

A Kernel for Semi-Supervised Learning With Multi-View Point Cloud Regularization

David S. Rosenberg, Vikas Sindhwani, Peter L. Bartlett, Partha Niyogi

May 29, 2009

SCOPE

In *semi-supervised learning* (SSL), we learn a predictive model from a collection of labeled data and a typically much larger collection of unlabeled data. These lecture notes present a framework called multi-view point cloud regularization (MVPCR) [5], which unifies and generalizes several semi-supervised kernel methods that are based on data-dependent regularization in reproducing kernel Hilbert spaces (RKHS). Special cases of MVPCR include co-regularized least squares (CoRLS) [7, 3, 6], manifold regularization (MR) [1, 8, 4], and graph-based semi-supervised learning. An accompanying theorem shows how to reduce any MVPCR problem to standard supervised learning with a new *multi-view kernel*.

RELEVANCE

RKHS techniques form the basis of many state-of-the-art supervised learning algorithms, such as support vector machines (SVMs), kernel ridge regression, and Gaussian processes. By plugging the new multi-view kernel into these, or any other standard kernel method, we can conveniently convert them to semi-supervised learning algorithms. Via the reduction of MVPCR to supervised RKHS learning, we can easily derive generalization error bounds using standard results. In particular, we generalize the bound given in [6] for CoRLS. From an experimental perspective, there are many interesting algorithms that fit into the MVPCR framework that have yet to be explored. As one example, we present *manifold co-regularization*, which directly combines the ideas in CoRLS and MR.

PROBLEM SETTING

We begin with some learning theory. For any *input* x from an input space \mathcal{X} , we suppose there is some *target* $y \in \mathbf{R}$ that we would like to predict. The “loss” incurred when we predict \hat{y} rather than y is given by a non-negative *loss function* $V(\hat{y}, y)$. Our goal is to find a prediction function whose average loss, or *risk*, is small. More formally, we assume that (x, y) pairs are drawn from a distribution $P_{\mathcal{X} \times \mathcal{Y}}$, and the risk of f is defined as the expected loss on a random pair: $R(f) = \mathbb{E}V(f(X), Y)$. Although we would like to minimize $R(f)$, in practice we cannot even compute it since $P_{\mathcal{X} \times \mathcal{Y}}$ is unknown. Instead, suppose we have a *training set* of pairs $(x_1, y_1), \dots, (x_\ell, y_\ell)$ sampled independently from $P_{\mathcal{X} \times \mathcal{Y}}$. Then we can minimize the *empirical risk*, which is defined as $\hat{R}_\ell(f) = \frac{1}{\ell} \sum_{i=1}^\ell V(f(x_i), y_i)$. By the law of large numbers, $\lim_{\ell \rightarrow \infty} \hat{R}_\ell(f) = R(f)$ with probability 1, so this is a plausible substitute. However, the *minimizer* of $\hat{R}_\ell(f)$ is not guaranteed to converge to the minimizer of $R(f)$ without additional constraints on the set of functions over which we are minimizing. Thus we constrain our minimization to some class of functions \mathcal{F} , and define the *empirical risk minimizer* and *risk minimizer*, respectively, by $\hat{f}_\ell = \arg \min_{f \in \mathcal{F}} \hat{R}_\ell(f)$ and $f_* = \arg \min_{f \in \mathcal{F}} R(f)$. For many \mathcal{F} , we will have $R(\hat{f}_\ell) \rightarrow R(f_*)$. With a finite training set, however, there is an inevitable gap between the risk of \hat{f}_ℓ and the risk of f_* . This gap is called *estimation error*, since \hat{f}_ℓ is only an “estimate” of the unknown function f_* . The speed with which the estimation error converges to zero is governed, in part, by the size of the class \mathcal{F} , with smaller classes giving faster convergence. As the ultimate performance benchmark, we consider the *Bayes prediction function*, defined as $y_* = \arg \min_f R(f)$, which minimizes the risk over all functions. The difference in risk between y_* (the best overall) and f_* (the best in \mathcal{F}) is called the *approximation error*. We can decompose the *excess risk* that \hat{f}_ℓ has over y_* using these two types of error:

$$R(\hat{f}_\ell) - R(y_*) = \underbrace{R(\hat{f}_\ell) - R(f_*)}_{\text{estimation error}} + \underbrace{R(f_*) - R(y_*)}_{\text{approximation error}}.$$

In practice, a convenient way to adjust the balance between approximation and estimation error is to use *Tikhonov regularization*, in which we solve $f_* = \arg \min_{f \in \mathcal{F}} \hat{R}_\ell(f) + \gamma \Omega(f)$, for some $\gamma > 0$ and some non-negative penalty function Ω . As γ increases, f_* is pulled towards a minimizer of $\Omega(f)$, which effectively limits the domain of optimization. Generally speaking, increasing γ will increase approximation error and decrease estimation error. Many popular

learning algorithms, including the SVM and kernel ridge regression, are Tikhonov regularization problems for which \mathcal{F} is an RKHS. An RKHS of functions from \mathcal{X} to \mathbf{R} is a Hilbert Space \mathcal{H} with a reproducing kernel, i.e. a function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbf{R}$ for which the following properties hold: (a) $k(x, \cdot) \in \mathcal{H}$ for all $x \in \mathcal{X}$, and (b) $\langle f, k(x, \cdot) \rangle_{\mathcal{H}} = f(x)$, for all $x \in \mathcal{X}$ and $f \in \mathcal{H}$. The most basic RKHS regularization problem is

$$\hat{f}_{\ell} = \arg \min_{f \in \mathcal{H}} \left[\hat{R}_{\ell}(f) + \gamma \|f\|_{\mathcal{H}}^2 \right]. \quad (1)$$

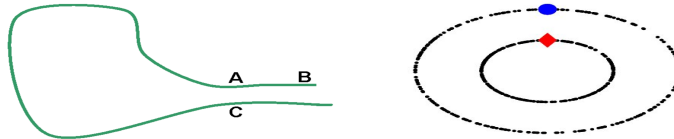
By the representer theorem, $\hat{f}_{\ell}(x) = \sum_{i=1}^{\ell} \alpha_i k(x, x_i)$ for some $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^{\ell}$, and thus (1) reduces to a finite-dimensional optimization over $\boldsymbol{\alpha}$. For the square loss, the optimal $\boldsymbol{\alpha}$ is a solution to $(K + \gamma I)\boldsymbol{\alpha} = \mathbf{y}$, where K is the kernel matrix defined by $K_{ij} = k(x_i, x_j)$, I is the identity matrix, and \mathbf{y} is the vector of training labels. We now discuss extensions of Tikhonov Regularization that are based on semi-supervised learning assumptions.

Manifold Smoothness and Cluster Assumptions: The input space \mathcal{X} often has a natural distance metric, such as the Euclidean distance when $\mathcal{X} = \mathbf{R}^d$. However, the input points themselves often suggest a different metric. For instance, suppose the input points lie on a one-dimensional manifold, as shown in Fig. 1(a). While the points A and C are close in Euclidean distance, they are far apart along the manifold. In Fig. 1(b), the manifold has two disjoint components, and while points on different components may be close in Euclidean distance, they are infinitely far apart along the manifold. The idea that the input distribution $P_{\mathcal{X}}$ may live on a low-dimensional manifold in \mathcal{X} is supported by many real-world problems. For example, in speech production, the articulatory organs can be modeled as a collection of tubes whose lengths and widths smoothly parameterize the low-dimensional manifold of speech signals. In vision, the images we get when viewing an object from different positions in \mathbf{R}^3 form a 3-dimensional submanifold in image space. The *manifold smoothness* assumption in semi-supervised learning (SSL) is that f_* is “smooth” with respect to the manifold underlying $P_{\mathcal{X}}$. Although we don’t generally know $P_{\mathcal{X}}$, in SSL we have a “point cloud” x_1, \dots, x_n sampled from $P_{\mathcal{X}}$. The intrinsic neighborhood structure of the manifold is approximated by the nearest neighbor graph on the point cloud. Let W be the adjacency matrix and define $\Omega_{\mathcal{X}}(f) = \frac{1}{2} \sum_{i,j} W_{ij} (f(x_i) - f(x_j))^2$. We can write this intrinsic smoothness measure as a quadratic form with the *Laplacian* matrix L of the graph, i.e. $\Omega_{\mathcal{X}}(f) = \mathbf{f}^T L \mathbf{f}$, where $\mathbf{f} = (f(x_1) \dots f(x_n))^T$ and $L = D - W$, where D is the diagonal degree matrix $D_{ii} = \sum_j W_{ij}$. We attain the *manifold regularization* (MR) algorithm

by adding $\Omega_{\mathcal{I}}(f)$ to the objective function of Eqn. (1):

$$\hat{f}_{\ell} = \arg \min_{f \in \mathcal{H}} \hat{R}_{\ell}(f) + \gamma \|f\|_{\mathcal{H}}^2 + \gamma_{\mathcal{I}} \Omega_{\mathcal{I}}(f). \quad (2)$$

As we make $\gamma_{\mathcal{I}}$ large, we push \hat{f}_{ℓ} towards the region of \mathcal{H} with small $\Omega_{\mathcal{I}}(f)$, i.e. towards those functions with high intrinsic smoothness. As we restrict \hat{f}_{ℓ} to a subset of \mathcal{H} , we reduce estimation error. If our manifold smoothness assumption is correct, then $\Omega_{\mathcal{I}}(f_*)$ should also be small, and the restriction will not increase approximation error.



(a) 1 connected component. (b) 2 connected components.

Multi-View Assumptions: In the multi-view approach to SSL, we have several classes of prediction functions, or “views.” This terminology arises in contexts where an input $x \in \mathcal{X}$ can be decomposed naturally as $x = (x^1, \dots, x^m)$, where each x^i represents a different “view” of the input x . With this decomposition of the input vector, we can define m views, where the i th view is a class of functions depending only on x^i and ignoring the other components of x . For example, suppose an input x is a clip from a video of a conference room. We divide x into an audio stream and a video stream, which we write as $x = (x^{\text{aud}}, x^{\text{vid}})$. Define our first view \mathcal{F}^{aud} to consist of prediction functions of the form $x \mapsto f(x^{\text{aud}})$, and our second view, \mathcal{F}^{vid} , to consist of functions of the form $x \mapsto f(x^{\text{vid}})$. Suppose the goal is to identify who is speaking in each video clip. Although it is certainly easier to identify who is speaking by using the x^{aud} and x^{vid} signals together, a person who is familiar with the voices and appearances of the individuals in the conference room could do quite well with just one of these signals. Thus it is reasonable to *assume that each of our views, both \mathcal{F}^{aud} and \mathcal{F}^{vid} , contains a function that makes the correct predictions.* Now suppose we have a very limited amount of training data, and the only time that Bob spoke, there was an accompanying sound of a truck passing outside in the audio stream x^{aud} , but no corresponding signal in the video track x^{vid} . Without additional information, it would be difficult to rule out a prediction function $f_{\text{bad}}^{\text{aud}} \in \mathcal{F}^{\text{aud}}$ that identifies Bob as the speaker whenever a truck passes. However, there is no evidence for a truck passing in the video signal, and thus there is no function in \mathcal{F}^{vid} that can consistently make the

same predictions as $f_{\text{bad}}^{\text{aud}}$. Since we assumed that each view contains a function that makes the correct predictions, and \mathcal{F}^{vid} does not contain any function that matches $f_{\text{bad}}^{\text{aud}}$, we can conclude that $f_{\text{bad}}^{\text{aud}}$ does not make the correct predictions. Thus by using the assumption that each view has a good function, we can prune out functions, such as $f_{\text{bad}}^{\text{aud}}$, that fit the training data but will not perform well in general. To effect this pruning in practice, we introduce a *co-regularization* function $\Omega_c(f^1, f^2)$ that measures the disagreement between f^1 and f^2 . Then, we solve

$$(\hat{f}_\ell^1, \hat{f}_\ell^2) = \arg \min_{f^1 \in \mathcal{H}^1, f^2 \in \mathcal{H}^2} \hat{R}_\ell \left(\frac{1}{2}(f^1 + f^2) \right) + \gamma_1 \|f^1\|_{\mathcal{H}^1}^2 + \gamma_2 \|f^2\|_{\mathcal{H}^2}^2 + \lambda \Omega_c(f^1, f^2), \quad (3)$$

for RKHSs \mathcal{H}^1 and \mathcal{H}^2 . The final prediction function is $\hat{\varphi}_\ell(x) = (\hat{f}_\ell^1(x) + \hat{f}_\ell^2(x))/2$. Taking $\Omega_c(f^1, f^2) = \sum_{i=1}^n (f^1(x_i) - f^2(x_i))^2$, we get the CoRLS [7, 3, 6] algorithm.

SOLUTION

Multi-View Point Cloud Regularization: We now consider a generalized Tikhonov regularization framework that subsumes the methods discussed above. Our views are RKHSs $\mathcal{H}^1, \dots, \mathcal{H}^m$ of real-valued functions on \mathcal{X} , with kernels k^1, \dots, k^m , respectively. Define $\mathcal{F} = \mathcal{H}^1 \times \dots \times \mathcal{H}^m$. We want to select one function from each view, say $f = (f^1, \dots, f^m) \in \mathcal{F}$, and to combine these functions into a single prediction function. We fix a vector of *view weights* $\mathbf{a} = (a_1, \dots, a_m) \in \mathbf{R}^m$, and define $u(f) = a_1 f^1 + \dots + a_m f^m$. The final prediction function is $\varphi(x) = u(f)(x) = a_1 f^1(x) + \dots + a_m f^m(x)$. We define the space of these prediction functions by $\tilde{\mathcal{H}} = u(\mathcal{F})$. Note that $\tilde{\mathcal{H}}$ may change with different settings of \mathbf{a} , in particular when entries of \mathbf{a} are set to zero. For any $f \in \mathcal{F}$, we denote the column vector of function evaluations on the point cloud by¹ $\underline{\mathbf{f}} = (f^1(x_1), \dots, f^1(x_n), \dots, f^m(x_1), \dots, f^m(x_n))^T \in \mathbf{R}^{mn}$. For any positive semidefinite (PSD) matrix $M \in \mathbf{R}^{mn \times mn}$, the objective function for MVPCR is

$$\arg \min_{\varphi \in \tilde{\mathcal{H}}} \min_{\{(f^1, \dots, f^m) : a_1 f^1 + \dots + a_m f^m = \varphi\}} \hat{R}_\ell(\varphi) + \sum_{i=1}^m \gamma_i \|f^i\|_{\mathcal{H}^i}^2 + \lambda \underline{\mathbf{f}}^T M \underline{\mathbf{f}}, \quad (4)$$

where $\gamma_1, \dots, \gamma_m > 0$ are RKHS norm regularization parameters, and $\lambda \geq 0$ is the point cloud norm regularization parameter. In the objective function above, $\tilde{\mathcal{H}}$ is a raw set of functions, without any additional structure. The main result of this paper endows $\tilde{\mathcal{H}}$ with an RKHS structure:

¹For the rest of this paper, we use bold face to indicate a finite dimensional column vector and an underline to indicate that a vector is the concatenation of a column vector associated with each of the m views.

Theorem 1. *There exists an inner product for which $\tilde{\mathcal{H}}$ is an RKHS with norm*

$$\|\varphi\|_{\tilde{\mathcal{H}}} = \sqrt{\min_{\{f: a_1 f^1 + \dots + a_m f^m = \varphi\}} \left[\sum_{i=1}^m \gamma_i \|f^i\|_{\mathcal{H}^i}^2 + \lambda \underline{\mathbf{f}}^T M \underline{\mathbf{f}} \right]} \quad (5)$$

and reproducing kernel function

$$\tilde{k}(z, x) = \sum_{j=1}^m \frac{a_j^2}{\gamma_j} k^j(z, x) - \lambda \underline{\mathbf{k}}_x^T \underline{\mathbf{A}} \underline{\mathbf{G}}^{-1} (I + \lambda M \underline{\mathbf{G}}^{-1} \mathcal{K})^{-1} M \underline{\mathbf{G}}^{-1} \underline{\mathbf{A}} \underline{\mathbf{k}}_z, \quad (6)$$

where we denote the point cloud kernel matrix for the j th view by $K^j = (k^j(x_i, x_k))_{i,k=1}^n$, \mathcal{K} is defined as the block diagonal matrix $\mathcal{K} = \text{diag}(K^1, \dots, K^m) \in \mathbf{R}^{mn \times mn}$,

$$\underline{\mathbf{A}} = \text{diag} \left(\underbrace{a_1, \dots, a_1}_{n \text{ times}}, \dots, \underbrace{a_m, \dots, a_m}_{n \text{ times}} \right) \quad \underline{\mathbf{G}} = \text{diag} \left(\underbrace{\gamma_1, \dots, \gamma_1}_{n \text{ times}}, \dots, \underbrace{\gamma_m, \dots, \gamma_m}_{n \text{ times}} \right),$$

and we denote the column vector of kernel evaluations between the point cloud and an arbitrary point $x \in \mathcal{X}$, for each kernel, by $\underline{\mathbf{k}}_x = (k^1(x_1, x), \dots, k^1(x_n, x), \dots, k^m(x_1, x), \dots, k^m(x_n, x))^T$.

For a proof, we point the reader to [5], where this theorem was first presented. We call the kernel given in Eqn. 6 the *multi-view kernel*. This theorem implies that the solution to the MVPCR problem in Eqn. (4) is exactly the solution to the standard RKHS regularization problem of Eqn. (1) over RKHS $\tilde{\mathcal{H}}$. Below, we use this reduction to derive complexity and generalization bounds for MVPCR as a consequence of well-known results for RKHS learning. This approach is much simpler than the “bare-hands” proof used for the special case of CoRLS in [6]. From an algorithmic perspective, since we have an explicit form for the multi-view kernel, we can easily plug it in to any standard kernel algorithm. For example, the kernel can be plugged into kernel logistic regression, Bayesian kernel methods such as Gaussian Processes, one-class SVMs, kernel PCA, etc., turning these algorithms into multi-view learners. We note that Thm. 1 generalizes the result of [8] for MR and of [9] for CoRLS.

Supervised Learning, MR, CoMR, and other Special Cases: It is easy to see that if we consider a single view, i.e. $m = a_1 = 1$, and set $\lambda = 0$, we get back the basic RKHS regularization problem of Eqn. (1). If we take $\lambda > 0$ and M to be the graph Laplacian, then we get back MR. To get CoRLS, we take $m = 2$, $a_1 = a_2 = 1/2$, and set M to the matrix,

$$M_c := \begin{pmatrix} I & -I \\ -I & I \end{pmatrix},$$

where each I is an $n \times n$ identity matrix. We can easily extend co-regularization to m views by taking M_c to be an $m \times m$ block matrix with $n \times n$ identity matrices on the diagonal, and $n \times n$ negative identity matrices off the diagonal. In particular, this recovers the multi-view generalization of co-regularization with least squares loss given in [3]. In [10], a kernel matrix is derived for co-regularized Gaussian processes. Their approach is transductive and does not provide predictions for points outside of the unlabeled training set. The multi-view kernel presented here (for $M = M_c$) not only recovers their kernel matrix when evaluated on the point cloud, but also possesses a natural out-of-sample extension to unseen data points. In addition, it generalizes to other loss functions, gives explicit control over view weights, and can incorporate more general data-dependent regularizers than the typical ℓ_2 -disagreement.

RADEMACHER COMPLEXITY AND GENERALIZATION BOUNDS

In the section “Problem Setting,” we discussed how we can trade off between approximation error and estimation error by changing the size of \mathcal{F} . We now discuss a precise measure of function class size. For a class of functions \mathcal{F} and a distribution on the domain \mathcal{X} , the *empirical Rademacher complexity* of \mathcal{F} for a sample $x_1, \dots, x_\ell \in \mathcal{X}$ is defined as $\hat{\mathfrak{R}}_\ell(\mathcal{F}) = \mathbb{E} \sup_{f \in \mathcal{F}} \frac{1}{\ell} \sum_{i=1}^{\ell} \sigma_i f(x_i)$, where the expectation is over the i.i.d. *Rademacher variables* $\sigma_1, \dots, \sigma_\ell$, which are distributed as $P(\sigma_i = 1) = P(\sigma_i = -1) = \frac{1}{2}$. For fixed σ_i ’s, the supremum selects the function $f \in \mathcal{F}$ that best “fits” the σ_i ’s, in the sense that when $\sigma_i = 1$, $f(x_i)$ is a large positive number, and when $\sigma_i = -1$, $f(x_i)$ is a large negative number. Thus if $\hat{\mathfrak{R}}_\ell(\mathcal{F})$ is large, then \mathcal{F} has functions that can fit most random noise sequences and may be prone to over-fitting the data. We make this statement precise with a well-known *generalization bound* [2], which bounds the worst case gap between risk and empirical risk in terms of $\hat{\mathfrak{R}}_\ell(\mathcal{F})$:

Theorem 2. *Suppose that the loss function $V(\cdot, y)$ is L -Lipschitz for every $y \in \mathbf{R}$ and $V(\cdot, \cdot) \in [a, b]$ for some $a < b$. Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$\sup_{f \in \mathcal{F}} \left| \hat{R}_\ell(f) - R(f) \right| \leq 2L\hat{\mathfrak{R}}_\ell(\mathcal{F}) + (b - a)\sqrt{\frac{2 \log(3/\delta)}{\ell}}.$$

We now derive bounds for MVPCR. It is straightforward to show that if $\hat{\varphi}_\ell$ is an MVPCR solution, then $\|\hat{\varphi}_\ell\|_{\tilde{\mathcal{H}}}^2 \leq r^2 := \hat{R}_\ell(0)$, where the 0 denotes the prediction function that always predicts 0. Thus we can consider the optimization in MVPCR to be over the norm ball $\tilde{\mathcal{H}}_r$ of radius r , rather than over all of $\tilde{\mathcal{H}}$. By a well-known result, $\hat{\mathfrak{R}}_\ell(\tilde{\mathcal{H}}_r) \leq \frac{r}{\ell} \sqrt{\text{tr } \tilde{K}}$, where \tilde{K} is the

kernel matrix for \tilde{k} on the labeled training points (c.f. [5] and references therein). To apply Thm. 2, $\tilde{\mathcal{H}}_r$ must be a fixed class of functions. However, in our setting $\tilde{\mathcal{H}}_r$ may be a random class of functions, even depending on the labeled data via the point cloud. Let us assume that the point cloud defining $\tilde{\mathcal{H}}$ is independent of the labeled data points. Then conditional on the point cloud, $\tilde{\mathcal{H}}_r$ is a deterministic class of functions. Plugging the bound on $\hat{\mathfrak{R}}_\ell(\tilde{\mathcal{H}}_r)$ into Thm. 2, we attain a generalization bound for any MVPCR algorithm. This also implies a bound on estimation error, since $R(\hat{f}_\ell) - R(f_*) \leq 2 \sup_{f \in \mathcal{F}} |\hat{R}_\ell(f) - R(f)|$, which is bounded by Thm. 2.

UNLABELED DATA IMPROVES THE BOUND

Here we present a result that shows how unlabeled data reduces the Rademacher complexity, and thus reduces the bound on estimation error, for the specific case of two-view CoRLS. Recall that the parameter λ controls the extent to which we enforce agreement between the prediction functions from each view: f^1 and f^2 . Let $\tilde{\mathcal{H}}(\lambda)$ denote the space of prediction functions for a particular value of λ . It has been shown in [9, 6] that the Rademacher complexity for a ball of radius r in $\tilde{\mathcal{H}}(\lambda)$ decreases with λ by an amount determined by $\Delta(\lambda) = \sum_{i=1}^{\ell} \rho^2(\gamma_1^{-1} \mathbf{k}_{Ux_i}^1, \gamma_2^{-1} \mathbf{k}_{Ux_i}^2)$, where $\mathbf{k}_{Ux_i}^1$ and $\mathbf{k}_{Ux_i}^2$ are $u \times 1$ column vectors whose j th entries are $k^1(x_i, x_{\ell+j})$ and $k^2(x_i, x_{\ell+j})$, respectively, and $\rho(\cdot, \cdot)$ is a metric on the space \mathbf{R}^u defined by $\rho^2(\mathbf{s}, \mathbf{t}) = \lambda(\mathbf{s} - \mathbf{t})'(I + \lambda S)^{-1}(\mathbf{s} - \mathbf{t})$, where $S = \gamma_1^{-1} K_{UU}^1 + \gamma_2^{-1} K_{UU}^2$ is a weighted sum of the unlabeled data kernel matrices. We see that the complexity reduction $\Delta(\lambda)$ grows with the ρ -distance between the two different (scaled) representations of the labeled points, where the measure of distance is determined by the unlabeled data.

MANIFOLD CO-REGULARIZATION

As an application of MVPCR, we present manifold co-regularization (CoMR), a multi-view version of MR. Roughly speaking, CoMR is two-view CoRLS with a particular choice of views. The *ambient view*, denoted by \mathcal{H}^A , is an RKHS of functions defined on the input space \mathcal{X} . As usual, the ‘‘smoothness’’ of a function $f \in \mathcal{H}^A$ is measured by the RKHS norm $\|f\|_{\mathcal{H}^A}$. The *intrinsic view*, denoted by \mathcal{H}^I , comprises functions whose domain is restricted to the point cloud $\mathcal{M} = \{x_1, \dots, x_n\}$. The measure of smoothness for a function $f \in \mathcal{H}^I$ is taken to be $\Omega_I(f) = \mathbf{f}^T M_I \mathbf{f}$, where M_I is a PSD matrix, such as the Laplacian matrix of the data adjacency graph. While in MR we look for a single function $f \in \mathcal{H}$ that has both small RKHS norm and

small $\Omega_{\mathcal{I}}(f)$, in CoMR we look for two separate functions, an $f^{\mathcal{I}} \in \mathcal{H}^{\mathcal{I}}$ with small $\Omega_{\mathcal{I}}(f)$ and an $f^{\mathcal{A}} \in \mathcal{H}^{\mathcal{A}}$ with small norm. We then ask only that $f^{\mathcal{A}}$ and $f^{\mathcal{I}}$ be close, as measured by a co-regularization term. Due to the difference in representations caused by differences in ambient and geodesic distances, by the discussion above we expect good reductions in the complexity of our co-regularized function space $\tilde{\mathcal{H}}$. For CoMR, we define the view combination function as $u(f^{\mathcal{A}}, f^{\mathcal{I}}) = a^{\mathcal{A}} f^{\mathcal{A}} + a^{\mathcal{I}} f^{\mathcal{I}}$, for any fixed $a^{\mathcal{A}}, a^{\mathcal{I}} \in \mathbf{R}$, and we define the space of final prediction functions as $\tilde{\mathcal{H}} = \{u(f^{\mathcal{A}}, f^{\mathcal{I}}) \mid f^{\mathcal{A}} \in \mathcal{H}^{\mathcal{A}}, f^{\mathcal{I}} \in \mathcal{H}^{\mathcal{I}}\}$. Then the CoMR optimization problem is written as: $\arg \min_{\varphi \in \tilde{\mathcal{H}}} \min_{(f^{\mathcal{A}}, f^{\mathcal{I}}) \in u^{-1}(\varphi)} \hat{R}_{\ell}(\varphi) + \gamma_{\mathcal{A}} \|f^{\mathcal{A}}\|_{\mathcal{H}^{\mathcal{A}}}^2 + \gamma_{\mathcal{I}} (\mathbf{f}^{\mathcal{I}})^T M_{\mathcal{I}} \mathbf{f}^{\mathcal{I}} + \lambda \underline{\mathbf{f}}^T M_{\mathcal{C}} \underline{\mathbf{f}}$, where $\underline{\mathbf{f}} = (f^{\mathcal{A}}(x_1), \dots, f^{\mathcal{A}}(x_n), f^{\mathcal{I}}(x_1), \dots, f^{\mathcal{I}}(x_n))^T$, and where $M_{\mathcal{C}}$, as before, is the co-regularization matrix for two views. If $M_{\mathcal{I}}$ is invertible, or if we make it so by adding a small ridge term, then $\mathcal{H}^{\mathcal{I}}$ is a finite-dimensional RKHS with norm $\|\mathbf{f}^{\mathcal{I}}\|_{\mathcal{I}} = \sqrt{(\mathbf{f}^{\mathcal{I}})^T M_{\mathcal{I}} \mathbf{f}^{\mathcal{I}}}$ and kernel function $k : \mathcal{M} \times \mathcal{M} \rightarrow \mathbf{R}$ given by the matrix $(M_{\mathcal{I}})^{-1}$. Thus our two views are proper RKHSs, and we may directly apply Thm. 1 to get an expression for a multi-view kernel on \mathcal{M} . Experiments in [9] showed that CoMR with $a^{\mathcal{I}} = a^{\mathcal{A}} = 1/2$ significantly outperforms MR on several learning tasks. Note that if $a^{\mathcal{I}} \neq 0$, then the final prediction function will only be defined on \mathcal{M} , since $f^{\mathcal{I}}$ is itself restricted to \mathcal{M} . However, if we take $a^{\mathcal{I}} = 0$, our final prediction function will be $f^{\mathcal{A}}$, which is defined on all of \mathcal{X} . For this special case, the CoMR objective function can be written more simply as $\arg \min_{f^{\mathcal{A}} \in \mathcal{H}^{\mathcal{A}}} \hat{R}_{\ell}(f^{\mathcal{A}}) + \gamma_{\mathcal{A}} \|f^{\mathcal{A}}\|_{\mathcal{H}^{\mathcal{A}}}^2 + \gamma_{\mathcal{I}} (\mathbf{f}^{\mathcal{A}})^T M_{\mathcal{I}} (I + \frac{\gamma_{\mathcal{I}}}{2\lambda} M_{\mathcal{I}})^{-1} \mathbf{f}^{\mathcal{A}}$. When we take $\lambda \rightarrow \infty$, the objective function reduces to MR. As $\lambda \rightarrow 0$, the dependence on the unlabeled data vanishes as the last term in the objective function goes to 0. Thus λ mediates between purely supervised learning at one limit and MR at the other limit. Note that when $a^{\mathcal{I}} = 0$, CoMR may be viewed as MR with a modified smoothness measure. See [5] for more details.

CONCLUSION

We have presented a new RKHS where Tikhonov regularization incorporates both smoothness and multi-view semi-supervised assumptions, and subsumes manifold regularization and co-regularization as special cases. Compared with early frameworks, MVPCR gives prediction functions with out-of-sample extensions, handles multiple views, and allows for arbitrary linear combinations of individual views, rather than simple averages. We expect that the multi-view kernel will allow convenient “plug-and-play” exploration of novel semi-supervised algorithms,

both in terms of implementation and performance bounds.

AUTHORS

David S. Rosenberg (drosen@stat.berkeley.edu), Sense Networks, NY

Vikas Sindhwani (vsindhw@us.ibm.com), IBM Research

Peter L. Bartlett (bartlett@cs.berkeley.edu), Computer Science Division and Dept. of Statistics, University of California, Berkeley

Partha Niyogi (niyogi@cs.uchicago.edu), Dept. of Computer Science, University of Chicago.

References

- [1] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR*, 7:2399–2434, 2006.
- [2] Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. Introduction to statistical learning theory. In *Advanced Lectures on Machine Learning*. Springer Berlin, 2004.
- [3] Ulf Brefeld, Thomas Gärtner, Tobias Scheffer, and Stefan Wrobel. Efficient co-regularised least squares regression. In *ICML*, volume 23, 2006.
- [4] Partha Niyogi. Manifold regularization and semi-supervised learning: Some theoretical analyses. Technical report, Department of Computer Science, University of Chicago, 2008.
- [5] David S. Rosenberg. *Semi-supervised learning with multiple views*. PhD thesis, University of California, Berkeley, 2008.
- [6] David S. Rosenberg and Peter L. Bartlett. The Rademacher complexity of co-regularized kernel classes. In *Int. Conf. on Artificial Intelligence and Statistics*, March 2007.
- [7] Vikas Sindhwani, Partha Niyogi, and Mikhail Belkin. A approach to semi-supervised learning with multiple views. In *Workshop on Learning with Multiple Views, ICML*, 2005.
- [8] Vikas Sindhwani, Partha Niyogi, and Mikhail Belkin. Beyond the point cloud: from transductive to semi-supervised learning. In *ICML*, volume 119, pages 824–831. ACM, 2005.
- [9] Vikas Sindhwani and David S. Rosenberg. An RKHS for multi-view learning and manifold co-regularization. In *International Conference on Machine Learning*, July 2008.
- [10] S. Yu, B. Krishnapuram, R. Rosales, H. Steck, and B. R. Rao. Bayesian co-training. In *NIPS 20*, pages 1665–1672. MIT Press, Cambridge, MA, 2008.