt{\frac{B}{\lambda}}$. Moreover, $\forall x, \theta \in \mathcal{X} \times (0, 1)$, $|h_{\mathcal{S}}^*(x)(\theta)| = |\langle h_{\mathcal{S}}^*(x), k_\Theta(\theta, \cdot)\rangle_{\mathcal{H}_{k_\Theta}}| \leqslant \|h_{\mathcal{S}}^*(x)\|_{\mathcal{H}_{k_\Theta}}\kappa_\Theta \leqslant \|h_{\mathcal{S}}^*\|_{\mathcal{H}_{k_\Theta}}\kappa_{\mathcal{X}}\kappa_\Theta$ which concludes the proof. □

Lemma S.9.5. *Assumption 4 is satisfied for $\xi = 2\left(B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}}\right)$.*

Proof. Let $\mathcal{S} = ((x_1, y_1), \dots, (x_n, y_n))$ be a training set and $h_{\mathcal{S}}^*$ be the output of our algorithm. $\forall (x, y) \in \mathcal{X} \times [-B, B]$, it holds that

$$\begin{aligned} \ell(y, h_{\mathcal{S}}^*, x) &= \frac{1}{m}\sum_{j=1}^{m} \max(\theta_j(y - h_{\mathcal{S}}^*(x)(\theta_j)), (\theta_j - 1)(y - h_{\mathcal{S}}^*(x)(\theta_j))) \leqslant \frac{2}{m}\sum_{j=1}^{m}|y - h_{\mathcal{S}}^*(x)(\theta_j)| \\ &\leqslant \frac{2}{m}\sum_{j=1}^{m}|y| + |h_{\mathcal{S}}^*(x)(\theta_j)| \leqslant 2\left(B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}}\right). \end{aligned}$$

□

Corollary S.9.6. *The* QR *learning algorithm defined in Eq. (10) is such that* $\forall n \geqslant 1$, $\forall \delta \in (0,1)$, *with probability at least* $1-\delta$ *on the drawing of the samples, it holds that*

$$\widetilde{R}(h_{\mathcal{S}}^*) \leqslant \widetilde{R}_{\mathcal{S}}(h_{\mathcal{S}}^*) + \frac{4\kappa_{\mathcal{X}}^2\kappa_{\Theta}^2}{\lambda n} + \left[\frac{8\kappa_{\mathcal{X}}^2\kappa_{\Theta}^2}{\lambda n} + 2\left(B + \kappa_{\mathcal{X}}\kappa_{\Theta}\sqrt{\frac{B}{\lambda}}\right)\right]\sqrt{\frac{\log(1/\delta)}{n}}. \qquad (27)$$

Proof. This is a direct consequence of Proposition S.9.2, Proposition S.9.1, Lemma S.9.3 and Lemma S.9.5. □

Definition S.9.3 (Hardy-Krause variation)**.** *Let* Π *be the set of subdivisions of the interval* $\Theta = [0,1]$. *A subdivision will be denoted* $\sigma = (\theta_1, \theta_2, \dots, \theta_p)$ *and* $f\colon \Theta \to \mathbb{R}$ *be a function. We call Hardy-Krause variation of the function* f *the quantity* $\sup_{\sigma\in\Pi} \sum_{i=1}^{p-1} |f(\theta_{i+1}) - f(\theta_i)|$.

Remark 3. *If* f *is continuous,* $V(f)$ *is also the limit as the mesh of* σ *goes to zero of the above quantity.*

In the following, let $f\colon \theta \mapsto \mathbf{E}_{X,Y}[\nu(\theta, Y, h_{\mathcal{S}}^*(X)(\theta))]$. This function is of primary importance for our analysis, since in the Quasi Monte-Carlo setting, the bound of Proposition 4.1 makes sense only if the function f has finite Hardy-Krause variation, which is the focus of the following lemma.

Lemma S.9.7. *Assume the boundeness of both scalar kernels* $k_{\mathcal{X}}$ *and* k_{Θ}. *Assume moreover that* k_{Θ} *is* $\mathcal{C}^1$ *and that its partial derivatives are uniformly bounded by some constant* C. *Then*

$$V(f) \leqslant B + \kappa_{\mathcal{X}}\kappa_{\Theta}\sqrt{\frac{B}{\lambda}} + 2\kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}. \qquad (28)$$

Proof. It holds that

$$\begin{aligned}
\sup_{\sigma\in\Pi} \sum_{i=1}^{p-1} |f(\theta_{i+1}) - f(\theta_i)| &= \sup_{\sigma\in\Pi} \sum_{i=1}^{p-1} \left| \int \nu(\theta_{i+1}, y, h_{\mathcal{S}}^*(x)(\theta_{i+1})) d\mathbf{P}_{X,Y} - \int \nu(\theta_i, y, h_{\mathcal{S}}^*(x)(\theta_i)) d\mathbf{P}_{X,Y} \right| \\
&= \sup_{\sigma\in\Pi} \sum_{i=1}^{p-1} \left| \int \nu(\theta_{i+1}, y, h_{\mathcal{S}}^*(x)(\theta_{i+1})) - \nu(\theta_i, y, h_{\mathcal{S}}^*(x)(\theta_i)) d\mathbf{P}_{X,Y} \right| \\
&\leqslant \sup_{\sigma\in\Pi} \sum_{i=1}^{p-1} \int |\nu(\theta_{i+1}, y, h_{\mathcal{S}}^*(x)(\theta_{i+1})) - \nu(\theta_i, y, h_{\mathcal{S}}^*(x)(\theta_i))| d\mathbf{P}_{X,Y} \\
&\leqslant \sup_{\sigma\in\Pi} \int \sum_{i=1}^{p-1} |\nu(\theta_{i+1}, y, h_{\mathcal{S}}^*(x)(\theta_{i+1})) - \nu(\theta_i, y, h_{\mathcal{S}}^*(x)(\theta_i))| d\mathbf{P}_{X,Y}.
\end{aligned}$$

The supremum of the integral is smaller than the integral of the supremum, as such

$$V(f) \leqslant \int V(f_{x,y}) d\mathbf{P}_{X,Y}, \qquad (29)$$

where $f_{x,y}\colon \theta \mapsto \nu(\theta, y, h_{\mathcal{S}}^*(x)(\theta))$ is the counterpart of the function f at point (x,y). To bound this quantity, let us first bound locally $V(f_{x,y})$. To that extent, we fix some (x,y) in the following. Since $f_{x,y}$ is continuous (because k_{Θ} is $\mathcal{C}^1$), then using Choquet (1969, Theorem 24.6), it holds that

$$V(f_{x,y}) = \lim_{|\sigma|\to 0} \sum_{i=1}^{p-1} |f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)|.$$

Moreover since $k \in \mathcal{C}^1$ and $\partial k_\theta = (\partial_1 k)(\cdot, \theta)$ has a finite number of zeros for all $\theta \in \times$, one can assume that in the subdivision considered afterhand all the zeros (in θ) of the residuals $y - h_{\mathcal{S}}^*(x)(\theta)$ are present, so that $y - h_{\mathcal{S}}^*(x)(\theta_{i+1})$ and $y - h_{\mathcal{S}}^*(x)(\theta_i)$ are always of the same sign. Indeed, if not, create a new, finer subdivision with this property and work with this one. Let us begin the proper calculation: let $\sigma = (\theta_1, \theta_2, \dots, \theta_p)$ be a subdivision of Θ, it holds that $\forall i \in \{1, \dots, p-1\}$:

$$|f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)| = |\max(\theta_{i+1}(y - h_S^*(x)(\theta_{i+1})), (1-\theta_{i+1})(y - h_S^*(x)(\theta_{i+1}))) - \max(\theta_i(y - h_S^*(x)(\theta_i)), (1-\theta_{i+1})(y - h_S^*(x)(\theta_i)))|.$$

We now study the two possible outcomes for the residuals:

- If $y - h(x)(\theta_{i+1}) \geqslant 0$ and $y - h(x)(\theta_i) \geqslant 0$ then

$$\begin{aligned}|f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)| &= |\theta_{i+1}(y - h_S^*(x)(\theta_{i+1})) - \theta_i(y - h_S^*(x)(\theta_i))| \\ &= |(\theta_{i+1} - \theta_i)y + (\theta_i - \theta_{i+1})h_S^*(x)(\theta_{i+1}) + \theta_i(h_S^*(x)(\theta_i) - h_S^*(x)(\theta_{i+1}))| \\ &\leqslant |(\theta_{i+1} - \theta_i)y| + |(\theta_i - \theta_{i+1})h_S^*(x)(\theta_{i+1})| + |\theta_i(h_S^*(x)(\theta_i) - h_S^*(x)(\theta_{i+1}))|.\end{aligned}$$

  From Lemma S.9.4, it holds that $h_S^*(x)(\theta_{i+1}) \leqslant \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}}$. Moreover,

$$\begin{aligned}|h_S^*(x)(\theta_i) - h_S^*(x)(\theta_{i+1})| &= \left|\langle h(x), k_\Theta(\theta_i,\cdot) - k_\Theta(\theta_{i+1},\cdot)\rangle_{\mathcal{H}_{k_\Theta}}\right| \\ &\leqslant \|h(x)\|_{\mathcal{H}_{k_\Theta}} \|k_\Theta(\theta_i,\cdot) - k_\Theta(\theta_{i+1},\cdot)\|_{\mathcal{H}_{k_\Theta}} \\ &\leqslant \kappa_{\mathcal{X}}\sqrt{\frac{B}{\lambda}}\sqrt{|k_\Theta(\theta_i,\theta_i) + k_\Theta(\theta_{i+1},\theta_{i+1}) - 2k_\Theta(\theta_{i+1},\theta_i)|} \\ &\leqslant \kappa_{\mathcal{X}}\sqrt{\frac{B}{\lambda}}\left(\sqrt{|k_\Theta(\theta_{i+1},\theta_{i+1}) - k_\Theta(\theta_{i+1},\theta_i)|} + \sqrt{|k_\Theta(\theta_i,\theta_i) - k_\Theta(\theta_{i+1},\theta_i)|}\right).\end{aligned}$$

  Since k_Θ is $\mathcal{C}^1$, with partial derivatives uniformly bounded by C, $|k_\Theta(\theta_{i+1},\theta_{i+1}) - k_\Theta(\theta_{i+1},\theta_i)| \leqslant C(\theta_{i+1} - \theta_i)$ and $|k_\Theta(\theta_i,\theta_i) - k_\Theta(\theta_{i+1},\theta_i)| \leqslant C(\theta_{i+1} - \theta_i)$ so that $|h_S^*(x)(\theta_i) - h_S^*(x)(\theta_{i+1})| \leqslant \kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}\sqrt{\theta_{i+1} - \theta_i}$ and overall

$$|f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)| \leqslant \left(B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}}\right)(\theta_{i+1} - \theta_i) + \kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}\sqrt{\theta_{i+1} - \theta_i}.$$

- If $y - h(x)(\theta_{i+1}) \leqslant 0$ and $y - h(x)(\theta_i) \leqslant 0$ then $|f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)| = |(1-\theta_{i+1})(y - h_S^*(x)(\theta_{i+1})) - (1-\theta_i)(y - h_S^*(x)(\theta_i))| \leqslant |h_S^*(x)(\theta_i) - h_S^*(x)(\theta_{i+1})| + |(\theta_{i+1} - \theta_i)y| + |(\theta_i - \theta_{i+1})h_S^*(x)(\theta_{i+1})| + |\theta_i(h_S^*(x)(\theta_i) - h_S^*(x)(\theta_{i+1}))|$ so that with similar arguments one gets

$$|f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)| \leqslant \left(B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}}\right)(\theta_{i+1} - \theta_i) + 2\kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}\sqrt{\theta_{i+1} - \theta_i}. \tag{30}$$

Therefore, regardless of the sign of the residuals $y - h(x)(\theta_{i+1})$ and $y - h(x)(\theta_i)$, one gets Eq. (30). Since the square root function has Hardy-Kraus variation of 1 on the interval $\Theta = [0,1]$, it holds that

$$\sup_{\sigma\in\Pi}\sum_{i=1}^{p-1}|f_{x,y}(\theta_{i+1}) - f_{x,y}(\theta_i)| \leqslant B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}} + 2\kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}.$$

Combining this with Eq. (29) finally gives

$$V(f) \leqslant B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}} + 2\kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}.$$

□

Lemma S.9.8. *Let* R *be the risk defined in Eq. (6) for the quantile regression problem. Assume that* $(\theta)_{j=1}^m$ *have been generated via the Sobol sequence and that* k_Θ *is* $\mathcal{C}^1$ *and that its partial derivatives are uniformly bounded by some constant* C*. Then*

$$|R(h_S^*) - \widetilde{R}(h_S^*)| \leqslant \left(B + \kappa_{\mathcal{X}}\kappa_\Theta\sqrt{\frac{B}{\lambda}} + 2\kappa_{\mathcal{X}}\sqrt{\frac{2BC}{\lambda}}\right)\frac{\log(m)}{m}. \tag{31}$$

Proof. Let $f\colon \theta \mapsto \mathbf{E}_{X,Y}[\nu(\theta, Y, h_S^*(X)(\theta))]$. It holds that $|R(h_S^*) - \widetilde{R}(h_S^*)| \leqslant V(f)\frac{\log(m)}{m}$ according to classical Quasi-Monte Carlo approximation results, where $V(f)$ is the Hardy-Krause variation of f. Lemma S.9.7 allows then to conclude.

□

Proof of Proposition 4.1. Combine Lemma S.9.8 and Corollary S.9.6 to get an asymptotic behaviour as $n, m \to \infty$. □

S.9.2 Cost-Sensitive Classification

In this setting, the cost is $\nu(\theta, y, h(x)(\theta)) = \left|\frac{\theta+1}{2} - \mathbb{1}_{\{-1\}(y)}\right| |1 - yh_\theta(x)|_+$ and the loss is

$$\ell\colon \begin{cases} \mathbb{R} \times \mathcal{H}_K \times \mathcal{X} & \to \mathbb{R} \\ (y, h, x) & \mapsto \frac{1}{m}\sum_{j=1}^m \left|\frac{\theta_j+1}{2} - \mathbb{1}_{\{-1\}}(y)\right| \left|1 - yh_{\theta_j}(x)\right|_+. \end{cases}$$

It is easy to verify in the same fashion as for QR that the properties above still hold, but with constants $\sigma = \kappa_\Theta$, $\beta = \frac{\kappa_{\mathcal{X}}^2\kappa_\Theta^2}{2\lambda n}$, $\xi = 1 + \frac{\kappa_{\mathcal{X}}\kappa_\Theta}{\sqrt{\lambda}}$. so that we get analogous properties to QR.

Corollary S.9.9. *The* CSC *learning algorithm defined in Eq. (10) is such that* $\forall n \geqslant 1$*,* $\forall \delta \in (0, 1)$*, with probability at least* $1 - \delta$ *on the drawing of the samples, it holds that*

$$\widetilde{R}(h_S^*) \leqslant \widetilde{R}_S(h_S^*) + \frac{\kappa_{\mathcal{X}}^2\kappa_\Theta^2}{\lambda n} + \left(\frac{2\kappa_{\mathcal{X}}^2\kappa_\Theta^2}{\lambda n} + 1 + \frac{\kappa_{\mathcal{X}}\kappa_\Theta}{\sqrt{\lambda}}\right)\sqrt{\frac{\log(1/\delta)}{n}}.$$

S.10 Experimental remarks

We present here more details on the experimental protocol used in the main paper as well as new experiments.

S.10.1 Alternative hyperparameters sampling

Many quadrature rules such as Monte-Carlo (MC) and QMC methods are well suited for Infinite Task Learning. For instance when Θ is high dimensional, MC is typically prefered over QMC, and vice versa. If Θ is one dimensional and the function to integrate is smooth enough then a Gauss-Legendre quadrature would be preferable. In Section 3.1 of the main paper we provide a unified notation to handle MC, QMC and other quadrature rules. In the case of

- MC: $w_j = \frac{1}{m}$ and $(\theta_j)_{j=1}^m \sim \mu^{\otimes m}$.
- QMC: $w_j = m^{-1}F^{-1}(\theta_j)$ and $(\theta_j)_{j=1}^m$ is a sequence with values in $[0, 1]^d$ such as the Sobol or Halton sequence, μ is assumed to be absolutely continuous w. r. t. the Lebesgue measure, F is the associated cdf.
- Quadrature rules: $((\theta_j, w_j'))_{j=1}^m$ is the indexed set of locations and weights produced by the quadrature rule, $w_j = w_j' f_\mu(\theta_j)$, μ is assumed to be absolutely continuous w. r. t. the Lebesgue measure, and f_μ denotes its corresponding probability density function.

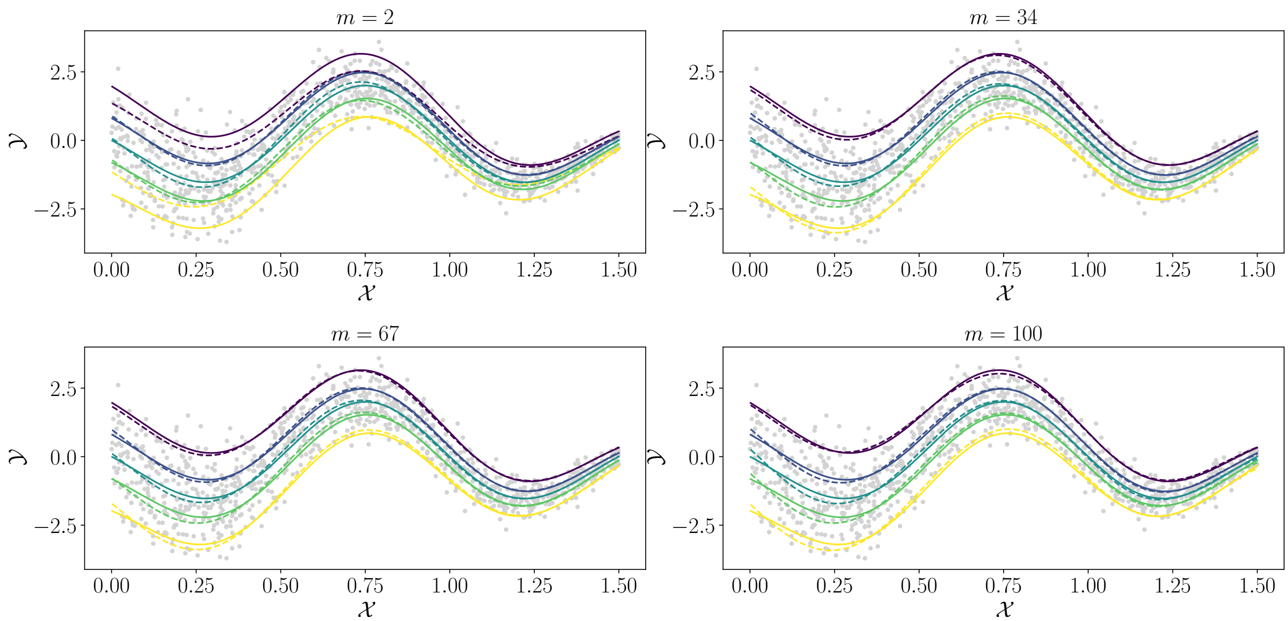

Figure S.4: Impact of the number of hyperparameters sampled.

S.10.2 Impact of the number of hyperparameters sampled

In the experiment presented on Fig. S.4, on the sine synthetic benchmark, we draw $n = 1000$ training points and study the impact of increasing m on the quality of the quantiles at $\theta \in \{0.05, 0.25, 0.5, 0.75, 0.95\}$. We notice that when $m \geqslant 34 \approx \sqrt{1000}$ there is little benefit to draw more m samples are the quantile curves do not change on the $n_{\text{test}} = 2000$ test points.

S.10.3 Smoothifying the cost function

The resulting κ-smoothed ($\kappa \in \mathbb{R}_+$) absolute value (ψ_1^κ) and positive part (ψ_+^κ) are as follows:

$$\psi_1^\kappa(p) := \left(\kappa|\cdot|\square\frac{1}{2}|\cdot|^2\right)(p) = \begin{cases} \frac{1}{2\kappa}p^2 & \text{if } |p| \leqslant \kappa \\ |p| - \frac{\kappa}{2} & \text{otherwise,} \end{cases}$$

$$\psi_+^\kappa(p) := \left(\kappa|\cdot|_+\square\frac{1}{2}|\cdot|^2\right)(p) = \begin{cases} \frac{1}{2\kappa}|p|_+^2 & \text{if } p \leqslant \kappa \\ p - \frac{\kappa}{2} & \text{otherwise.} \end{cases}$$

where $\square$ is the classical infimal convolution (Bauschke et al., 2011). All the smoothified loss functions used in this paper have been gathered in Table S.2.

Remarks

- Minimizing the κ-smoothed pinball loss

$$v_{\theta,\kappa}(y, h(x)) = |\theta - \mathbb{1}_{\mathbb{R}_-}(y - h(x))|\psi_1^\kappa(y - h(x)),$$

  yields the quantiles when $\kappa \to 0$, the expectiles as $\kappa \to +\infty$. The intermediate values are known as M-quantiles (Breckling et al., 1988).
- In practice, the absolute value and positive part can be approximated by a smooth function by setting the smoothing parameter κ to be a small positive value; the optimization showed a robust behaviour w. r. t. this choice with a random coefficient initialization.

Impact of the Huber loss support The influence of the κ parameter is illustrated in Fig. S.5. For this experiment, 10000 samples have been generated from the sine wave dataset described in Section 5, and the model

Table S.2: Examples for objective (8). ψ_1^κ, ψ_+^κ: κ-smoothed absolute value and positive part. $h_x(\theta) := h(x)(\theta)$.

	LOSS	PENALTY
QUANTILE	$\int_{[0,1]} \left\lvert \theta - \mathbb{1}_{\mathbb{R}_-}(y - h_x(\theta)) \right\rvert \lvert y - h_x(\theta) \rvert d\mu(\theta)$	$\lambda_{nc} \int_{[0,1]} \left\lvert -\frac{dh_x}{d\theta}(\theta) \right\rvert_+ d\mu(\theta) + \frac{\lambda}{2} \lVert h \rVert^2_{\mathcal{H}_K}$
M-QUANTILE (SMOOTH)	$\int_{[0,1]} \left\lvert \theta - \mathbb{1}_{\mathbb{R}_-}(y - h_x(\theta)) \right\rvert \psi_1^\kappa (y - h_x(\theta)) d\mu(\theta)$	$\lambda_{nc} \int_{(0,1)} \psi_+^\kappa \left(-\frac{dh_x}{d\theta}(\theta) \right) d\mu(\theta) + \frac{\lambda}{2} \lVert h \rVert^2_{\mathcal{H}_K}$
EXPECTILES (SMOOTH)	$\int_{[0,1]} \left\lvert \theta - \mathbb{1}_{\mathbb{R}_-}(y - h_x(\theta)) \right\rvert (y - h_x(\theta))^2 d\mu(\theta)$	$\lambda_{nc} \int_{(0,1)} \left\lvert -\frac{dh_x}{d\theta}(\theta) \right\rvert_+^2 d\mu(\theta) + \frac{\lambda}{2} \lVert h \rVert^2_{\mathcal{H}_K}$
COST-SENSITIVE	$\int_{[-1,1]} \left\lvert \frac{\theta+1}{2} - \mathbb{1}_{\{-1\}}(y) \right\rvert \lvert 1 - yh_x(\theta) \rvert_+ d\mu(\theta)$	$\frac{\lambda}{2} \lVert h \rVert^2_{\mathcal{H}_K}$
COST-SENSITIVE (SMOOTH)	$\int_{[-1,1]} \left\lvert \frac{\theta+1}{2} - \mathbb{1}_{\{-1\}}(y) \right\rvert \psi_+^\kappa (1 - yh_x(\theta)) d\mu(\theta)$	$\frac{\lambda}{2} \lVert h \rVert^2_{\mathcal{H}_K}$
LEVEL-SET	$\int_{[\epsilon,1]} -t(\theta) + \frac{1}{\theta} \lvert t(\theta) - h_x(\theta) \rvert_+ d\mu(\theta)$	$\frac{1}{2} \int_{[\epsilon,1]} \lVert h(\cdot)(\theta) \rVert^2_{\mathcal{H}_{k_\mathcal{X}}} d\mu(\theta) + \frac{\lambda}{2} \lVert t \rVert^2_{\mathcal{H}_{k_b}}$

have been trained on 100 quantiles generated from a Gauss-Legendre Quadrature. When κ is large the expectiles are learnt (dashed lines) while when κ is small the quantiles are recovered (the dashed lines on the right plot match the theoretical quantiles in plain lines). It took 225s (258 iteration, and 289 function evaluations) to train for $\kappa = 1 \cdot 10^1$, 1313s for $\kappa = 1 \cdot 10^{-1}$ (1438 iterations and 1571 function evaluations), 931s for $\kappa = 1e^{-3}$ (1169 iterations and 1271 function evaluations) and 879s for $\kappa = 0$ (1125 iterations and 1207 function evaluations). We used a GPU Tensorflow implementation and run the experiments in float64 on a computer equipped with a GTX 1070, and intel i7 7700 and 16GB of DRAM.

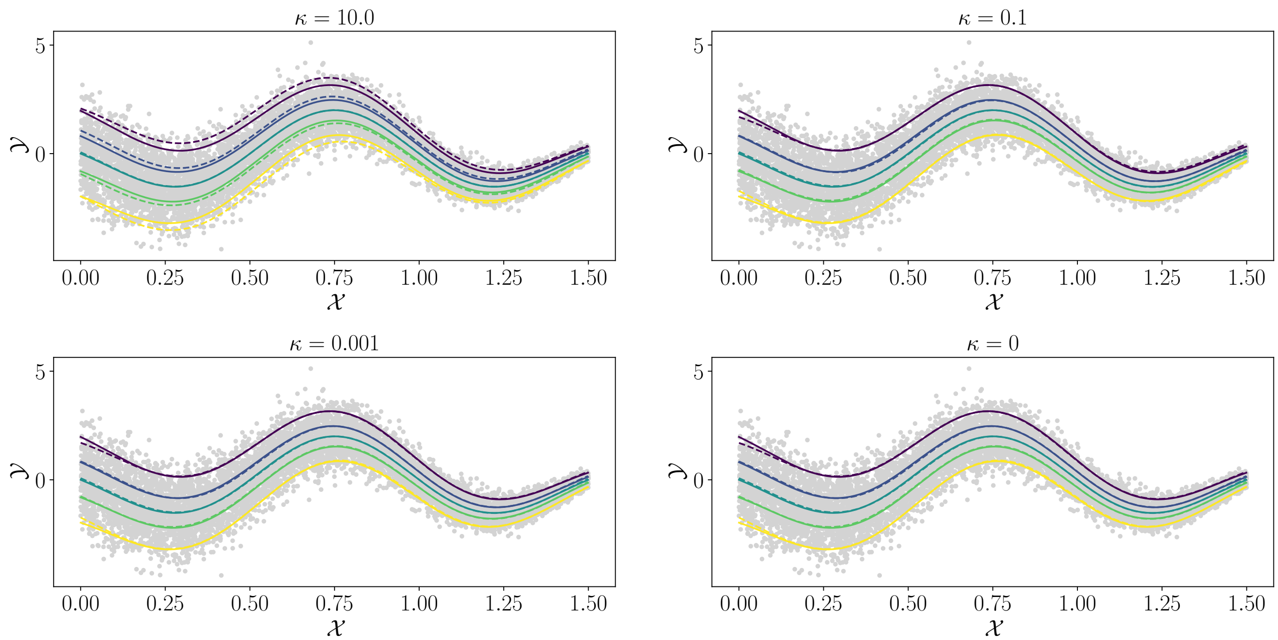

Figure S.5: Impact of the Huber loss smoothing of the pinball loss for differents values of κ.

S.10.4 Experimental protocol for QR

In this section, we give additional details regarding the choices being made while implementing the ITL method for ∞-QR.

QR real datasets For ∞-QR, $k_\mathcal{X}$, k_Θ were Gaussian kernels. We set a bias term $k_b = k_\Theta$. The hyperparameters optimized were λ, the weight of the ridge penalty, $\sigma_\mathcal{X}$, the input kernel parameter, and $\sigma_\Theta = \sigma_b$, the output kernel parameter. They were optimized in the (log)space of $\left[10^{-6}, 10^6\right]^3$. The non-crossing constraint

λ_{nc} was set to 1. The model was trained on the continuum $\Theta = (0, 1)$ using QMC and Sobol sequences. For all datasets we draw $m = 100$ quantiles form a Sobol sequence

For JQR we similarly chose two Gaussian kernels. The optimized hyperparameters were the same as for ∞-QR. The quantiles learned were $\theta \in \{ 0.1, 0.3, 0.5, 0.7, 0.9 \}$.

For the IND-QR baseline, we trained independently a non-paramatric quantile estimator as described in Takeuchi et al. (2006). A Gaussian kernel was used and its bandwidth was optimized in the (log)space of $\left[10^{-6}, 10^{6}\right]$. No non-crossing was enforced.

S.10.5 Experiments with CSC

In this section we provide numerical illustration concerning the CSC problem. We used the Iris UCI dataset with 4 attributes and 150 samples, the two synthetic SCIKIT-LEARN (Pedregosa et al., 2011) datasets TWO-MOONS (noise=0.4) and CIRCLES (noise=0.1) with both 2 attributes and 1000 samples and a third synthetic SCIKIT-LEARN dataset TOY (class sep=0.5) with 20 features (4 redundant and 10 informative) and $n = 1000$ samples.

As detailed in Section 2, Cost-Sensitive Classification on a continuum $\Theta = [-1, 1]$ that we call Infinite Cost-Sensitive Classification (∞-CSC) can be tackled by our proposed technique. In this case, the hyperparameter θ controls the tradeoff between the importance of the correct classification with labels -1 and $+1$. When $\theta = -1$, class -1 is emphasized; the probability of correctly classified instances with this label (called specificity) is desired to be 1. Similarly, for $\theta = +1$, the probability of correct classification of samples with label $+1$ (called sensitivity) is ideally 1.

To illustrate the advantage of (infinite) joint learning we used two synthetic datasets CIRCLES and TWO-MOONS and the UCI IRIS dataset. We chose $k_{\mathcal{X}}$ to be a Gaussian kernel with bandwidth $\sigma_{\mathcal{X}} = (2\gamma_{\mathcal{X}})^{(-1/2)}$ the median of the Euclidean pairwise distances of the input points (Jaakkola et al., 1999). k_Θ is also a Gaussian kernel with bandwidth $\gamma_\Theta = 5$. We used $m = 20$ for all datasets. As a baseline we trained independently 3 Cost-Sensitive Classification classifiers with $\theta \in \{ -0.9, 0, 0.9 \}$. We repeated 50 times a random $50 - 50\%$ train-test split of the dataset and report the average test error and standard deviation (in terms of sensitivity and specificity)

Our results are illustrated in Table S.3. For $\theta = -0.9$, both independent and joint learners give the desired 100% specificity; the joint Cost-Sensitive Classification scheme however has significantly higher sensitivity value (15% vs 0%) on the dataset CIRCLES. Similar conclusion holds for the $\theta = +0.9$ extreme: the ideal sensitivity is reached by both techniques, but the joint learning scheme performs better in terms of specificity (0% vs 12%) on the dataset CIRCLES.

Table S.3: ∞-CSC vs Independent (IND)-CSC. Higher is better.

DATASET	METHOD	$\theta = -0.9$		$\theta = 0$		$\theta = +0.9$	
		SENSITIVITY	SPECIFICITY	SENSITIVITY	SPECIFICITY	SENSITIVITY	SPECIFICITY
TWO-MOONS	IND	0.3 ± 0.05	0.99 ± 0.01	0.83 ± 0.03	0.86 ± 0.03	0.99 ± 0	0.32 ± 0.06
	∞-CSC	0.32 ± 0.05	0.99 ± 0.01	0.84 ± 0.03	0.87 ± 0.03	1 ± 0	0.36 ± 0.04
CIRCLES	IND	0 ± 0	1 ± 0	0.82 ± 0.02	0.84 ± 0.03	1 ± 0	0 ± 0
	∞-CSC	0.15 ± 0.05	1 ± 0	0.82 ± 0.02	0.84 ± 0.03	1 ± 0	0.12 ± 0.05
IRIS	IND	0.88 ± 0.08	0.94 ± 0.06	0.94 ± 0.05	0.92 ± 0.06	0.97 ± 0.05	0.87 ± 0.06
	∞-CSC	0.89 ± 0.08	0.94 ± 0.05	0.94 ± 0.06	0.92 ± 0.05	0.97 ± 0.04	0.90 ± 0.05
TOY	IND	0.51 ± 0.06	0.98 ± 0.01	0.83 ± 0.03	0.86 ± 0.03	0.97 ± 0.01	0.49 ± 0.07
	∞-CSC	0.63 ± 0.04	0.96 ± 0.01	0.83 ± 0.03	0.85 ± 0.03	0.95 ± 0.02	0.61 ± 0.04

References

Bauschke, H. H. and P. L. Combettes (2011). *Convex analysis and monotone operator theory in Hilbert spaces.* Springer (cit. on pp. 11, 14, 19).

Bousquet, O. and A. Elisseeff (2002). "Stability and generalization." In: *Journal of Machine Learning Research* 2, pp. 499–526 (cit. on p. 14).

Breckling, J. and R. Chambers (1988). "M-quantiles." In: *Biometrika* 75.4, pp. 761–771 (cit. on p. 19).

Carmeli, C. et al. (2010). "Vector valued reproducing kernel Hilbert spaces and universality." In: *Analysis and Applications* 8 (1), pp. 19–61 (cit. on p. 12).

Choquet, G. (1969). *Cours d'analyse: Tome II. Topologie*. Masson et Cie. (cit. on p. 16).

Jaakkola, T., M. Diekhans, and D. Haussler (1999). "Using the Fisher kernel method to detect remote protein homologies." In: *ISMB*. Vol. 99, pp. 149–158 (cit. on p. 21).

Li, Y., Y. Liu, and J. Zhu (2007). "Quantile regression in reproducing kernel Hilbert spaces." In: *Journal of the American Statistical Association* 102.477, pp. 255–268 (cit. on p. 10).

Pedregosa, F. et al. (2011). "Scikit-learn: Machine learning in Python." In: *Journal of Machine Learning Research* 12.Oct, pp. 2825–2830 (cit. on p. 21).

Takeuchi, I. et al. (2006). "Nonparametric quantile estimation." In: *Journal of Machine Learning Research* 7, pp. 1231–1264 (cit. on p. 21).

Zhou, D.-X. (2008). "Derivative reproducing properties for kernel methods in learning theory." In: *Journal of computational and Applied Mathematics* 220.1-2, pp. 456–463 (cit. on p. 12).

Ziemer, W. P. (2012). *Weakly differentiable functions: Sobolev spaces and functions of bounded variation*. Vol. 120. Springer Science & Business Media (cit. on p. 12).