Semi-Supervised Learning in Reproducing Kernel Hilbert Spaces Using Local Invariances

Wee Sun Lee[1,2], Xinhua Zhang[1,2], and Yee Whye Teh[1]

[1] Department of Computer Science, National University of Singapore.
[2] Singapore-MIT Alliance, E4-4-10, 4 Engineering Drive 3, Singapore 117576.

Abstract. We propose a framework for semi-supervised learning in reproducing kernel Hilbert spaces using local invariances that explicitly characterize the behavior of the target function around both labeled and unlabeled data instances. Such invariances include: invariance to small changes to the data instances, invariance to averaging across a small neighbourhood around data instances, and invariance to local transformations such as translation and rotation. These invariances are approximated by minimizing loss functions on derivatives and local averages of the functions. We use a regularized cost function, consisting of the sum of loss functions penalized with the squared norm of the function, and give a representer theorem showing that an optimal function can be represented as a linear combination of a finite number of basis functions. For the representer theorem to hold, the derivatives and local averages are required to be bounded linear functionals in the reproducing kernel Hilbert space. We show that this is true in the reproducing kernel Hilbert spaces defined by Gaussian and polynomial kernels.

1 Introduction

Semi-supervised learning is the problem of learning from labeled and unlabeled training data. It is important in application domains where labeled data is scarce but unlabeled data can be easily obtained. Accurate assumptions on the relationship between the data distribution and the target function are essential for semi-supervised learning; without such assumptions, unlabeled data would contribute no information to the learning process [10]. However with incorrect assumptions, the outcome of using unlabeled data in learning can actually be worse than not using the unlabeled data—the algorithm may select a function that fits the data distribution assumption well but does badly on the labeled data loss function, when it is unable to do both well.

In this paper, we consider two types of local invariance assumptions that are often suggested by prior knowledge in various domains. The first type of assumption is that the target function does not change much in the neighbourhood of each observed data instance. This is reasonable when instances from the same class are clustered together and away from instances from different classes. Since the function is allowed to change more rapidly away from the observed data instances, the decision boundary is encouraged to fall in regions of low data density. We consider two methods for implementing this type of assumption: the first restricts the gradient of the function to be small at the observed data instances, while the second restricts the function value at each data instance to be similar to the average value across a small neighbourhood of that instance.

The second type of invariance assumption used in this paper is invariance to certain local transformations. Examples of useful transformations include translational and rotational invariances in vision problems such as handwritten digit recognition. We utilize this type of local transformation invariance assumption by assuming that the gradient of the function is small along directions of transformation at each data instance.

To reduce the risk of not being able to approximate the target function well, we use powerful reproducing kernel Hilbert spaces of functions for learning. It turns out that incorporating the desired local invariances into learning can be elegantly done in this approach by treating derivatives and local averages as linear functionals in those spaces. The cost function that we minimize consists of the sum of loss functions on these linear functionals, the usual loss functions on the labeled training data, and a regularization penalty based on the squared norm of the function. We give a representer theorem, showing that we can represent an optimal function that minimizes the cost using a finite number of basis functions when the linear functionals are bounded. Furthermore, with convex loss functions, the resulting optimization problem is convex. We then show that the linear functionals that we use are bounded in the reproducing kernel Hilbert spaces defined by the Gaussian and polynomial kernels, allowing their use within the framework.

Previous works on semi-supervised learning have explored the assumption that the decision boundary should pass through regions of low data density. Transductive support vector machines tries to label the unlabeled examples in such a way that margin of the resulting function is large. This process is computationally expensive and is performed using a heuristic in [3]. Methods based on minimizing the cut in a graph with edges representing similarity between examples are proposed in [2, 11, 4]. These methods only produce labels on unlabeled data that are present during training and not on future unseen examples. The manifold regularization method in [1] uses a regularizer based on a graph Laplacian, extending graph based methods [11], and allowing a hypothesis function to be produced. Like the method in [1], the methods proposed in this paper results in convex optimization problems and produce hypothesis functions that can be used for future predictions. In fact, the terms in the graph Laplacian regularizer can be expressed as bounded linear functionals, allowing the method to be put within the framework proposed here. We are not aware of any previous work on using local transformation invariance for semi-supervised learning although it has been used in supervised learning [7]. A similar representer theorem for bounded linear functional is provided in [8], but not in the context of semi-supervised learning.

We give some mathematical preliminaries in Section 2, the representer theorem in Section 3, and show that the linear functionals used are bounded in Section 4. A preliminary experiment on a simple synthetic data set using the gradient functional shows encouraging result and is described in Section 5. We discuss other potential applications for the techniques in this paper in Section 6.

2 Preliminaries

2.1 Kernels

Definition 1 (Positive Definite Kernel Matrix). *Given a function $k : X^2 \to \mathbb{R}$ and $x_1, \ldots, x_l \in X$, the matrix K where $K_{ij} = k(x_i, x_j)$ is called the kernel matrix of k with respect to $x_1, \ldots, x_l \in \mathbb{R}$. If $\sum_{i,j=1}^{l} \alpha_i \alpha_j K_{ij} \geq 0$ for all $\alpha_1, \ldots, \alpha_l \in \mathbb{R}$. we say the kernel matrix is* positive definite.

Note that we follow the convention in [6] of using the term positive definite even though the inequality is not strict.

Definition 2 (Positive Definite Kernel Function). *A function $k : X^2 \to \mathbb{R}$ is called a* positive definite kernel *if for all $l \in \mathbb{N}$ and all $x_1, \ldots, x_l \in X$, the kernel matrix K formed by $K_{ij} = k(x_i, x_j)$ is symmetric positive definite.*

We will refer to positive definite kernel functions simply as *kernels*.

Example 1 (Kernels). The most commonly used nonlinear kernels are the Gaussian and polynomial kernels. The Gaussian kernel, defined on $\mathbb{R}^d \times \mathbb{R}^d$, is $k(x_1, x_2) = \exp\left(-\frac{1}{2\sigma_k^2}\|x_1 - x_2\|^2\right)$. The polynomial kernel and the homogeneous polynomial kernel of degree m, defined on $\mathbb{R}^d \times \mathbb{R}^d$, are $k(x_1, x_2) = (x_1 \cdot x_2 + 1)^m$ and $k(x_1, x_2) = (x_1 \cdot x_2)^m$ respectively.

2.2 Reproducing Kernel Hilbert Space

Given a positive definite kernel, we can use the functions $k(x, \cdot)$, $x \in X$ to construct a normed space by defining an appropriate inner product. We define the vector space by taking linear combinations of the functions

$$f(\cdot) = \sum_{i=1}^{l} \alpha_i k(x_i, \cdot)$$

for arbitrary $l \in \mathbb{N}$, $\alpha_i \in \mathbb{R}$ and $x_1, \ldots, x_l \in X$. The inner product between f and $g = \sum_{j=1}^{l'} \beta_j k(x'_j, \cdot)$ is defined as

$$\langle f, g \rangle = \sum_{i=1}^{l} \sum_{j=1}^{l'} \alpha_i \beta_j k(x_i, x'_j).$$

This definition can be shown to satisfy the properties of an inner product, namely, symmetry ($\langle f, g \rangle = \langle g, f \rangle$), linearity ($\langle af + bg, h \rangle = a\langle f, h \rangle + b\langle g, h \rangle$) and positive definiteness ($\langle f, f \rangle \geq 0$; and $\langle f, f \rangle = 0$ implies $f = \mathbf{0}$) [6].

With this definition, we have

$$\|f\|^2 = \langle f, f \rangle = \sum_{i,j=1}^{l} \alpha_i \alpha_j k(x_i, x_j) \geq 0.$$

Another useful property of this space is the fact that k is a *reproducing kernel*, that is

$$f(x) = \langle k(x, \cdot), f \rangle,$$

which follows from the definition of the inner product.

With the inner product on the vector space, we have obtained a *pre-Hilbert* space. We complete the space by adding the limit points of convergent sequences to form a Hilbert space, usually called a *reproducing kernel Hilbert space (RKHS)*.

2.3 Operators and Functionals

Definition 3 (Linear Operator and Functional). *A* linear operator T *is a mapping from a vector space X to a vector space Y, such that for all $x, y \in X$ and scalar α,*

$$T(x + y) = Tx + Ty$$
$$T(\alpha x) = \alpha T x.$$

If the range $Y \subseteq \mathbb{R}$, the operator is called a functional*.*

Definition 4 (Bounded Linear Operator). *Let $T : X \to Y$ be a linear operator on a normed spaces X and Y. The operator T is said to be bounded if there exists some $c > 0$ such that for all $x \in X$*

$$\|Tx\| \leq c\|x\|.$$

The smallest value of c such that the inequality holds for all nonzero $x \in X$ is called the norm of the operator and denoted $\|T\|$.

Bounded linear operators defined on X can be extended to the completion of the space such that its norm is preserved [5]. Throughout this paper, we define our bounded linear functionals on the pre-Hilbert space constructed with the kernels and use its extension on the completion of the space.

Example 2. For each x in a reproducing kernel Hilbert space H the linear functional $f \mapsto \langle f, k(x, \cdot) \rangle = f(x)$ is bounded since $|\langle f, k(x, \cdot) \rangle| \leq \|k(x, \cdot)\| \|f\| = k(x, x)^{1/2} \|f\|$ by the Cauchy-Schwarz inequality.

In fact, the bounded linear functionals on a Hilbert space H are in 1-1 correspondence with elements $x \in H$, as shown by Riesz's Theorem (see [5]).

Theorem 1 (Riesz). *Every bounded linear functional L on a Hilbert space H can be represented in terms of an inner product*

$$L(x) = \langle x, z \rangle$$

where the representer *of the functional, z, has norm*

$$\|z\| = \|L\|$$

and is uniquely determined by L.

When H is a reproducing kernel Hilbert space the representer of the functional has the form

$$z(x) = \langle z, k(x, \cdot) \rangle = L(k(x, \cdot)).$$

Riesz's theorem will be useful for our paper since it allows us to represent functionals related to local invariances as elements of the reproducing kernel Hilbert space.

3 Representer Theorem

We wish to learn a target function f both from labeled data and from local invariances extracted from labeled and unlabeled data. Let $(x_1, y_1), \ldots, (x_l, y_l)$ be the labeled training data, and $l_2(y, f(x))$ be the loss function on f when training input x is labeled as y. We measure deviations from local invariances around each labeled or unlabeled input instance, and express these as bounded linear functionals $L_{l+1}(f), \ldots, L_n(f)$ on the reproducing kernel Hilbert space H. The linear functionals are associated with another loss function $l_1(L_i(f))$ penalizing violations of the local invariances. As an example, the derivative of f with respect to an input feature at some training instance x is a linear functional in f, and the loss function penalizes large values of the derivative at x. Section 4 describes other local invariances we can consider and show that these can be expressed as bounded linear functionals. Finally, we place a squared loss function $\|f\|^2$ as a regularization term, penalizing functions with large norms. Putting these loss functions together, we set out to find the function minimizing the cost

$$\sum_{i=1}^{l} l_2(y_i, f(x_i)) + \rho_2 \sum_{i=l+1}^{n} l_1\left(L_i(f)\right) + \rho_1 \|f\|^2.$$

where $\rho_1, \rho_2 > 0$ are the relative strengths of the loss functions. Reasonable examples of l_2 include the logistic loss, hinge loss and squared loss while examples of l_1 include the squared loss, absolute loss and ϵ-insensitive loss. These are all convex loss functions and result in convex optimization problems for finding the optimal f.

In the following, we derive a representer theorem showing that the solution of the optimization problem lies in the span of a finite number of functions associated with the labeled data and the functionals. Similar results are available in [8].

Theorem 2. *Let L_i, $i = l+1, \ldots, n$, be bounded linear functionals in the reproducing kernel Hilbert space H defined by the kernel k. The solution of the optimization problem*

$$g = \mathit{argmin}_{f \in H} \sum_{i=1}^{l} l_2(y_i, f(x_i)) + \rho_2 \sum_{i=l+1}^{n} l_1\left(L_i(f)\right) + \rho_1 \|f\|^2$$

for $\rho_1, \rho_2 > 0$ can be expressed as

$$g(\cdot) = \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot)$$

where z_i is the representer of L_i. Furthermore, the parameters $\alpha = [\alpha_1, \ldots, \alpha_n]^T$ can be obtained by minimizing

$$\sum_{i=1}^{l} l_2(y_i, f(x_i)) + \rho_2 \sum_{i=l+1}^{n} l_1\left(L_i(f)\right) + \rho_1 \alpha^T K \alpha \tag{1}$$

where $f = \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot)$ and $K_{i,j} = \langle k'_i, k'_j \rangle$ where $k'_i = k(x_i, \cdot)$ if $i \leq l$ and $k'_i = z_i(\cdot)$ otherwise.

Proof. By Riesz's Theorem, the linear functional $L_i(f)$ can be represented as an inner product

$$L_i(f) = \langle f, z_i \rangle$$

where z_i is a member of the Hilbert space and $z_i(x) = \langle z_i, k(x, \cdot)\rangle$.

We now claim that the solution of the optimization problem can be represented as

$$\sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot).$$

To see this, any function f in the RKHS can be represented as

$$f(\cdot) = \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot) + f_\perp(\cdot)$$

where $f_\perp(x)$ is in the orthogonal complement of the span of $k(x_i, \cdot)$ for $1 \le i \le l$ and of $z_i(\cdot)$ for $l+1 \le i \le n$. Each of the terms that contains the loss function l_2 depends only on $f(x_k)$ which can be written as

$$\begin{aligned} f(x_k) &= \langle f(\cdot), k(x_k, \cdot)\rangle \\ &= \sum_{i=1}^{l} \alpha_i \langle k(x_i, \cdot), k(x_k, \cdot)\rangle + \sum_{i=l+1}^{n} \alpha_i \langle z_i(\cdot), k(x_k, \cdot)\rangle + \langle f_\perp(x), k(x_k, \cdot)\rangle \\ &= \sum_{i=1}^{l} \alpha_i k(x_i, x_k) + \sum_{i=l+1}^{n} \alpha_i z_i(x_k). \end{aligned}$$

Hence, each of those terms depends only on the projection of the function onto the span.

Similarly, each of the terms that contain the loss function l_1 depends only on the projection of the function onto the span.

$$\begin{aligned} L_k(f) = \langle f, z_k \rangle &= \sum_{i=1}^{l} \alpha_i \langle k(x_i, \cdot), z_k\rangle + \sum_{i=l+1}^{n} \alpha_i \langle z_i, z_k\rangle + \langle f_\perp(x), z_k\rangle \\ &= \langle \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i, z_k\rangle. \end{aligned}$$

Since all the values in the loss functions l_1 and l_2 depends only on the component that lies in the span, any function that has components in the orthogonal complement has higher cost than its projection onto the span. Hence, the solution of the optimization problem must lie in the span of the desired functions.

Finally, note that the squared norm of the function

$$f(\cdot) = \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot)$$

can be written as $\alpha^T K \alpha$ where $K_{i,j} = \langle k'_i, k'_j \rangle$ where $k'_i = k(x_i, \cdot)$ if $i \le l$ and $k'_i = z_i(\cdot)$ otherwise. □

In practice, learning machines such as the support vector machine often use an additional constant value (bias) that is not penalized in the optimization. The following more general version of the representer theorem covers this case as well. The proof is similar to that of Theorem 2.

Theorem 3. *Let L_i, $i = l+1, \ldots, n$, be bounded linear functionals in the reproducing kernel Hilbert space H defined by the kernel k. Let F be the span of a fixed set of basis functions ϕ_j, $j = 1, \ldots, m$. The solution of the optimization problem*

$$g = argmin_{f=f_1+f_2, f_1 \in F, f_2 \in H} \sum_{i=1}^{l} l_2(y_i, f(x_i)) + \rho_2 \sum_{i=l+1}^{n} l_1\left(L_i(f)\right) + \rho_1 \|f_2\|^2$$

for $\rho_1, \rho_2 > 0$ can be expressed as

$$g(\cdot) = \sum_{j=1}^{m} w_j \phi_j(\cdot) + \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot)$$

where z_i is the representer of L_i. Furthermore, the parameters $w = [w_1, \ldots, w_m]^T$ and $\alpha = [\alpha_1, \ldots, \alpha_n]^T$ can be obtained by minimizing

$$\sum_{i=1}^{l} l_2(y_i, f(x_i)) + \rho_2 \sum_{i=l+1}^{n} l_1\left(L_i(f)\right) + \rho_1 \alpha^T K \alpha$$

where $f = \sum_{j=1}^{m} w_j \phi_j(\cdot) + \sum_{i=1}^{l} \alpha_i k(x_i, \cdot) + \sum_{i=l+1}^{n} \alpha_i z_i(\cdot)$ and $K_{i,j} = \langle k'_i, k'_j \rangle$ where $k'_i = k(x_i, \cdot)$ if $i \leq l$ and $k'_i = z_i(\cdot)$ otherwise.

4 Local Invariances as Bounded Linear Functionals

4.1 Derivatives

Let x^i be the i-th component of the vector x. The following theorem from functional analysis shows that derivatives as well as the transformation and local average functionals described later are all bounded on reproducing kernel Hilbert spaces defined by polynomial kernels.

Theorem 4 (see [5]). *Let X be a finite dimensional normed space. Then every linear functional on X is bounded.*

Corollary 1. *The linear functional $L_{x_{i,j}}(f) = \left.\frac{\partial f(x)}{\partial x^j}\right|_{x_i}$ is bounded in the reproducing kernel Hilbert spaces defined by the polynomial kernel and the homogeneous polynomial kernel.*

For polynomial kernel $k(x, y) = (x \cdot y + 1)^n$, we have

$$z_{x_{i,j}}(x) = n(x \cdot x_i + 1)^{n-1} x^j$$

and

$$\langle z_{x_{i,j}}, z_{x_{p,q}} \rangle = \begin{cases} (n-1)n(x_p \cdot x_i + 1)^{n-2} x_p^j x_i^q & \text{if } j \neq q \text{ and} \\ (n-1)n(x_p \cdot x_i + 1)^{n-2} x_p^j x_i^q + n(x_p \cdot x_i + 1)^{n-1} & \text{if } j = q. \end{cases}$$

For Gaussian kernels, we will need the following known results.

Lemma 1 (see [6]). *The following are properties of kernels:*

- *if k_1 and k_2 are kernels, and $\alpha_1, \alpha_2 \geq 0$, then $\alpha k_1 + \alpha_2 k_2$ is a kernel.*
- *if $k_1, k_2, \ldots$ are kernels, and $k(x, x') = \lim_{n\to\infty} k_n(x, x')$ exists for all x, x', then k is a kernel.*

Theorem 5. *The linear functional $L_{x_{i,j}}(f) = \left.\frac{\partial f(x)}{\partial x^j}\right|_{x_i}$ is bounded in the reproducing kernel Hilbert space defined by Gaussian kernels with $||L_{x_{i,j}}|| = 1/\sigma$.*

Proof. We will first show that the theorem holds for those f with

$$f(x) = \sum_{k=1}^{l} \alpha_k \exp\left(-\frac{1}{2\sigma^2}\|t_k - x\|^2\right)$$

for some l and $t_1, \ldots, t_l \in \mathbb{R}^d$. We have

$$\left.\frac{\partial f(x)}{\partial x^j}\right|_{x_i} = \sum_{k=1}^{l} \alpha_k \exp\left(-\frac{1}{2\sigma^2}\|t_k - x_i\|^2\right) \frac{1}{\sigma^2}(t_k^j - x_i^j)$$

and so

$$\left(\left.\frac{\partial f(x)}{\partial x^j}\right|_{x_i}\right)^2 = \sum_{k_1,k_2} \alpha_{k_1}\alpha_{k_2} \exp\left(-\frac{1}{2\sigma^2}\left(\|t_{k_1} - x_i\|^2 + \|t_{k_2} - x_i\|^2\right)\right) \cdot \frac{1}{\sigma^4}(t_{k_1}^j - x_i^j)(t_{k_2}^j - x_i^j).$$

On the other hand, we have by definition that

$$\begin{aligned} \|f\|^2 &= \sum_{k_1,k_2} \alpha_{k_1}\alpha_{k_2} \exp\left(-\frac{1}{2\sigma^2}\|t_{k_1} - t_{k_2}\|^2\right) \\ &= \sum_{k_1,k_2} \alpha_{k_1}\alpha_{k_2} \exp\left(-\frac{1}{2\sigma^2}\left(\|t_{k_1} - x_i\|^2 + \|t_{k_2} - x_i\|^2\right)\right) \\ &\qquad \cdot \exp\left(\frac{1}{\sigma^2}(t_{k_1}^j - x_i^j)^T(t_{k_2}^j - x_i^j)\right). \end{aligned}$$

where we have used the identity

$$\|t_{k_1} - t_{k_2}\|^2 = \|t_{k_1} - x_i\|^2 + \|t_{k_2} - x_i\|^2 - 2(t_{k_1} - x_i)^T(t_{k_2} - x_i).$$

If $\|f\|^2 = 0$, we have the zero function which has zero gradient as well. Suppose $\|f\|^2 > 0$. We will find values of c such that

$$\left(\frac{\partial f(x)}{\partial x^j}\bigg|_{x_i} \right)^2 \leq c\|f\|^2.$$

Substituting the above expansions into the inequality, we get

$$\sum_{k_1,k_2} \alpha_{k_1}\alpha_{k_2} \exp\left(-\frac{1}{2\sigma^2}\left(\|t_{k_1} - x_i\|^2 + \|t_{k_2} - x_i\|^2 \right) \right)$$
$$\cdot \left(c\exp\left(\frac{1}{\sigma^2}(t^j_{k_1} - x^j_i)^T(t^j_{k_2} - x^j_i) \right) - \frac{1}{\sigma^4}(t^j_{k_1} - x^j_i)(t^j_{k_2} - x^j_i) \right) \geq 0$$

Let $\beta_k = \alpha_k \exp\left(-\frac{1}{2\sigma^2}\|t_k - x_i\|^2\right)$ and $u_k = \frac{t_k - x_i}{\sigma}$. The above simplifies to

$$\sum_{k_1,k_2} \beta_{k_1}\beta_{k_2} \left(c\exp\left(u^T_{k_1}u_{k_2}\right) - \frac{1}{\sigma^2}u^j_{k_1}u^j_{k_2} \right) \geq 0. \tag{2}$$

We now show that when $c \geq 1/\sigma^2$ the expression within the parentheses above, as a function of u_{k_1} and u_{k_2}, is a positive definite kernel. As a result the inequality of (2) will hold for any β_k and c is an upper bound on the norm of the linear functional $\frac{\partial f(x)}{\partial x^j}\Big|_{x_i}$.

Expanding the exponential term using its Taylor series, we get

$$\begin{aligned}
&c\exp(u^T_{k_1}u_{k_2}) - \frac{1}{\sigma^2}u^j_{k_1}u^j_{k_2} \\
=&c + cu^T_{k_1}u_{k_2} - \frac{1}{\sigma^2}u^j_{k_1}u^j_{k_2} + \frac{c}{2!}(u^T_{k_1}u_{k_2})^2 + \frac{c}{3!}(u^T_{k_1}u_{k_2})^3 + \cdots \\
=&c + \left(c - \frac{1}{\sigma^2} \right) u^T_{k_1}u_{k_2} + \frac{1}{\sigma^2}\sum_{j' \neq j} u^{j'}_{k_1}u^{j'}_{k_2} + \frac{c}{2!}(u^T_{k_1}u_{k_2})^2 + \frac{c}{3!}(u^T_{k_1}u_{k_2})^3 + \cdots
\end{aligned}$$

Since $c \geq 1/\sigma^2$, all the coefficients are positive. Each of the terms $(u^T_{k_1}u_{k_2})^n$ is a homogeneous polynomial kernel hence is positive definite. The $\sum_{j' \neq j} u^{j'}_{k_1}u^{j'}_{k_2}$ term is positive definite as well. Since the sum converges, by Lemma 1, the kernel in the parentheses of (2) is a positive definite kernel.

In summary, we have shown that

$$\left| \frac{\partial f(x)}{\partial x^j}\bigg|_{x_i} \right| \leq \frac{1}{\sigma}\|f(x)\|$$

for all

$$f(x) = \sum_{k=1}^{l} \alpha_k \exp\left(-\frac{1}{2\sigma^2}\|t_k - x\|^2 \right).$$

Since the number of terms l is arbitrary, we can extend the linear functional to the Hilbert space while retaining the norm, thus $||L_{x_{i,j}}|| \leq 1/\sigma$.

Next, we show that $\|L_{x_{i,j}}\| \geq 1/\sigma$, so necessarily $\|L_{x_{i,j}}\| = 1/\sigma$. Without loss of generality, suppose $j = 1$ and $c < 1/\sigma^2$. We consider a simple family of f with $l = 2$, and $u_1 = (x, 0, \ldots, 0)^T$ and $u_2 = (-x, 0, \ldots, 0)^T$. The left hand side of (2) is

$$\left(c \exp\left(x^2\right) - \frac{1}{\sigma^2}x^2\right)\beta_1^2 + 2\left(c \exp\left(-x^2\right) + \frac{1}{\sigma^2}x^2\right)\beta_1\beta_2 \\ + \left(c \exp\left(x^2\right) - \frac{1}{\sigma^2}x^2\right)\beta_2^2 \tag{3}$$

Treated as a quadratic function in (β_1, β_2), the discriminant is

$$4c^2\left(\exp\left(-x^2\right) + \exp\left(x^2\right)\right)\left(\exp\left(-x^2\right) - \exp\left(x^2\right) + \frac{2}{c\sigma^2}x^2\right).$$

When $x = 0$, the discriminant is 0. Denote $r(x) = \exp\left(-x^2\right) - \exp\left(x^2\right) + \frac{2}{c\sigma^2}x^2$, then the derivative $r'(x) = 2x\left(-\exp\left(-x^2\right) - \exp\left(x^2\right) + \frac{2}{c\sigma^2}\right)$. As $c < 1/\sigma^2$, there exists $\delta > 0$, such that $r'(x) > 0$ for all $x \in (0,\ \delta)$. Thus $r(x)$ and the discriminant are positive for $x \in (0,\ \delta)$, in which case the function (3) crosses 0. So there are values of β_1, β_2 such that (3) is less than 0, contradicting the inequality (2). Therefore c cannot be less than $1/\sigma^2$ and $||L_{x_{i,j}}|| \geq 1/\sigma$. $\square$

Finally we can evaluate the representer of the derivative functional for the Gaussian kernel $k(x, y) = \exp(-\frac{1}{2\sigma^2}\|x - y\|^2)$,

$$z_{x_{i,j}}(x) = \frac{1}{\sigma^2}(x^j - x_i^j)\exp(-\frac{1}{2\sigma^2}\|x - x_i\|^2).$$

We also have

$$\langle z_{x_{i,j}}, z_{x_{p,q}}\rangle = \begin{cases} -\frac{1}{\sigma^4}(x_i^j - x_p^j)(x_i^q - x_p^q)\exp(-\frac{1}{2\sigma^2}\|x_i - x_p\|^2) & \text{if } j \neq q \text{ and} \\ \frac{1}{\sigma^4}(\sigma^2 - (x_i^j - x_p^j)^2)\exp(-\frac{1}{2\sigma^2}\|x_i - x_p\|^2) & \text{if } j = q. \end{cases}$$

4.2 Transformation Invariance

Invariance to known local transformations of input has been used successfully in supervised learning [7]. Here we show that transformation invariance can be handled in our framework for semi-supervised learning in reproducing kernel Hilbert spaces, by showing that gradients with respect to the transformations are bounded linear functionals.

We require a differentiable function g that maps points from a space X to $\mathbb{R}$. Next, we consider a family of bijective transformations $t_\alpha : X \mapsto X$, parametrized by α and differentiable in both α and X. We use t_α to define a family of operators $s(g, \alpha) = g \circ t_\alpha^{-1}$ that takes in a function g and outputs another function. Finally, we sample a fixed number of locations in X to obtain a vector, $\boldsymbol{g}$, that is presented to the learning algorithm.

Example 3. As an example, consider an image g to be a function that maps points in the plane $\mathbb{R}^2$ to the intensity of the image at that point. Digital images are discretization of real images where we sample at fixed pixel locations of the function g to obtain a fixed sized vector. In practice, given the digital image, the image function is reconstructed by convolving it with a two dimensional Gaussian function [7]. Translation can now be represented as an operator t_α:

$$t_\alpha : \begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} x + \alpha_x \\ y + \alpha_y \end{pmatrix}.$$

The function $s(g, \alpha)(x, y)$ gives us the intensity at location (x, y) of the image translated by an amount (α_x, α_y). Finally the translated image is digitized by evaluating $s(g, \alpha)$ at the same set of pixel locations. Notice that for a fixed g, the digital image, $\boldsymbol{g}$, is a vector valued function of α.

The following result allows us to use derivatives with respect to each of the parameters in α within the framework.

Theorem 6. *The derivatives* $\frac{\partial f(\boldsymbol{g}(\alpha))}{\partial \alpha_i}\Big|_{\alpha=0}$ *with respect to each of the parameters in* α *are bounded linear functionals when derivatives with respect to each component of the vector function* g *is a bounded linear functional.*

Proof. Using the chain rule and the fact that the derivatives with respect to each component of the vector function is a bounded linear functional, we have

$$\begin{aligned}
|L_{g,i}(f)| &= \left| \frac{\partial f(\boldsymbol{g}(\alpha))}{\partial \alpha_i}\Big|_{\alpha=0} \right| \\
&= \left| \sum_{j=1}^{J} \frac{\partial f(\boldsymbol{g}(\alpha))}{\partial \boldsymbol{g}^j(\alpha)} \frac{\partial \boldsymbol{g}^j(\alpha)}{\partial \alpha_i}\Big|_{\alpha=0} \right| \\
&\leq \sum_{j=1}^{J} \left| \frac{\partial \boldsymbol{g}^j(\alpha)}{\partial \alpha_i}\Big|_{\alpha=0} \right| \left| \frac{\partial f(\boldsymbol{g}(\alpha))}{\partial \boldsymbol{g}^j(\alpha)}\Big|_{\alpha=0} \right| \\
&\leq \sum_{j=1}^{J} \left| \frac{\partial \boldsymbol{g}^j(\alpha)}{\partial \alpha_i}\Big|_{\alpha=0} \right| C_j \|f\|. \\
&= C\|f\|.
\end{aligned}$$

□

Corollary 2. *The derivatives* $\frac{\partial f(\boldsymbol{g}(\alpha))}{\partial \alpha_i}\Big|_{\alpha=0}$ *with respect to each of the parameters in* α *are bounded linear functionals in the reproducing kernel Hilbert spaces defined by the polynomial and Gaussian kernels.*

4.3 Local Averaging

Using gradients to enforce the local invariance that the target function does not change much around data instances increases the number of basis functions by a factor of d

where d is the number of gradient directions that we use. The optimization problem can become computationally expensive if d is large. When we do not have useful information about the invariant directions, it may be useful to have methods that do not increase the number of basis functions by much. We consider linear functionals

$$L_{x_i}(f) = \int_X f(\tau)p(x_i - \tau)d\tau - f(x_i)$$

where $p(\cdot)$ is a probability density function centred at zero. Minimizing a loss with such linear functionals will favour functions whose local averages given by the integral are close to the function values at data instances. If $p(\cdot)$ is selected to be a low pass filter, the function should be smoother and less likely to change in regions with more data points but is less constrained to be smooth in regions where the data points are sparse. Hence, such loss functions may be appropriate when we believe that data instances from the same class are clustered together.

To use the framework we have developed, we need to select the probability density $p(\cdot)$ and the kernel k such that $L_{x_i}(f)$ is a bounded linear functional. In addition, for efficient implementation, we require that the linear functional be efficiently evaluated. We show that the Gaussian kernel together with the Gaussian density function satisfies the required properties.

Theorem 7. *The linear functional $L_{x_i}(f) = \int_X f(\tau)p(x_i-\tau)d\tau - f(x_i)$ is bounded in the reproducing kernel Hilbert space defined by a Gaussian kernel for any probability density $p(\cdot)$.*

Proof. Using the reproducing kernel property and the Cauchy-Schwarz inequality,

$$|f(x)| = |\langle f, k(x,\cdot)\rangle| \leq \|f\|\|k(x,\cdot)\| = \|f\|k(x,x)^{1/2} = \|f\|.$$

Now we have

$$\begin{aligned} L_{x_i}(f) &= \int_X f(\tau)p(x_i - \tau)d\tau - f(x_i) \\ &\leq \int_X |f(\tau)|p(x_i - \tau)d\tau + \|f\| \\ &\leq \|f\| \int_X p(x_i - \tau)d\tau + \|f\| \\ &= 2\|f\|. \end{aligned}$$

□

To complete all the calculations efficiently, we need to be able to efficiently evaluate $z_{x_i}(x) = \langle z_{x_i}, k(x,\cdot)\rangle = L_{x_i}(k(x,\cdot))$ and $\langle z_{x_i}, z_{x_j}\rangle = L_{x_i} z_{x_j}$ where z_{x_i} is the representer of the local average functional L_{x_i}. Recall that the Gaussian kernel is defined as $k(x_1, x_2) = \exp\left(-\frac{1}{2\sigma_k^2}\|x_1 - x_2\|^2\right)$ while the Gaussian density is $p(x) = \frac{1}{(2\pi)^{d/2}\sigma_p^d} \exp\left(-\frac{1}{2\sigma_p^2}\|x\|^2\right)$. Since the convolution of a Gaussian density with another

Gaussian density is a Gaussian density, we have

$$\begin{aligned}
z_{x_i}(x) &= L_{x_i}(k(x,\cdot)) \\
&= \int_X k(x,\tau)p(x_i-\tau)d\tau - k(x_i,x) \\
&= (2\pi)^{d/2}\sigma_k^d \int_X \frac{1}{(2\pi)^{d/2}\sigma_k^d} \exp\left(-\frac{1}{2\sigma_k^2}\|x-\tau\|^2\right) \\
&\qquad \frac{1}{(2\pi)^{d/2}\sigma_p^d} \exp\left(-\frac{1}{2\sigma_p^2}\|x_i-\tau\|^2\right) d\tau - k(x_i,x) \\
&= \frac{\sigma_k^d}{(\sigma_k+\sigma_p)^d} \exp\left(-\frac{1}{2(\sigma_k+\sigma_p)^2}\|x_i-x\|^2\right) - \exp\left(-\frac{1}{2\sigma_k^2}\|x_i-x\|^2\right)
\end{aligned}$$

Finally, $\langle z_{x_i}, z_{x_j}\rangle = L_{x_i}(z_{x_j})$ can also be efficiently evaluated as

$$\begin{aligned}
L_{x_i}z_{x_j} =& (2\pi)^{d/2}\sigma_k^d \int_X \left[\frac{1}{(2\pi)^{d/2}(\sigma_k+\sigma_p)^d} \exp\left(-\frac{1}{2(\sigma_k+\sigma_p)^2}\|x_j-x\|^2\right)\right. \\
&\qquad \left. - \frac{1}{(2\pi)^{d/2}\sigma_k^d} \exp\left(-\frac{1}{2\sigma_k^2}\|x_j-x\|^2\right)\right] \\
&\qquad \frac{1}{(2\pi)^{d/2}\sigma_p^d} \exp\left(-\frac{1}{2\sigma_p^2}\|x_i-\tau\|^2\right) d\tau - z_{x_j}(x_i) \\
=& \frac{\sigma_k^d}{(\sigma_k+2\sigma_p)^d} \exp\left(-\frac{1}{2(\sigma_k+2\sigma_p)^2}\|x_i-x_j\|^2\right) \\
&- \frac{\sigma_k^d}{(\sigma_k+\sigma_p)^d} \exp\left(-\frac{1}{2(\sigma_k+\sigma_p)^2}\|x_i-x_j\|^2\right) - z_{x_j}(x_i)
\end{aligned}$$

5 Experimental Result

We experimented with the "two moon" dataset shown in Figure 1, with only two labeled data instances (red circle and blue square). We used the Gaussian kernel with $\sigma = 0.12$ and the gradient invariances. Equation (1) is minimized with the hinge loss for l_2 and the ϵ-insensitive loss for l_1. This quadratic programming problem can be reformulated into its dual which has only bound constraints, allowing it to be solved more efficiently as follows: minimize

$$\begin{aligned}
&\frac{1}{2}\sum_{i,j=l+1}^{l+n} p_{ij}\left(\alpha_i'-\alpha_i\right)\left(\alpha_j'-\alpha_j\right) + \frac{1}{2}\sum_{i,j=1}^{l} p_{ij}y_iy_j\beta_i\beta_j + \sum_{i=l+1}^{l+n}\sum_{j=1}^{l} p_{ij}y_j\left(\alpha_i'-\alpha_i\right)\beta_j \\
&\quad + \epsilon\sum_{i=l+1}^{l+n}\left(\alpha_i'+\alpha_i\right) - \sum_{i=1}^{l}\beta_i
\end{aligned}$$

subject to $\alpha_i, \alpha_i' \in [0, \rho_2/(2\rho_1)], \beta_j \in [0, 1/(2\rho_1)]$, for all $i = l+1, \ldots, l+n$, and $j = 1, \ldots, l$, where p_{ij} are $k(x_i, x_j)$, $z_i(x_j)$, $z_j(x_i)$ or $\langle z_i(\cdot), z_j(\cdot)\rangle$ depending on (i, j).

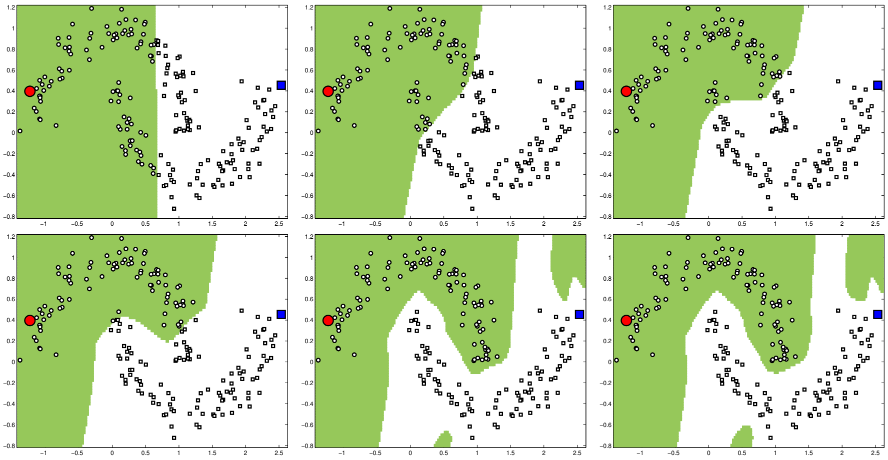

Fig. 1. The decision boundary for the two moon dataset with two labeled instances, $\rho_2 = 0$ (top left), $\rho_2 = 0.001$ (top center), $\rho_2 = 0.005$ (top right), $\rho_2 = 0.01$ (bottom left), $\rho_2 = 0.1$ (bottom center) and $\rho_2 = 10$ (bottom right).

We use $\rho_1 = 1$, $\epsilon = 0.001$ and $\rho_2 \in \{0, 0.001, 0.005, 0.01, 0.1, 10\}$ in the experiment. The proportion of basis functions with non-zero coefficients are 0.5, 36.6, 50.9, 61.7, 77.4, and 74.1 percent respectively. We also tried the squared loss for both l_1 and l_2 with similar classification results. The result shows that the method can be made to work in this simple case. Further empirical work with real datasets is required before we can tell whether the method is practically interesting.

6 Discussion

Another example utilizing loss functions on linear functionals that fits within the framework that we are using is the graph Laplacian regularizer [1]. The graph Laplacian regularizer can be written as $\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}(f(x_i) - f(x_j))^2$ where w_{ij} is a measure of similarity between x_i and x_j. If we define the linear functional as $\langle f, k(x_i, \cdot) - k(x_j, \cdot)\rangle$ and loss function $w_{ij}(\cdot)^2$ for each (x_i, x_j) pair, the optimization problem can be put into the same form as the other examples in this paper. However, it is simpler to use the usual representer theorem [6] for this problem, as we only use the values of the functions in the optimization problem.

It is interesting to ask whether the invariances used in this paper are better exploited by using global invariance constraints on the function class, thus eliminating the need for unlabeled data. We note that enforcing global invariance can degrade the discrimination power of the function class, for example, the digit '6' would be indistinguishable from the digit '9' under global rotational invariance [7]. In contrast, local invariances does not prevent the function from changing in regions of low data density.

Enforcing the local invariance constraints in reproducing kernel Hilbert spaces is also likely to be helpful in the case of supervised learning, where no additional unlabeled data is available. An example of this is provided in [7] where translation, rotation and various other constraints were used in supervised learning for handwritten digit recognition using neural networks in a framework similar to that used in this paper. Another potential application of the techniques in this paper is to the problem of imbalanced data learning [9], where we may wish to keep the decision boundary further away from instances of the minority class and closer to the instances of majority class.

Local transformation invariances such as translation and rotation invariances are widely used in visual tasks. Finding invariances that are useful for other application domains, e.g., language processing, would also be interesting.

References

1. M. Belkin, P. Niyogi, and V. Sindhwani. On manifold regularization. In *AISTATS*, 2005.
2. Avrim Blum and Shuchi Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proc. 18th International Conf. on Machine Learning*, pages 19–26. Morgan Kaufmann, San Francisco, CA, 2001.
3. Thorsten Joachims. Transductive inference for text classification using support vector machines. In *Proc. 16th International Conf. on Machine Learning*, pages 200–209. Morgan Kaufmann, San Francisco, CA, 1999.
4. Thorsten Joachims. Transductive learning via spectral graph partitioning. In *Proc. 20th International Conf. on Machine Learning*, 2003.
5. Erwin Kreyszig. *Introductory Functional Analysis with Applications*. Wiley, 1989.
6. Bernhard Schölkopf and Alex Smola. *Learning with Kernels*. MIT Press, 2001.
7. Patrice Simard, Yann LeCun, John S. Denker, and Bernard Victorri. Transformation invariance in pattern recognition-tangent distance and tangent propagation. In *Neural Networks: Tricks of the Trade*, pages 239–27, 1996.
8. Grace Wahba. An introduction to model building with reproducing kernel Hilbert spaces. Statistics Department TR 1020, University of Wisconsin-Madison, 2000.
9. Gang Wu and Edward Y. Chang. Adaptive feature-space conformal transformation for imbalanced-data learning. In *Proc. 20th International Conf. on Machine Learning*, pages 816–823, 2003.
10. Tong Zhang and Frank J. Oles. A probability analysis on the value of unlabeled data for classification problems. In *Proc. 17th International Conf. on Machine Learning*, pages 1191–1198. Morgan Kaufmann, San Francisco, CA, 2000.
11. Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *International Conference on Machine Learning*, pages 912–919, 2003.