Convex Representation Learning for Generalized Invariance in Semi-Inner-Product Space

Yingyi Ma [1] Vignesh Ganapathiraman [1] Yaoliang Yu [2] Xinhua Zhang [1]

Abstract

Invariance (defined in a general sense) has been one of the most effective priors for representation learning. Direct factorization of parametric models is feasible only for a small range of invariances, while regularization approaches, despite improved generality, lead to nonconvex optimization. In this work, we develop a *convex* representation learning algorithm for a variety of generalized invariances that can be modeled as semi-norms. Novel Euclidean embeddings are introduced for kernel representers in a semi-inner-product space, and approximation bounds are established. This allows invariant representations to be learned efficiently and effectively as confirmed in our experiments, along with accurate predictions.

1. Introduction

Effective modeling of structural priors has been the workhorse of a variety of machine learning algorithms. Such priors are available in a rich supply, including invariance (Simard et al., 1996; Ferraro and Caelli, 1994), equivariance (Cohen and Welling, 2016; Graham and Ravanbakhsh, 2019), disentanglement (Bengio et al., 2013; Higgins et al., 2017), homophily/heterophily (Eliassi-Rad and Faloutsos, 2012), fairness (Creager et al., 2019), correlations in multiple views and modalities (Wang et al., 2015; Kumar et al., 2018), etc.

In this paper we focus on "generalized invariance", where certain relationship holds irrespective of certain changes in data. This extends traditional settings that are limited to, *e.g.*, transformation and permutation. For instance, in multilabel classification there are semantic or logical relationships between classes which hold for any input. Common examples include mutual exclusion and implication (Mirzazadeh et al., 2015a; Deng et al., 2012). In mixup (Zhang et al., 2018), a convex interpolation of a pair of examples is postulated to yield the same interpolation of output labels.

While conventional wisdom learns models whose prediction accords with these structures, recent developments show that it can be more effective to learn structure-encoding representations. Towards this goal, the most straightforward approach is to directly parameterize the model. For example, deep sets model permutation invariance via an additive decomposition (Zaheer et al., 2017), convolutional networks use sparse connection and parameter sharing to model translational invariance, and a similar approach has been developed for equivariance (Ravanbakhsh et al., 2017). Although they simplify the model and can enforce invariance over the *entire* space, their applicability is very restricted, because most useful structures do not admit a readily decomposable parameterization. As a result, most invariance/equivariance models are restricted to permutations and group based diffeomorphism.

In order to achieve significantly improved generality and flexibility, the regularization approach can be leveraged, which penalizes the violation of pre-specified structures. For example, Rifai et al. (2011) penalizes the norm of the Jacobian matrix to enforce contractivity, conceivably a generalized type of invariance. Smola (2019) proposed using a max-margin loss over all transformations (Teo et al., 2007). However, for most structures, regularization leads to a nonconvex problem. Despite the recent progress in optimization for deep learning, the process still requires a lot of trial and error. Therefore a convex learning procedure will be desirable, because besides the convenience in optimization, it also offers the profound advantage of decoupling parameter optimization from problem specification: poor learning performance can only be ascribed to a poor model architecture, not to poor local minima.

Indeed convex invariant representation learning has been studied, but in limited settings. Tangent distance kernels (Haasdonk and Keysers, 2002) and Haar integration ker-

[1] University of Illinois at Chicago [2] University of Waterloo and Vector Institute. Correspondence to: Xinhua Zhang <zhangx@uic.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

nels are engineered to be invariant to a group of transformations (Raj et al., 2017; Mroueh et al., 2015; Haasdonk and Burkhardt, 2007), but it relies on sampling for tractable computation and the sample complexity is $O(d/\epsilon^2)$ where d is the dimension of the underlying space. Bhattacharyya et al. (2005) treated all perturbations within an ellipsoid neighborhood as invariances, and it led to an expensive second order cone program (SOCP). Other distributionally robust formulations also lead to SOCP/SDPs (Rahimian and Mehrotra, 2019). The most related work is Ma et al. (2019), which warped a reproducing kernel Hilbert space (RKHS) by linear functionals that encode the invariances. However, in order to keep the warped space an RKHS, their applicability is restricted to *quadratic* losses on linear functionals.

In practice, however, there are many invariances that cannot be modeled by quadratic penalties. For example, the logical relationships between classes impose an ordering in the discriminative output (Mirzazadeh et al., 2015a), and this can hardly be captured by quadratic forms. Similarly, when a large or infinite number of invariances are available, measuring the maximum violation makes more sense than their sum, and it is indeed the principle underlying adversarial learning (Madry et al., 2018). Again this is not amenable to quadratic forms.

Our goal, therefore, is to develop a *convex* representation learning approach that efficiently incorporates generalized invariances as semi-norm functionals. Our first contribution is to show that compared with linear functionals, semi-norm functionals encompass a much broader range of invariance (Sections 5 and 6).

Our key tool is the semi-inner-product space (s.i.p., Lumer, 1961), into which an RKHS can be warped by augmenting the RKHS norm with semi-norm functionals. A specific example of s.i.p. space is the reproducing kernel Banach space (Zhang et al., 2009), which has been used for ℓ_p regularization in, *e.g.*, kernel SVMs, and suffers from high computational cost (Salzo et al., 2018; Der and Lee, 2007; Bennett and Bredensteiner, 2000; Hein et al., 2005; von Luxburg and Bousquet, 2004; Zhou et al., 2002). A s.i.p. space extends RKHS by relaxing the underlying inner product into a semi-inner-product, while retaining the important construct: *kernel function*. To our best knowledge, s.i.p. space has yet been applied to representation learning.

Secondly, we developed efficient computation algorithms for solving the regularized risk minimization (RRM) with the new s.i.p. norm (Section 3). Although Zhang et al. (2009) established the representer theorem from a pure mathematical perspective, no practical algorithm was provided and ours is the first to fill this gap.

However, even with this progress, RRMs still do not provide invariant representations of data instances; it simply learns a discriminant function by leveraging the representer theorem (which does hold in the applications we consider). So our third contribution, as presented in Section 4, is to learn and extract representations by embedding s.i.p. kernel representers in Euclidean spaces. This is accomplished in a *convex* and efficient fashion, constituting a secondary advantage over RRMs which is not convex in the dual coefficients. Different from Nyström or Fourier linearization of kernels in RKHS, the kernel representers in a s.i.p. space carry interestingly different meanings and expressions in primal and dual spaces. Finally, our experiments demonstrate that the new s.i.p.-based algorithm learns more predictive representations than strong baselines.

2. Preliminaries

Suppose we have an RKHS $\mathcal{H} = (\mathcal{F}, \langle\cdot,\cdot\rangle_{\mathcal{H}}, k)$ with $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}}$, inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ and kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Our goal is to renorm $\mathcal{H}$ hence *warp the distance metric* by adding a functional R that induces desired structures.

2.1. Existing works on invariance modeling by RKHS

Smola and Schölkopf (1998) and Zhang et al. (2013) proposed modeling invariances by bounded linear functionals in RKHS. Given a function f, the graph Laplacian is $\sum_{ij} w_{ij}(f(x_i) - f(x_j))^2$, and obviously $f(x_i) - f(x_j)$ is bounded and linear. Transformation invariance can be characterized by $\frac{\partial}{\partial\alpha}|_{\alpha=0} f(I(\alpha))$, where $I(\alpha)$ stands for the image after applying an α amount of rotation, translation, etc. It is again bounded and linear. By Riesz representation theorem, a bounded linear functional can be written as $\langle z_i, f\rangle_{\mathcal{H}}$ for some $z_i \in \mathcal{H}$.

Based on this view, Ma et al. (2019) took a step towards representation learning. By adding $R(f)^2 := \sum_i \langle z_i, f\rangle^2_{\mathcal{H}}$ to the RKHS norm square, the space is warped to favor f that respects invariance, *i.e.*, small magnitude of $\langle z_i, f\rangle$. They showed that it leads to a new RKHS with a kernel

$$k^\circ(x_1, x_2) = k(x_1, x_2) - z(x_1)^\top (I + K_z)^{-1} z(x_2), \quad (1)$$

where $z(x) = (z_1(x), \ldots, z_m(x))^\top$ and $K_z = (\langle z_i, z_j\rangle)_{i,j}$.

Although the kernel representer of k° offers a new invariance aware representation, the requirement that the resulting space remains an RKHS forces the penalties in R to be quadratic on $\langle z_i, f\rangle$, significantly limiting its applicability to a broader range of invariances such as total variation $\int_x |f'(x)| \mathrm{d}x$. Our goal is to relax this restriction by enabling *semi-norm* regularizers with new tools in functional analysis, and illustrate its applications in Sections 5 and 6.

2.2. Semi-inner-product spaces

We first specify the range of regularizer R considered here.

Assumption 1. *We assume that $R : \mathcal{F} \to \mathbb{R}$ is a semi-norm. Equivalently, $R : \mathcal{F} \to \mathbb{R}$ is convex and $R(\alpha f) = |\alpha|\, R(f)$ for all $f \in \mathcal{F}$ and $\alpha \in \mathbb{R}$ (absolute homogeneity). Furthermore, we assume R is closed (i.e., lower semicontinuous)* w.r.t. *the topology in $\mathcal{H}$.*

Since R is closed convex and its domain is the entire Hilbert space $\mathcal{H}$, R must be continuous. By exempting R from being induced by an inner product, we enjoy substantially improved flexibility in modeling various regularities.

For most learning tasks addressed below, it will be convenient to directly construct R from the specific regularity. However, in some context it will also be convenient to constructively explicate R in terms of support functions.

Proposition 1. *$R(f)$ satisfies Assumption 1 if, and only if, $R(f) = \sup_{g \in S} \langle f, g\rangle_{\mathcal{H}}$, where $S \subseteq \mathcal{H}$ is bounded in the RKHS norm and is symmetric ($g \in S \Leftrightarrow -g \in S$).*

The proof is in Appendix **??**. Using R, we arrive at a new norm defined by

$$\|f\|_{\mathcal{B}} := \sqrt{\|f\|_{\mathcal{H}}^2 + R(f)^2}, \tag{2}$$

thanks to Assumption 1. It is immediately clear from Proposition 1 that $\|f\|_{\mathcal{H}} \leq \|f\|_{\mathcal{B}} \leq C\,\|f\|_{\mathcal{H}}$, for some constant $C > 0$ that bounds the norm of S. In other words, the two norms $\|\cdot\|_{\mathcal{H}}$ and $\|\cdot\|_{\mathcal{B}}$ are equivalent, hence in particular the norm $\|\cdot\|_{\mathcal{B}}$ is complete. We thus arrive at a Banach space $\mathcal{B} = (\mathcal{F}, \|\cdot\|_{\mathcal{B}})$. Note that both $\mathcal{H}$ and $\mathcal{B}$ have the same underlying vector space $\mathcal{F}$—the difference is in the norm or distance metric. To proceed, we need to endow more structures on $\mathcal{B}$.

Definition 1 (Strict convexity)**.** *A normed vector space $(\mathcal{F}, \|\cdot\|)$ is strictly convex if for all $\mathbf{0} \neq f, g \in \mathcal{F}$,*

$$\|f + g\| = \|f\| + \|g\| \tag{3}$$

implies $g = \alpha f$ for some $\alpha \geq 0$. Equivalently, if the unit ball $\mathsf{B} := \{f \in \mathcal{F} : \|f\| \leq 1\}$ is strictly convex.

Using the parallelogram law it is clear that the Hilbert norm $\|\cdot\|_{\mathcal{H}}$ is strictly convex. Moreover, since summation preserves strict convexity, it follows that the new norm $\|\cdot\|_{\mathcal{B}}$ is strictly convex as well.

Definition 2 (Gâteaux differentiability)**.** *A normed vector space $(\mathcal{F}, \|\cdot\|)$ is Gâteaux differentiable if for all $\mathbf{0} \neq f, g \in \mathcal{F}$, there exists the directional derivative*

$$\lim_{t \in \mathbb{R}, t \to 0} \tfrac{1}{t}(\|f + tg\| - \|f\|). \tag{4}$$

We remark that both strict convexity and Gâteaux differentiability are algebraic but not topological properties of the norm. In other words, two equivalent (in terms of topology) norms may not be strictly convex or Gâteaux differentiable at the same time. For instance, the ℓ_2-norm on $\mathbb{R}^d$ is both strictly convex and Gâteaux differentiable, while the equivalent ℓ_1-norm is not.

Recall that $\mathcal{B}^*$ is the dual space of $\mathcal{B}$, consisting of all continuous linear functionals on $\mathcal{B}$ and equipped with the dual norm $\|F\|_{\mathcal{B}^*} = \sup_{\|f\|_{\mathcal{B}} \leq 1} F(f)$. The dual space of a normed (reflexive) space is Banach (reflexive).

Definition 3. *A Banach space $\mathcal{B}$ is reflexive if the canonical map $\jmath : \mathcal{B} \to \mathcal{B}^{**}$, $f \mapsto \langle \cdot; \jmath f\rangle := \langle f; \cdot\rangle$ is onto, where $\langle f; F\rangle$ is the (bilinear) duality pairing between dual spaces. Here $\cdot$ is any element in $\mathcal{B}^*$.*

Note that reflexivity is a topological property. In particular, equivalent norms are all reflexive if any one of them is. As any Hilbert space $\mathcal{H}$ is reflexive, so is the equivalent norm $\|\cdot\|_{\mathcal{B}}$ in (2).

Theorem 1 (Borwein and Vanderwerff 2010, p. 212-213)**.** *A Banach space $\mathcal{B}$ is strictly convex (Gâteaux differentiable) if its dual space $\mathcal{B}^*$ is Gâteaux differentiable (strictly convex). The converse is true too if $\mathcal{B}$ is reflexive.*

Combining Proposition 1 and Theorem 1, we see that R, hence $\|\cdot\|_{\mathcal{B}}$, is Gâteaux differentiable if (the closed convex hull of) the set S in Proposition 1 is strictly convex.

We are now ready to define a semi-inner-product (s.i.p.) on a normed space $(\mathcal{F}, \|\cdot\|)$. We call a bivariate mapping $[\cdot, \cdot] : \mathcal{F} \times \mathcal{F} \to \mathbb{R}$ a s.i.p. if for all $f, g, h \in \mathcal{F}$ and $\lambda \in \mathbb{R}$,

- additivity: $[f + g, h] = [f, h] + [g, h]$
- homogeneity: $[\lambda f, g] = [f, \lambda g] = \lambda [f, g]$,
- norm-inducing: $[f, f] = \|f\|^2$,
- Cauchy-Schwarz: $[f, g] \leq \|f\| \cdot \|g\|$.

We note that an s.i.p. is additive in its second argument iff it is an inner product (by simply verifying the parellelogram law). Lumer (1961) proved that s.i.p. does exist on every normed space. Indeed, let the subdifferential $J = \partial \frac{1}{2} \|\cdot\|_{\mathcal{B}}^2 : \mathcal{B} \rightrightarrows \mathcal{B}^*$ be the (multi-valued) duality mapping. Then, any selection $j : \mathcal{B} \to \mathcal{B}^*, f \mapsto j(f) \in J(f)$ with the convention that $j(\mathbf{0}) = \mathbf{0}$ leads to a s.i.p.:

$$[f, g] := \langle f; j(g)\rangle\,. \tag{5}$$

Indeed, from definition, for any $f \neq \mathbf{0}$, $j(f) = \|f\| F$, where $\|F\|^* = 1$ and $\langle f; F\rangle = \|f\|$. A celebrated result due to Giles (1967) revealed the uniqueness of s.i.p. if the norm $\|\cdot\|$ is Gâteaux differentiable, and later Faulkner (1977) proved that the (unique) mapping j is onto iff $\mathcal{B}$ is reflexive. Moreover, j is 1-1 if $\mathcal{B}$ is strictly convex (like in (2)), as was shown originally in Giles (1967).

Let us summarize the above results.

Definition 4. *A Banach space $\mathcal{B}$ is called a s.i.p. space iff it is reflexive, strictly convex, and Gâteaux differentiable. Clearly, the dual $\mathcal{B}^*$ of a s.i.p. space is s.i.p. too.*

Theorem 2 (Riesz representation)**.** *Let $\mathcal{B}$ be a s.i.p. space. Then, for any continuous linear functional $f^* \in \mathcal{B}^*$, there exists a unique $f \in \mathcal{B}$ such that*

$$f^* = [\cdot, f] = j(f), \quad \textit{and} \quad \|f\| = \|f^*\|_{\mathcal{B}^*}. \tag{6}$$

From now on, we identify the duality mapping j with the star operator $f^* := j(f)$. Thus, we have a unique way to represent all continuous functionals on a s.i.p. space. Conveniently, the unique s.i.p. on the dual space follows from (5): for all $f^*, g^* \in \mathcal{B}^*$,

$$[f^*, g^*] := [g, f] = \langle g; f^* \rangle \,, \tag{7}$$

from which one easily verifies all properties of an s.i.p. Some literature writes $[f^*, g^*]_{\mathcal{B}^*}$, $[g, f]_{\mathcal{B}}$, $\langle g; f^* \rangle_{\mathcal{B}^*}$, and $\langle f; g^* \rangle_{\mathcal{B}}$ to explicitize where the operations take place. We simplify these notations by omitting subscripts when the context is clear, but still write $\|f\|_{\mathcal{B}}$ and $\|f^*\|_{\mathcal{B}^*}$.

Finally, fix $x \in \mathcal{X}$ and consider the evaluation (linear) functional $\mathsf{ev}_x : \mathcal{B} \to \mathbb{R}$, $f \mapsto f(x)$. When ev_x is continuous (which indeed holds for our norm (2)), Theorem 2 implies the existence of a unique $G_x \in \mathcal{B}$ such that

$$f(x) = \mathsf{ev}_x(f) = [f, G_x] = [G_x^*, f^*]. \tag{8}$$

Varying $x \in \mathcal{X}$ we obtain a unique *s.i.p. kernel* $G : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that $G_x := G(\cdot, x) \in \mathcal{B}$. Thus, using s.i.p. we obtain the reproducing property:

$$f(x) = [f, G(\cdot, x)], \; G(x, y) = [G(\cdot, y), G(\cdot, x)]. \tag{9}$$

Different from a reproducing kernel in RKHS, G is not necessarily symmetric or positive semi-definite.

3. Regularized Risk Minimization

In this section we aim to provide a computational device for the following regularized risk minimization (RRM) problem:

$$\min_{f \in \mathcal{H}} \;\; \ell(f) + \|f\|_{\mathcal{H}}^2 + R(f)^2. \tag{10}$$

where $\ell(f)$ is the empirical risk depending on discriminant function values $\{f(x_j)\}_{j=1}^n$ for training examples $\{x_j\}$. Clearly, this objective is equivalent to

$$\min_{f \in \mathcal{B}} \;\; \ell(f) + \|f\|_{\mathcal{B}}^2 \,. \tag{11}$$

Remark 1. *Unlike the usual treatment in reproducing kernel Banach spaces (RKBS) (*e.g. *Zhang et al., 2009), we only require $\mathcal{B}$ to be reflexive, strictly convex and Gâteaux differentiable, instead of the much more demanding uniform convexity and smoothness. This more general condition not only suffices for our subsequent results but also simplifies the presentation. A similar definition like ours was termed pre-RKBS in Combettes et al. (2018).*

Zhang et al. (2009, Theorem 2) established the representer theorem for RKBS: the optimal f for (11) has its dual form

$$f^* = \sum\nolimits_j c_j G_{x_j}^*, \tag{12}$$

where $\{c_j\}$ are real coefficients. To optimize $\{c_j\}$, we need to substitute (12) into (11), which in turn requires evaluating i) $\|f\|_{\mathcal{B}}^2$, which equals $\|f^*\|_{\mathcal{B}^*}^2$; ii) $f(x)$, which, can be computed through inverting the star operator as follows:

$$\begin{aligned}
\|f^*\|_{\mathcal{B}^*} &= \max_{\|h\|_{\mathcal{B}} \le 1} \langle h; f^* \rangle \\
&= \max_{\|h\|_{\mathcal{B}} \le 1} \sum\nolimits_j c_j \left\langle h; G_{x_j}^* \right\rangle \\
&= \max_{h : \|h\|_{\mathcal{H}}^2 + R(h)^2 \le 1} \sum\nolimits_j c_j h(x_j),
\end{aligned}$$

where the last equality is due to (8) and (5). The last maximization step operates in the RKHS $\mathcal{H}$, and thanks to the strict convexity of $\| \cdot \|_{\mathcal{B}}$, it admits the unique solution

$$h = f / \|f\|_{\mathcal{B}} = f / \|f^*\|_{\mathcal{B}^*}, \tag{13}$$

because $\langle f; f^* \rangle = \|f\|_{\mathcal{B}} \|f^*\|_{\mathcal{B}^*}$, and $\mathcal{B}$ is a s.i.p. space.

We summarize this computational inverse below:

Theorem 3. *If $f^* = \sum_j c_j G_{x_j}^*$, then $f = \|f\|_{\mathcal{B}} \, f^\circ$, where*

$$f^\circ := \arg \max_{h : \|h\|_{\mathcal{H}}^2 + R(h)^2 \le 1} \sum_j c_j h(x_j), \tag{14}$$

$$\|f\|_{\mathcal{B}} = \sum\nolimits_j c_j f^\circ(x_j) = \left\langle f^\circ, \sum\nolimits_j c_j k(x_j, \cdot) \right\rangle_{\mathcal{H}}. \tag{15}$$

In addition, the argmax in (14) is attained uniquely.

In practice, we first compute f° by solving (14), and then f can be evaluated at different x without redoing any optimization. As a special case, setting $f^* = G_x^*$ allows us to evaluate the kernel $G_x = G_x^\circ(x) G_x^\circ$.

Specialization to RKHS. When $R(f)^2 = \sum_i \langle z_i, f \rangle_{\mathcal{H}}^2$, $\|\cdot\|_{\mathcal{B}}$ is induced by an inner product, making $\mathcal{B}$ an RKHS. Now we can easily recover (1) by applying Theorem 3, because the optimization in (14) with $f^* = G_x^*$ is

$$\max_{h \in \mathcal{H}} \; h(x), \quad \text{s.t.} \quad \|h\|_{\mathcal{H}}^2 + \sum\nolimits_k \langle z_k, h \rangle_{\mathcal{H}}^2 \le 1, \tag{16}$$

and its unique solution can be easily found in closed form:

$$G_x^\circ = \frac{k(\cdot, x) - (z_1, \ldots, z_m)(I + K_z)^{-1} z(x)}{(k(x, x) - z(x)^\top (I + K_z)^{-1} z(x))^{1/2}}. \tag{17}$$

Plugging into $G_x = G_x^\circ(x) G_x^\circ$, we recover (1).

Overall, the optimization of (11) may no longer be convex in $\{c_j\}$, because $f(x)$ is generally not linear in $\{c_j\}$ even though f^* is (since the star operator is not linear). In practice, we can initialize $\{c_j\}$ by training without $R(f)$ (*i.e.*, setting $R(f)$ to 0). Despite the nonconvexity, we have achieved a new solution technique for a broad class of inverse problems, where the regularizer is a semi-norm.

4. Convex Representation Learning by Euclidean Embedding

Interestingly, our framework—which so far only learns a predictive model—can be directly extended to learn structured *representations* in a *convex* fashion. In representation learning, one identifies an "object" for each example x, which, in our case, can be a function in $\mathcal{F}$ or a vector in Euclidean space. Such a representation is supposed to have incorporated the prior invariances in R, and can be directly used for other (new) tasks such as supervised learning without further regularizing by R. This is different from the RRM in Section 3, which, although still enjoys the representer theorem in the applications we consider, only seeks a discriminant function f without providing a new representation for each example.

Our approach to convex representation learning is based on Euclidean embeddings (a.k.a. finite approximation or linearization) of the kernel representers in a s.i.p. space, which is analogous to the use of RKHS in extracting useful features. However, different from RKHS, G_x and G_x^* play different roles in a s.i.p. space, hence require *different* embeddings in $\mathbb{R}^d$. For any $f \in \mathcal{B}$ and $g^* \in \mathcal{B}^*$, we will seek their Euclidean embeddings $\iota(f)$ and $\iota^*(g^*)$, respectively. Note ι^* is just a notation, not to be interpreted as "the adjoint of ι."

We start by identifying the properties that a reasonable Euclidean embedding should satisfy intuitively. Motivated by the bilinearity of $\langle \cdot ; \cdot \rangle_{\mathcal{B}}$, it is natural to require

$$\langle f; g^* \rangle_{\mathcal{B}} \approx \langle \iota(f), \iota^*(g^*) \rangle, \quad \forall f \in \mathcal{B}, g^* \in \mathcal{B}^*, \tag{18}$$

where $\langle \cdot, \cdot \rangle$ stands for Euclidean inner product. As $\langle \cdot ; \cdot \rangle_{\mathcal{B}}$ is bilinear, ι and ι^* should be linear on $\mathcal{B}$ and $\mathcal{B}^*$ respectively. Also note $\iota^*((f+g)^*) \neq \iota^*(f^*) + \iota^*(g^*)$ in general.

Similar to the linearization of RKHS kernels, we can apply invertible transformations to ι and ι^*. For example, doubling ι while halving ι^* makes no difference. We will just choose one representation out of them. It is also noteworthy that in general, $\|\iota(f)\|$ (Euclidean norm) approximates $\|f\|_{\mathcal{H}}$ instead of $\|f\|_{\mathcal{B}}$. (18) is the only property that our Euclidean embedding needs to satisfy.

We start by embedding the unit ball $\mathsf{B} := \{f \in \mathcal{F} : \|f\|_{\mathcal{B}} \leq 1\}$. Characterizing R by support functions as in Proposition 1, a natural Euclidean approximation of $\|\cdot\|_{\mathcal{B}}$ is

$$\|v\|^2_{\tilde{\mathcal{B}}} := \|v\|^2 + \max\nolimits_{g \in S} \langle v, \tilde{g} \rangle^2, \quad \forall\, v \in \mathbb{R}^d, \tag{19}$$

where $\tilde{g}$ is the Euclidean embedding of g in the original RKHS, designed to satisfy that $\langle \tilde{f}, \tilde{g} \rangle \approx \langle f, g \rangle_{\mathcal{H}}$ for all $f, g \in \mathcal{H}$ (or a subset of interest). Commonly used embeddings include Fourier (Rahimi and Recht, 2008), hash (Shi et al., 2009), Nyström (Williams and Seeger, 2000), etc. For example, given landmarks $\{z_i\}_{i=1}^n$ sampled from $\mathcal{X}$, the Nyström approximation for a function $f \in \mathcal{H}$ is

$$\tilde{f} = K_z^{-1/2}(f(z_1), \ldots, f(z_n))^\top \tag{20}$$

$$\text{where} \quad K_z := [k(z_i, z_j)]_{i,j} \in \mathbb{R}^{n \times n}. \tag{21}$$

Naturally, the dual norm of $\|\cdot\|_{\tilde{\mathcal{B}}}$ is

$$\|u\|_{\tilde{\mathcal{B}}^*} := \max\nolimits_{v : \|v\|_{\tilde{\mathcal{B}}} \leq 1} \langle u, v \rangle, \quad \forall\, u \in \mathbb{R}^d. \tag{22}$$

Clearly the unit ball of $\|\cdot\|_{\tilde{\mathcal{B}}}$ and $\|\cdot\|_{\tilde{\mathcal{B}}^*}$ are also symmetric, and we denote them as $\tilde{\mathsf{B}}$ and $\tilde{\mathsf{B}}^*$, respectively.

$$\begin{tikzcd}
\mathcal{B} \arrow[r, "\iota"] \arrow[d, shift left, "j"] & \tilde{\mathcal{B}} \arrow[d, shift left, "\tilde{j}"] \\
\mathcal{B}^* \arrow[r, "\iota^*"] \arrow[u, shift left, "j^{-1}"] & \tilde{\mathcal{B}}^* \arrow[u, shift left, "\tilde{j}^{-1}"]
\end{tikzcd}$$

Figure 1: The commutative diagram for our embeddings.

As shown in Figure 1, we have the following commutative diagram. Let $j : \mathcal{B} \to \mathcal{B}^*$ be the star operator and j^{-1} its inverse, and similarly for $\tilde{j} : \tilde{\mathcal{B}} \to \tilde{\mathcal{B}}^*$ and its inverse $\tilde{j}^{-1}$. Then, it is natural to require

$$\iota = \tilde{j}^{-1} \circ \iota^* \circ j, \tag{23}$$

where $\tilde{j}^{-1}$ can be computed for any $u := \iota^*(f^*)$ via a Euclidean counterpart of Theorem 3:

$$\tilde{j}^{-1}(u) := \|u\|_{\tilde{\mathcal{B}}^*} \cdot \arg\max\nolimits_{v \in \tilde{\mathsf{B}}} \langle v, u \rangle. \tag{24}$$

The argmax is unique because $\|\cdot\|_{\tilde{\mathcal{B}}}$ is strictly convex.

At last, how can we get $\iota^*(f^*)$ in the first place? We start from the simpler case where f^* has a kernel expansion as in (12).[1] Here, by the linearity of ι^*, it will suffice to compute $\iota^*(G_x^*)$. By Theorem 3,

$$G_x = G_x^\circ(x) G_x^\circ, \quad \text{where} \quad G_x^\circ := \arg\max\nolimits_{h \in \mathsf{B}} h(x)$$

is uniquely attained. Denoting $k_y := k(\cdot, y)$, it follows

$$\langle G_x; G_y^* \rangle_{\mathcal{B}} \overset{\text{by (8)}}{=} G(x,y) = \langle G_x, k_y \rangle_{\mathcal{H}} = \langle G_x^\circ(x) G_x^\circ, k_y \rangle_{\mathcal{H}}.$$

So by comparing with (18), it is natural to introduce

$$\iota^*(G_y^*) := \tilde{k}_y, \tag{25}$$

$$\iota(G_x) := G_x^\circ(x) \tilde{G}_x^\circ \approx \left\langle \tilde{G}_x^\circ, \tilde{k}_x \right\rangle \tilde{G}_x^\circ, \tag{26}$$

$$\text{where} \quad \tilde{G}_x^\circ := \arg\max\nolimits_{v \in \tilde{\mathsf{B}}} \left\langle v, \tilde{k}_x \right\rangle. \tag{27}$$

[1] We stress that although the kernel expansion (12) is leveraged to *motivate* the design of ι^*, the underpinning foundation is that the span of $\{G_x^* : x \in \mathcal{X}\}$ is dense in $\mathcal{B}^*$ (Theorem 4). The representer theorem (Zhang et al., 2009, Theorem 2), which showed that the solution to (11) must be in the form of (12), is *not* relevant to our construction.

The last optimization is *convex* and can be solved very efficiently because, thanks to the positive homogeneity of R, it is equivalent to

$$\min_v \left\{ \|v\|^2 + \max_{g \in S} \langle v, \tilde{g} \rangle^2 \right\} \quad s.t. \quad v^\top \tilde{k}_x = 1. \quad (28)$$

Detailed derivation and proof are relegated to Appendix **??**. To solve (28), LBFGS with projection to a hyperplane (which has a trivial closed-form solution) turned out to be very efficient in our experiment. Overall, the construction of $\iota(f)$ and $\iota^*(f^*)$ for f^* from (12) proceeds as follows:

1. Define $\iota^*(G_x^*) = \tilde{k}_x$;
2. Define $\iota^*(f^*) = \sum_i \alpha_i \tilde{k}_{x_i}$ for $f^* = \sum_i \alpha_i G_{x_i}^*$;
3. Define $\iota(f)$ based on $\iota^*(f^*)$ by using (23).

In the next subsection, we will show that these definitions are sound, and both ι and ι^* are linear. However, the procedure may still be inconvenient in computation, because f needs to be first dualized to f^*, which in turn needs to be expanded into the form of (12). Fortunately, our representation learning only needs to compute the embedding of G_x, bypassing all these computational challenges.

4.1. Analysis of Euclidean Embeddings

The previous derivations are based on the necessary conditions for (18) to hold. We now show that ι and ι^* are well-defined, and are linear. To start with, denote the base Euclidean embedding on $\mathcal{H}$ by $T : \mathcal{H} \to \mathbb{R}^d$, where $T(f) = \tilde{f}$. Then by assumption, T is linear and $\tilde{k}_x = T(k(\cdot, x))$.

Theorem 4. *$\iota^*(f^*)$ is well defined for all $f^* \in \mathcal{B}^*$, and $\iota^* : \mathcal{B}^* \to \mathbb{R}^d$ is linear. That is,*

a) If $f^* = \sum_i \alpha_i G_{x_i}^* = \sum_j \beta_j G_{z_j}^*$ are two different expansions of f^*, then $\sum_i \alpha_i \tilde{k}_{x_i} = \sum_j \beta_j \tilde{k}_{z_j}$.

b) The linear span of $\{G_x^* : x \in \mathcal{X}\}$ is dense in $\mathcal{B}^*$. So extending the above to the whole $\mathcal{B}^*$ is straightforward thanks to the linearity of T.

We next analyze the linearity of ι. To start with, we make two assumptions on the Euclidean embedding of $\mathcal{H}$.

Assumption 2 (surjectivity). *For all $v \in \mathbb{R}^d$, there exists a $g_v \in \mathcal{H}$ such that $\tilde{g}_v = v$.*

Assumption 2 does not cost any generality, because it is satisfied whenever the d coordinates of the embedding are linearly independent. Otherwise, this can still be enforced easily by projecting to an orthonormal basis of $\{\tilde{g} : g \in \mathcal{H}\}$.

Assumption 3 (lossless). *$\left\langle \tilde{f}, \tilde{g} \right\rangle = \langle f, g \rangle_{\mathcal{H}}$ for all $f, g \in \mathcal{H}$. This is possible when,* e.g., *$\mathcal{H}$ is finite dimensional.*

Theorem 5. *$\iota : \mathcal{B} \to \mathbb{R}^d$ is linear under Assumptions 2 & 3.*

Although Theorems 4 and 5 appear intuitive, the proof for the latter is rather nontrivial and is deferred to Appendix **??**. Some lemmas there under Assumptions 2 and 3 may be of interest too, hence highlighted here.

1. $\langle \iota(f), \iota^*(g^*) \rangle = \langle f; g^* \rangle, \forall\, f \in \mathcal{B}, g^* \in \mathcal{B}^*$.
2. $\|g\|_{\mathcal{B}} = \|g^*\|_{\mathcal{B}^*} = \|\iota^*(g^*)\|_{\tilde{\mathcal{B}}^*} = \|\iota(g)\|_{\tilde{\mathcal{B}}}, \forall\, g \in \mathcal{B}$.
3. $\tilde{\mathsf{B}} = \iota(\mathsf{B}) := \{\iota(f) : \|f\|_{\mathcal{B}} \le 1\}$.
4. $\tilde{\mathsf{B}}^* = \iota^*(\mathsf{B}^*) := \{\iota^*(g^*) : \|g^*\|_{\mathcal{B}^*} \le 1\}$.
5. $\max_{v \in \tilde{\mathsf{B}}} \langle v, \iota^*(g^*) \rangle = \max_{f \in \mathsf{B}} \langle f; g^* \rangle, \ \ \forall g^* \in \mathcal{B}^*$.

4.2. Analysis under Inexact Euclidean Embedding

When Assumption 3 is unavailable, Theorem 4 still holds, but the linearity of ι has to be relaxed to an approximate sense. To analyze it, we first rigorously quantify the inexactness of the Euclidean embedding T. Consider a subspace based embedding, such as Nyström approximation. Here T satisfies that there exists a countable set of orthonormal bases $\{e_i\}_{i=1}^\infty$ of $\mathcal{H}$, such that

1. $Te_k = 0$ for all $k > d$,
2. $\langle Tf, Tg \rangle = \langle f, g \rangle_{\mathcal{H}}, \ \ \forall f, g \in V := \mathrm{span}\{e_1, \ldots, e_d\}$.

Clearly the Nyström approximation in (20) satisfies these conditions, where $d = n$, and $\{e_1, \ldots, e_d\}$ is any orthonormal basis of $\{k_{z_1}, \ldots, k_{z_d}\}$ (assuming d is no more than the dimensionality of $\mathcal{H}$).

Definition 5. *$f \in \mathcal{H}$ is called ϵ-approximable by T if*

$$\left\| f - \sum\nolimits_{i=1}^d \langle f, e_i \rangle_{\mathcal{H}} e_i \right\|_{\mathcal{H}} \le \epsilon. \quad (29)$$

In other words, the component of f in $V^\perp$ is at most ϵ.

Theorem 6 (The proof is in Appendix **??**). *Let $f, g \in \mathcal{F}$ and $\alpha \in \mathbb{R}$. Then $\iota(\alpha f_1) = \alpha \iota(f_1)$. If f, g, and all elements in S are ϵ-approximable by T, then*

$$|\langle \iota(f), \iota^*(g^*) \rangle - \langle f; g^* \rangle| = O(\sqrt{\epsilon}) \quad (30)$$

$$\|\iota(f+g) - \iota(f) - \iota(g)\| = O(\sqrt{\epsilon}). \quad (31)$$

To summarize, the primal embedding $\iota(G_x)$ as defined in (26) provides a new feature representation that incorporates structures in the data. Based on it, a simple linear model can be trained to achieve the desired regularities in prediction. We now demonstrate its flexibility and effectiveness on two example applications.

5. Application 1: Mixup

Mixup is a data augmentation technique (Zhang et al., 2018), where a pair of training examples x_i and x_j are randomly selected, and their convex interpolation is postulated to yield the same interpolation of output labels. In particular, when $y_i \in \{0,1\}^m$ is the one-hot vector encoding the class that x_i belongs to, the loss for the pair is

$$\mathbb{E}_\lambda[\ell(f(\underbrace{\lambda x_i + (1-\lambda)x_j}_{=:\,\tilde{x}_\lambda}),\ \underbrace{\lambda y_i + (1-\lambda)y_j}_{=:\,\tilde{y}_\lambda})]. \tag{32}$$

Existing literature relies on stochastic optimization, with a probability pre-specified on λ. This is somewhat artificial. Changing expectation to maximization appears more appealing, but no longer amenable to stochastic optimization.

To address this issue and to learn representations that incorporate mixup prior while also accommodating classification with multiclass or even structured output, we resort to a joint kernel $k((x,y),(x',y'))$, whose simplest form is decomposed as $k^x(x,x')k^y(y,y')$. Here k^x and k^y are separate kernels on input and output respectively. Now a function $f(x,y)$ learned from the corresponding RKHS quantifies the "compatibility" between x and y, and the prediction can be made by $\arg\max_y f(x,y)$. In this setting, the $R(f)$ for mixup regularization can leverage the ℓ_p norm of $g_{ij}(\lambda) := \frac{\partial}{\partial\lambda} f(\tilde{x}_\lambda, \tilde{y}_\lambda)$ over $\lambda \in [0,1]$, effectively accounting for an infinite number of invariances.

Theorem 7. *$R_{ij}(f) := \|g_{ij}(\lambda)\|_p$ satisfies Assumption 1 for all $p \in (1,\infty)$. The proof is in Appendix* **??**.

Clearly taking *expectation* or *maximization* over all pairs of n training examples still satisfies Assumption 1. In our experiment, we will use the ℓ_∞ norm, which despite not being covered by Theorem 7, is directly amenable to the embedding algorithm. More specifically, for each pair (x,y) we need to embed $k((\cdot,\cdot),(x,y))$ as a $d \times m$ matrix. This is different from the conventional setting where each example x employs one feature representation shared for all classes; here the representation changes for different classes y. To this end, we need to first embed each invariance $g_{ij}(\lambda)$ by

$$Z_\lambda^{ij} := \tfrac{\partial}{\partial\lambda}\big(\tilde{k}_{\hat{x}_\lambda}\tilde{y}_\lambda^\top\big) = \Big(\tfrac{\partial}{\partial\lambda}\tilde{k}_{\hat{x}_\lambda}\Big)\tilde{y}_\lambda^\top + \tilde{k}_{\hat{x}_\lambda}(y_i - y_j)^\top.$$

Letting $\langle A,B\rangle = \mathrm{tr}(A^\top B)$ and $\|V\|_{\mathsf{F}}^2 = \langle V,V\rangle$, the Euclidean embedding $\iota(G_{x,y})$ can be derived by solving (28):

$$\min_{V\in\mathbb{R}^{d\times m}} \left\{\alpha\,\|V\|_{\mathsf{F}}^2 + \frac{1}{n^2}\sum_{ij}\max_{\lambda\in[0,1]}\left\langle V, Z_\lambda^{ij}\right\rangle^2\right\} \tag{33}$$

$$s.t.\ \left\langle V, \tilde{k}_x y^\top\right\rangle = 1. \tag{34}$$

Although the maximization over λ in (33) is not concave, it is 1-D and a grid style search can solve it globally with $O(\frac{1}{\epsilon})$ complexity. In practice, a local solver like L-BFGS almost always found its global optimum in 10 iterations.

6. Application 2: Embedding Inference for Structured Multilabel Prediction

In output space, there is often prior knowledge about pairwise or multi-way relationships between labels/classes. For example, if an image represents a cat, then it must represent an animal, but not a dog (assuming there is at most one object in an image). Such logic relationships of implication and exclusion can be highly useful priors for learning (Mirzazadeh et al., 2015a; Deng et al., 2012). One way to leverage it is to perform inference at test time so that the predicted multilabel conforms to these logic. However, this can be computation intensive at test time, and it will be ideal if the predictor has already accounted for these logic, and at test time, one just needs to make binary decisions (relevant/irrelevant) for each individual category separately. We aim to achieve this by learning a representation that embeds this structured prior.

To this end, it is natural to employ the joint kernel framework. We model the implication relationship of $y_1 \to y_2$ by enforcing $f(x,y_2) \geq f(x,y_1)$, which translates to a penalty on the amount by which $f(x,y_1)$ is above $f(x,y_2)$

$$[f(x,y_1) - f(x,y_2)]_+, \quad \text{where } [z]_+ = \max\{0,z\}. \tag{35}$$

To model the mutual exclusion relationship of $y_1 \leftrightsquigarrow y_2$, intuitively we can encourage that $f(x,y_1) + f(x,y_2) \leq 0$, *i.e.*, a higher likelihood of being a cat demotes the likelihood of being a dog. It also allows both y_1 and y_2 to be irrelevant, *i.e.*, both $f(x,y_1)$ and $f(x,y_2)$ are negative. This amounts to another sublinear penalty on f: $[f(x,y_1) + f(x,y_2)]_+$. To summarize, letting $\tilde{p}$ be the empirical distribution, we can define $R(f)$ by

$$R(f)^2 := \mathop{\mathbb{E}}_{x\sim\tilde{p}}\Big[\max_{y_1\to y_2}[f(x,y_1) - f(x,y_2)]_+^2 \tag{36}$$

$$+ \max_{y_1\leftrightsquigarrow y_2}[f(x,y_1) + f(x,y_2)]_+^2\Big]. \tag{37}$$

It is noteworthy that although $R(f)$ is positively homogeneous and convex (hence sublinear), it is no longer absolutely homogeneous and therefore not satisfying Assumption 1. However, the embedding algorithm is still applicable without change. It will be interesting to study the presence of kernel function G in spaces "normed" by sublinear functions. We leave it for future work.

7. Experiments

Here we highlight the major results and experiment setup. Details on data preprocessing, experiment setting, optimization, and additional results are given in Appendix **??**.

Table 1: Test accuracy of minimizing empirical risk on binary classification tasks.

	SVM	Warping	Dual	Embed
4 v.s. 9	97.1	98.0	97.6	97.8
2 v.s. 3	98.4	99.1	98.7	98.9

7.1. Sanity check for s.i.p. based methods

Our first experiment tries to test the effectiveness of optimizing the regularized risk (11) with respect to the dual coefficients $\{c_j\}$ in (12). We compared 4 algorithms: SVM with Gaussian kernel; Warping which incorporates transformation invariance by kernel warping as described in Ma et al. (2019); Dual which trains the dual coefficients $\{c_j\}$ by LBFGS to minimize empirical risk as in (11); Embed which finds the Euclidean embeddings by convex optimization as in (28), followed by a linear classifier. The detailed derivation of the gradient in $\{c_j\}$ for Dual is relegated to Appendix **??**.

Four transformation invariances were considered, including rotation, scaling, and shifts to the left and upwards. Warping summed up the square of $\frac{\partial}{\partial \alpha}|_{\alpha=0} f(I(\alpha))$ over the four transformations, while Dual and Embed took their max as the $R(f)^2$. To ease the computation of derivative, we resorted to finite difference for all methods, with two pixels for shifting, 10 degrees for rotation, and 0.1 unit for scaling. No data augmentation was applied.

All algorithms were evaluated on two binary classification tasks: 4 v.s. 9 and 2 v.s. 3, both sampling 1000 training and 1000 test examples from the MNIST dataset.

Since the square loss on the invariances used by Warping makes good sense, the purpose of this experiment is *not* to show that the s.i.p. based methods are better in this setting. Instead we aim to perform a sanity check on a) good solutions can be found for the nonconvex optimization over the dual variables in Dual, b) the Euclidean embedding of s.i.p. representers performs competitively. As Table 1 shows, both checks turned out affirmative, with Dual and Embed delivering similar accuracy as Warping. In addition, Embed achieved higher accuracy than dual optimization, suggesting that the learned representations have well captured the invariances and possess better predictive power.

7.2. Mixup

We next investigated the performance of Embed on mixup.

Datasets. We experimented with three image datasets: MNIST, USPS, and Fashion MNIST, each containing 10 classes. From each dataset, we drew $n \in \{500, 1000\}$ examples for training and n examples for testing. Based on the training data, p number of pairs were drawn from it. Both Vanilla and Embed used Gaussian RKHS, along with Nyström approximation whose landmark points consisted of the entire training set. The vanilla mixup optimizes the objective (32) averaged over all sampled pairs. Following Zhang et al. (2018), The λ was generated from a Beta distribution, whose parameter was tuned to optimize the performance. Again, Embed was trained with a linear classifier.

Algorithms. We first ran mixup with stochastic optimization where pairs were drawn on the fly. Then we switched to batch training of mixup (denoted as Vanilla), with the number of sampled pair increased from $p = n$, $2n$, up to $5n$. It turned out when $p = 4n$, the performance already matches the best test accuracy of the online stochastic version, which generally witnesses much more pairs. Therefore we also varied p in $\{n, 2n, 4n\}$ when training Embed. each setting was evaluated 10 times with randomly sampled training and test data. The mean and standard deviation are reported in Table 2.

Results. As Table 2 shows, Embed achieves higher accuracy than Vanilla on almost all datasets and combinations of n and p. The margin tends to be higher when the training set size (n and p) is smaller. Besides, Vanilla achieves the highest accuracy at $p = 4n$.

7.3. Structured multilabel prediction

Finally, we validate the performance of Embed on *structured multilabel prediction* as described in Section 6, showing that it is able to capture the structured relationships between the class labels (implication and exclusion) in a hierarchical multilabel prediction task.

Datasets. We conducted experiments on three multilabel datasets where additional information is available about the hierarchy in its class labels (link): Enron (Klimt and Yang, 2004), WIPO (Rousu et al., 2006), Reuters (Lewis et al., 2004). Implication constraints were trivially derived from the hierarchy, and we took siblings (of the same parent) as *exclusion* constraints. For each dataset, we experimented with $100/100, 200/200, 500/500$ randomly drawn train/test examples.

Algorithms. We compared Embed with two baseline algorithms for multilabel classification: a multilabel SVM with RBF kernel (ML-SVM), and an SVM that incorporates the hierarchical label constraints (HR-SVM) (Vateekul et al., 2012). No inference is conducted at test time, such as removing violations of implications or exclusions known a priori.

Results. Table 3 reports the accuracy on the three train/test splits for each of the datasets. Clearly, Embed outperforms both the baselines in most of the cases.

Table 2: Test accuracy on mixup classification task based on 10 random runs.

Dataset		$n = 500$			$n = 1000$		
	p	n	$2n$	$4n$	n	$2n$	$4n$
MNIST	Vanilla	90.16±1.40	90.93±1.01	91.40±1.04	91.00±1.17	92.01±1.21	92.48±1.03
	Embed	**91.36±1.41**	**91.90±1.08**	**92.11±1.01**	**92.51±1.01**	**92.79±0.98**	**93.03±1.00**
USPS	Vanilla	90.54±1.28	91.76±1.14	92.40±1.25	93.87±1.19	94.72±1.12	95.32±1.13
	Embed	**92.46±1.24**	**93.02±1.12**	**93.21±1.14**	**94.74±0.97**	**95.11±0.94**	**95.67±0.96**
Fashion MNIST	Vanilla	79.37±3.11	81.15±2.08	81.72±1.96	82.53±1.49	83.13±1.36	83.69±1.31
	Embed	**81.56±2.27**	**82.16±1.56**	**82.52±1.49**	**83.28±1.48**	**84.07±1.32**	**84.34±1.31**

Table 3: Test accuracy on multilabel prediction with logic relationship

Dataset	Embed			ML-SVM			HR-SVM		
	100	200	500	100	200	500	100	200	500
Enron	**96.2**±0.3	**95.7**±0.2	**94.7**±0.2	92.7±0.4	91.8±0.4	91.0±0.3	93.1±0.3	92.5±0.3	92.0±0.2
Reuters	**95.7**±1.4	**97.2**±1.2	**98.0**±0.4	94.2±1.4	95.1±1.3	95.2±1.2	95.1±1.2	**97.3**±1.3	97.7±1.3
WIPO	**98.6**±0.1	**98.4**±0.1	98.4±0.1	98.1±0.3	98.2±0.2	98.3±0.1	98.3±0.1	**98.5**±0.1	**98.7**±0.2

8. Conclusions and Future Work

In this paper, we introduced a new framework of representation learning where an RKHS is turned into a semi-inner-product space via a semi-norm regularizer, broadening the applicability of kernel warping to *generalized* invariances, *i.e.*, relationships that hold irrespective of certain changes in data. For example, the mixup regularizer enforces smooth variation irrespective of the interpolation parameter λ, and the structured multilabel regularizer enforces logic relationships between labels regardless of input features. Neither of them can be modeled convexly by conventional methods in transformation invariance, and the framework can also be directly applied to non-parametric transformations (Pal et al., 2017). An efficient Euclidean embedding algorithm was designed and its theoretical properties are analyzed. Favorable experimental results were demonstrated for the above two applications.

This new framework has considerable potential of being applied to other invariances and learning scenarios. For example, it can be directly used in maximum mean discrepancy and the Hilbert–Schmidt independence criterion, providing efficient algorithms that complement the mathematical analysis in Fukumizu et al. (2011). It can also be applied to convex deep neural networks (Ganapathiraman et al., 2018; 2016), which convexify multi-layer neural networks through kernel matrices of the hidden layer outputs.

Other examples of generalized invariance include *convex* learning of: a) node representations in large networks that are robust to topological perturbations (Zügner et al., 2018). The exponential number of perturbation necessitates max instead of sum; b) equivariance based on the largest deviation under swapped transformations over the input domain (Ravanbakhsh et al., 2017); and c) co-embedding multiway relations that preserve co-occurrence and affinity between groups (Mirzazadeh et al., 2015b).

Acknowledgements

We thank the reviewers for their constructive comments. This work is supported by Google Cloud and NSF grant RI:1910146. YY thanks NSERC and the Canada CIFAR AI Chairs program for funding support.

References

Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

K. P. Bennett and E. J. Bredensteiner. Duality and geometry in SVM classifiers. In P. Langley, editor, *International Conference on Machine Learning (ICML)*, pages 57–64, San Francisco, California, 2000. Morgan Kaufmann Publishers.

C. Bhattacharyya, K. S. Pannagadatta, and A. J. Smola. A second order cone programming formulation for classifying missing data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 153–160, 2005.

J. M. Borwein and J. D. Vanderwerff. *Convex Functions: Constructions, Characterizations and Counterexamples*. Cambridge University Press, 2010.

T. S. Cohen and M. Welling. Group Equivariant Convolutional Networks. In *International Conference on Machine Learning (ICML)*, 2016.

P. L. Combettes, S. Salzo, and S. Villa. Regularized learning scheme in feature Banach spaces. *Analysis and Applications*, 16(1):1–54, 2018.

E. Creager, D. Madras, J.-H. Jacobsen, M. Weis, K. Swersky, T. Pitassi, and R. Zemel. Flexibly fair representation learning by disentanglement. In *International Conference on Machine Learning (ICML)*, 2019.

J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-scale object classification using label relation graphs. In *European Conference on Computer Vision (ECCV)*, 2012.

R. Der and D. Lee. Large-margin classification in Banach spaces. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2007.

T. Eliassi-Rad and C. Faloutsos. Discovering roles and anomalies in graphs:theory and applications. *Tutorial at SIAM International Conference on Data Mining (ICDM)*, 2012.

G. D. Faulkner. Representation of linear functionals in a Banach space. *Rocky Mountain Journal of Mathematics*, 7(4):789–792, 1977.

M. Ferraro and T. M. Caelli. Lie transformation groups, integral transforms, and invariant pattern recognition. *Spatial Vision*, 8:33–44, 1994.

K. Fukumizu, G. R. Lanckriet, and B. K. Sriperumbudur. Learning in Hilbert vs. Banach spaces: A measure embedding viewpoint. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2011.

V. Ganapathiraman, X. Zhang, Y. Yu, and J. Wen. Convex two-layer modeling with latent structure. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

V. Ganapathiraman, Z. Shi, X. Zhang, and Y. Yu. Inductive two-layer modeling with parametric bregman transfer. In *International Conference on Machine Learning (ICML)*, 2018.

J. R. Giles. Classes of semi-inner-product spaces. *Transactions of the American Mathematical Society*, 129(3): 436–446, 1967.

D. Graham and S. Ravanbakhsh. Equivariant entity-relationship networks. *arXiv:1903.09033*, 2019.

B. Haasdonk and H. Burkhardt. Invariant kernel functions for pattern analysis and machine learning. *Machine Learning*, 68(1):35–61, 2007.

B. Haasdonk and D. Keysers. Tangent distance kernels for support vector machines. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 2, pages 864–868. IEEE, 2002.

M. Hein, O. Bousquet, and B. Schölkopf. Maximal margin classification for metric spaces. *J. Comput. System Sci.*, 71:333–359, 2005.

I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. β-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*, 2017.

B. Klimt and Y. Yang. The enron corpus: A new dataset for email classification research. In *European Conference on Machine Learning (ECML)*, pages 217–226. Springer, 2004.

A. Kumar, P. Sattigeri, K. Wadhawan, L. Karlinsky, R. Feris, W. T. Freeman, and G. Wornell. Co-regularized alignment for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research (JMLR)*, 5(Apr): 361–397, 2004.

link. Multilabel dataset. `https://sites.google.com/site/hrsvmproject/datasets-hier`.

G. Lumer. Semi-inner-product spaces. *Transactions of the American Mathematical Society*, 100:29–43, 1961.

Y. Ma, V. Ganapathiraman, and X. Zhang. Learning invariant representations with kernel warping. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

F. Mirzazadeh, S. Ravanbakhsh, N. Ding, and D. Schuurmans. Embedding inference for structured multilabel prediction. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015a.

F. Mirzazadeh, M. White, A. György, and D. Schuurmans. Scalable metric learning for co-embedding. In *European Conference on Machine Learning (ECML)*, 2015b.

Y. Mroueh, S. Voinea, and T. Poggio. Learning with group invariant features: A kernel perspective. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

D. K. Pal, A. A. Kannan, G. Arakalgud, and M. Savvides. Max-margin invariant features from transformed unlabeled data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

A. Rahimi and B. Recht. Random features for large-scale kernel machines. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems (NeurIPS)*. MIT Press, Cambridge, MA, 2008.

H. Rahimian and S. Mehrotra. Distributionally robust optimization: A review. *arXiv:1908.05659*, 2019.

A. Raj, A. Kumar, Y. Mroueh, P. Thomas Fletcher, and B. Schoelkopf. Local group invariant representations via orbit embeddings. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

S. Ravanbakhsh, J. Schneider, and B. Poczos. Equivariance Through Parameter-Sharing. In *International Conference on Machine Learning (ICML)*, 2017.

S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *International Conference on Machine Learning (ICML)*, 2011.

J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor. Kernel-based learning of hierarchical multilabel classification models. *Journal of Machine Learning Research (JMLR)*, 7(Jul):1601–1626, 2006.

S. Salzo, L. Rosasco, and J. Suykens. Solving ℓ^p-norm regularization with tensor kernels. In A. Storkey and F. Perez-Cruz, editors, *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 84, 2018.

Q. Shi, J. Petterson, G. Dror, J. Langford, A. Smola, and S. Vishwanathan. Hash kernels for structured data. *Journal of Machine Learning Research (JMLR)*, 10:2615–2637, 2009.

P. Simard, Y. LeCun, J. S. Denker, and B. Victorri. Transformation invariance in pattern recognition-tangent distance and tangent propagation. In *Neural Networks: Tricks of the Trade*, pages 239–274, 1996.

A. Smola. Sets and symmetries. *NeurIPS Workshop on Sets & Partitions*, 2019.

A. J. Smola and B. Schölkopf. On a kernel-based method for pattern recognition, regression, approximation and operator inversion. *Algorithmica*, 22:211–231, 1998.

C. H. Teo, A. Globerson, S. Roweis, and A. Smola. Convex learning with invariances. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2007.

P. Vateekul, M. Kubat, and K. Sarinnapakorn. Top-down optimized svms for hierarchical multi-label classification: A case study in gene function prediction. *Intelligent Data Analysis*, 2012.

U. von Luxburg and O. Bousquet. Distance-based classification with lipschitz functions. *Journal of Machine Learning Research*, 5:669–695, 2004.

W. Wang, R. Arora, K. Livescu, and J. Bilmes. On deep multi-view representation learning. In *International Conference on Machine Learning (ICML)*, 2015.

C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2000.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. Smola. Deep sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

H. Zhang, Y. Xu, and J. Zhang. Reproducing kernel Banach spaces for machine learning. *Journal of Machine Learning Research (JMLR)*, 10:2741–2775, 2009.

H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

X. Zhang, W. S. Lee, and Y. W. Teh. Learning with invariance via linear functionals on reproducing kernel hilbert space. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2013.

D. Zhou, B. Xiao, H. Zhou, and R. Dai. Global geometry of svm classifiers. Technical Report 30-5-02, Institute of Automation, Chinese Academy of Sciences, 2002.

D. Zügner, A. Akbarnejad, and S. Günnemann. Adversarial attacks on neural networks for graph data. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 2847–2856. ACM, 2018.