Linear Manifold Regularization for Large Scale Semi-supervised Learning

Vikas Sindhwani VIKASS@CS.UCHICAGO.EDU
Partha Niyogi NIYOGI@CS.UCHICAGO.EDU
Mikhail Belkin MISHA@CS.UCHICAGO.EDU
Department of Computer Science, University of Chicago, Hyde Park, Chicago, IL 60637

Sathiya Keerthi SELVARAK@YAHOO-INC.COM
Yahoo! Research Labs, 210 S Delacey Avenue, Pasadena, CA 91105

Abstract

The enormous wealth of unlabeled data in many applications of machine learning is beginning to pose challenges to the designers of semi-supervised learning methods. We are interested in developing linear classification algorithms to efficiently learn from massive partially labeled datasets. In this paper, we propose Linear Laplacian Support Vector Machines and Linear Laplacian Regularized Least Squares as promising solutions to this problem.

1. Introduction

A single web-crawl by search engines like Yahoo and Google indexes billions of webpages. Only a very small fraction of these web-pages can be hand-labeled and assembled into topic directories. The remaining web-pages form a massive collection of unlabeled documents. Text categorization is among an increasing range of applications that stand to immensely benefit from the development of large scale semi-supervised learning methods.

In this paper, we propose linear semi-supervised classification algorithms for dealing with large datasets. Linear methods have traditionally played a pivotal role in the development of machine learning algorithms. They have been pervasively deployed in information retrieval, data analysis and pattern recognition systems.

Our algorithms are rooted in a general framework for semi-supervised learning called *Manifold Regularization* (Belkin, Niyogi & Sindhwani, 2004; Sindhwani, Niyogi and Belkin, 2005). In this framework, unlabeled data is incorporated within a geometrically motivated regularizer. We specialize this framework for linear classification, and focus on the problem of dealing with large amounts of data.

This paper is organized as follows: In section 2, we setup the problem of linear semi-supervised learning and discuss some prior work. The general Manifold Regularization framework is reviewed in section 3. In section 4, we discuss our methods. Section 5 outlines some research in progress.

2. Linear Semi-supervised Learning

The problem of linear semi-supervised clasification is setup as follows: We are given l labeled examples $\{x_i, y_i\}_{i=1}^{l}$ and u unlabeled examples $\{x_i\}_{i=l+1}^{l+u}$, where the patterns $x \in X \subset \mathcal{R}^d$ are d-dimensional vectors and the labels $y_i \in \{-1, +1\}$ represent two classes. We are interested in developing tractable algorithms to learn a linear classifier $f(x) = sign(w^T x + b)$ given by a weight vector w and a threshold b, in the case where u is very large. We can hope for tractability when the size of the data matrix is still manageable, e.g when dealing with low dimensional problems, or with very sparse high-dimensional problems (e.g text categorization).

A number of recent efforts have considered semi-supervised extensions of well-established supervised methods e.g Support Vector Machines (SVM).

Transductive SVMs (Joachims, 1999) implement the following philosophy for using unlabeled data to choose a weight vector: Find a weight vector and a labeling of the unlabeled examples so that the data (both labeled and unlabeled examples) is separated with maximum margin. This requires a joint optimization of

Appearing in *Proc. of the 22^{st} ICML Workshop on Learning with Partially Classified Training Data*, Bonn, Germany, August 2005. Copyright 2005 by the author(s).

the SVM objective function over possible choices of binary-valued labels on the unlabeled data and the weight vector.

Transductive SVM:

$$(w^*, b^*) = \underset{\substack{w,b \\ y_{l+1}\cdots\ y_{l+u}}}{\operatorname{argmin}} \frac{1}{2}w^T w + C_l \sum_{i=1}^{l} V(y_i, w^T x_i + b) + C_u \sum_{i=l+1}^{l+u} V(y_i, w^T x_i + b)$$

where $V(y, f) = (1 - yf)_+$ is the hinge loss and C_l, C_u are real-valued parameters that balance the hinge loss between the labeled and unlabeled examples. Thus, TSVM minimizes regularized hinge loss over choices of labels for the unlabeled examples. This optimization is implemented in (Joachims, 1999) by first using an inductive SVM to label the unlabeled data and then iteratively solving SVM quadratic programs. At each step labels are switched to improve the objective function. This procedure is susceptible to local minima and requires an unknown, possibly large number of label switches before converging. Thus, this approach does not seem to be well-suited to large scale problems.

Semi-supervised SVMs (S^{3}VM) (Bennett & Demirez, 1998) use a similar objective function optimized using mixed integer programming – Assume both positive and negative labels for each labeled examples and choose the label that incurs smaller total hinge loss:

Semi-supervised SVM:

$$(w^*, b^*) = \underset{w,b}{\operatorname{argmin}} \frac{1}{2}w^T w + C_l \sum_{i=1}^{l} V(y_i, w^T x_i + b) + C_u \sum_{i=l+1}^{l+u} \min\left[V(-1, w^T x_i + b), V(+1, w^T x_i + b)\right]$$

The 1-norm of the weight vector is preferred in (Bennett & Demirez, 1998); the optimization found to be intractable even for moderate amounts of unlabeled data. (Fung & Mangasarian, 2001) reformulate this approach as a concave minimization problem which is solved by a successive linear approximation algorithm. These methods are applied to relatively small sized problems.

3. Manifold Regularization

Before discussing our algorithms, we briefly discuss the intuition and algorithmic framework of Manifold Regularization. The success of semi-supervised learning rests on how much information unlabeled examples carry about the distribution of labels in the pattern space. Frequently, unlabeled examples may help in identifying clusters or a low-dimensional manifold structure along which labels can be assumed to vary smoothly. These are the *cluster* and *manifold assumption* respectively.

The Manifold regularization framework extends the classical framework of regularization in Reproducing Kernel Hilbert Spaces (RKHS) to exploit the geometry of the marginal distribution, as estimated by unlabeled data, when these assumptions are satisfied. Unlabeled data is incorporated via a regularization term (in addition to the RKHS norm) whose role is to penalize functions for changing rapidly along a manifold or within a cluster.

The empirical estimate of these underlying structures can be encoded as a graph whose vertices are the labeled and unlabeled data points and whose edge weights $\{W_{ij}\}_{i,j=1}^{l+u}$ represent appropriate pairwise similarity relationships between examples. Given this graph, one can bias learning in an RKHS towards functions that vary smoothly along the edges (along the underlying structure). The following optimization problem is solved over an RKHS $\mathcal{H}_K$ with kernel function $K(x, y)$ and loss function V:

Manifold Regularization:

$$f^* = \underset{f \in \mathcal{H}_K}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(x_i, y_i, f) + \gamma_A \|f\|_K^2 + \gamma_I \sum_{i,j=1}^{l+u} (f(x_i) - f(x_j))^2 W_{ij}$$

Here, γ_A, γ_I are regularization parameters that control the RKHS norm and the *intrinsic* norm respectively.

Defining, $\hat{f} = [f(x_1), \ldots, f(x_{l+u})]^T$, and L as the Laplacian matrix of the graph, given by $L = D - W$ where D is the diagnonal degree matrix given by $D_{ii} = \sum_{j=1}^{l+u} W_{ij}$, we re-write the above problem as:

$$f^* = \underset{f \in \mathcal{H}_K}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(x_i, y_i, f) + \gamma_A \|f\|_K^2 + \gamma_I \hat{f}^T L \hat{f}$$

One can also take powers of the Laplacian L^p to define the graph regularizer. For more on graph regularization, see (Belkin,Matveeva & Niyogi; Kondor & Lafferty,2002).

A version of the Representer theorem can be easily proved that shows that the minimizer has the following form:

$$f(x) = \sum_{i=1}^{l+u} \alpha_i K(x_i, x) \qquad (1)$$

This reduces the optimization problem in manifold regularization into a finite dimensional problem of estimating the $(l+u)$ expansion coefficients α_i. The algorithms that estimate these coefficients and construct the optimal function are called Laplacian RLS and Laplacian SVM for the squared loss and hinge loss respectively. Both algorithms involve inverting $(l+u) \times (l+u)$ dense Gram matrices so that the complexity of a naive implementation is $O(l+u)^3$ which is prohibitive for large datasets. Experiments showing state-of-the-art semi-supervised learning performance with small to moderate number of unlabeled examples is reported in (Sindhwani, Niyogi and Belkin, 2005).

4. Linear Manifold Regularization

Linear manifold regularization specializes the above algorithms to linear functions resulting in the following optimization problem:

$$(w^*, b^*) = \operatorname*{argmin}_{w,b} \gamma_A w^T w + \gamma_I w^T X^T L X w$$
$$+ \frac{1}{l} \sum_{i=1}^{l} V(y_i, w^T x_i + b) \qquad (2)$$

where $X = [x_1 \dots x_{l+u}]^T$ is the $(l+u) \times\ d$ data matrix with rows as training examples. In contrast to TSVM and S^{3}VM, this is a minimization of a convex problem whose solution can be readily obtained.

When the data dimensionality d is much larger than the number of examples $(l+u)$ and there is not much sparsity, it is advantageous to simply use the general RKHS algorithms described in Section 3, with the linear kernel $K(x,y) = x^T y$. However, we are interested in the case where $(l+u)$ is large but the dimensionality d or the avergage number of non-zero entries, $\bar{d}$, in a feature vector is relatively small. Also, note that the graph Laplacian is typically highly sparse with a small number of entries, k, per column. Instead of looking at the dual variables α_i in the large expansion in (1), we will directly approach the problem in primal variables w.

4.1. Linear Laplacian RLS

Taking V to be the squared loss $V(y, w^T x + b) = (y - w^T x - b)^2$ and setting the gradient of the objective function to 0, we immediately obtain a linear system that can be solved to obtain the desired weight vector:

$$(X_l^T X_l + \gamma_A l I + \gamma_I l X^T L X) w = X_l^T Y \qquad (3)$$

Here X_l is the submatrix of X corresponding to labeled examples and Y is the vector of labels. This is a $d \times\ d$ system which can be easily solved when d is small. When d is large but feature vctors are highly sparse with $\bar{d}$ number of non-zero entries, we can employ Conjugate Gradient (CG) methods to solve this system. CG techniques are Krylov methods that solve a system $Ax = b$ by repeatedly multiplying a candidate solution z by A. In the case of linear Laplacian RLS, we can construct the matrix-vector product in the LHS of (3) in time $O(n(\bar{d}+pk))$,where p (typically very small) is the power of the Laplacian matrix (if L^p is used as the graph regularizer), and k (typically small) is average number of entries per column in L. This is done by using an intermediate vector Xw and appropriately forming sparse matrix-vector products. Thus, the algorithm can employ very well-developed methods for efficiently obtaining the solution.

4.2. Linear Laplacian SVM

To solve the optimization problem 4 for the hinge loss, we adopt a different strategy (which also works for Linear Laplacian RLS). We can rewrite problem 4 as:

$$(w^*, b^*) = \operatorname*{argmin}_{w,b} \gamma_A w^T T^2 w + \frac{1}{l} \sum_{i=1}^{l} V(y_i, w^T x_i + b)$$
$$\text{where} \quad T^2 = (\gamma_A I + \gamma_I X^T L X)$$

Changing variables by $\tilde{w} = Tw$ and $\tilde{x} = T^{-1}x$, we can convert the above problem into a standard SVM running only on the labeled examples that are preprocessed with T^{-1} which is defined using unlabeled data:

$$(\tilde{w}^*, b^*) = \operatorname*{argmin}_{\tilde{w},b} \tilde{w}^T \tilde{w} + \frac{1}{l} \sum_{i=1}^{l} V(y_i, \tilde{w}^T \tilde{x}_i + b)$$

When d is small, the preprocessing matrix is a small $d \times\ d$ matrix, and the reparameterized SVM runs only on a small number of labeled examples. After solving this SVM, we obtain the solution $w^* = T^{-1}\tilde{w}^*$.

For L2-SVMs defined by the loss function $V(y,f) = (1 - yf)_+^2$, there has been recent work on designing large scale linear SVMs (Keerthi & DeCoste, 2005). At the core of this algorithm are RLS iterations implemented using conjugate gradient techniques. In cojunction with efficient CG-based techniques, this algorithm can also be modified to incorporate unlabeled data via a graph regularizer.

5. Research in Progress

We have discussed some ideas towards linear methods for utilizing large amounts of unlabeled data. We are currently interested in efficient implementation of variations of these ideas, and exploring applications in real-world classification tasks.

Also of interest is the issue of model selection. The problem of efficiently probing the space of solutions for families of shifted Tikhonov regularization problems generated by the regularization parameter has received some attention in numerical computing literature e.g in (Frommer & Maass, 1999). These methods can possibly be adapted for choosing the parameters γ_A, γ_I.

A class of techniques for data subset selection and low-rank kernel approximation can also be applied for efficient non-linear manifold regularization.

Finally, another direction is the development of semi-supervised feature selection methods building on these ideas, using the weight vector of the linear semi-supervised classifier. This has natural applications in text classification.

References

Belkin M., Matveeva I., Niyogi P. (2004) *Regression and Regularization on Large Graphs* COLT 2004.

Belkin M., Niyogi P. & Sindhwani V., (2004). *Manifold Regularization : A Geometric Framework for Learning for Examples* Technical Report, Univ. of Chicago, Department of Computer Science, TR-2004-06

K. Bennett and A. Demirez (1998), *Semi-Supervised Support Vector Machines* NIPS 1998

Frommer & Maass (1999), *Fast CG-based methods for Tikhonov-Phillips regularization.* SIAM Journal of Scientific Computing, 20(5), 1831-1850

G. Fung and O. Mangasarian (2001), *Semi-Supervised Support Vector Machines for Unlabeled Data Classification* Optimization Methods and Software 15, 2001, 29-44

S. S. Keerthi, D. DeCoste (2005) *A Modified Finite Newton Method for Fast Solution of Large Scale Linear SVMs* 6(Mar):341–361, 2005.

R. I. Kondor and J. Lafferty (2002) *Diffusion Kernels on Graphs and Other Discrete Input Spaces* ICML 2002

Joachims T., (1999). *Transductive Inference for Text Classification using Support Vector Machines* ICML 1999.

Joachims T. (2003), *Transductive Learning via Spectral Graph Partitioning* (ICML) 2003

V. Sindhwani, P. Niyogi and M. Belkin, *Beyond the point cloud: from Transductive to Semi-supervised Learning* Under submission, 2005