ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

Learning with hard constraints as a limit case of learning with soft constraints

Giorgio Gnecco[1] and Marco Gori[2] and Stefano Melacci[2] and Marcello Sanguineti[3]

1- IMT - School for Advanced Studies - DYSCO
Piazza S. Francesco 19, Lucca - Italy
giorgio.gnecco@imtlucca.it

2- University of Siena - DIISM
Via Roma 56, Siena - Italy
{marco, mela}@dii.unisi.it

3- University of Genoa - DIBRIS
Via all'Opera Pia 13, Genoa - Italy
marcello.sanguineti@unige.it

Abstract. We refer to the framework of learning with mixed hard/soft pointwise constraints considered in *Gnecco et al., IEEE TNNLS, vol. 26, pp. 2019-2032, 2015*. We show that the optimal solution to the learning problem with hard bilateral and linear pointwise constraints stated therein can be obtained as the limit of the sequence of optimal solutions to the related learning problems with soft bilateral and linear pointwise constraints, when the penalty parameter tends to infinity.

1 Introduction

In this work, we investigate constrained machine-learning problems in an environment based on both hard constraints (i.e, constraints whose violation is not allowed) and soft constraints (i.e., constraints whose violation is admissible, at the cost of some penalization). From a computational point of view, dealing with hard constraints is usually more difficult than dealing with soft constraints. So, often, hard-constrained optimization problems are solved by replacing them with sequences of soft-constrained optimization problems, whose penalty terms increase when increasing the index in each sequence. The motivation behind this technique is that, under suitable conditions, the optimal solutions of these soft-constrained optimization problems tend to those of the original hard-constrained optimization problems [1, Section 10.11]. Likewise in [2], we focus here on the case of pointwise constraints (i.e., constraints that are associated with a finite set of examples, in which each element of the set defines one such constraint). This is motivated by the fact that pointwise constraints are very often used in machine learning problems, since they are able to model very general learning conditions. As an extension of that work, we show that, under the framework considered therein, the optimal solution to the learning problem with hard bilateral and linear pointwise constraints can be obtained as the limit of the sequence of optimal solutions to the related learning problems with soft bilateral and linear pointwise constraints, when the penalty parameter tends to infinity.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

The paper is organized as follows. In Section 2, we detail the problem of learning from examples with bilateral pointwise constraints, in the general case in which soft and hard constraints are both present in the problem formulation. Section 3 compares the optimal solutions to the two problems obtained in the particular cases in which there are only hard linear constraints, or only soft linear constraints, and studies the limit behavior of the optimal solution to the soft constrained learning problem, when the penalty parameter tends to infinity. Finally, Section 4 is a short discussion.

2 Learning with mixed hard and soft pointwise constraints

We model an intelligent agent operating on a subset $\mathcal{X}$ of the feature space $\mathbb{R}^d$ as one implementing a vector-valued function $f := [f_1, \ldots, f_n]' \in \mathcal{F}$, where $\mathcal{F}$ denotes a suitable function space from $\mathcal{X}$ to $\mathbb{R}^n$. We assume the availability of prior knowledge in the form of constraints on f, which are expressed as follows:

$$\forall x^{(h)} \in \mathcal{X}_H \subseteq \mathcal{X} : \phi_i(x^{(h)}, f(x^{(h)})) = 0, i = 1, \ldots, n\,. \tag{1}$$

Here, it is assumed that the set $\mathcal{X}_H := \{x^{(1)}, x^{(2)}, \ldots, x^{(|\mathcal{X}_H|)}\}$ has a finite number of elements, the functions ϕ_i are scalar-valued. Finally, n denotes the number of constraints (for simplicity, the same as the number of components of f), which have the form (1). We use the expression *hard bilateral pointwise constraints* to denote constraints of the form (1). The term "hard" depends on the fact that any such constraint cannot be violated. In the following, we denote by $\mathcal{C}$ any collection of constraints of the form (1). Instead, we use the expression *soft constraints* to denote constraints whose violation is tolerated, at the cost of some penalization. A typical case of soft constraints arising in machine learning is the one associated with a finite labeled set, which models the classical framework of learning from supervised examples. These can be expressed in terms of $V(f(\tilde{x}^{(\kappa)}), \tilde{y}^{(\kappa)})$, where $V : \mathbb{R}^n \times \mathbb{R}^n \mapsto [0, +\infty)$ is a differentiable *loss function* (i.e., a non-negative function satisfying $V(z, z) = 0$, $\forall z \in \mathbb{R}^n$), $\tilde{x}^{(\kappa)}$ belongs to a finite subset $\mathcal{X}_S := \{\tilde{x}^{(1)}, \tilde{x}^{(2)}, \ldots, \tilde{x}^{(|\mathcal{X}_S|)}\} \subseteq \mathcal{X}$, and $\tilde{y}^{(\kappa)} \in \mathbb{R}^n$ denotes its label.

In the following, we assume $\mathcal{X} = \mathbb{R}^d$. For what concerns the choice of the function space $\mathcal{F}$, we consider the case in which, for any $j \in \mathbb{N}_n := \{1, \ldots, n\}$ and a positive integer k, the j-th component $f_j : \mathcal{X} \to \mathbb{R}$ of f is an element of the Sobolev space $\mathcal{W}^{k,2}(\mathcal{X})$, which is the subset of $\mathcal{L}^2(\mathcal{X})$ whose elements have weak partial derivatives up to the order k with finite $\mathcal{L}^2(\mathcal{X})$-norms. Concluding, we set

$$\mathcal{F} := \underbrace{\mathcal{W}^{k,2}(\mathcal{X}) \times \ldots \times \mathcal{W}^{k,2}(\mathcal{X})}_{n \text{ times}}\,. \tag{2}$$

Moreover, we assume $k > \frac{d}{2}$, because, under this condition, every element of $\mathcal{W}^{k,2}(\mathcal{X})$ admits a continuous representative, on which the constraints (1) can be evaluated unambiguously, and $\mathcal{F}$ is a Reproducing Kernel Hilbert Space. These are consequences of the Sobolev Embedding Theorem [3, Chapter 4].

We introduce a seminorm $\| f \|_{P,\gamma}$ on $\mathcal{F}$, through the pair (P, γ), where $P := [P_0, \ldots, P_{l-1}]'$ is a (vector-valued) finite-order differential operator of order

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

k with l components, and $\gamma \in \mathbb{R}^n$ is a fixed vector with positive entries. Let $\| f_j \|_P^2 := < Pf_j, Pf_j > := \sum_{r=0}^{l-1} \int_{\mathcal{X}} (P_r f_j(x) P_r f_j(x)) dx$, $\| f \|_{P,\gamma}^2 := \sum_{j=1}^n \gamma_j < Pf_j, Pf_j >$, and $\mu \geq 0$ a fixed non-negative constant. Given a subset $\mathcal{X}_S \subset \mathcal{X}$ with finite cardinality $|\mathcal{X}_S|$ and a set of labeled examples $(f(\tilde{x}^{(\kappa)}), \tilde{y}^{(\kappa)})$, $\kappa = 1, \ldots, |\mathcal{X}_S|$, we denote by

$$\mathcal{E}_s(f) \quad := \quad \frac{1}{2} \| f \|_{P,\gamma}^2 + \frac{\mu}{|\mathcal{X}_S|} \sum_{\kappa=1}^{|\mathcal{X}_S|} V(f(\tilde{x}^{(\kappa)}), \tilde{y}^{(\kappa)}) \tag{3}$$

the objective functional to be optimized (which includes the soft pointwise constraints), and by $\mathcal{F}_\mathcal{C} \subseteq \mathcal{F}$ the subset of functions belonging to the function space $\mathcal{F}$ (see (2)) that are also compatible with a given collection $\mathcal{C}$ of hard pointwise constraints having the expressions (1). We consider the following problem.

Problem LMPC (Learning with Mixed Pointwise Constraints). *The optimization problem of determining a constrained (either local or global) minimizer f^o of $\mathcal{E}_s$ over $\mathcal{F}_\mathcal{C}$ is referred to as learning from the soft pointwise constraints associated with $\mathcal{E}_s$ and the hard pointwise constraint collection $\mathcal{C}$.*

In the following, we focus on the case in which the operator P is invariant under spatial translation, and has constant coefficients. For a function u and a multi-index α with d non-negative entries α_j, we write $D^\alpha u$ to denote $\frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d}} u$, where $|\alpha| := \sum_{j=1}^d \alpha_j$. Then, we assume for the generic component P_i of the operator P the form $P_i = \sum_{|\alpha| \leq k} b_{i,\alpha} D^\alpha$,where the $b_{i,\alpha}$'s are suitable real constants. We also define the formal adjoint of P as the operator $P^\star := [P_0^\star, \ldots, P_{l-1}^\star]'$ whose i-th component $P_i^\star$ has the form $P_i^\star := \sum_{|\alpha| \leq k} (-1)^{|\alpha|} b_{i,\alpha} D^\alpha$. For simplicity, in the following we assume for P the form $Pf := [Pf_1, Pf_2, \ldots, Pf_n]'$, i.e., we use an overloaded notation. Finally, we define the operators $L := (P^\star)' P$ and, using again an overloaded notation, $\gamma L := [\gamma_1 L, \ldots, \gamma_n L]'$.

The next result is a simplified statement of [2, Theorem 4.1], and gives a representation for an optimal solution f^o to Problem LMPC. We use the following terminology. For two vector-valued functions $u^{(1)}$ and $u^{(2)}$ with the same number of components, we denote by $u^{(1)} * u^{(2)}$ the vector-valued function v whose first component is the convolution of the corresponding components of $u^{(1)}$ and $u^{(2)}$, the second component is the convolution of the corresponding components of $u^{(1)}$ and $u^{(2)}$, and so on, i.e., $v_j := (u^{(1)} * u^{(2)})_j := u_j^{(1)} * u_j^{(2)}$ for every j. A free-space *Green's function* associated with a linear differential operator O is a solution g to the distributional differential equation $Og = \delta$, where δ is the Dirac delta centered at the origin. Finally, we let $\gamma^{-1} g := [\gamma_1^{-1} g, \ldots, \gamma_n^{-1} g]'$.

Theorem 2.1. (Representer theorem for Problem LMPC). *Let us consider an intelligent agent minimizing the functional (3) on $\mathcal{F}$ and satisfying a given set of hard bilateral pointwise constraints expressed by*

$$\forall x^{(h)} \in \mathcal{X}_H, \quad \phi_i(x^{(h)}, f(x^{(h)})) = 0 \, , i = 1, \ldots, n \, ,$$

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

in which $\phi_i \in \mathcal{C}^1(\mathcal{X} \times \mathbb{R}^n)$. *Let* f^o *be any constrained local minimizer of the functional (3). Assume that for any* $x^{(h)}$ *belonging to the finite set* $\mathcal{X}_H$, *the Jacobian matrix* $\frac{\partial(\phi_1,\ldots,\phi_m)}{\partial(f_1^o,\ldots,f_m^o)}$, *evaluated in* $x^{(h)}$, *is invertible, Let also* $\mathcal{X}_H \cap \mathcal{X}_S = \emptyset$, L *be invertible on* $\mathcal{W}^{k,2}(\mathcal{X})$, *and assume the existence of a free-space Green's function* g *of* L *that belongs to* $\mathcal{W}^{k,2}(\mathcal{X})$. *Then, there exist* $|\mathcal{X}_H|$ *vectors* $\lambda^{(h)} \in \mathbb{R}^n$, *each one associated with a point* $x^{(h)} \in \mathcal{X}_H$, *such that* f^o *satisfies*

$$f^o(\cdot) = \gamma^{-1} g(\cdot) * \left(\sum_{i=1}^{n} \omega_i^H(\cdot) + \sum_{\kappa=1}^{|\mathcal{X}_S|} \omega_\kappa^S(\cdot) \right) , \tag{4}$$

in which

$$\omega_i^H(\cdot) := - \sum_{h=1}^{|\mathcal{X}_H|} \lambda_i^{(h)} \delta(\cdot - x^{(h)}) \nabla_f \phi_i(x^{(h)}, f^o(x^{(h)})) ,$$

$$\omega_\kappa^S(\cdot) := - \frac{\mu}{|\mathcal{X}_S|} \delta(\cdot - \tilde{x}^{(\kappa)}) \nabla_f V(f^o(\tilde{x}^{(\kappa)}), \tilde{y}^{(\kappa)}) , \tag{5}$$

$\lambda_i^{(h)}$ *denotes the* i*-th component of* $\lambda^{(h)}$, *and* $\nabla_f \phi_i$ *denotes the gradient w.r.t. the second vector argument* f *of the function* ϕ_i.

The distributions ω_i^H and ω_κ^S are named, respectively, the *reaction of the* i*-th hard pointwise constraint*, and the *reaction of the* κ*-th soft pointwise constraint.*

3 Hard constraints as limit cases of soft constraints

As a case study, we consider Problem LMPC under the assumption of hard bilateral and linear pointwise constraints expressed by

$$\forall x^{(h)} \in \mathcal{X}_H, \quad \phi_i(x^{(h)}, f(x^{(h)})) = f(x^{(h)}) - y_i^{(h)} = 0 , \quad i = 1, \ldots, n . \tag{6}$$

One can notice that, in this case, the nonsingularity of the Jacobian matrix, which is required in Theorem 2.1, holds true, since it takes the form of the identity matrix. Moreover, beside the hard bilateral and linear pointwise constraints (6), other soft pointwise constraints expressed in terms of the quadratic loss

$$V(f(\tilde{x}^{(\kappa)}), \tilde{y}^{(\kappa)}) = \frac{1}{2} \sum_{j=1}^{n} \left(f_j(\tilde{x}^{(\kappa)}) - \tilde{y}_j^{(\kappa)} \right)^2 \tag{7}$$

may or may not be present in the problem formulation. Such constraints can be interpreted as soft versions of the hard bilateral and linear pointwise constraints (6). Finally, to simplify the notation, we set $\gamma_j = \bar{\gamma} > 0$, for every $j = 1, \ldots, n$.

In the following, under the assumptions of Theorem 2.1, we investigate the optimal solution to Problem LMPC with hard and soft constraints of the forms (6) and (7), respectively, in the two degenerate cases (which were only briefly mentioned in [2]) in which *(i)* there are only hard constraints; *(ii)* there are only

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

soft constraints. Then, we compare the two cases.

Only soft constraints. For the case *(i)*, the expression (4) reduces to:

$$f_j^o(\cdot) \quad = \quad -\bar{\gamma}^{-1} \sum_{\kappa=1}^{|\mathcal{X}_S|} \frac{\mu}{|\mathcal{X}_S|} g(\cdot - \tilde{x}^{(\kappa)}) \left(f_j^o(\tilde{x}^{(\kappa)}) - \tilde{y}_j^{(\kappa)} \right) . \tag{8}$$

Moreover, the evaluation of (8) on the set $\mathcal{X}_S$ makes it possible to reduce the problem of finding the unknown coefficients $f_j^o(\tilde{x}^{(\kappa)})$ to the one of solving the linear system $\left(\bar{\gamma}^{-1} \frac{\mu}{|\mathcal{X}_S|} G_{SS} + I_{|\mathcal{X}_S|} \right) \tilde{F}_S^{(j)} = \bar{\gamma}^{-1} \frac{\mu}{|\mathcal{X}_S|} G_{SS} \tilde{Y}_S^{(j)}$, where we have defined the following matrices:

$$\tilde{F}_S := (\tilde{F}_S^{(1)} | \tilde{F}_S^{(2)} | \cdots | \tilde{F}_S^{(n)}) := \begin{pmatrix} f_1^o(\tilde{x}^{(1)}) & f_2^o(\tilde{x}^{(1)}) & \cdots & f_n^o(\tilde{x}^{(1)}) \\ f_1^o(\tilde{x}^{(2)}) & f_2^o(\tilde{x}^{(2)}) & \cdots & f_n^o(\tilde{x}^{(2)}) \\ \cdots & \cdots & \cdots & \cdots \\ f_1^o(\tilde{x}^{(\mathbf{J}\mathcal{X}_S\mathbf{J})}) & f_2^o(\tilde{x}^{(\mathbf{J}\mathcal{X}_S\mathbf{J})}) & \cdots & f_n^o(\tilde{x}^{(\mathbf{J}\mathcal{X}_S\mathbf{J})}) \end{pmatrix} ,$$

$$\tilde{Y}_S := (\tilde{Y}_S^{(1)} | \tilde{Y}_S^{(2)} | \cdots | \tilde{Y}_S^{(n)}) := \begin{pmatrix} \tilde{y}_1^{(1)} & \tilde{y}_2^{(1)} & \cdots & \tilde{y}_n^{(1)} \\ \tilde{y}_1^{(2)} & \tilde{y}_2^{(2)} & \cdots & \tilde{y}_n^{(2)} \\ \cdots & \cdots & \cdots & \cdots \\ \tilde{y}_1^{(\mathbf{J}\mathcal{X}_S\mathbf{J})} & \tilde{y}_2^{(\mathbf{J}\mathcal{X}_S\mathbf{J})} & \cdots & \tilde{y}_n^{(\mathbf{J}\mathcal{X}_S\mathbf{J})} \end{pmatrix} ,$$

$I_{|\mathcal{X}_S|}$ denotes the $|\mathcal{X}_S| \times |\mathcal{X}_S|$ identity matrix, and $G_{SS} \in \mathbb{R}^{|\mathcal{X}_S| \times |\mathcal{X}_H|}$ is defined in terms of its elements as $G_{SS,r,s} := g(\tilde{x}^{(r)} - \tilde{x}^{(s)})$. Hence, $\tilde{F}_S^{(j)} = \left(\bar{\gamma}^{-1} \frac{\mu}{|\mathcal{X}_S|} G_{SS} + I_{|\mathcal{X}_S|} \right)^{-1} \bar{\gamma}^{-1} \frac{\mu}{|\mathcal{X}_S|} G_{SS} \tilde{Y}_S^{(j)}$. Moreover, when G_{SS} is invertible, one has (making explicit the dependence of $\tilde{F}_S^{(j)}$ on μ) $\lim_{\mu \to +\infty} \tilde{F}_S^{(j)}(\mu) = \tilde{Y}_S^{(j)}$.

Only hard constraints. For the case *(ii)*, the expression (4) reduces to:

$$f_j^o(\cdot) \quad = \quad -\bar{\gamma}^{-1} \sum_{h=1}^{|\mathcal{X}_H|} \lambda_j^{(h)} g(\cdot - x^{(h)}) . \tag{9}$$

Moreover, the evaluation of (9) on the set $\mathcal{X}_H$ allows one to find the unknown coefficients $\lambda_j^{(h)}$ by solving the linear system $F_H^{(j)} = Y_H^{(j)}$, $\bar{\gamma}^{-1} G_{HH} \Lambda^{(j)} = -F_H^{(j)}$, where we have defined the following matrices:

$$\begin{aligned} F_H &:= (F_H^{(1)} | F_H^{(2)} | \cdots | F_H^{(n)}) \\ &:= \begin{pmatrix} f_1^o(x^{(1)}) & f_2^o(x^{(1)}) & \cdots & f_n^o(x^{(1)}) \\ f_1^o(x^{(2)}) & f_2^o(x^{(2)}) & \cdots & f_n^o(x^{(2)}) \\ \cdots & \cdots & \cdots & \cdots \\ f_1^o(x^{(\mathbf{J}\mathcal{X}_H\mathbf{J})}) & f_2^o(x^{(\mathbf{J}\mathcal{X}_H\mathbf{J})}) & \cdots & f_n^o(x^{(\mathbf{J}\mathcal{X}_H\mathbf{J})}) \end{pmatrix} , \end{aligned}$$

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

$$
\begin{aligned}
Y_H :=& (Y_H^{(1)} | Y_H^{(2)} | \cdots | Y_H^{(n)}) \\
:=& \begin{pmatrix} y_1^{(1)} & y_2^{(1)} & \cdots & y_n^{(1)} \\ y_1^{(2)} & y_2^{(2)} & \cdots & y_n^{(2)} \\ \cdots & \cdots & \cdots & \cdots \\ y_1^{(\mathbf{J}\mathbf{X}_H\mathbf{J})} & y_2^{(\mathbf{J}\mathbf{X}_H\mathbf{J})} & \cdots & y_n^{(\mathbf{J}\mathbf{X}_H\mathbf{J})} \end{pmatrix}, \\
\Lambda :=& (\Lambda^{(1)} | \Lambda^{(2)} | \cdots | \Lambda^{(n)}) \\
:=& \begin{pmatrix} \lambda_1^{(1)} & \lambda_2^{(1)} & \cdots & \lambda_n^{(1)} \\ \lambda_1^{(2)} & \lambda_2^{(2)} & \cdots & \lambda_n^{(2)} \\ \cdots & \cdots & \cdots & \cdots \\ \lambda_1^{(\mathbf{J}\mathbf{X}_H\mathbf{J})} & \lambda_2^{(\mathbf{J}\mathbf{X}_H\mathbf{J})} & \cdots & \lambda_n^{(\mathbf{J}\mathbf{X}_H\mathbf{J})} \end{pmatrix},
\end{aligned}
$$

and $G_{HH} \in \mathbb{R}^{|\mathcal{X}_H| \times |\mathcal{X}_H|}$ (assumed to be invertible) is defined in terms of its elements as $G_{HH,r,s} := g(x^{(r)} - x^{(s)})$. Hence, one gets $\Lambda^{(j)} = -\bar{\gamma} G_{HH}^{-1} Y_H^{(j)}$.

Comparison of the two cases, and limit behavior. In the following, we assume the set $\mathcal{X}_S$ for the case *(i)* to be equal to the set $\mathcal{X}_H$ for the case *(ii)* (this is not in contrast with the assumptions of Theorem 2.1, since here we are considering two different instances of the problem), $\tilde{Y}_S = Y_H$, and $G_{SS} = G_{HH}$ invertible. Then, a comparison of the expressions (8) and (9) shows that, when μ tends to $+\infty$, the optimal solution (8) of the problem with only soft constraints tends to the optimal solution (9) with only hard constraints, provided that $\lim_{\mu\to\infty} \frac{\mu}{|\mathcal{X}_H|}(\tilde{F}_S^{(j)}(\mu) - \tilde{Y}_S^{(j)}) = \Lambda^{(j)}$. This is indeed the case, since the analysis above shows that

$$
\begin{aligned}
\lim_{\mu\to\infty} \frac{\mu}{|\mathcal{X}_H|}(\tilde{F}_S^{(j)}(\mu) - \tilde{Y}_S^{(j)}) &= \lim_{\mu\to\infty} -\bar{\gamma} G_{SS}^{-1} \tilde{F}_S^{(j)}(\mu) = -\bar{\gamma} G_{SS}^{-1} \tilde{Y}_S^{(j)} \\
&= -\bar{\gamma} G_{HH}^{-1} Y_H^{(j)} = \Lambda^{(j)} . \qquad (10)
\end{aligned}
$$

4 Discussion

We have shown that, in the learning framework of [2], the case of learning from hard constraints only can be interpreted as a limit case of learning from soft constraints only (see [2, Fig. 2 (a) and (c)] for simulation results in both cases). An extension is expected for the learning problem with mixed constraints [2, Fig. 2 (b)], when all the hard constraints are replaced by soft constraints.

References

[1] D. G. Luenberger. *Optimization by Vector Space Methods.* Wiley & Sons, 1969.

[2] G. Gnecco, M. Gori, S. Melacci, and M. Sanguineti. Learning with mixed hard/soft constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 26:2019–2032, 2015.

[3] R. A. Adams and J. F. Fournier. *Sobolev Spaces.* Academic Press, 2003.