Multiple Kernel Testing for SVM-based System Identification

Matthew Higgs[*]
University College London
m.higgs@cs.ucl.sc.uk

John Shawe-Taylor
University College London
jst@cs.ucl.sc.uk

Abstract

We apply methods of multiple kernel learning to the problem of system identification for multi-dimensional temporal data. Rather than building a full probabilistic model, we take a computationally simple approach that uses out of the box machine learning methods. We attempt to learn the covariance function of a stochastic process via multiple kernel learning. We achieve promising preliminary results and the work suggests an abundance of future theoretical work. We hope to draw on the theory of SVM methods to give a principled learning theory style description of system identification in stochastic processes.

1 Introduction

Inference and learning in dynamical systems has long been studied and there exists a large number of probabilistic methods to tackle the problem. Whether the method is linear Gaussian (Kalman filter) or nonlinear (extended Kalman filter, particle filter, variational) the standard method for model selection generally follows the (regularised) maximum likelihood approach. A notion of generalisation is not well defined for learning a dynamical system, and methods such as cross validation can be impractical due to strong model dependencies and insufficient (non-iid) data. Multiple kernel learning (MKL) presents an alternative approach to model selection in classification and regression and has been developed as an extension to the standard one-kernel SVM. Here we consider applying a MKL ν-SVM to the problem of parameter estimation in a dynamical system.

Given a set of observations $\{\mathbf{y}_i \in \mathbb{R}^d\}_{i=1}^n$ at times $\{t_1, \ldots, t_n\}$ we would like to build a model of the generating process. In this paper this is done by utilising the relation between kernels and differential operators or, in the Gaussian process setting, between covariance functions and stochastic differential equations. We fix a set of positive definite *test kernels* $\{\mathbf{K}_j : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}^d \times \mathbb{R}^d, j = 1, \ldots, p\}$ and perform multiple kernel learning with a ν-SVM sub-loop. The outputs of the algorithm (kernel-weightings, support-vectors, predicted values) allow us to draw conclusions about the underlying process. We therefore dub the method *multiple kernel testing*. We first need generalise MKL to the $\mathbb{R}^d$-valued ν-SVM [3]. The $\mathbb{R}^d$-valued ν-SVM has a dual form analogous to the $\mathbb{R}$-valued ν-SVM, and the standard two-step MKL optimisation procedure of [4] is applicable. In section 3 we apply the $\mathbb{R}^d$-valued ν-SVM MKL algorithm to the problem of learning the covariance function of a stochastic process. We present two simple examples with promising results. The examples are linear and we give a short description of how the method can be extended to non-linear systems.

Notation: We use bold lower case for vectors and bold uppercase for matrices. Row $\mathbf{v}^\top$ is the transpose of $\mathbf{v}$ and we will use $\mathbf{v} \cdot \mathbf{u} := \mathbf{v}^\top \mathbf{u}$ when the vectors are decorated with superscripts.

[*]Center for Computational Statistics and Machine Learning

2 Vector-valued MKL ν-SVM

We consider a convex combination of positive definite kernels from the set $\{\mathbf{K}_j : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}^d \times \mathbb{R}^d, j = 1, \ldots, p\}$, such that

$$\mathbf{K}_{\boldsymbol{\eta}}(s,t) := \sum_{j=1}^{p} \eta_j \mathbf{K}_j(s,t), \quad \boldsymbol{\eta} \in \mathbb{R}_+^p, \quad ||\boldsymbol{\eta}||_1 = 1. \tag{1}$$

Let $||\cdot||_p$ denote the Euclidean p-norm. Let $\mathcal{H}_j : \mathbb{R}_+ \to \mathbb{R}^d$ be the RKHS associated with kernel $\mathbf{K}_j$. Assume $\boldsymbol{\eta}$ is fixed and define $||\cdot||^2_{\mathcal{H}_{\boldsymbol{\eta}}} := \sum_{j=1}^{p} \eta_j ||\cdot||^2_{\mathcal{H}_j}$. Using ϵ-insensitive loss $L_{\epsilon,p}(\mathbf{y}, \hat{\mathbf{y}}) := \max(0, ||\mathbf{y} - \hat{\mathbf{y}}||_p - \epsilon)$ and $||\cdot||^2_{\mathcal{H}_{\boldsymbol{\eta}}}$-norm regularisation, we obtain a regression function

$$f(t) = \sum_{i=1}^{n} \mathbf{K}_{\boldsymbol{\eta}}(t_i, t)(\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-) + \mathbf{b} \tag{2}$$

where $\{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^- \in \mathbb{R}^d\}_{i=1}^p$ solve the dual problem (derived in appendix)

$$\begin{aligned}
\max_{\{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^-\}} \quad & \sum_{i=1}^{n} (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-)^\top \mathbf{y}_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-)^\top \mathbf{K}_{\boldsymbol{\eta}}(\mathbf{x}_i, \mathbf{x}_j)(\boldsymbol{\gamma}_j^+ - \boldsymbol{\gamma}_j^-) \\
\text{subject to} \quad & \sum_{i=1}^{n} (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-) = \mathbf{0}, \\
& \sum_{i=1}^{n} ||\boldsymbol{\gamma}_i^{(\pm)}||_{\frac{p}{p-1}} \leq \nu, \\
& ||\boldsymbol{\gamma}_i^{(\pm)}||_{\frac{p}{p-1}} \leq \frac{C}{n}.
\end{aligned} \tag{3}$$

Notation $(\pm)$ implies the predicate holds for superscripts '$+$' and '$-$'. Bias $\mathbf{b} \in \mathbb{R}^d$ and tube width ϵ can be obtained from the KKT conditions (see [3] and [7]). Let $J(\boldsymbol{\eta}, \{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^-\})$ denote the objective function in (3). Following the simple method of [2] we compute

$$\frac{\partial J(\boldsymbol{\eta}, \{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^-\})}{\partial \eta_k} = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-)^\top, \mathbf{K}_k(t_i, t_j)(\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-). \tag{4}$$

In an iterative fashion, we obtain $\{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^-\}^{(k)}$ for $\boldsymbol{\eta}^{(k)}$ and update $\boldsymbol{\eta}^{(k+1)} = \pi_\triangle\big(\boldsymbol{\eta}^{(k)} - \mu_k \nabla_\eta J(\boldsymbol{\eta}^{(k)}, \{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^-\}^{(k)})\big)$, where $\pi_\triangle$ projects onto the unit simplex. The step size μ_k is determined using Armijo's rule.

3 Multiple kernel testing

We consider applying the above multiple kernel learning algorithm to the problem of system identification. We are given a set of observations $\{\mathbf{y}_i \in \mathbb{R}^d\}_{i=1}^n$ at times $\{t_1, \ldots, t_n\}$. We consider a set of positive definite kernels $\mathbf{K}_j$ corresponding to the covariance function $\mathbf{K}_j$ of a particular Gaussian process. In the following examples, observations are generated from the underlying processes and we analyse whether parameters of the processes can be can be estimated by learning the covariance function using multiple kernel learning.

Toy example (1): We consider the Ornstein-Uhlenbeck (OU) process given by

$$dX = -\gamma X dt + \sigma dW, \tag{5}$$

where $\gamma > 0$ is a drift parameter and σdW the scaled Wiener measure. A typical path sampled from (5) can be observed in figure 3. We would like to fit mean-path's exponential decay rate γ to some data. For suitable initial conditions, the corresponding covariance function $K(s,t)$ is given by

$$K_\gamma(s,t) = \frac{\sigma^2}{2\gamma} e^{-\gamma|s-t|}. \tag{6}$$

In experiment we removed the $\frac{1}{\gamma}$ term so that K_γ has range $[0, \frac{\sigma^2}{2}]$ for all $\gamma > 0$. Parameter γ was fixed to 1 in (5) and observations were drawn according to a iid Gaussian noise model. A MKL ν-SVM ($\nu = 0.3$, $C = 10$) was applied to the set of kernels $\{K_\gamma(s,t) : \gamma \in \{10^{-i/4}\}_{i=8}^{-6}\}$. At each gradient step the kernel weights were projected back onto the unit-simplex. In figure 1 we see a stem plot of the output weight η for each kernel. The peak corresponds to the high kernel-weights. The bold stem plot corresponds to the weight on the kernel with the same γ as the one used to generate the data from (5).

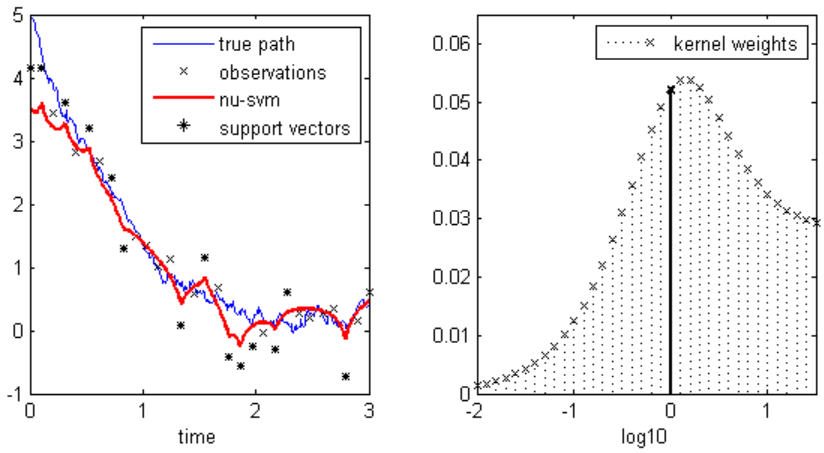

Figure 1: (**LH**) Path from OU-process, noisy observations and MKL ν-SVM results. (**RH**) MKL output kernel-weight η against $\log_{10}(\gamma)$ (**bold** stem is kernel with γ value same as (5)).

Toy example (2): We consider the damped oscillator with system noise described the second order system

$$\begin{bmatrix} dX \\ dV \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\lambda^2 & -2\lambda \end{bmatrix} \begin{bmatrix} dX \\ dV \end{bmatrix} dt + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \sigma dW. \tag{7}$$

The deterministic dynamics of the mean path evolve according to $\ddot{x} - 2\lambda\dot{x} - \lambda^2 x = 0$, an over-damped oscillator. A typical path sampled from (7) can be observed in figure 2. The covariance function corresponding to (7) is the Mátern kernel of order $2/3$ given by

$$K_\lambda(s,t) \propto \sigma^2(\lambda|s-t|)^{2/3} B_{2/3}(\lambda|s-t|), \tag{8}$$

where $B_{2/3}$ is the modified Bessel-function of second kind. Parameter λ in (7) was fixed to 0.2 and observations were drawn according to a iid Gaussian noise model. A ν-SVM ($\nu = 0.1$, $C = 10$) with multiple kernel learning was applied to the set of kernels $\{K_\lambda(s,t) : \lambda \in \{10^{-i/4}\}_{i=8}^{-6}\}$. At each gradient step the kernel weights were projected back onto the unit-simplex. In figure 2 we see a stem plot of the output weight η for each kernel. The bold stem plot corresponds to the weight on the kernel with the same λ as the one used to generate the data from (7). Both estimates in figures 1 and 2 are based on observations from only one sample path and therefore susceptible to instabilities. Robust estimation is performed on multiple sample paths.

3.1 Non-stationary, non-linear systems

The given examples are only linear, but nonlinear and/or non-stationary systems can also be dealt with approximately using *localised* MKL. We introduce a parameterised set of time-dependent kernel-weights $\{\varphi_j : \mathbb{R}_+ \times \boldsymbol{\Theta} \to \mathbb{R}, j = 1, \ldots, p\}$, where $\boldsymbol{\Theta}$ is a matrix of parameters. The selection of kernels can be seen as a multi-class classification problem and the gaiting model of [5] can be applied. Define

$$\mathbf{K}_{\boldsymbol{\Theta}}(s,t) = \sum_{j=1}^{p} \varphi_j(t|\boldsymbol{\Theta}) \varphi_j(s|\boldsymbol{\Theta}) \mathbf{K}_j(s,t). \tag{9}$$

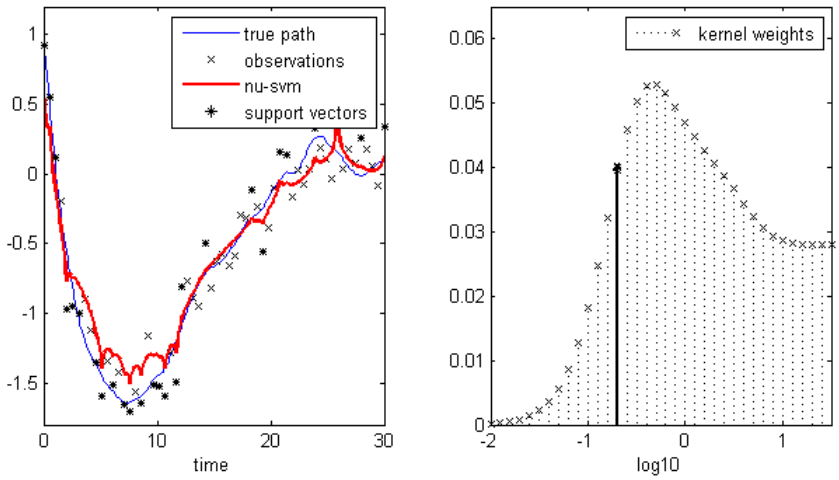

Figure 2: (**LH**) Path from stochastic damped-oscillator, noisy observations and MKL ν-SVM results. (**RH**) MKL output kernel-weight η against $\log_{10}(\lambda)$ (**bold** stem is kernel with λ value same as (7)).

For the case when φ_j is constant for all $i = 1, \ldots, p$, we have $\eta_j = \varphi_j^2$ and retrieve standard MKL. The gaiting model of [5] assumes linear separability between the regions of kernel choice. In the setting of non-stationary systems, trajectories may repeat certain behaviours numerous time. This would require us to use multiples of the same kernel in the test set. It is therefore desirable to look for kernel weights $\{\varphi_j\}$ that allow us to reuse kernels. One possibility is to define

$$\varphi_j(t|\mathbf{\Theta}) = \sum_{i=1}^{n} \Theta_{i,j}\psi_i(t_i - t), \quad \sum_{k=1}^{p} \Theta_{i,k} = 1, \tag{10}$$

where $\{\psi_i\}$ is a set of radial basis functions. The method of describing non-linear behaviour by a non-stationary kernel is very similar to the linear variational approximation of a nonlinear SDE given in [1]. As an example take the highly nonlinear double-well system

$$dX \quad = \quad f(X)dt + \sigma dW_t, \tag{11}$$

where $f(x) = 4x(\theta - x^2), \quad \theta > 0$. The behaviour of the system primarily depends on the stability of the three equilibrium points $0, \pm\sqrt{\theta}$. Linearising around each point generates three local OU processes (one with a possibly hazardous kernel). An affective localised MKL algorithm should choose a kernel at time t that is similar to the covariance function of the corresponding closest local equilibrium.

4 Conclusions

In this paper we have identified a new application for multiple kernel learning. We have provided preliminary results, and the technique suggests a large number of future theoretical avenues. The algorithm is simple and we would like to experiment with more advanced MKL methods. We would also like to consider the following points. One interpretation of the kernel weights is that they represent the probabilities that a sample path was drawn from one of a finite set of GPs. This kind of probabilistic description may not be compatible with the MKL ν-SVM algorithm, and we should maybe instead consider questions about the relevance of the support vectors. For example whether they tell us anything about the underlying dynamics. One possibility is that the support vectors represent optimal boundary conditions, in between which the inferred path follows the natural path of the covariance function, and the support vectors come in to play when the mean path needs to deviate due to data. This would go someway to relating the predictive path of the ν-SVM to the posterior mean of GP regression. The full version of the paper looks at multi-dimensional experiments and finalising the approach to nonlinear systems. As with the linear case the computational methods already exist and the focus is on applications and the interpretation of results.

References

[1] Cdric Archambeau, Dan Cornford, D. Lawrence, Anton Schwaighofer, and Joaquin Quionero C. Gaussian process approximations of stochastic differential equations. In *Journal of Machine Learning Research Workshop and Conference Proceedings*, page 2007, 2007.

[2] Olivier Bousquet and Daniel J. L. Herrmann. On the complexity of learning the kernel matrix. In *In Advances in Neural Information Processing Systems 15*, pages 399–406. MIT Press, 2003.

[3] Mark Brudnak. Vector-valued support vector regression. In *IEEE International Joint Conference on Neural Networks*, 2006.

[4] Olivier Chapelle, Vladimir Vapnik, Olivier Bousquet, and Sayan Mukherjee. Choosing multiple parameters for support vector machines. *Machine Learning*, 46(1–3):131–159, 2002.

[5] Mehmet Gönen and Ethem Alpaydn. Localized multiple kernel regression. In *ICPR 2010*, 2010.

[6] Charles A. Micchelli and Massimiliano A. Pontil. On learning vector-valued functions. *Neural Comput.*, 17(1):177–204, 2005.

[7] Bernhard Schoelkopf, Alex J. Smola, Robert C. Williamson, and Peter L. Bartlett. New support vector algorithms, 2000.

A Derivation of $\mathbb{R}^d$-valued ν-SVM

We avoid any implicit feature mapping. Let $\mathcal{X}$ denote a topological set and let $\mathbb{R}^d$ denote the d-dimensional real Euclidean-space equipped with p-norm $||\cdot||_p$. Let $\mathbf{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^d \times \mathbb{R}^d$ denote a positive definite kernel and let $\boldsymbol{\mathcal{H}}$ denote its reproducing kernel Hilbert space of $\mathbb{R}^d$-valued functions, with norm $||\cdot||_{\boldsymbol{\mathcal{H}}}$. Define $L_{\epsilon,p}(\mathbf{y}, \hat{\mathbf{y}}) := \max(0, ||\mathbf{y} - \hat{\mathbf{y}}||_p - \epsilon)$. Given a set of observations $\{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{X} \times \mathbb{R}^d\}_{i=1}^n$ we look to solve

$$\min_{\mathbf{f} \in \boldsymbol{\mathcal{H}}, \mathbf{b} \in \mathcal{Y}, \epsilon \geq 0} \quad \frac{1}{2}||\mathbf{f}||^2_{\boldsymbol{\mathcal{H}}} + \frac{C}{n}\sum_{i=1}^n L_{\epsilon,p}(\mathbf{y}_i, \mathbf{f}(\mathbf{x}_i) + \mathbf{b}) + \nu\epsilon. \tag{12}$$

To help smooth (12) we introduce slack variables $\{\xi_i \in \mathbb{R}_+\}_{i=1}^n$, and rewrite (12) as

$$\begin{aligned} \min_{\mathbf{f} \in \boldsymbol{\mathcal{H}}, \mathbf{b} \in \mathcal{Y}, \epsilon, \{\xi_i\}} \quad & \frac{1}{2}||\mathbf{f}||^2_{\boldsymbol{\mathcal{H}}} + \frac{C}{n}\sum_{i=1}^n \xi_i + \nu\epsilon \\ \text{subject to} \quad & ||\mathbf{y}_i - \mathbf{f}(\mathbf{x}_i) - \mathbf{b}||_p \leq \epsilon + \xi_i, \quad \epsilon, \xi_i \geq 0. \end{aligned} \tag{13}$$

To move to the dual requires us to differentiate the p-norm with respect to $\mathbf{f}$ and $\mathbf{b}$. To keep notation cleaner we introduce additional slack variables $\{\boldsymbol{\zeta}_i^+, \boldsymbol{\zeta}_i^- \in \mathbb{R}^d_+\}_{i=1}^n$ and rewrite (13) as

$$\begin{aligned} \min_{\mathbf{f}, \mathbf{b}, \epsilon, \{\xi_i\}, \{\boldsymbol{\zeta}_i^+, \boldsymbol{\zeta}_i^-\}} \quad & \frac{1}{2}||\mathbf{f}||^2_{\boldsymbol{\mathcal{H}}} + \frac{C}{n}\sum_{i=1}^n \xi_i + \nu\epsilon \\ \text{subject to} \quad & ||\boldsymbol{\zeta}_i^+ + \boldsymbol{\zeta}_i^-||_p \leq \epsilon + \xi_i, \quad \epsilon, \xi_i \geq 0, \\ & \boldsymbol{\zeta}_i^+ - \mathbf{y}_i + \mathbf{f}(\mathbf{x}_i) + \mathbf{b} \in \mathbb{R}^d_+, \quad \boldsymbol{\zeta}_i^+ \in \mathbb{R}^d_+, \\ & \boldsymbol{\zeta}_i^- + \mathbf{y}_i - \mathbf{f}(\mathbf{x}_i) - \mathbf{b} \in \mathbb{R}^d_+, \quad \boldsymbol{\zeta}_i^- \in \mathbb{R}^d_+. \end{aligned} \tag{14}$$

Introducing Lagrange multipliers $\alpha_i, \beta, \eta_i \geq 0$ and $\boldsymbol{\theta}_i^+, \boldsymbol{\theta}_i^-, \boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^- \in \mathbb{R}^d_+$, the Lagrangian of (14) is given by

$$\begin{aligned} \mathcal{L} \quad = \quad & \frac{1}{2}||\mathbf{f}||^2_{\boldsymbol{\mathcal{H}}} + \frac{C}{n}\sum_{i=1}^n \xi_i + \nu\epsilon - \sum_{i=1}^n \alpha_i(\epsilon + \xi_i - ||\boldsymbol{\zeta}_i^+ + \boldsymbol{\zeta}_i^-||_p) - \beta\epsilon - \sum_{i=1}^n \eta_i\xi_i - \sum_{i=1}^n \boldsymbol{\theta}_i^+ \cdot \boldsymbol{\zeta}_i^+ \\ & - \sum_{i=1}^n \boldsymbol{\gamma}_i^+ \cdot (\boldsymbol{\zeta}_i^+ - \mathbf{y}_i + \mathbf{f}(\mathbf{x}_i) + \mathbf{b}) - \sum_{i=1}^n \boldsymbol{\theta}_i^- \cdot \boldsymbol{\zeta}_i^- - \sum_{i=1}^n \boldsymbol{\gamma}_i^- \cdot (\boldsymbol{\zeta}_i^- + \mathbf{y}_i - \mathbf{f}(\mathbf{x}_i) - \mathbf{b}). \\ \text{s.t.} \quad & \alpha_i, \beta, \eta_i \geq 0, \quad \boldsymbol{\theta}_i^+, \boldsymbol{\theta}_i^-, \boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^- \in \mathbb{R}^d_+. \end{aligned}$$

Before differentiating $\mathcal{L}$ with respect to the primal variables, we recall by the representer theorem (see, for example [6]) that the minimiser $\mathbf{f}^*$ of $\mathcal{L}$ is given by

$$\mathbf{f}^*(\cdot) = \sum_{i=1}^n \mathbf{K}(\mathbf{x}_i, \cdot)\mathbf{c}_i^*, \tag{15}$$

for some $\{\mathbf{c}_i^* \in \mathbb{R}^d\}_{i=1}^n$. Replacing $\mathbf{f}(\cdot)$ by $\sum_{i=1}^n \mathbf{K}(\mathbf{x}_i, \cdot)\mathbf{c}_i$ in $\mathcal{L}$ gives squared-norm term $||\mathbf{f}||_{\mathcal{H}}^2 = \sum_{i=1}^n \sum_{j=1}^n \mathbf{c}_i^\top \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)\mathbf{c}_j$. Then, differentiating $\mathcal{L}$ with respect to $\{\mathbf{c}_i\}, \mathbf{b}, \epsilon, \{\xi_i\}, \{\boldsymbol{\zeta}_i^+, \boldsymbol{\zeta}_i^-\}$ gives the Euler-Lagrange equations

$$\begin{aligned}
\mathbf{c}_i &= (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-) && (16)\\
\sum_{i=1}^n (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-) &= \mathbf{0} && (17)\\
\sum_{i=1}^n \alpha_i &= \nu - \beta && (18)\\
\eta_i &= \frac{C}{n} - \alpha_i && (19)\\
\boldsymbol{\theta}_i^{(\pm)} &= \alpha_i \nabla_{\boldsymbol{\zeta}_i^{(\pm)}} \left(||\boldsymbol{\zeta}_i^+ + \boldsymbol{\zeta}_i^-||_p \right) - \boldsymbol{\gamma}_i^{(\pm)} && (20)
\end{aligned}$$

where $(\pm)$ implies the predicate holds for superscripts '$+$' and '$-$'. Inserting (16)-(20) and lemma 1 into $\mathcal{L}$ we get

$$\mathcal{L} = \sum_{i=1}^n (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-)^\top \mathbf{y}_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-)^\top \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)(\boldsymbol{\gamma}_j^+ - \boldsymbol{\gamma}_j^-). \quad (21)$$

Using (18)-(19), we have

$$\begin{aligned}
\beta \geq 0 \quad &\Rightarrow \quad \sum_{i=1}^n \alpha_i \leq \nu, && (22)\\
\eta_i \geq 0 \quad &\Rightarrow \quad \alpha_i \leq \frac{C}{n}, && (23)
\end{aligned}$$

and using (20) with lemma 2 we have

$$\boldsymbol{\theta}_i^{(\pm)} \in \mathbb{R}_+^d \quad \Rightarrow \quad 0 \leq ||\boldsymbol{\gamma}_i^{(\pm)}||_q \leq \alpha_i, \quad (24)$$

where $q = \frac{p}{p-1}$. Combing (24) with (22) and (23) we get constraints, where we maximise the dual (21) over $\{\boldsymbol{\gamma}_i^+, \boldsymbol{\gamma}_i^-\}$,

$$\begin{aligned}
\text{subject to} \qquad & \sum_{i=1}^n (\boldsymbol{\gamma}_i^+ - \boldsymbol{\gamma}_i^-) = \mathbf{0}\\
& \sum_{i=1}^n ||\boldsymbol{\gamma}_i^{(\pm)}||_q \leq \nu\\
& ||\boldsymbol{\gamma}_i^{(\pm)}||_q \leq \frac{C}{n}.
\end{aligned}$$

B Technical lemmas

Lemma 1. *For any $\mathbf{v}, \mathbf{u} \in \mathbb{R}^d$ it holds that*

$$||\mathbf{v} + \mathbf{u}||_p = \left(\nabla_{\mathbf{v}} ||\mathbf{v} + \mathbf{u}||_p \right) \cdot \mathbf{v} + \left(\nabla_{\mathbf{u}} ||\mathbf{v} + \mathbf{u}||_p \right) \cdot \mathbf{u}. \quad (25)$$

Proof.

$$\begin{aligned}
A &= \left(\nabla_{\mathbf{v}} ||\mathbf{v} + \mathbf{u}||_p \right) \cdot \mathbf{v} + \left(\nabla_{\mathbf{u}} ||\mathbf{v} + \mathbf{u}||_p \right) \cdot \mathbf{u}\\
&= \frac{||\mathbf{v} + \mathbf{u}||_p}{||\mathbf{v} + \mathbf{u}||_p^p} (\mathbf{v} + \mathbf{u})^{p-1} \cdot \mathbf{v} + \frac{||\mathbf{v} + \mathbf{u}||_p}{||\mathbf{v} + \mathbf{u}||_p^p} (\mathbf{v} + \mathbf{u})^{p-1} \cdot \mathbf{u}\\
&= ||\mathbf{v} + \mathbf{u}||_p.
\end{aligned}$$

□

Lemma 2 (Proof, see [3])**.** *For any $\mathbf{v} \in \mathbb{R}^d$ and any $p \in \mathbb{N}$ and $q = \frac{p}{p-1}$, it holds that*

$$\left|\left| \nabla_{\mathbf{v}} ||\mathbf{v}||_p \right|\right|_q = 1. \quad (26)$$