Duality in RKHSs with Infinite Dimensional Outputs: Application to Robust Losses

Pierre Laforgue [1] **Alex Lambert** [1] **Luc Brogat-Motte** [1] **Florence d'Alché-Buc** [1]

Abstract

Operator-Valued Kernels (OVKs) and associated vector-valued Reproducing Kernel Hilbert Spaces provide an elegant way to extend scalar kernel methods when the output space is a Hilbert space. Although primarily used in finite dimension for problems like multi-task regression, the ability of this framework to deal with infinite dimensional output spaces unlocks many more applications, such as functional regression, structured output prediction, and structured data representation. However, these sophisticated schemes crucially rely on the kernel trick in the output space, so that most of previous works have focused on the square norm loss function, completely neglecting robustness issues that may arise in such surrogate problems. To overcome this limitation, this paper develops a duality approach that allows to solve OVK machines for a wide range of loss functions. The infinite dimensional Lagrange multipliers are handled through a *Double Representer Theorem*, and algorithms for ϵ-insensitive losses and the Huber loss are thoroughly detailed. Robustness benefits are emphasized by a theoretical stability analysis, as well as empirical improvements on structured data applications.

1. Introduction

Due to increasingly available streaming and network data, learning to predict complex objects such as structured outputs or time series has attracted a great deal of attention in machine learning. Extending the well known kernel methods devoted to non-vectorial data (Hofmann et al., 2008), several kernel-based approaches have emerged to deal with complex output data. While Structural SVM and variants cope with discrete structures (Tsochantaridis et al., 2005; Joachims et al., 2009) through structured losses, Operator-Valued Kernels (OVKs) and vector-valued Reproducing Kernel Hilbert Spaces (vv-RKHSs, Micchelli and Pontil (2005); Carmeli et al. (2006; 2010)) provide a unique framework to handle both functional and structured outputs. Vv-RKHSs are classes of functions that map an arbitrary input set $\mathcal{X}$ to some output Hilbert space $\mathcal{Y}$ (Senkene and Tempel'man, 1973; Caponnetto et al., 2008). Primarily used with finite dimensional outputs ($\mathcal{Y} = \mathbb{R}^p$) to solve multi-task regression (Micchelli and Pontil, 2005; Baldassarre et al., 2012) and multiple class classification (Dinuzzo et al., 2011), OVK methods have further been exploited to handle outputs in infinite dimensional Hilbert spaces. This has unlocked numerous applications, such as functional regression (Kadri et al., 2010; 2016), structured prediction (Brouard et al., 2011; Kadri et al., 2013), infinite quantile regression (Brault et al., 2019), or structured data representation learning (Laforgue et al., 2019). Nonetheless, these sophisticated schemes often come along with a basic loss function: the output space squared norm, neglecting desirable properties such as parsimony and robustness.

In nonparametric modeling, model parsimony boils down to data sparsity, *e.g.* reducing the number of training data points on which the model relies to make a prediction. Such a property is highly valuable (Hastie et al., 2015): not only does it prevent overfitting but it also alleviates the inherent computational load of optimization and prediction, allowing to scale to larger datasets. Another appealing property of a regression tool is robustness to outliers (Huber, 1964; Zhu et al., 2008). Real data may suffer from incorrect feature measurements and spurious annotations, leading to training datasets contaminated with outliers. Then, minimizing the squared loss is inappropriate as the least-squares estimates behave poorly when the residuals distribution is not normal, but rather heavy-tailed. In (scalar) kernel methods, these two properties – data sparsity and robustness to outliers – are imposed through the choice of appropriate losses. Data sparsity is leveraged by using ϵ-insensitive losses, exploited in the well known Support Vector Regression (Drucker et al., 1997) while robust regression (Fung and Mangasarian, 2000) can be obtained by minimizing the Huber loss function (Huber, 1964). Driven by three emblematic learning tasks,

[1] LTCI, Télécom Paris, Institut Polytechnique de Paris, France. Correspondence to: Pierre Laforgue <pierre.laforgue@telecom-paris.fr>.

Proceedings of the 37th International Conference on Machine Learning, Online, PMLR 119, 2020.

structured prediction, functional regression, and structured data representation, we propose a general duality framework that enables sparse data regression and robust regression, even when working in vv-RKHSs with infinite-dimensional outputs. Although extensively used within scalar kernel methods, very few attempts have been made to adapt duality to vv-RKHSs. In Brouard et al. (2016b), dualization is presented, but only used in the maximum margin regression scenario. Sangnier et al. (2017) consider a wider class of loss functions, including ϵ-insensitive losses to leverage data sparsity, but only in the case of matrix-valued kernels (Álvarez et al., 2012), for which the dual problem is finite dimensional. For a general OVK however, the dual problem is to be solved over $\mathcal{Y}^n$, and is intractable without additional work when $\mathcal{Y}$ is infinite dimensional. We first notice that the extensions of ϵ-insensitive losses and the Huber loss to general Hilbert space are (still) expressed as convolutions of simpler losses whose Fenchel-Legendre (FL) transforms are known. Inspired by this remark, we identify general conditions on the OVKs and FL transforms to establish a *Double Representer Theorem* allowing to work with matrix parameterized representations. In particular, a careful use of the duality principle considerably broadens the range of loss functions for which OVK solutions are computable. The present work thus aims at developing a comprehensive methodology to solve these dual problems.

The rest of the paper is organized as follows. In Section 2, we introduce OVKs, recall the general formulation of dual problems for OVK machines, and derive their solvable finite dimensional reformulation. Section 3 is devoted to specific instantiations of this problem for ϵ-insensitive losses and the Huber loss, with algorithms duly explicited. In Section 4, we apply our framework to induce sparsity and robustness into structured prediction, functional regression, and structured data representation. Proofs are postponed to the Appendix.

2. Learning in vv-RKHSs

After reminders on OVKs and vv-RKHS learning theory, this section exposes the duality approach for the regularized empirical risk minimization problem in vv-RKHSs. Two strategies are then detailed to solve infinite dimensional dual problems, either under an assumption on the kernel, or by approximating the dual. In the following, $\mathcal{Y}$ is assumed to be a separable Hilbert space.

Definition 1. *An OVK is an application $\mathcal{K}\colon \mathcal{X}\times\mathcal{X} \to \mathcal{L}(\mathcal{Y})$, that satisfies the following two properties for all $n \in \mathbb{N}^*$:*

1) $\forall (x, x') \in \mathcal{X} \times \mathcal{X}, \qquad \mathcal{K}(x, x') = \mathcal{K}(x', x)^{\#}$,

2) $\forall\ (x_i, y_i)_{i=1}^n \in (\mathcal{X} \times \mathcal{Y})^n, \ \sum_{i,j=1}^n \langle y_i, \mathcal{K}(x_i, x_j) y_j\rangle_{\mathcal{Y}} \geqslant 0$,

with $\mathcal{L}(E)$ the set of bounded linear operators on vector space E, and $A^{\#}$ the adjoint of any operator A.

A simple example of OVK is the *separable kernel.*

Definition 2. *$\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathcal{L}(\mathcal{Y})$ is a* separable kernel *iff there exist a scalar kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and a positive semi-definite operator $A \in \mathcal{L}(\mathcal{Y})$ such that for all $(x, x') \in \mathcal{X}^2$ it holds: $\quad \mathcal{K}(x, x') = k(x, x')A$.*

Similarly to scalar-valued kernels, an OVK can be uniquely associated to a functional space from $\mathcal{X}$ to $\mathcal{Y}$: its vv-RKHS.

Theorem 1. *Let $\mathcal{K}$ be an OVK, and for $x \in \mathcal{X}$, let $\mathcal{K}_x\colon y \mapsto \mathcal{K}_x y \in \mathcal{F}(\mathcal{X}, \mathcal{Y})$ the linear operator such that: $\forall x' \in \mathcal{X},\ (\mathcal{K}_x y)(x') = \mathcal{K}(x', x)y$. Then, there is a unique Hilbert space $\mathcal{H}_{\mathcal{K}} \subset \mathcal{F}(\mathcal{X}, \mathcal{Y})$ the vv-RKHS associated to $\mathcal{K}$ such that $\forall x \in \mathcal{X}$ it holds:*

(i) $\mathcal{K}_x$ spans the space $\mathcal{H}_{\mathcal{K}}$ ($\forall y \in \mathcal{Y}\colon \mathcal{K}_x y \in \mathcal{H}_{\mathcal{K}}$)

(ii) $\mathcal{K}_x$ is bounded for the uniform norm

(iii) $\forall f \in \mathcal{H}_{\mathcal{K}},\ f(x) = \mathcal{K}_x^{\#} f$ (reproducing property)

Given a sample $\mathcal{S} = \{(x_i, y_i)_{i=1}^n\} \in (\mathcal{X} \times \mathcal{Y})^n$ of n i.i.d. realizations of a generic random variable (X, Y), an OVK $\mathcal{K} : \mathcal{X}\times\mathcal{X} \to \mathcal{L}(\mathcal{Y})$, a convex loss function $\ell : \mathcal{Y}\times\mathcal{Y} \to \mathbb{R}$, and a regularization parameter $\Lambda > 0$, the general form of an OVK-based learning problem is to find $\hat{h}$ that solves:

$$\min_{h\in\mathcal{H}_{\mathcal{K}}} \frac{1}{n}\sum_{i=1}^n \ell(h(x_i), y_i) + \frac{\Lambda}{2}\|h\|^2_{\mathcal{H}_{\mathcal{K}}}. \tag{1}$$

Similarly to scalar ones, a crucial tool in operator-valued kernel methods is the *Representer Theorem*, ensuring that $\hat{h}$ actually pertains to a reduced subspace of $\mathcal{H}_{\mathcal{K}}$.

Theorem 2. *(Theorem 4.2 in Micchelli and Pontil (2005)) There exists $(\hat{\alpha}_i)_{i=1}^n \in \mathcal{Y}^n$ such that*

$$\hat{h} = \frac{1}{\Lambda n}\sum_{i=1}^n \mathcal{K}(\cdot, x_i)\hat{\alpha}_i.$$

Although Theorem 2 drastically downscales the search domain (from $\mathcal{H}_{\mathcal{K}}$ to $\mathcal{Y}^n$), it gives no further information about the $(\hat{\alpha}_i)_{i=1}^n$. One way to gain insight about these coefficients is to perform Problem (1)'s dualization, with the notation $\ell_i : y \in \mathcal{Y} \mapsto \ell(y, y_i)$ for any $i \leqslant n$.

Theorem 3. *(Appendix B in Brouard et al. (2016b)) The solution to Problem (1) is given by*

$$\hat{h} = \frac{1}{\Lambda n}\sum_{i=1}^n \mathcal{K}(\cdot, x_i)\hat{\alpha}_i, \tag{2}$$

with $(\hat{\alpha}_i)_{i=1}^n \in \mathcal{Y}^n$ the solutions to the dual problem

$$\min_{(\alpha_i)_{i=1}^n\in\mathcal{Y}^n} \sum_{i=1}^n \ell_i^\star(-\alpha_i) + \frac{1}{2\Lambda n}\sum_{i,j=1}^n \langle \alpha_i, \mathcal{K}(x_i, x_j)\alpha_j\rangle_{\mathcal{Y}}\,, \tag{3}$$

where $f^\star : \alpha \in \mathcal{Y} \mapsto \sup_{y\in\mathcal{Y}} \langle \alpha, y\rangle_{\mathcal{Y}} - f(y)$ denotes the Fenchel-Legendre transform of a function $f : \mathcal{Y} \to \mathbb{R}$.

Refer to Appendix A.1 for Theorem 3's proof, that has been reproduced for self-containedness. Dualization brings in additional information about the optimal coefficients (notice nonetheless that Theorem 2 holds true for a much wider class of problems). As it is, Problem (3) is however of little interest, since the optimization must be performed on the infinite dimensional space $\mathcal{Y}^n$. Depending on the problem, we propose two solutions: either using a *Double Representer Theorem*, or by approximating Problem (3).

Notation. If $\mathcal{K}$ is identity decomposable (i.e. $\mathcal{K} = k\,\mathbf{I}_{\mathcal{Y}}$), K^X and K^Y denote the input and output gram matrices. For any matrix M, $M_{i:}$ represents its i^{th} line, and $\|M\|_{p,q}$ its $\ell_{p,q}$ row wise mixed norm, *i.e.* the ℓ_q norm of the ℓ_p norms of its lines. χ_S denotes the characteristic function of a set S, null on S and equal to $+\infty$ otherwise, $f \,\Box\, g$ is the infimal convolution of f and g (Bauschke et al., 2011), $(f \,\Box\, g)(x) = \inf_y f(y) + g(x-y)$. Finally, $\#S$ is the cardinality of any set S, and $\|\cdot\|_{\text{op}}$ the operator norm.

2.1. The Double Representer Theorem

In order to make Problem (3) solvable, we need assumptions on the loss and the kernel. Let $\mathbf{Y}$ denote $\text{span}(y_i,\ i \leqslant n)$. Assumptions 1 and 2 characterize admissible losses through conditions on their Fenchel-Legendre (FL) transforms. They are standard for kernel methods, and ensure computability by stipulating that only dot products are involved.

Assumption 1. *$\forall i \leqslant n,\ \forall (\alpha^{\mathbf{Y}}, \alpha^{\perp}) \in \mathbf{Y} \times \mathbf{Y}^{\perp}$, it holds $\ell_i^\star(\alpha^{\mathbf{Y}}) \leqslant \ell_i^\star(\alpha^{\mathbf{Y}} + \alpha^{\perp})$.*

Assumption 2. *$\forall i \leqslant n, \exists L_i : \mathbb{R}^{n+n^2} \to \mathbb{R}$ such that for all $\boldsymbol{\omega} = (\omega_j)_{j \leqslant n} \in \mathbb{R}^n, \quad \ell_i^\star(-\sum_{j=1}^n \omega_j\, y_j) = L_i(\boldsymbol{\omega}, K^Y)$.*

Regarding the OVK, the key point is Assumption 3. Roughly speaking, $\mathbf{Y}$ is what we *see* and *know* about output space $\mathcal{Y}$, while $\mathbf{Y}^{\perp}$ represents the part we *ignore*. What we need is an OVK somewhat *aligned* with the outputs, in the sense that the little we know about $\mathcal{Y}$ should be preserved through $\mathcal{K}$. As for Assumption 4, it helps simplifying the computations.

Assumption 3. *$\forall i, j \leqslant n$, $\mathbf{Y}$ is invariant by $\mathcal{K}(x_i, x_j)$, i.e. $\forall y \in \mathcal{Y},\ \ y \in \mathbf{Y} \Rightarrow \mathcal{K}(x_i, x_j)y \in \mathbf{Y}$.*

Remark 1. *It is important to notice that we do not need Assumption 3 to hold true for every collection $\{y_i\}_{i \leqslant n} \in \mathcal{Y}^n$. It rather constitutes an a posteriori condition to ensure that the kernel is aligned with the training sample at hand. If $\mathcal{Y}$ is finite dimensional, one may hope that with sufficiently many outputs, then $\mathbf{Y}$ spans $\mathcal{Y}$, and every matrix-valued kernel then fits. If $\mathcal{Y}$ is infinite dimensional, identity-decomposable kernels are admissible (which despite simple expressions may describe nontrivial dependences in infinite dimensional spaces). Moreover, separable kernels with operators similar to the empirical covariance $\sum_i y_i \otimes y_i$ (Kadri et al., 2013) are also eligible, opening the door to ad-hoc and learned kernels, see Appendix A.8 for further examples.*

Assumption 4. *There exist $T \geqslant 1$, and for every $t \leqslant T$ admissible scalar kernels $k_t : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ as well as positive semi-definite operators $A_t \in \mathcal{L}(\mathcal{Y})$, such that for all $(x, x') \in \mathcal{X}^2$ it holds: $\mathcal{K}(x, x') = \sum_{t=1}^T k_t(x, x') A_t$.*

Under Assumption 4, K_t^X and K_t^Y denote the matrices such that $[K_t^X]_{ij} = k_t(x_i, x_j)$, $[K_t^Y]_{ij} = \langle y_i, A_t y_j \rangle_{\mathcal{Y}}$. Notice that it is by no means restrictive, since every shift-invariant OVK can be approximated arbitrarily closely by kernels satisfying Assumption 4. Furthermore, if for all $t \leqslant T$, A_t keeps $\mathbf{Y}$ invariant, then Assumption 3 is directly fulfilled. Under these assumptions, Theorem 4 proves that the optimal coefficients lie in $\mathbf{Y}^n$, ensuring the solutions computability.

Theorem 4. *Let $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be a loss function with Fenchel-Legendre transforms satisfying Assumptions 1 and 2, and $\mathcal{K}$ be an OVK verifying Assumption 3. Then, the solution to Problem (1) is given by*

$$\hat{h} = \frac{1}{\Lambda n} \sum_{i,j=1}^{n} \mathcal{K}(\cdot, x_i)\, \hat{\omega}_{ij}\, y_j, \tag{4}$$

with $\hat{\Omega} = [\hat{\omega}_{ij}] \in \mathbb{R}^{n \times n}$ the solution to the dual problem

$$\min_{\Omega \in \mathbb{R}^{n \times n}} \ \sum_{i=1}^{n} L_i\left(\Omega_{i:}, K^Y\right) + \frac{1}{2\Lambda n} \mathbf{Tr}\left(\tilde{M}^\top (\Omega \otimes \Omega)\right),$$

with M the n^4 tensor such that $M_{ijkl} = \langle y_k, \mathcal{K}(x_i, x_j) y_l \rangle_{\mathcal{Y}}$, and $\tilde{M}$ its rewriting as a $n^2 \times n^2$ block matrix. If kernel $\mathcal{K}$ further satisfies Assumption 4, then tensor M simplifies to $M_{ijkl} = \sum_{t=1}^T [K_t^X]_{ij} [K_t^Y]_{kl}$, and the problem rewrites

$$\min_{\Omega \in \mathbb{R}^{n \times n}} \ \sum_{i=1}^{n} L_i\left(\Omega_{i:}, K^Y\right) + \frac{1}{2\Lambda n} \sum_{t=1}^{T} \mathbf{Tr}\left(K_t^X \Omega K_t^Y \Omega^\top\right). \tag{5}$$

See Appendix A.2 for the proof. This theorem can be seen as a *Double Representer Theorem*, since both theorems share analogous proofs and consequences: a search domain reduction, respectively from $\mathcal{H}_{\mathcal{K}}$ to $\mathcal{Y}^n$, and $\mathcal{Y}^n$ to $\mathbb{R}^{n \times n}$.

Remark 2. *The* Double Representer Theorem *emphasizes that only the knowledge of the n^4 tensor M is required to make OVK problems in infinite dimensional output spaces computable. Although it might seem prohibitive at first sight, one has to keep in mind that, like for scalar kernel methods, a first n^2 cost is needed to use (input) kernels with infinite dimensional feature maps, while the second n^2 cost allows for handling infinite dimensional outputs. In the case of a decomposable kernel, one has $M_{ijkl} = K_{ij}^X K_{kl}^Y$. One only needs two n^2 matrices, recovering the scalar complexity.*

We now present a non-exhaustive list of admissible losses (one may refer to Appendix A.3 for the proof).

Proposition 1. *The following losses have Fenchel-Legendre transforms verifying Assumptions 1 and 2:*

- $\ell_i(y) = f(\langle y, z_i \rangle)$, $z_i \in Y$ *and* $f : \mathbb{R} \to \mathbb{R}$ *convex. This encompasses maximum-margin regression, obtained with* $z_i = y_i$ *and* $f(t) = \max(0, 1-t)$.
- $\ell(y) = f(\|y\|)$, $f : \mathbb{R}_+ \to \mathbb{R}$ *convex increasing s.t.* $t \mapsto \frac{f'(t)}{t}$ *is continuous over* $\mathbb{R}_+$. *This includes all power functions* $\frac{\lambda}{\eta}\|y\|_{\mathcal{Y}}^{\eta}$ *for* $\eta > 1$ *and* $\lambda > 0$.
- $\forall \lambda > 0$, *with* $\mathcal{B}_\lambda$ *the centered ball of radius* λ,
  - $\ell(y) = \lambda\|y\|$,
  - $\ell(y) = \lambda\|y\|\log(\|y\|)$,
  - $\ell(y) = \chi_{\mathcal{B}_\lambda}(y)$,
  - $\ell(y) = \lambda(\exp(\|y\|) - 1)$.
- $\ell_i(y) = f(y - y_i)$, $f^\star$ *verifying Assumptions 1 and 2.*
- *Any infimal convolution involving functions satisfying Assumptions 1 and 2. This encompasses* ϵ*-insensitive losses (Sangnier et al., 2017), the Huber loss (Huber, 1964), and generally all Moreau or Pasch-Hausdorff envelopes (Moreau, 1962; Bauschke et al., 2011).*

2.2. Approximating the Dual Problem

If Assumption 3 is not satisfied, another way to get a finite dimensional decomposition similar to that of Theorem 4 is to approximate the dual problem. This may be done by restricting the dual variables to suitable finite dimensional subsets of $\mathcal{Y}$, if the following hypothesis on kernel $\mathcal{K}$ holds.

Assumption 5. *The kernel* $\mathcal{K} = k \cdot A$ *is a separable OVK, with* A *a compact operator.*

Recalling that A is by design self adjoint and positive, its compactness then allows for a spectral decomposition: there exists an orthonormal basis $(\psi_j)_{j=1}^{\infty}$ of $\mathcal{Y}$, and some positive $(\lambda_j)_{j=1}^{\infty}$, ordered in a non-increasing fashion and converging to zero, such that $A = \sum_{j=1}^{\infty} \lambda_j \psi_j \otimes \psi_j$ (Osborn, 1975).

Using such a basis, one can say that there exists $(\hat{\omega}_i)_{i=1}^n \in \ell^2(\mathbb{R})^n$ such that $\forall i \leqslant n, \hat{\alpha}_i = \sum_{j=1}^{\infty} \hat{\omega}_{ij}\psi_j$. Since this leads to an infinite size representation of the dual variables, the idea is then to restrict the search space to the eigenspace associated to the m largest eigenvalues of A, for some $m > 0$. Let $\widetilde{\mathcal{Y}}_m$ denote $\text{span}(\{\psi_j\}_{j=1}^m)$, and $S = \text{diag}(\lambda_j)_{j=1}^m$. An approximated dual problem reads

$$\min_{(\alpha_i)_{i=1}^n \in \widetilde{\mathcal{Y}}_m^n} \sum_{i=1}^n \ell_i^\star(-\alpha_i) + \frac{1}{2\Lambda n} \sum_{i,j=1}^n \langle \alpha_i, \mathcal{K}(x_i, x_j)\alpha_j \rangle_{\mathcal{Y}}, \tag{6}$$

We now state a condition similar to Assumption 2, which makes the solution to Problem (6) computable.

Assumption 6. $\forall i \leqslant n, \exists L_i : \mathbb{R}^{2m} \to \mathbb{R}$ *such that* $\forall\, \boldsymbol{\omega} = (\omega_j)_{j \leqslant m} \in \mathbb{R}^m$, $\ell_i^\star(-\sum_{j=1}^m \omega_j\, \psi_j) = L_i(\boldsymbol{\omega}, R_{i:})$, *with* $R \in \mathbb{R}^{n \times m}$ *the matrix such that* $R_{ij} = \langle y_i, \psi_j \rangle_{\mathcal{Y}}$.

Remark 3. *Assumption 6 is similar to Assumption 2, except that the output Gram matrix* K^Y *is replaced by matrix* R *storing the dot products between the orthonormal family* $\{\psi_j\}_{j=1}^m$ *and the outputs. In particular, all losses explicited in Proposition 1 have FL transforms verifying Assumption 6.*

Theorem 5. *Let* $\mathcal{K}$ *be an OVK meeting Assumption 5 and* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ *be a loss function with FL transforms satisfying Assumption 6. Then, Problem (6) is equivalent to*

$$\min_{\Omega \in \mathbb{R}^{n \times m}} \sum_{i=1}^n L_i\left(\Omega_{i:}, R_{i:}\right) + \frac{1}{2\Lambda n} \mathbf{Tr}\left(K^X \Omega S \Omega^\top\right). \tag{7}$$

Denoting by $\hat{\Omega} = [\hat{\omega}_{ij}] \in \mathbb{R}^{n \times m}$ *the solution to Problem (7), the associated predictor is finally given by*

$$\hat{h} = \frac{1}{\Lambda n} \sum_{i=1}^n \sum_{j=1}^m k(\cdot, x_i)\, \lambda_j\, \hat{\omega}_{ij}\, \psi_j, \tag{8}$$

Remark 4. *The rationale behind the above approximation is that under compactness of* A, *Equation (8) constitutes a reasonable approximation of Equation (2). Notice that Kadri et al. (2016) use a truncated spectral decomposition of the operator to implement a functional version of Kernel Ridge Regression, without resorting to dualization however.*

3. Application to Robust Losses

We now instantiate Theorem 4's dual problem for three loss functions encouraging data sparsity and robustness. They write as infimal convolutions, and are thus hardly tractable in the primal. Their dual problems enjoy simple resolution algorithms that are thoroughly detailed. A stability analysis is also carried out to highlight the hyperparameters impact.

3.1. Complete Dual Resolution for Three Robust Losses

As a first go, we recall the important notion of ϵ-insensitive losses. Following in the footsteps of Sangnier et al. (2017), we extend them in a natural way from $\mathbb{R}^p$ to any Hilbert space $\mathcal{Y}$. To avoid additional notation, in this subsection ℓ denotes the loss taken w.r.t. one argument (previously ℓ_i).

Definition 3. *Let* $\ell : \mathcal{Y} \to \mathbb{R}_+$ *be a convex loss such that* $\ell(0) = 0$, *and* $\epsilon > 0$. *The* ϵ*-insensitive version of* ℓ, *denoted* ℓ_ϵ, *is defined by* $\ell_\epsilon(y) = (\ell \,\square\, \chi_{\mathcal{B}_\epsilon})(y)$, *or again:*

$$\forall y \in \mathcal{Y},\ \ \ell_\epsilon(y) = \begin{cases} 0 & \textit{if } \|y\|_{\mathcal{Y}} \leqslant \epsilon \\ \inf\limits_{\|d\|_{\mathcal{Y}} \leqslant 1} \ell(y - \epsilon d) & \textit{otherwise} \end{cases}.$$

In other terms, $\ell_\epsilon(y)$ is the smallest value of ℓ within the ball of radius ϵ centered at y. As revealed by the next definition, natural choices for ℓ yield extensions of celebrated scalar loss functions to infinite dimensional Hilbert spaces.

Definition 4. *If $\ell = \|\cdot\|_{\mathcal{Y}}$, then $\|\cdot\|_{\mathcal{Y},\epsilon} = \max(\|\cdot\|_{\mathcal{Y}} - \epsilon, 0)$, and the related problem is the natural extension of ϵ-SVR.*

If $\ell = \|\cdot\|^2_{\mathcal{Y}}$, then $\|\cdot\|^2_{\mathcal{Y},\epsilon} = \max(\|\cdot\|_{\mathcal{Y}} - \epsilon, 0)^2$, and the related problem is called the ϵ-insensitive Ridge regression.

The third framework that nicely falls into our resolution methodology is the Huber loss regression (Huber, 1964). Tailored to induce robustness, the Huber loss function does not feature convolution with $\chi_{\mathcal{B}_\epsilon}$ but rather between the first two powers of the Hilbert norm (that used in Definition 4).

Definition 5. *The Huber loss of parameter κ is given by $\ell_{H,\kappa}(y) = (\kappa\|\cdot\|_{\mathcal{Y}} \,\square\, \frac{1}{2}\|\cdot\|^2_{\mathcal{Y}})(y)$, or again:*

$$\forall y \in \mathcal{Y}, \;\; \ell_{H,\kappa}(y) = \begin{cases} \frac{1}{2}\|y\|^2_{\mathcal{Y}} & \text{if } \|y\|_{\mathcal{Y}} \leqslant \kappa \\ \kappa\left(\|y\|_{\mathcal{Y}} - \frac{\kappa}{2}\right) & \text{otherwise} \end{cases} .$$

Due to its asymptotic behavior as $\|\cdot\|_{\mathcal{Y}}$, the Huber loss is useful when the training data is heavy tailed or contains outliers. Illustrations of Definitions 4 and 5's loss functions in one and two dimensions are available in Appendix B. Interestingly, Problem (5) for these three losses – and an identity decomposable kernel – admits a very nice writing, allowing for an efficient resolution.

Theorem 6. *If $\mathcal{K} = k\,\mathbf{I}_{\mathcal{Y}}$, the solutions to the ϵ-Ridge regression, κ-Huber regression, and ϵ-SVR primal problems*

$$(P1) \quad \min_{h \in \mathcal{H}_{\mathcal{K}}} \; \frac{1}{2n}\sum_{i=1}^{n} \|h(x_i) - y_i\|^2_{\mathcal{Y},\epsilon} + \frac{\Lambda}{2}\|h\|^2_{\mathcal{H}_{\mathcal{K}}},$$

$$(P2) \quad \min_{h \in \mathcal{H}_{\mathcal{K}}} \; \frac{1}{n}\sum_{i=1}^{n} \ell_{H,\kappa}(h(x_i) - y_i) + \frac{\Lambda}{2}\|h\|^2_{\mathcal{H}_{\mathcal{K}}},$$

$$(P3) \quad \min_{h \in \mathcal{H}_{\mathcal{K}}} \; \frac{1}{n}\sum_{i=1}^{n} \|h(x_i) - y_i\|_{\mathcal{Y},\epsilon} + \frac{\Lambda}{2}\|h\|^2_{\mathcal{H}_{\mathcal{K}}},$$

are given by Equation (4), with $\hat{\Omega} = \hat{W}V^{-1}$, and $\hat{W}$ the solution to the respective finite dimensional dual problems

$$(D1) \quad \min_{W \in \mathbb{R}^{n \times n}} \; \frac{1}{2}\|AW - B\|^2_{\text{Fro}} + \epsilon\,\|W\|_{2,1},$$

$$(D2) \quad \min_{W \in \mathbb{R}^{n \times n}} \; \frac{1}{2}\|AW - B\|^2_{\text{Fro}}, \qquad \text{s.t.} \quad \|W\|_{2,\infty} \leqslant \kappa,$$

$$(D3) \quad \min_{W \in \mathbb{R}^{n \times n}} \; \frac{1}{2}\|AW - B\|^2_{\text{Fro}} + \epsilon\,\|W\|_{2,1}, \qquad \text{s.t.} \quad \|W\|_{2,\infty} \leqslant 1,$$

with V, A, B such that: $VV^\top = K^Y$, $A^\top A = \frac{K^X}{\Lambda n} + \mathbf{I}_n$ (or $A^\top A = K^X/(\Lambda n)$ for the ϵ-SVR), and $A^\top B = V$.

Theorem 6's proof is detailed in Appendix A.5. If $\mathcal{K}$ is not identity decomposable, but only satisfies Assumption 4, the dual problems do not admit compact writings such as those of Theorem 6. Nonetheless, they are still easily solvable, and the standard Ridge regression is recovered for $\epsilon = 0$ or $\kappa = +\infty$. This is discussed at length in the Appendix.

Problem $(D1)$ is a Multi-Task Lasso problem (Obozinski et al., 2010). It can be solved by Projected Gradient Descent (PGD), that involves the Block Soft Thresholding operator such that $\text{BST}(x, \tau) = (1 - \tau/\|x\|)_+\, x$. Problem $(D2)$ is a constrained least square problem, that also admits a resolution through PGD, but with the Projection operator such that $\text{Proj}(x, \tau) = \min(\tau/\|x\|, 1)\, x$. Finally, Problem $(D3)$ combines both non-smooth terms and consequently both projection steps. Given a stepsize η, and T a number of epoch, the algorithms are detailed in Algorithm 1. Note that $\tilde{K}$'s Singular Value Decomposition is not necessary, since the computations only involve $A^\top A = \tilde{K}$ and $A^\top B = V$.

Algorithm 1 Projected Gradient Descents (PGDs)

```
input : Gram matrices $K^X$, $K^Y$, parameters $\Lambda$, $\epsilon$, $\kappa$
init  : $\tilde{K} = \frac{1}{\Lambda n}K^X + \mathbf{I}_n$ (or $\tilde{K} = \frac{1}{\Lambda n}K^X$ for $\epsilon$-SVR),
        $K^Y = VV^\top$, $W = \mathbf{0}_{\mathbb{R}^{n \times n}}$
for epoch from $1$ to $T$ do
    // gradient step
    $W = W - \eta(\tilde{K}W - V)$
    // projection step
    for row $i$ from $1$ to $n$ do
        $W_{i:} = \text{BST}(W_{i:}, \epsilon)$                   // if Ridge or SVR
        $W_{i:} = \text{Proj}(W_{i:}, \kappa \text{ or } 1)$      // if Huber or SVR
return $W$
```

3.2. Approximate Dual Resolution with Huber Loss

In this section we solve Problem (6) for the Huber loss and $\mathcal{Y} = L^2[\Theta, \mu]$, with Θ a compact set endowed with measure μ. A classical choice of OVK is then $\mathcal{K} = k_{\mathcal{X}} \cdot T_k$, $k_{\mathcal{X}}$ being a scalar kernel over the inputs, and T_k the integral operator associated to a scalar kernel $k\colon \Theta \times \Theta \to \mathbb{R}$ defined for all $g \in L^2[\Theta, \mu]$ by $T_k g = \int_\Theta k(\cdot, \theta)g(\theta)\mathrm{d}\mu(\theta)$. Continuity of k grants compactness of T_k, allowing for the methodology presented in Section 2.2. In the following, $(\lambda_j, \psi_j)_{j=1}^m$ denotes the eigendecomposition of T_k, which is dependent both in k and μ, and can be obtained by solving a differential equation derived from the eigenvalue problem. However, given that the optimal kernel k is unknown, one can choose a Hilbertian basis $\{\psi_j\}_{j=1}^\infty$ of $L^2[\Theta, \mu]$ and a non-increasing summable sequence $(\lambda_j)_{j=1}^\infty \in \mathbb{R}_+^*$ to construct the kernel k, which gives direct access to T_k's eigendecomposition.

Theorem 7. *For an OVK $\mathcal{K} = k_{\mathcal{X}}\, T_k$, an approximate solution to the Huber loss regression problem*

$$\min_{h \in \mathcal{H}_{\mathcal{K}}} \; \frac{1}{n}\sum_{i=1}^{n} \ell_{H,\kappa}(h(x_i) - y_i) + \frac{\Lambda}{2}\|h\|^2_{\mathcal{H}_{\mathcal{K}}},$$

is given by Equation (8), with $\hat{\Omega}$ the solution to the following constrained quadratic problem (with R as in Assumption 6), that can be tackled by PGD in the spirit of Algorithm 1:

$$\begin{aligned} \min_{\Omega \in \mathbb{R}^{n \times m}} \quad & \mathbf{Tr}\left(\frac{1}{2}\Omega\Omega^\top + \frac{1}{2\Lambda n}K^X \Omega S \Omega^\top - \Omega R^\top\right), \\ \text{s.t.} \quad & \|\Omega\|_{2,\infty} \leqslant \kappa. \end{aligned} \tag{9}$$

Remark 5. *When κ is large, one recovers the unconstrained Ridge regression problem, whose solution enjoys a closed form expression, and for which a resolution method based on an approximation of the inverse of the integral operator T_k was presented in Kadri et al. (2016).*

3.3. Stability Analysis

Algorithm stability is a notion introduced by Bousquet and Elisseeff (2002). It links the *stability* of an algorithm, *i.e.* how removing a training observation impacts the algorithm output, to the algorithm generalization capacity, *i.e.* how far the empirical risk of the algorithm output is to its true risk. The rationale behind this approach is that standard analyses of Empirical Risk Minimization rely on a crude approximation consisting in bounding the empirical process $\sup_{h \in \mathcal{H}} |\hat{\mathcal{R}}_n(h) - \mathcal{R}(h)|$. Indeed, considering a supremum over the whole hypothesis set seems very pessimistic, as decision functions with high discrepancy $|\hat{\mathcal{R}}_n(h) - \mathcal{R}(h)|$ would hopefully not be selected by the algorithm. However, the limitation of stability approaches lies in that algorithms performances are never compared to an optimal solution h^*. Nevertheless, their capacity to deal with OVK machines without making the trace-class assumption (as opposed to Rademacher-based strategies, see *e.g.* Maurer and Pontil (2016)) make them particularly well suited to our setting. In the footsteps of Audiffren and Kadri (2013), we now derive stability bounds for our algorithms, which are all the more relevant as they make explicit the role of hyperparameters. For any algorithm A, $h_{A(\mathcal{S})}$ and $h_{A(\mathcal{S}^{\setminus i})}$ denote the decision functions output by the algorithm, respectively trained on samples $\mathcal{S}$ and $\mathcal{S}^{\setminus i} = \mathcal{S} \backslash \{(x_i, y_i)\}$. Notice that symmetry among observations in Problem (1) cancels the impact of i. Formally, algorithm stability states as follows.

Definition 6. *(Bousquet and Elisseeff, 2002) Algorithm A has stability β if for any sample $\mathcal{S}$, and any $i \leqslant \#\mathcal{S}$, it holds: $\sup_{(x,y) \in \mathcal{X} \times \mathcal{Y}} |\ell(h_{A(\mathcal{S})}(x), y) - \ell(h_{A(\mathcal{S}^{\setminus i})}(x), y)| \leqslant \beta$.*

Assumption 7. *There exists $M > 0$ such that for any sample $\mathcal{S}$ and any realization $(x, y) \in \mathcal{X} \times \mathcal{Y}$ of (X, Y) it holds: $\ell(h_{A(\mathcal{S})}(x), y) \leqslant M$.*

Theorem 8. *(Bousquet and Elisseeff, 2002) Let A be an algorithm with stability β and loss function satisfying Assumption 7. Then, for any $n \geqslant 1$ and $\delta \in]0, 1[$ it holds with probability at least $1 - \delta$:*

$$\mathcal{R}(h_{A(\mathcal{S})}) \leqslant \hat{\mathcal{R}}_n(h_{A(\mathcal{S})}) + 2\beta + (4n\beta + M)\sqrt{\frac{\ln(1/\delta)}{2n}}.$$

Stability for OVK machines such as in Problem (1) may be derived from the following two assumptions.

Assumption 8. *There exists $\gamma > 0$ such that for any input observation $x \in \mathcal{X}$ it holds: $\|\mathcal{K}(x, x)\|_{op} \leqslant \gamma^2$.*

Assumption 9. *There exists $C > 0$ such that for any point $(x, y) \in \mathcal{X} \times \mathcal{Y}$, any sample $\mathcal{S}$, and any $i \leqslant \#\mathcal{S}$, it holds: $|\ell(h_{\mathcal{S}}(x), y) - \ell(h_{\mathcal{S}^{\setminus i}}(x), y)| \leqslant C\|h_{\mathcal{S}}(x) - h_{\mathcal{S}^{\setminus i}}(x)\|_{\mathcal{Y}}$.*

Theorem 9. *(Audiffren and Kadri, 2013) If Assumptions 8 and 9 hold, then the algorithm returning the solution to Problem (1) has β stability with $\beta \leqslant C^2\gamma^2/(\Lambda n)$.*

In order to get generalization bounds, we shall now derive constants M and C of Assumptions 7 and 9 respectively. This is usually done under the following assumption.

Assumption 10. *There exists $M_{\mathcal{Y}} > 0$ such that for any realization $y \in \mathcal{Y}$ of Y it holds: $\|y\|_{\mathcal{Y}} \leqslant M_{\mathcal{Y}}$.*

Remark 6. *It should be noticed that in structured prediction or structured data representation this assumption is directly fulfilled with $M_{\mathcal{Y}} = 1$. Indeed, outputs (and potentially inputs) are actually some $y_i = \phi(z_i)$, with ϕ the canonical feature map associated to a scalar kernel, so that it suffices to choose a normalized kernel to satisfy Assumption 10.*

Theorem 10. *Under Assumption 10, algorithms previously described satisfy Assumptions 7 and 9 with constants M and C as detailed in Figure 1.*

4. Applications and Numerical Experiments

In this section, we discuss some applications unlocked by vv-RKHSs with infinite dimensional outputs. In particular, structured prediction, structured representation learning, and functional regression are formally described, and numerical experiments highlight the benefits of the losses introduced.

4.1. Application to Structured Output Prediction

Assume one is interested in learning a predictive decision rule f from a set $\mathcal{X}$ to a complex structured space $\mathcal{Z}$. To bypass the absence of norm on $\mathcal{Z}$, one may design a (scalar) kernel k on $\mathcal{Z}$, whose canonical feature map $\phi : z \mapsto k(\cdot, z)$ transforms any element of $\mathcal{Z}$ into an element of the (scalar) RKHS associated to k, denoted $\mathcal{Y}$ ($= \mathcal{H}_k$). Learning a predictive model f from $\mathcal{X}$ to $\mathcal{Z}$ boils down to learning a surrogate vector-valued model h from $\mathcal{X}$ to $\mathcal{Y}$, which is searched for in the vv-RKHS $\mathcal{H}_{\mathcal{K}}$ associated to an OVK $\mathcal{K}$ by solving the following regularized empirical problem.

$$\hat{h} = \underset{h \in \mathcal{H}_{\mathcal{K}}}{\operatorname{argmin}} \ \frac{1}{n}\sum_{i=1}^n \ell(h(x_i), \phi(z_i)) + \frac{\Lambda}{2}\|h\|^2_{\mathcal{H}_{\mathcal{K}}}. \tag{10}$$

Once $\hat{h}$ is learned, the predictions in $\mathcal{Z}$ are produced through a pre-image problem $f(x) = \operatorname{argmin}_{z \in \mathcal{Z}} \ell(\phi(z), \hat{h}(x))$. This approach called Input Output Kernel Regression has

	M	C
ϵ-SVR	$\sqrt{M_{\mathcal{Y}} - \epsilon}\left(\frac{\sqrt{2}\gamma}{\sqrt{\Lambda}} + \sqrt{M_{\mathcal{Y}} - \epsilon}\right)$	1
ϵ-Ridge	$(M_{\mathcal{Y}} - \epsilon)^2\left(1 + \frac{2\sqrt{2}\gamma}{\sqrt{\Lambda}} + \frac{2\gamma^2}{\Lambda}\right)$	$2(M_{\mathcal{Y}} - \epsilon)\left(1 + \frac{\gamma\sqrt{2}}{\sqrt{\Lambda}}\right)$
κ-Huber	$\kappa\sqrt{M_{\mathcal{Y}} - \frac{\kappa}{2}}\left(\frac{\gamma\sqrt{2\kappa}}{\sqrt{\Lambda}} + \sqrt{M_{\mathcal{Y}} - \frac{\kappa}{2}}\right)$	κ

Figure 1. Algorithms Constants

$$\begin{tikzcd}
 & \phi(z) \in \mathcal{Y} \arrow[d, dashed, bend left, "\text{argmin}"] \\
x \in \mathcal{X} \arrow[ur, "\hat{h} \in \mathcal{H}_{\mathcal{K}}"] \arrow[r, "f"'] & z \in \mathcal{Z} \arrow[u, "\phi"]
\end{tikzcd}$$

Figure 2. Output Kernel Regression

been studied in several works (Brouard et al., 2011; Kadri et al., 2013). As an instance of the general Output Kernel Regression scheme of Figure 2, it belongs to the family of Surrogate Approaches for structured prediction (see *e.g.* Ciliberto et al. (2016)). While previous works have focused on identity decomposable kernels only, with the squared loss or hinge loss (Brouard et al., 2016b), our general framework allows for many more losses. The use of an ϵ-insensitive loss in Problem (10), in particular, seems adequate as it is a surrogate task, and inducing small mistakes that do not harm the inverse problem, while improving generalization, sounds as a suitable compromise. We thus advocate to solve structured prediction in vv-RKHSs by using losses more sophisticated than the squared norm. In the following, the variants of IOKR are called accordingly to the loss they minimize: ϵ-SV-IOKR, ϵ-Ridge-IOKR, and Huber-IOKR.

YEAST dataset. Although our approach's main strength of is to predict infinite dimensional outputs, we start with a simpler standard structured prediction dataset composed of 14-dimensional outputs (the so-called YEAST dataset Finley and Joachims (2008)) described in the Supplements, on which comparisons and interpretations are easier. We have collected results from Finley and Joachims (2008) and Belanger and McCallum (2016), and benchmarked our three algorithms. Hyperparameters Λ, ϵ, κ have been selected among geometrical grids by cross-validation on the train dataset solely, and performances evaluated on the same test set as the above publications. Results in terms of Hamming error are reported in Figure 6, with significant improvements for the ϵ-Ridge-IOKR and Huber-IOKR. Furthermore, in order to highlight the interactions between our two ways of regularizing, *i.e.* the RKHS norm and the ϵ-insensitivity, we have plotted the ϵ-Ridge-IOKR Mean Square Errors (the Hamming before clamping) and solution sparsity with respect to Λ for ϵ varying from $1e$-5 to 1.5 (Figures 3 and 4): Λ and ϵ seem to act as competitive regularizations. When Λ is small, the regularization in ϵ is efficient, as solution with the best MSE is obtained for ϵ around 0.6. Conversely, when Λ is big, no sparsity is induced, and having a high ϵ induces too much regularization. Similar graphs for the ϵ-SVR and κ-Huber are available in the Supplements, that highlight the superiority of the approaches for a wide range of hyperparameters. A linear output kernel was used, such that solving the inverse problem boils down to clamping.

Metabolite dataset. Regarding the infinite dimensional outputs, we have considered the metabolite identification problem (Schymanski et al., 2017), in which one aims at predicting molecules from their mass spectra. For this task, Ridge-IOKR is the state-of-the-art approach, corresponding to our ϵ-Ridge-IOKR with $\epsilon = 0$. Given the high number of constraints, Structured SVMs are not tractable as confirmed by our tests using the Pystruct lib implementation (Müller and Behnke, 2014). This was already noticed in Belanger and McCallum (2016) (14 is the maximum output dimension on which SSVMs were tested), and the implementation we tried indeed yielded very poor results despite prolonged training ($5\%, 31\%, 45\%$ top-k errors). We thus investigated the advantages of substituting the standard Ridge Regression for its ϵ-insensitive version or a Huber regression. Outputs (*i.e.* metabolites) are embedded in an infinite dimensional Hilbert space through a Tanimoto-Gaussian kernel with 0.72 bandwidth. The dataset, presented in the Supplements and described at length in Brouard et al. (2016a), is composed of 6974 mass spectra, while algorithms are compared through the top-k accuracies, $k = 1, 10, 20$. Two Λ's have been picked for their interesting behavior: one that yields the best performance for Ridge-IOKR, and the second that gives the best overall scores (hyperparameters ϵ and κ being chosen to produce the best scores each time). Again, results of Table 1 show improvements due to robust losses that are all the more important as the norm regularization is low, with an improvement on the best overall score.

Table 1. Top 1 / 10 / 20 test accuracies (%)

Λ	$1e$-6	$1e$-4
Ridge-IOKR	35.7 | 79.9 | 86.6	38.1 | 82.0 | 88.9
ϵ-Ridge-IOKR	37.1 | 81.7 | 88.3	36.3 | 81.2 | 87.9
Huber-IOKR	**38.3** | **82.2** | **89.1**	37.7 | 81.9 | 88.8

4.2. Structured Representation Learning

Extracting vectorial representations from structured inputs is another task that can be tackled in vv-RKHSs (Laforgue et al., 2019). This is a relevant approach in many cases: when complex data are uniquely available under the form of a similarity matrix for instance, for preserving privacy, or when deep neural networks fail to tackle structured objects

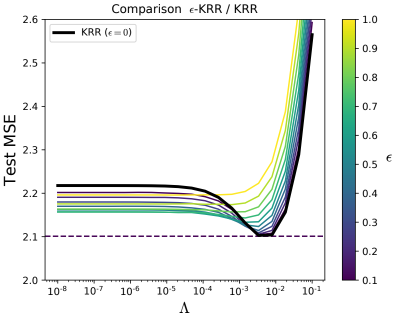

Figure 3. Test MSE w.r.t. Λ

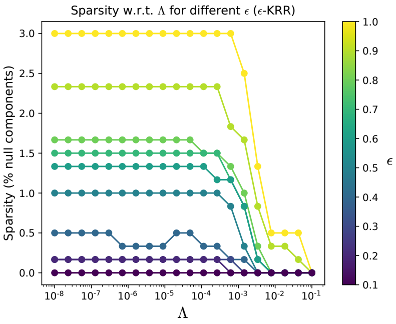

Figure 4. Sparsity w.r.t. Λ

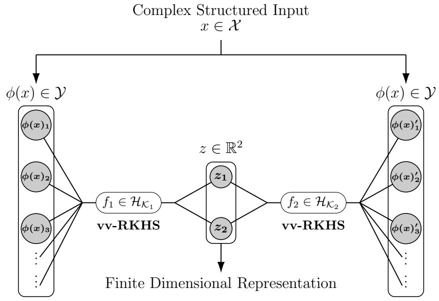
Complex Structured Input
$x \in \mathcal{X}$
$\phi(x) \in \mathcal{Y}$
$\phi(x)_1$
$\phi(x)_2$
$\phi(x)_3$
$f_1 \in \mathcal{H}_{\mathcal{K}_1}$
vv-RKHS
$z \in \mathbb{R}^2$
z_1
z_2
$f_2 \in \mathcal{H}_{\mathcal{K}_2}$
vv-RKHS
$\phi(x) \in \mathcal{Y}$
$\phi(x)'_1$
$\phi(x)'_2$
$\phi(x)'_3$
Finite Dimensional Representation

Figure 5. 2-Layer Kernel Autoencoder

SSVM	20.2
SPEN	20.0
ϵ-RIDGE-IOKR	**19.0**
HUBER-IOKR	19.1
ϵ-SV-IOKR	21.1

Figure 6. YEAST Hamming errors

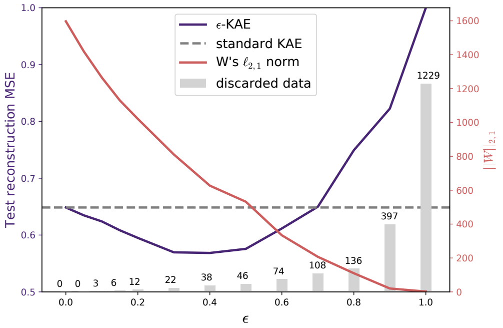

Figure 7. Reconstruction error w.r.t. ϵ

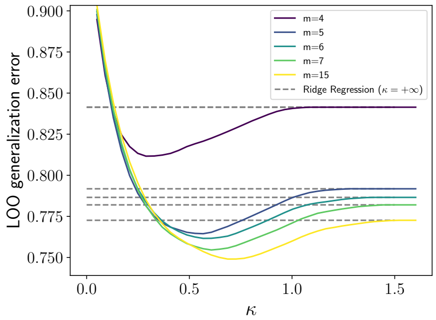

Figure 8. LOO error w.r.t. κ

as raw data. Embedding data into a Hilbert space makes sense. Then, composing functions in vv-RKHSs results in a Kernel Autoencoder (KAE, Figure 5) that outputs finite codes by minimizing the (regularized) discrepancy:

$$\frac{1}{2n}\sum_{i=1}^{n}\|\phi(x_i) - f_2 \circ f_1(\phi(x_i))\|_{\mathcal{Y}}^2 + \Lambda\,\mathrm{Reg}(f_1, f_2). \quad (11)$$

Again, this reconstruction loss is not the real goal, but rather a proxy to make the internal representation meaningful. Therefore, all incentives to use ϵ-insensitive losses or the Huber loss still apply. The inferred ϵ-KAE and Huber-KAE, obtained by changing the loss function in Problem (11), are optimized as follows: the first layer coefficients are updated by Gradient Descent, while the second ones are reparametrized into W_2 and updated through PGD (instead of KRR closed form for standard KAEs). This has been applied to a drug dataset, introduced in Su et al. (2010) as an extract from the NCI-Cancer database. As shown in Figure 7, the ϵ-insensitivity improves the generalization while inducing sparsity. The ϵ-insensitive framework is thus particularly promising in the context of Autoencoders.

4.3. Function-to-Function Regression

Regression with both inputs and outputs of functional nature is a challenging problem at the crossroads of Functional Data Analysis (Ramsay and Silverman, 2007) and Machine Learning (Kadri et al., 2016). While Functional Linear Modeling is the most common approach to address function-to-function regression, nonparametric approaches based on vv-RKHSs have emerged, that rely on the minimization of a squared loss. However, robustness to abnormal functions is particularly meaningful in a field where data come from sensors and are used to monitor physical assets. To the best of our knowledge, robust regression has only been tackled in the context of Functional Linear Models (Kalogridis and Van Aelst, 2019). We propose here to highlight the relevance of OVK machines learned with a Huber loss by solving Problem (9) for various levels κ.

Lip acceleration from EMG dataset. We consider the problem of predicting lip acceleration among time from electromyography (EMG) signals (Ramsay and Silverman, 2007). The dataset consists of 32 records of the lower lip trajectory over 641 timestamps, and the associated EMG records, augmented with 4 outliers to assess the robustness of our approach. Usefulness of minimizing the Huber loss is illustrated in Figure 8 by computing the Leave-One-Out (LOO) error associated to each model for various values of m. For each m, as κ grows larger than a threshold, the constraint on $\|\Omega\|_{2,\infty}$ becomes void and we recover the Ridge Regression solution. The kernel chosen is given by $k_{\mathcal{X}}(x_1, x_2) = \int_0^1 \exp\left(|x_1(\theta) - x_2(\theta)|\right)\mathrm{d}\theta$, with $(\psi_j)_{j=1}^m$ being the harmonic basis in sine and cosine of $L^2[0,1]$, and $(\lambda_j)_{j=1}^m = (1/(j+1)^2)_{j=1}^m$.

4.4. Related Work

Another application of the presented results, both theoretical and computational, is the generalization of the *loss trick*, see *e.g.* Ciliberto et al. (2016). In the context of Output Kernel Regression, the latter stipulates that for suitable losses, the decoding expresses in terms of loss evaluations. The work by Luise et al. (2019) has extended this trick to penalization schemes different from the natural vv-RKHS norm. Our findings, and the double expansion in particular, suggest that the loss trick can still be used with other surrogate loss functions than the squared norm, opening the door to a wide range of applications.

5. Conclusion

This work presents a versatile framework based on duality to learn OVK machines with infinite dimensional outputs. The case of convolved losses (*e.g.* ϵ-insensitive, Huber) is thoroughly tackled, from algorithmic procedures to stability analysis. This offers novel ways to enforce sparsity and robustness when learning within vv-RKHSs, opening an avenue for new applications on structured and functional data (*e.g.* anomaly detection, robust prediction). Future research directions could feature a calibration study of these novel surrogate approaches, or the introduction of kernel approximations such as random Fourier features, that would benefit our framework twice: both in input and in output.

Acknowledgment. This work has been partially funded by the industrial chair *Data Science & Artificial Intelligence for Digitalized Industry and Services* from Télécom Paris. Authors would like to thank Olivier Fercoq for his insightful discussions and helpful comments.

References

Álvarez, M. A., Rosasco, L., and Lawrence, N. D. (2012). Kernels for vector-valued functions: a review. *Foundations and Trends in Machine Learning*, 4(3):195–266.

Audiffren, J. and Kadri, H. (2013). Stability of multi-task kernel regression algorithms. In *Asian Conference on Machine Learning*, pages 1–16.

Baldassarre, L., Rosasco, L., Barla, A., and Verri, A. (2012). Multi-output learning via spectral filtering. *Machine Learning*, 87(3):259–301.

Bauschke, H. H., Combettes, P. L., et al. (2011). *Convex analysis and monotone operator theory in Hilbert spaces*, volume 408. Springer.

Belanger, D. and McCallum, A. (2016). Structured prediction energy networks. In *International Conference on Machine Learning*, pages 983–992.

Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. *Journal of machine learning research*, 2(Mar):499–526.

Brault, R., Lambert, A., Szabo, Z., Sangnier, M., and d'Alché-Buc, F. (2019). Infinite task learning in rkhss. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1294–1302.

Brouard, C., d'Alché-Buc, F., and Szafranski, M. (2011). Semi-supervised penalized output kernel regression for link prediction. In *International Conference on Machine Learning (ICML)*, pages 593–600.

Brouard, C., Shen, H., Dührkop, K., d'Alché-Buc, F., Böcker, S., and Rousu, J. (2016a). Fast metabolite identification with input output kernel regression. *Bioinformatics*, 32(12):28–36.

Brouard, C., Szafranski, M., and D'Alché-Buc, F. (2016b). Input output kernel regression: supervised and semi-supervised structured output prediction with operator-valued kernels. *The Journal of Machine Learning Research*, 17(1):6105–6152.

Caponnetto, A., Micchelli, C. A., Pontil, M., and Ying, Y. (2008). Universal multitask kernels. *Journal of Machine Learning Research*, 9:1615–1646.

Carmeli, C., De Vito, E., and Toigo, A. (2006). Vector valued reproducing kernel hilbert spaces of integrable functions and mercer theorem. *Analysis and Applications*, 4(04):377–408.

Carmeli, C., De Vito, E., Toigo, A., and Umanitá, V. (2010). Vector valued reproducing kernel hilbert spaces and universality. *Analysis and Applications*, 8(01):19–61.

Ciliberto, C., Rosasco, L., and Rudi, A. (2016). A consistent regularization approach for structured prediction. In *Advances in Neural Information Processing Systems (NIPS) 29*, pages 4412–4420.

Dinuzzo, F., Ong, C., Gehler, P., and Pillonetto, G. (2011). Learning output kernels with block coordinate descent. In *International Conference on Machine Learning (ICML)*, pages 49–56.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A. J., and Vapnik, V. (1997). Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161.

Finley, T. and Joachims, T. (2008). Training structural svms when exact inference is intractable. In *Proceedings of the 25th international conference on Machine learning*, pages 304–311.

Fung, G. and Mangasarian, O. L. (2000). Data selection for support vector machine classifiers. In Ramakrishnan, R., Stolfo, S. J., Bayardo, R. J., and Parsa, I., editors, *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, Boston, MA, USA, August 20-23, 2000*, pages 64–70. ACM.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC.

Hofmann, T., Schoelkopf, B., and Smola, A. J. (2008). Kernel methods in machine learning. *Ann. Statist.*, 36(3):1171–1220.

Huber, P. J. (1964). Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pages 73–101.

Joachims, T., Hofmann, T., Yue, Y., and Yu, C.-N. (2009). Predicting structured objects with support vector machines. *Commun. ACM*, 52(11):97–104.

Kadri, H., Duflos, E., Preux, P., Canu, S., and Davy, M. (2010). Nonlinear functional regression: a functional rkhs approach. In Teh, Y. W. and Titterington, M., editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 374–380. PMLR.

Kadri, H., Duflos, E., Preux, P., Canu, S., Rakotomamonjy, A., and Audiffren, J. (2016). Operator-valued kernels for learning from functional response data. *Journal of Machine Learning Research*, 17(20):1–54.

Kadri, H., Ghavamzadeh, M., and Preux, P. (2013). A generalized kernel approach to structured output learning. In *International Conference on Machine Learning (ICML)*, pages 471–479.

Kalogridis, I. and Van Aelst, S. (2019). Robust functional regression based on principal components. *Journal of Multivariate Analysis*, 173:393 – 415.

Laforgue, P., Clémençon, S., and d'Alché-Buc, F. (2019). Autoencoding any data through kernel autoencoders. In *Artificial Intelligence and Statistics*, pages 1061–1069.

Luise, G., Stamos, D., Pontil, M., and Ciliberto, C. (2019). Leveraging low-rank relations between surrogate tasks in structured prediction. *arXiv preprint arXiv:1903.00667*.

Maurer, A. and Pontil, M. (2016). Bounds for vector-valued function estimation. *arXiv preprint arXiv:1606.01487*.

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

Moreau, J. J. (1962). Fonctions convexes duales et points proximaux dans un espace hilbertien. *Comptes rendus hebdomadaires des séances de l'Académie des sciences*, 255:2897–2899.

Müller, A. C. and Behnke, S. (2014). pystruct - learning structured prediction in python. *Journal of Machine Learning Research*, 15:2055–2060.

Obozinski, G., Taskar, B., and Jordan, M. I. (2010). Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252.

Osborn, J. E. (1975). Spectral approximation for compact operators. *Mathematics of computation*, 29(131):712–725.

Ramsay, J. O. and Silverman, B. W. (2007). *Applied functional data analysis: methods and case studies*. Springer.

Sangnier, M., Fercoq, O., and d'Alché-Buc, F. (2017). Data sparse nonparametric regression with ϵ-insensitive losses. In *Asian Conference on Machine Learning*, pages 192–207.

Schymanski, E., Ruttkies, C., and Krauss, M. e. a. (2017). Critical assessment of small molecule identification 2016: automated methods. *J Cheminform*, 9:22.

Senkene, E. and Tempel'man, A. (1973). Hilbert spaces of operator-valued functions. *Lithuanian Mathematical Journal*, 13(4):665–670.

Su, H., Heinonen, M., and Rousu, J. (2010). Structured output prediction of anti-cancer drug activity. In *IAPR International Conference on Pattern Recognition in Bioinformatics*, pages 38–49. Springer.

Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of machine learning research*, 6(Sep):1453–1484.

Zhu, J., Hoi, S. C. H., and Lyu, M. R. (2008). Robust regularized kernel regression. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(6):1639–1644.