Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 09-012

A Unified View of Graph-based Semi-Supervised Learning: Label Propagation, Graph-Cuts, and Embeddings

Amrudin Agovic and Arindam Banerjee

May 12, 2009

A Unified View of Graph-based Semi-Supervised Learning: Label Propagation, Graph-Cuts, and Embeddings

Amrudin Agovic
aagovic@cs.umn.edu
University of Minnesota

Arindam Banerjee
banerjee@cs.umn.edu
University of Minnesota

Abstract

Recent years have seen a growing number of graph-based semi-supervised learning methods. While the literature currently contains several of these methods, their relationships with one another and with other graph-based data analysis algorithms remain unclear. In this paper, we present a unified view of graph-based semi-supervised learning. Our framework unifies three important and seemingly unrelated approaches to semi-supervised learning, viz label propagation, graph cuts and manifold embeddings. We show that most existing label propagation methods solve a special case of a generalized label propagation (GLP) formulation which is a constrained quadratic program involving a graph Laplacian. Different methods arise simply based on the choice of the Laplacian and the nature of the constraints. Further, we show that semi-supervised graph-cut problems can also be viewed and solved as special cases of the GLP formulation. In addition, we show that semi-supervised non-linear manifold embedding methods also solve variants of the GLP problem and propose a novel family of semi-supervised algorithms based on existing embedding methods. Finally, we present comprehensive empirical performance evaluation of the existing label propagation methods as well as the new ones derived from manifold embedding. The new family of embedding based label propagation methods are found to be competitive on several datasets.

1 Introduction

Semi-supervised learning is becoming a crucial part of data mining, since the gap between the total amount of data being collected in several problem domains and the amount of labeled data available for predictive modeling is ever increasing. Semi-supervised learning methods typically make assumptions about the problem based on which predictions are made on the unlabeled data [6]. A commonly used assumption, called the smoothness assumption, is that nearby

points should have the same label. The assumption can be instantiated in several ways, and that has lead to several different algorithms for semi-supervised learning [22, 21, 18].

Graph-based semi-supervised learning algorithms are an instantiation of the smoothness assumption. In such a setting, a graph is constructed where each vertex corresponds to a point, and the edge connecting two vertices typically has a weight proportional to the proximity of the two points [22, 21]. Then, labels are "propagated" along the weighted edges to get predictions on the unlabeled data. Recent years have seen significant interest in the design of label propagation algorithms for graph-based semi-supervised learning. In spite of the progress, there is limited understanding on the relationships between the different label propagation methods, as well as their relationships to other graph-based methods including those for graph-cuts and non-linear manifold embeddings. Further, empirical evaluation and comparison among the methods have been rather limited.

In this paper, we present a unified view of graph-based semi-supervised learning methods. Our framework unifies three important and seemingly unrelated approaches to semi-supervised learning, viz lapel propagation, graph cuts and manifold embeddings. We show that most existing label propagation methods solve a special case of a generalized label propagation (GLP) formulation which is a constrained quadratic program involving a graph Laplacian [8, 17]. Different methods arise simply based on the choice of the Laplacian and the nature of the constraints. Further, we show that semi-supervised graph-cut problems can also be viewed and solved as special cases of the GLP formulation. In addition, we show that semi-supervised non-linear manifold embedding methods also solve variants of the GLP problem. Based on this observation, we effectively provide a recipe for converting embedding methods to semi-supervised learning methods, and introduce a novel family of embedding based label propagation methods. Our analysis also reveals direct relationships among existing methods, e.g., label propagation for semi-supervised unnormalized graph cuts is the same as Gaussian fields [22] and semi-supervised Laplacian eigenmaps [3]. Finally, we present comprehensive empirical performance evaluation of the existing label propagation methods as well as the new ones derived from manifold embedding. Among other things, we demonstrate that the new class of embedding based label propagation methods are competitive on several datasets.

The rest of the paper is organized as follows. We review background material in Section 2. In Section 3, we introduce the GLP formulation and present an unified view of existing label propagation methods. In Section 4, we show the relationship between semisupervised graph-cuts and the GLP formulation. We discuss semisupervised manifold embedding and introduce a set of embedding based label propagation methods in Section 5. We present empirical results in Section 6 and conclude in Section 7.

2 Background

In this section we review necessary background on graph-based semi-supervised learning and graph Laplacians.

2.1 Graph-based Semi-Supervised Learning

Let $D = \{(x_1, y_1), \ldots, (x_\ell, y_\ell), x_{\ell+1}, \ldots, x_{\ell+u}\}$ be partially labeled dataset for classification, where only ℓ out of the $n = (\ell + u)$ points have labels, and $y_i \in \{-1, +1\}$ for $i = 1, \ldots, \ell$.[1] Let $G = (V, E)$ be an undirected graph over the points, where each vertex v_i corresponds to a datapoint x_i, and each edge in E has a non-negative weight $w_{ij} \geq 0$. The weight w_{ij} typically reflects the similarity between x_i and x_j, and is assumed to be computed in a suitable application dependent manner. Given the partially labeled dataset D and the similarity graph G, our objective is to learn a function $f \in \mathbb{R}^n$, which associates each vertex to a discriminant score f_i and a final prediction $\text{sign}(f_i)$ for classification. The problem has been extensively studied in the recent past [16, 18, 6, 22, 21].

2.2 Graph Laplacians

Let $G = (V, E)$ be an undirected weighted graph with weights $w_{ij} \geq 0$, and let D be a diagonal matrix with $D_{ii} = \sum_j w_{ij}$. In the existing literature, there are three related matrices that are called the graph Laplacian, and there does not appear to be a consensus on the nomenclature [17]. These three matrices are intimately related, and we will use all of them in our analysis. The *unnormalized graph Laplacian* L_u is defined as:

$$L_u = D - W \ . \tag{1}$$

The following property of the unnormalized graph Laplacian is important for our analysis: For any $f \in R^n$, we have

$$f^t L_u f = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2 \ . \tag{2}$$

The matrix L_u is a symmetric and positive semidefinite. There are also two normalized graph Laplacians in the literature [8], respectively given by:

$$L_r = D^{-1} L_u = I - D^{-1} W \ , \tag{3}$$

$$L_s = D^{-1/2} L_u D^{-1/2} = I - D^{-1/2} W D^{-1/2} \ . \tag{4}$$

For the symmetrically normalized graph Laplacian, the following property holds: For any $f \in \mathbb{R}^n$, we have

$$f^t L_s f = \frac{1}{2} \sum_{i,j} w_{ij} \left(\frac{f_i}{\sqrt{D_{ii}}} - \frac{f_j}{\sqrt{D_{jj}}} \right)^2 \ . \tag{5}$$

[1] While we focus on the 2-class case for ease of exposition, the extensions to multi-class are mostly straightforward. We report results on multi-class problems in Section 6.

We refer the reader to [13, 8, 17] for further details on Laplacians and their properties.

3 A Unified View of Label Propagation

In this section, we present a unified view of several label propagation formulations as a constrained optimization problem involving a quadratic form of the Laplacian where the constraints are obtained from the labeled data.

3.1 Generalized Label Propagation

The Generalized Label Propagation (GLP) formulation considers a graph-based semi-supervised learning setting as described in Section 2.1. Let W be the symmetric weight matrix and L be a corresponding graph Laplacian. Note that L may be any of the Laplacians discussed in Section 2, and we will see how different label propagation formulations result out of specific choices of the Laplacian. Let $f \in \mathbb{R}^n$, where $n = \ell + u$, be the predicted score on each data point x_i, $i = 1, \ldots, n$; the predicted label on x_i can be obtained as $\text{sign}(f_i)$. The generalized label propagation (GLP) problem can be formulated as follows:

$$\min_{f \in \mathcal{S}} f^T L f, \quad s.t. \ \sum_{i=1}^{\ell} (f_i - y_i)^2 \leq \epsilon \ , \tag{6}$$

where $\epsilon \geq 0$ is a constant and $\mathcal{S} \subseteq \mathbb{R}^n$. For most existing formulations $\mathcal{S} = \mathbb{R}^n$ whereas for a few $\mathcal{S} = \{f | f \in \mathbb{R}^n, f \perp \mathbb{1}\}$ where $\mathbb{1}$ is the all ones vector. The Lagrangian for the GLP problem is given by $L(f, \mu) = f^T L f + \mu \sum_{i=1}^{\ell} (f_i - y_i)^2$, where $\mu \geq 0$ is the Lagrangian multiplier. Some variants assume $y_i = 0$ for $i = (\ell + 1), \ldots, n$, so the constraint will be of the form $\sum_{i=1}^{n} (f_i - y_i)^2 \leq \epsilon$. Assuming the Laplacian to be symmetric, which is true for L_u and L_s, the first order necessary conditions are given by $(L + \mu I) f = \mu y$, where I is the identity matrix. Several existing methods work with the special case $\epsilon = 0$, which makes the constraints binding so that $\sum_{i=1}^{\ell} (f_i - y_i)^2 = 0$ and $f_i = y_i$ on the labeled points. The first order conditions for the special case is given by $Lf = 0$. In the next several sections, we show how most of the existing label propagation methods for semi-supervised learning can be derived directly as a special case of the GLP formulation or closely related to it with special case choices of the Laplacian L, the constant ϵ, and the subspace $\mathcal{S}$.

3.2 Gaussian Fields (GF)

Motivated by the assumption that neighboring points in a graph will have similar labels in Gaussian fields, the following energy function is considered [22]:

$$E(f) = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2 \ . \tag{7}$$

The GF method computes labels by minimizing the energy function $E(f)$ with respect to f under the contraint that $f_i = y_i$ for all labeled points. As observed in [22], the energy function is harmonic, i.e., it is twice continuously differentiable and it satisfies Laplace's equation [10]. From the harmonic property of the energy function it follows that the predicted labels will satisfy: $f = D^{-1}Wf$. In terms block matrices corresponding to labeled and unlabeled points we have:

$$\begin{bmatrix} D_{\ell\ell} & 0 \\ 0 & D_{uu} \end{bmatrix} \begin{bmatrix} f_\ell \\ f_u \end{bmatrix} = \begin{bmatrix} W_{\ell\ell} & W_{\ell u} \\ W_{u\ell} & W_{uu} \end{bmatrix} \begin{bmatrix} f_\ell \\ f_u \end{bmatrix} \ .$$

Since $f_\ell = y_\ell$ due to the constraints,[2] the above system can be simplified to get a closed form for f_u given by

$$f_u = (D_{uu} - W_{uu})^{-1} W_{ul} y_l \ . \tag{8}$$

We can interpret the objective function in Gaussian Fields as a special case of the GLP problem in (6). In particular, using the identity in (2) and noting that the constraints on the labeled points are binding, GF can be seen as a special case of GLP with $L = L_u$ and $\epsilon = 0$, i.e.,

$$\min_{f \in \mathbb{R}^n} \ f^T L_u f \ , \quad s.t. \ \ \sum_{i=1}^{\ell} (f_i - y_i)^2 \leq 0 \ . \tag{9}$$

3.3 Tikhonov Regularization (TIKREG)

Given a partially labeled data set, TIKREG [2] is an algorithm for regularized regression on graphs, where the objective is to infer a function f over the graph. The objective function for TIKREG is given by

$$\min_f \ \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2 + \frac{1}{\gamma \ell} \sum_{i=1}^{\ell} (f_i - y_i)^2 \tag{10}$$

with the constraint that $f \perp \ $, i.e., f lies in the orthogonal subspace of $\ $, the all ones vector. The parameter γ is assumed to be a real-valued number. A closed form solution for the above problem is obtained [2] as:

$$f = (\ell \gamma L_u + I_k)^{-1} (\hat{y} + \mu \ ) \tag{11}$$

where $\hat{y} = (y_1, y_2, \ldots, y_l, 0, \ldots, 0)$, $I_k = diag(1, \ldots, 1, 0, \ldots, 0)$ with the number of ones equal to the number of labeled points. The orthogonality constraint on f is enforced through the lagrange multiplier μ, which is optimally computed as:

$$\mu = -\frac{^T(\ell \gamma L_u + I_k)^{-1} \hat{y}}{^T(\ell \gamma L_u + I_k)^{-1}} \ . \tag{12}$$

The objective function can be viewed as a special case of the GLP objective in (6). As before, the first term is $f^T L_u f$, where L_u is the unnormalized Laplacian.

[2] We abuse notation and denote $[f_1, \ldots, f_\ell]^T$ by f_ℓ (similarly for y_ℓ) and $[f_{(\ell+1)}, \ldots, f_n]^T$ by f_u in the sequel.

The second term corresponds to the constraint $\sum_i (f_i - y_i)^2 \leq \epsilon$, in (6) where $1/\gamma\ell$ is the optimal Lagrange multiplier corresponding to the constraint. In other words, if $\epsilon(1/\gamma\ell)$ is the constraint value that leads to the optimal Lagrange multiplier of $1/\gamma\ell$, the TIKREG problem can be seen as a special case of GLP:

$$\min_{f \in \mathbb{R}^n, f \perp \mathbb{1}} f^T L_u f \ , \quad s.t. \quad \sum_{i=1}^{\ell} (f_i - y_i)^2 \leq \epsilon(1/\ell\gamma) \ . \tag{13}$$

3.4 Local and Global Consistency (LGC)

The Local and Global Consistency (LGC) approach [21] gives an alternative graph based regularization framework for semi-supervised learning. In particular, the LGC is formulated based on the following objective function [21]:

$$\min_f \ \frac{1}{2} \left(\sum_{i,j=1}^{n} w_{ij} \left(\frac{1}{\sqrt{D_{ii}}} f_i - \frac{1}{\sqrt{D_{jj}}} f_j \right)^2 + \mu \sum_{i=1}^{n} (f_i - y_i)^2 \right) \ , \tag{14}$$

with $\mu > 0$ as the regularization parameter. Note that LGC assumes that there is a valid y_i for all points; operationally, the $y_i, i = 1, \ldots, \ell$ is set to the true given label, whereas $y_i, i = \ell + 1, \ldots, n$ is set to 0. The problem is solved using an iterative label propagation algorithm. Given a weight matrix W among the points, the weights are normalized to obtain $S = D^{-1/2} W D^{-1/2}$ with D diagonal and $D_{ii} = \sum_j w_{ij}$. Starting from an initial guess $f^{(0)}$, the iterative algorithm proceeds with the following updates:

$$f^{(t+1)} = \alpha S f^{(t)} + (1 - \alpha) y \ , \tag{15}$$

where $\alpha \in (0, 1)$. As shown in [21], this update equation converges to $f^* = (1 - \alpha)(I - \alpha S)^{-1} y$, which can be shown to optimize the objective function in (14) when $\alpha = 1/(1 + \mu)$. We now show that the LGC formulation is a special case of the GLP formulation in (6). From the identity involving the normalized Laplacian in (5), then the LGC can be seen as a special case of GLP as follows:

$$\min_f \ f^T L_s f \ , \quad s.t. \quad \sum_{i=1}^{n} (f_i - y_i)^2 \leq \epsilon(\mu) \ , \tag{16}$$

where $\epsilon(\mu)$ is the constant corresponding to the optimal Lagrange multiplier μ. Note that since in LGC, one starts with an initial label $y_i, i = 1, \ldots, n$, the constraint involves terms corresponding to all the points.

3.5 Related Methods

We review three other methods, viz cluster kernels, Gaussian random walks, and local neighborhood propagation for graph-based semi-supervised learning which are closely related to the GLP framework.

3.5.1 Cluster Kernels (CK)

The main idea in cluster kernels [7] is to embed the data into a lower dimensional space based on its cluster structure and then subsequently build a classifier on the low-dimensional data. If K denotes a suitable kernel on the data space, the embedding method focuses on the k primary eigenvectors of the symmetrized matrix $D^{-1/2}KD^{-1/2}$. If K corresponds to the edge weights on the graph $G = (V, E)$ between the points, i.e., $K = W$, then the embedding corresponds to the k eigenvectors of the symmetrized Laplacian $L_s = I - D^{-1/2}WD^{-1/2}$ corresponding to the smallest k eigenvalues. In particular, for $k = 1$, the embedding is given by the eigenvector corresponding to the smallest eigenvalue of L_s which is the solution to LGC in absence of any semi-supervision. CK trains a suitable (linear) classifier on the low-dimensional embedding to obtain the final predictions.

3.5.2 Gaussian Random Walks EM (GWEM)

Consider a random walk on the graph with transition probability $P = D^{-1}W$. The GWEM method [16] works with the m-step transition probability matrix P^m so that the probability of going from x_i to x_j is given by $p_{m|0}(x_j|x_i) = (P^m)_{ij}$. The random walk is assumed to start with uniform probability from any one of the nodes, so $P(x_i) = 1/n$. Using Bayes rule, one can obtain the posterior probabilities $P_{0|m}(x_i|x_j)$. Now, each point is assumed to have a (possibly unknown) distribution $p(y|x_i)$ over the class labels. For any point x_j, the posterior probability of class label y is given by $P(y_j = c|x_j) = \sum_i P(y_i = c|x_i)p_{0|m}p(x_i|x_j)$. The prediction is based on $y_j = \text{argmax}_c\ P(y_j = c|x_j)$. Now, since $P(y|x_i)$ is unknown for the unlabeled points, an EM algorithm can be used to alternately maximize the log-posterior probability of known labels on the labeled points $\sum_{k=1}^{\ell} \log P(y_k|x_k) = \sum_{k=1}^{\ell} \log \sum_{i=1}^{N} P(y_i|x_i)P_{0|m}(x_i|x_k)$. The EM algorithm alternates between the E step which estimates $P(x_i|x_k, y_k) \propto P(y_k|x_i)P_{0|m}(x_i|x_k)$, where k denotes an index over labeled points, and the M step, which computes $P(y = c|x_i) = \sum_{k:y_k=c}^{\ell} P(x_i|x_k, y_k)/\sum_{h=1}^{\ell} P(x_i|x_h, y_h)$.

We now show that GWEM can be interpreted in terms of spectral decomposition of a suitable asymmetrically normalized Laplacian L_r as in (3). For a fixed number of steps m for the random walk, let $Z^T = P^m = (D^{-1}W)^m$. Note that Z^T itself is a transition probability matrix, and $Z_{ij} = P_{m|0}(x_i|x_j)$. Let D_Z be a diagonal matrix such that $D_{Z,ii} = \sum_j Z_{ij}$. Since the prior probability $P(x_i) = 1/n$, by Bayes rule we have $P_{0|m}(x_j|x_i) = P_{m|0}(x_i|x_j)/\sum_{i'} P_{m|0}(x_{i'}|j) = (D_Z^{-1}Z)_{ij}$. Let $f_j = P(y_j|x_j)$. When the EM algorithm converges we will have: $f = D_Z^{-1}Zf \Rightarrow (I - D_Z^{-1}Z)f = 0$, where $f_i = y_i$ for the labeled points. Since $D_Z^{-1}Z$ is a transition probability matrix, from (3) we note that $(I - D^{-1}Z)$ can be viewed as a asymmetrically normalized Laplacian L_r so that $L_r f = 0$. Finally, since $D_Z f = Zf$ resembles the fixed point equation for GFs, a block decomposition as in (8) yields $f_u = (D_{z,uu} - Z_{uu})^{-1}Z_{u\ell}y_\ell$.

3.5.3 Linear Neighborhood Propagation (LNP)

Linear Neighborhood Propagation (LNP) [18] is another recent approach, which differs from the other methods as LNP computes a stochastic transition matrix U directly from the data. In particular, one computes a probability distribution over neighboring points so that their expectation best approximates the point under consideration: $\min_{\mathbf{u}_i} \left\|x_i - X_i^N \mathbf{u}_i\right\|^2$, where $\mathbf{u}_i$ is probability distribution over the neighbors of x_i and X_i^N is a matrix each of whose columns is a neighbor of x_i. Once the transition probability matrix U is computed, the semisupervised learning problem is posed as follows:

$$\min_{f \in \mathbb{R}} \sum_{i,j} u_{ij}(f_i - f_j)^2 + \mu \sum_{i=1}^{n} (f_i - y_i)^2 \ , \tag{17}$$

where, similar to LGC [21], the labels $y_i, i = 1, \ldots, \ell$ are set to their true values, and the unknown labels $y_i, i = \ell + 1, \ldots, n$ are set to 0. Similar to LGC, the LNP problem is solved by an iterative label propagation algorithm. Starting from an initial guess $f^{(0)}$, the iterative algorithm proceeds with the following updates:

$$f^{(t+1)} = \alpha U f^{(t)} + (1 - \alpha) y \ , \tag{18}$$

where $\alpha = 1/(1 + \mu) \in (0, 1)$. The updates are the same as in (15) for LGC [21] with the difference that U is not normalized symmetrically, but is a transition probability matrix of a random walk. In spite of the similarities, a careful consideration of the analysis in [18] reveals that update equation in (18) does not solve the problem in (17). On convergence, the iterative updates in (18) leads to $f = (I - \alpha U)^{-1}(1 - \alpha)y$. On the other hand, setting derivatives of (17) to zero leads to $f = \left(I - \alpha(U + U^T)/2\right)^{-1}(1 - \alpha)y$. The issue arises in the analysis [18] when one assumes $[(I - U) + (I - U)^T]f \approx 2(I - U)f$, which is not true unless U is symmetric. For empirical evaluation, we use the iterative updates in (18).

3.6 Label Propagation and Green's Function

We briefly describe an interesting relationship between label propagation and the discrete Green's function. Green's functions are typically used to convert inhomogenous partial differential equations with boundary conditions into an integral problem. In particular the inverse Laplace operator with the zero mode removed can be interpreted as a Green's function for the discrete Laplace operator [9]. Let $\mathcal{G} = L^\dagger$ be the generalized inverse of the Laplacian L. The solutions for both GF and GWEM can be expressed as: $f_u = (D_{uu} - W_{uu})^{-1} W_{ul} y_l = L_u^\dagger z_u$ where $z_u = W_{ul} y_l$. Discarding the zero mode of L_u, we have $f_u \approx \mathcal{G}_u z_u$. As argued in [9], discarding the zero mode is important to ensure that the Green's function exists; further, it does not affect the final result. Then f_u can be viewed as a solution to a partial differential equation with boundary value constraints. The interpretation is intuitive if the labeled points are treated as electric charges.

In particular one assumes labeled points to be postive and negative charges. Using the Green's function one then computes the influence of these charges on unlabeled points [9]. For methods such as LGC and LNP the solution has the form $f = (I - A/(1+\mu))^{-1} \mu y/(1+\mu)$, with $A = D^{-1/2} W D^{-1/2}$ for LGC and $A = U$ for LNP. Considering the strong regularization limit as $\mu \to 0$ and removing the zero mode in L we obtain: $f = L^{\dagger} y \approx \mathcal{G} y$.

4 Semi-Supervised Graph Cuts

We now demonstrate how label propagation formulations can be viewed as solving a relaxed version of semi-supervised graph-cut problems. Let $G = (V, E)$ be a weighted undirected graph with weight matrix W. If V_1, V_2 is a partitioning of V, i.e., $V_1 \cap V_2 = \emptyset, V_1 \cup V_2 = V$, then the value of the cut implied by the partitioning (V_1, V_2) is given by: $cut(V_1, V_2) = \frac{1}{2} \sum_{v_i \in V_1, v_j \in V_2} w_{ij}$. The minimum cut problem is to find a partitioning (V_1, V_2) such that $cut(V_1, V_2)$ is minimized. Due to practical reasons, one often works with a normalized cut objective, such as the ratio-cut [11] or normalized-cut [15], which encourage the partitions V_1, V_2 to be more balanced. The objective for ratio-cut is $Rcut(V_1, V_2) = \frac{cut(V_1,V_2)}{|V_1|} + \frac{cut(V_2,V_1)}{|V_2|}$. The objective for normalized-cut is similar, however it normalizes cuts by the weight of the edges in each partition. Letting $Vol(V) = \sum_{i \in V} D_{ii}$, we have: $Ncut(V_1, V_2) = \frac{cut(V_1,V_2)}{Vol(V_1)} + \frac{cut(V_2,V_1)}{Vol(V_2)}$.

While the graph-cut problems outlined above are unsupervised, given labels on some of the nodes, one can construct a semi-supervised graph cut problem that respects the labeling [4, 5]. Let A_1 be the subset of vertices with label +1, and A_2 be the subset with label -1. Clearly, A_1 and A_2 are disjoint subsets of V. The *semi-supervised unnormalized cut* problem can be posed as follows: Find a partitioning (V_1, V_2) such that $cut(V_1, V_2)$ is minimized subject to the constraint $A_1 \subseteq V_1, A_2 \subseteq V_2$. In order to achieve balanced cuts, we also consider semi-supervised versions of the ratio-cut (or normalized-cut) problem. In particular, the *semi-supervised ratio-cut* problem can be posed as follows: Find a partitioning (V_1, V_2) such that $Rcut(V_1, V_2)$ is minimized subject to the constraint $A_1 \subseteq V_1, A_2 \subseteq V_2$. Similarly, one can pose the semi-supervised normalized-cut problem using $Ncut(V_1, V_2)$ instead of $Rcut(V_1, V_2)$ above. The problems outlined above are NP-hard, and there has been some work on developing polynomial-time approximation schemes (PTASs) for related problems [4, 5]. In this section, we show that relaxed versions of these problems lead to special cases of the GLP formulation for a suitable choice of the Laplacian L and the constraint ϵ, and hence can be solved using label propagation methods.

4.1 Semi-Supervised Unnormalized Cut

Consider a graph partitioning given by V_1 and V_2. Let f be defined as follows

$$f_i = \begin{cases} 1 & \text{if } v_i \in V_1 \\ -1 & \text{if } v_i \in V_2 \ . \end{cases} \tag{19}$$

From (2), we now have

$$f^t L_u f = \frac{1}{2}\sum_{i,j=1}^{n} \mathbf{w}_{ij}(f_i - f_j)^2 = 4cut(V_1, V_2) \ . \tag{20}$$

For any given disjoint sets A_1, A_2 which constitute the semi-supervision, we construct constraints on the labels as $y_i = +1$ if $v_i \in A_1$ and $y_i = -1$ if $v_i \in A_2$. Then, for all nodes in the labeled set, i.e., $v_i \in A_1 \cup A_2 = \mathcal{L}$, we have the constraint that $f_i = y_i$. Then, the semi-supervised unnormalized cut problem can be written as:

$$\min_{V_1, V_2} f^t L_u f, \quad s.t.\ f_i \text{ is as in (19)}, \ \ \forall v_i \in \mathcal{L}, f_i = y_i \ . \tag{21}$$

By relaxing the problem such that $f \in \mathbb{R}^n$ and noting that the constraint above is equivalent to $\sum_{i=1}^{\ell}(f_i - y_i)^2 \leq 0$, we obtain the following formulation:

$$\min_{f \in \mathbb{R}^n} f^t L_u f, \quad s.t.\ \sum_{i=1}^{\ell}(f_i - y_i)^2 \leq 0 \tag{22}$$

Clearly, the objective function is a special case of our GLP formulation using an unnormalized graph Laplacian and $\epsilon = 0$. In particular, the above is *exactly the same* as the formulation for Gaussian Fields [22] described in section (3.2).

4.2 Semi-Supervised Ratio Cut

In the context of the ratio cut problem, consider again a graph partitioning given by V_1 and V_2. Let f be defined as

$$f_i = \begin{cases} +\sqrt{|V_2|/|V_1|} & \text{if } v_i \in V_1 \\ -\sqrt{|V_1|/|V_2|} & \text{if } v_i \in V_2 \ . \end{cases} \tag{23}$$

Now, following (2), we can express

$$f^T L_u f = \frac{1}{2}\sum_{i,j} w_{ij}(f_i - f_j)^2 = |V| Rcut(V_1, V_2) \tag{24}$$

where $|V|$ is a constant. From the predefined values of f we can see that $f^T \ = 0$, and $||f||^2 = n$. The objective function for the semi-supervised ratio-cut problem can therefore be expressed as:

$$\min_{V_1, V_2} f^T L_u f, \ \ s.t.\ f \perp \ , \ ||f||^2 = n, \ f \text{ as in (23)}, \ \forall v_i \in \mathcal{L}, \ f_i = y_i \ . \tag{25}$$

We relax the problem and perform the optimization over $f \in \mathbb{R}^n$ such that $f \perp \ $. Note that in the unsupervised case, i.e., $\mathcal{L} = \emptyset$, the empty set, the solution to the problem is simply the second eigenvector of L corresponding

to the second smallest eigenvalue. Now, relaxing the constraint[3] on $||f||$ and allowing f_i to mildly deviate from y_i on $v_i \in \mathcal{L}$, we get the following problem:

$$\min_{f \in \mathbb{R}^n} f^T L_u f, \quad s.t. \quad f \perp \quad , \quad \sum_{i=1}^{\ell} (f_i - y_i)^2 \leq \epsilon \ , \tag{26}$$

which is exactly the problem TIKREG solves [2]. The key difference between the relaxed unnormalized formulation in (22) and the normalized formulation in (26) is the constraint $f \perp \quad \Rightarrow \sum_i f_i = 1$, which ensures f lies in the subspace of $\mathbb{R}^n$ orthogonal to $\quad$. The balancing constraint ensures the total score on positive predictions is the same as that on the negative predictions.

4.3 Semi-Supervised Normalized Cut

In the context of normalized cut, for a graph partitioning given by V_1 and V_2, let f be defined as follows

$$f_i = \begin{cases} \sqrt{vol(V_2)/vol(V_1)} & \text{if } v_i \in V_1 \\ -\sqrt{vol(V_1)/vol(V_2)} & \text{if } v_i \in V_2 \ . \end{cases} \tag{27}$$

Following an analysis similar to that of ratio cut, a semi-supervised normalized cut can be posed as the following optimization problem:

$$\min_{V_1, V_2} f^t L_u f, \quad s.t.\ Df \perp \quad , \ f^T D f = vol(V) \ , \quad f \text{ as in (27)}, \forall v_i \in \mathcal{L}, f_i = y_i \ . \tag{28}$$

First, we relax the problem and perform the optimization over $f \in \mathbb{R}^n$ such that $f \perp \quad$. With $g = D^{1/2} f$, the relaxed problem is

$$\min_{g \in \mathbb{R}^n} g^T D^{-1/2} L_u D^{-1/2} g \ \ s.t.\ g \perp D^{1/2} \ , \ ||g||^2 = vol(V), \ \forall v_i \in \mathcal{L}, g_i = D^{1/2} y_i. \tag{29}$$

Note that if $\mathcal{L} = \emptyset$, then the solution to the problem is simply the second eigenvector of the symmetrically normalized Laplacian $L_s = D^{-1/2} L_u D^{-1/2}$ corresponding to the second smallest eigenvalue. Now, relaxing the constraint on $||g||$ and allowing g_i to mildly deviate from $D^{1/2} y_i$ on $v_i \in \mathcal{L}$, we get the following problem:

$$\min_{g \in \mathbb{R}^n} g^T L_s g \ s.t.\ g \perp D^{1/2} \ , \ \sum_{i=1}^{\ell} ||g_i - D^{1/2} y_i||^2 \leq \epsilon \ . \tag{30}$$

Algorithms for the above formulation have not been explored in the literature. The formulation is nearest to that of CK, but not the same since CK is a heterogeneous method which uses the normalized Laplacian for embedding, and then applies a classification algorithm on the embedding. It is also similar to

[3]Since the final prediction depends on sign(f_i), the norm constraint $||f||^2 = n$ does not have an effect on the accuracy.

LGC [21], although the constraint in LGC includes all points with $y_i = 0$ for $i = (\ell + 1), \ldots, n$ and does not involve the $D^{1/2}$ scaling on y_i in the constraint.

There has been notable attempts in the literature to directly solve some of the semi-supervised graph cut problems [4, 5]. Among such methods, the spectral graph transducer (SGT) [12] solves a problem closely related to the semi-supervised ratio cut problem, and reduces to TIKREG [2] under certain assumptions.

5 Semi-Supervised Embedding

In this section, we show how label propagation methods can be viewed as doing semi-supervised embedding. The geometric perspective helps in identifying relationships between existing embedding and label propagation methods, e.g., between Laplacian Eigenmaps [3] and Gaussian Fields [22]. More generally, we derive a new family of label propagation methods based on existing embedding methods, including Locally Linear Embedding (LLE) [14], Local Tangent Space Alignment (LTSA) [20] and Laplacian Eigenmaps (LE) [3]. While all such methods can be seen as a special case of the GLP formulation, they differ in the details—in particular, in the choice of the postive semi-definite matrix L and nature of constraints. Since our exposition is focussed on two class classification, the embedding will always be on $\mathbb{R}$, a one dimensional space.

5.1 Non-linear Manifold Embedding

Manifold embedding methods obtain a lower dimensional representation of a given dataset such that some suitable neighborhood structures are preserved. In this section we briefly review three popular embedding methods and demonstrate that their semi-supervised generalizations solve a variant of the GLP formulation.

Locally Linear Embedding (LLE): In LLE, the assumption is that each point in the high-dimensional space can be accurately approximated by a locally linear region. In particular, the neighborhood dependencies are estimated by solving $\min_W \sum_i ||x_i - \sum_{j \in \mathcal{N}_i} w_{ij} x_j||^2$, such that $\sum_{j \in \mathcal{N}_i} w_{ij} = 1$, where $\mathcal{N}_i$ is the set of neighboring points of x_i. Then W is used to reconstruct the points in a lower-dimensional space by solving:

$$\min_{f \in \mathbb{R}} \sum_i ||f_i - \sum_j w_{ij} f_j||^2 \ , \quad s.t. \ \ f \perp \ \ , \ ||f||^2 = n \ . \tag{31}$$

Letting $M = (I - W)^T (I - W)$, which is positive semi-definite and can be viewed as an iterated Laplace operator [3], we can rewrite the objective function as:

$$\min_f \ \ f^T M f \ \ , \quad s.t. \ \ f \perp \ \ , \ ||f||^2 = n \ . \tag{32}$$

Laplacian Eigenmaps (LE): LE is based on the correspondence between the graph Laplacian and the Laplace Beltrami operator [3]. The symmetric weights

between neighboring points are typically computed using the RBF kernel as $w_{ij} = \exp(-||x_i - x_j||^2/\sigma^2)$. Then W is used to reconstruct the points in a lower-dimensional space by solving:

$$\min_f \ \frac{1}{2}\sum_{i,j} w_{ij}(f_i - f_j)^2 \ , \quad s.t. \ \ f \perp D \quad , \ \ f^T D f = I \ . \tag{33}$$

Using (2), the objective function is $f^T L_u f$. Letting $g = D^{1/2} f$, with $M = L_s = D^{-1/2} L_u D^{-1/2}$ we can express the objective function as

$$\min_g \ g^T M g \ , \quad s.t. \ \ g \perp D^{1/2} \quad , \ \ ||g||^2 = 1 \ . \tag{34}$$

Local Tangent Space Alignment (LTSA): In LTSA, the tangent space at each point is approximated using local neighborhoods and a global embedding is obtained by aligning the local tangent spaces. If X_i^N denotes the matrix of neighbors of x_i, then it can be shown [20] that the principal components of X_i^N give an approximation to the tangent space of the embedding f_i. Let $g_{i1}, \ldots, g_{ik}$ be the top k principal components for X_i^N. Let $G_i = [e/\sqrt{k}, g_{i1}, \ldots, g_{id}]^T$. If $\mathcal{N}_i$ are the indices of the neighbors of x_i, submatrices of the alignment matrix M are computed as $M(\mathcal{N}_i, \mathcal{N}_i) \leftarrow M(\mathcal{N}_i, \mathcal{N}_i) + I - G_i G_i^T$ for $i = 1, \ldots, n$. Finally, using M, which is guaranteed to be positive semidefinite, an embedding is subsequently obtained by minimizing the alignment cost:

$$\min_f \ f^T M f \ , \quad s.t. \ f \perp \quad , \ \ ||f||^2 = n. \tag{35}$$

We refer the reader to [20] for a detailed analysis of LTSA.

5.2 Semi-Supervised Embedding

In this section, we consider two variants of semi-supervised embedding and its applications to label propagation. The variants differ in whether they consider the constraints associated with the corresponding unsupervised embedding problem. As discussed in Section 5.1, there are typically two types of constraints: $f \perp A$, where $A = I$ or $D^{1/2}$, and $||f||^2 = c$, a constant. Since the prediction is based on $\text{sign}(f_i)$, the norm constraint does not play any role in a classification setting, and will be ignored for our analysis. The two variants we consider are based on whether $f \perp A$ is enforced or not, in addition to the constraints coming from the partially labeled data.

5.2.1 Unconstrained Semi-Supervised Embedding

Following [19], we want to obtain an embedding $f = [f_\ell \ \ f_u]^T$, where the exact embeddings of the first ℓ points are known and given by y_ℓ.[4] The objective for semi-supervised embedding is given by

$$\min_f \ f^T M f \ , \quad s.t \ \ f_\ell = y_\ell \ , \tag{36}$$

[4]While the constraints can be relaxed to consider $\sum_{i=1}^{\ell}(f_i - y_i)^2 \leq \epsilon$, we do not focus on the general case here.

where M is a suitable positive semi-definite matrix. Since f_ℓ is fixed, the problem can be cast in terms of block matrices as:

$$\min_{f_u} \begin{bmatrix} f_\ell^T & f_u^T \end{bmatrix} \begin{bmatrix} M_{\ell\ell} & M_{\ell u} \\ M_{u\ell} & M_{uu} \end{bmatrix} \begin{bmatrix} f_\ell \\ f_u \end{bmatrix} \ , \tag{37}$$

Setting the first derivative to zero one obtains:

$$f_u = -M_{uu}^{-1} M_{u\ell} y_\ell \ . \tag{38}$$

In the context of label propagation for a 2-class classification setting, we will have $y_i = +1$ or $y_i = -1$ for $i = 1, \ldots, \ell$. In other words, *the labeled points are being embedded to its true class label, and the rest will be embedded while trying to maintain the neighborhood structure.* For LLE, $M = (I - W)^T (I - W)$ and we call the corresponding label propagation algorithm LLELP. Similarly, for LTSA, M is as discussed in Section 5.1, and the corresponding algorithm will be called LTSALP. For unconstrained LE from (33), $M = L_u$, and the corresponding algorithm will be called LELP. For LELP, since $M = L_u$, the unnormalized Laplacian, from (38) we have

$$\begin{aligned} f_u &= -L_{uu}^{-1} L_{u\ell} y_\ell = -(D_{uu} - W_{uu})^{-1} (D_{u\ell} - W_{u\ell}) y_\ell \\ &= (D_{u\ell} - W_{u\ell})^{-1} W_{u\ell} y_\ell \ , \end{aligned}$$

since $D_{u\ell} = 0$ as D is a diagonal matrix. We note that the solution is exactly the same as that for GF as in (8) implying *the equivalence of GF and LELP.*

5.2.2 Constrained Semi-Supervised Embedding

In this section, we consider embedding problems when the orthogonality constraint of the form $f \perp A$ is enforced. In particular, we consider the following problem

$$\min_f \ f^T M f \ \ , \quad s.t. \ \ \sum_i^\ell (f_i - y_i)^2 \leq \epsilon \ , \ \ f \perp A \ \ , \tag{39}$$

where $A = I$ for LLE and LTSA, and $A = D^{1/2}$ for LE. Let α and μ be the Lagrange multipliers for the two constraints respectively. The first order necessary conditions obtained from the Lagrangian corresponding to (39) yields

$$f = (M + \alpha I_k)^{-1} (\alpha y + \mu A \ /2) \tag{40}$$

Since ${}^T A^T f = 0$, a direct calculation gives the optimal Lagrange multiplier as

$$\mu = -2\alpha \frac{{}^T A^T (M + \alpha I_k)^{-1} y}{{}^T A^T (M + \alpha I_k)^{-1} A} \ . \tag{41}$$

For LLE, $M = (I - W)^T (I - W)$ and $A = I$, and we call the corresponding algorithm LLELPC. For LTSA, M is as discussed in Section 5.1 and $A = I$, and we call the corresponding algorithm LTSALPC. For LE as in (34), $M = L_{sym} = I - D^{1/2} W D^{1/2}$ and $A = D^{1/2}$ and we call the corresponding algorithm LELPC.

6 Experiments

In this section we provide an empirical evaluation of label propagation methods discussed in this paper. To our knowledge, this is the most comprehensive empirical comparison of the methods till date. Our experiments are divided into two parts: First we compare seven methods on 12 benchmark data sets in terms of their accuracy; later, we take a closer look at the performance of the new label propagation methods obtained from the perspective of semi-supervised manifold embedding. For the first set of experiments, the methods we consider include 6 standard methods: GF, LNP, CK, GWEM, TIKREG and LGC, as discussed in Section 3. In addition, we include LTSALP, the novel approach based on semi-supervised LTSA embedding. For the second set of experiments, the methods we consider are the 6 embedding based methods introduced in Section 5.

Methodology: We conducted our experiments on 12 well known benchmark data sets. They include the following 7 UCI data sets: Ionoshpere, Hepatitis, Cancer, Pima, Wine, USPS (1-4 only), and Letter (E and F only). In addition we also ran experiments on 5 text datasets, which are all subsets of the 20Newsgroup data set: Different100, Similar100, Same100, Different1000 and Same1000 [1]. Each dataset contains a subset of 3 newsgroups with varying degree of difficulty for classification. Different100 (1000) includes alt.atheism, rec.sport.baseball, and sci.space, and are hence easy to separate; Same100 (1000) includes comp.graphics, comp.os.ms-windows, comp.windows.x, and are difficult to separate; whereas Similar100 includes talk.politics.guns, talk. politics.mideast, talk.politics.misc, and are moderately difficult to separate. For each method and each data set we ran five-fold semi-supervised cross validation. In particular, the training points were chosen from four folds and the test error was measured on the fifth fold. All points were used to construct the neighborhood graph. Further, for each approach involving parameters, e.g., CK using SVMs with RBF kernel, parameter values were selected by cross-validation.

6.1 Experimental Results

The performance of 6 methods from the literature as well as the novel LTSALP in shown in Figures 1 and 3 (left panel).[5] All results reported are on the test set. To avoid clutter, we display the results for the top five methods based on average test-set error in Figure 1. We make the following observations based on the results: (i) There is no dominating method across all datasets. However, there are a set of methods that seem to be consistently among the top few, and the performance of the top few methods are typically close; (ii) The new method LTSALP is the top performing method in most of the UCI datasets. However, it performs poorly on the text datasets possibly indicating that the idea of aligning the tangent spaces may have to be suitably modified for sparse high-dimensional datasets; (iii) TIKREG is among the most consistent meth-

[5] We could not include plots on all datasets due to space constraints.

ods on the text datasets. While it is typically among the top 5 on the UCI data sets, it is not the top performing method; (iv) In spite of an issue with its formulation, LNP is found to be quite consistent across several UCI and text datasets; (v) When CK does well, it outperforms all the other methods. However, it does not have a consistent performance, which probably can be addressed by more thorough cross-validation over its parameter choices; (vi) GF seems to be slow starter, not performing well for small number of labeled points, but improving significantly as the number of labeled points increase; and (vii) GWEM does not demonstrate a consistent performance and seems quite sensitive to the predefined number of steps in the random walk. In summary, while several of the methods perform well on some datasets, the embedding based novel method LTSALP seems to outperform the existing methods in most of the UCI datasets. Further, LNP and TIKREG seems to be quite consistent in terms of their performance respectively on UCI and text datasets.

6.2 Semi-Supervised Embedding Methods

We now compare the six label propagation methods based on semi-supervised manifold embedding. In particular we examine both variants of Laplacian Eigenmaps based Label Propagation (LELP,LELPC), Locally Linear Embedding based Label Propagation (LLELP,LLELPC), and Local Tangent Space Alignment based Label Propagation (LTSALP,LTSALPC). Note that LTSALP has already been compared to the existing methods in Figures 1 and 3 to serve as a reference. We compare the methods on the UCI datasets, and show representative plots in Figure 2.[6] Based on our experiments, we observe that LELP and LTSALP performed most consistently well, while LLELP did well only on specific data sets such as Wine. The performance of LLELP seems to be more sensitive to the geometry of a data set. The effect of the orthogonality constraint on these methods can be understood when comparing the restults for Cancer and Hepatitis in Figure 2. While the performance is clearly affected by the constraints, it does not necessarily result in improved performance. For example, the constraints lead to better performance in Hepatitis but worse performance in Cancer.

For the embedding methods, the quality of semi-supervised label propagation seems to depend on how well the unsupervised embedding preserves the class separation. Figure 5 illustrates the difference between unsupervised embedding and semi-supervised embedding on Wine. Note that the unsupervised embedding obtained from LLE and LTSA maintains the class separation better than LE for this particular dataset. When semisupervision is added, the embedding obtained from all the methods changes suitably. The class separation is most clear in LLELP followed by LTSALP and LELP, a fact reflected in the test-set error rates in Figure 3 (right panel). In general, label propagation based on a semi-supervised embedding method works well if the geometric structure of the class labels is well aligned with the biases of the embedding method.

[6] We report results on 2-class problems in Figure 2. Since Wine is a 3-class dataset, we constructed a 2-class subset Wine(2) for these experiments.

7 Conclusions

In this paper, we have developed a unified perspective to a large set graph-based semi-supervised learning methods. We have demonstrated that almost all such methods minimize a quadratic form involving a graph Laplacian under suitable constraints primarily derived from the label information. Further, we showed that semi-supervised approaches to graph-cuts or manifold embeddings lead to the same computational problem. We have effectively provided a recipe for converting embedding methods to label propagation algorithms. Our extensive empirical evaluation reveals that while there are no clear winners in terms of performance, certain methods seem consistent across several datasets including some of the new embedding based methods introduced in this paper. Our analysis improves our theoretical understanding of an important class of methods in semi-supervised learning, introduces several new methods to solve the problem, and, to our knowledge, provides the first comprehensive empirical performance evaluation of this family of models. The competitive performance of the embedding based label propagation methods such as LTSALP provide strong incentive to investigate the family of methods further.

References

[1] A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra. Clustering on the unit hypersphere using von Mises-Fisher distributions. *JMLR*, 6:1345–1382, 2005.

[2] M. Belkin, I. Matveeva, and P. Niyogi. Regularization and semi-supervised learning on large graphs. In *COLT*, 2004.

[3] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

[4] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *ICML*, 2001.

[5] A. Blum, J. Lafferty, M. R. Rwebangira, and R. Reddy. Semi-supervised learning using randomized mincuts. In *ICML*, 2004.

[6] O. Chapelle, B. Scholkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, 2006.

[7] O. Chapelle, J. Weston, and B. Schölkopf. Cluster kernels for semi-supervised learning. In *NIPS*, 2003.

[8] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

[9] C. Ding, H. D. Simon, R. Jin, and T. Li. A learning framework using green's function and kernel regularization with application to recommender system. In *KDD*, 2007.

[10] P. G. Doyle and L. J. Snell. *Random walks and electric networks*. Mathematical Assn of America, 1984.

[11] L. Hagen and A. Kahng. New spectral methods for ratio cut partitioning and clustering. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pages 1074–1085, 1992.

[12] T. Joachims. Transductive learning via spectral graph partitioning. In *ICML*, 2003.

[13] B. Mohar. The laplacian spectrum of graphs. In *Graph Theory, Combinatorics, and Apps*, Wiley, 1991.

[14] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.

[15] J. Shi and J. Malik. Normalized cuts and image segmentation. In *CVPR*, 1997.

[16] M. Szummer and T. Jaakkola. Partially labeled classification with markov random walks. In *NIPS*, 2001.

[17] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

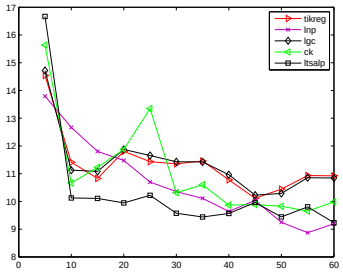
[18] F. Wang and C. Zhang. Label propagation through linear neighborhoods. In *ICML*, 2006.

[19] X. Yang, H. Fu, H. Zha, and J. Barlow. Semi-supervised nonlinear dimensionality reduction. In *ICML*, 2006.

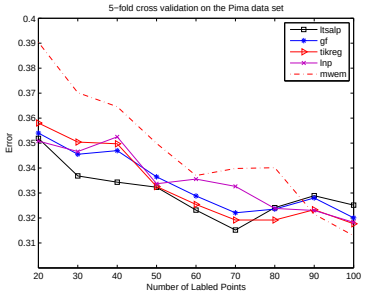
[20] Z. Zhang and H. Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM J. on Scientific Computing*, 26(1):313–338, 2005.

[21] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS*, 2003.

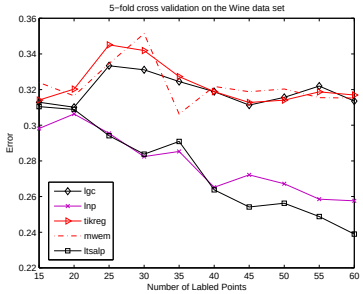
[22] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In *ICML*, 2003.



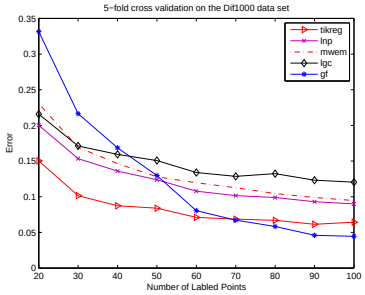
(a) Cancer



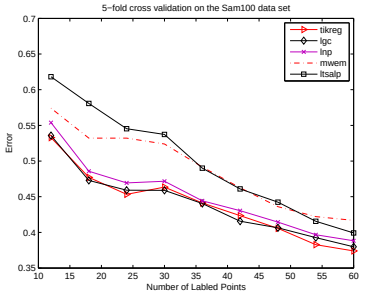
(b) Pima



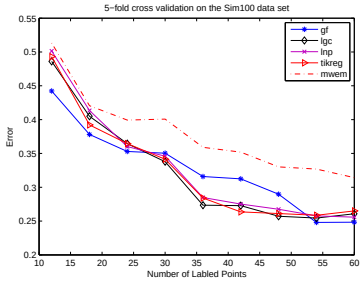
(c) Wine



(d) Different1000

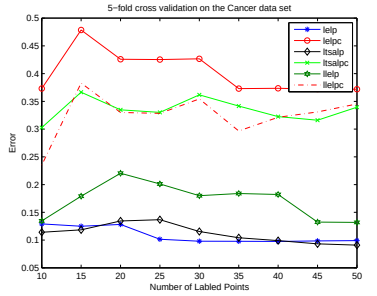


(e) Same100

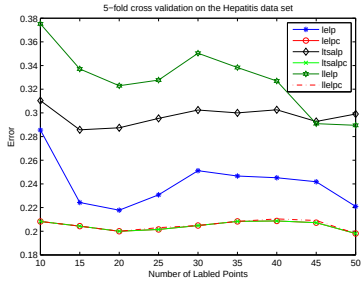


(f) Similar100

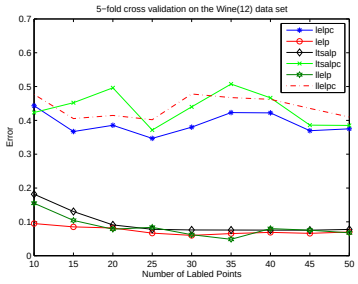
Figure 1: Performance on UCI and text datasets. Each plot shows the top 5 methods out of 6 standard methods: GF, LNP, CK, GWEM, TIKREG, LGC, and the new method LTSALP.



(a) Cancer



(b) Hepatitis



(c) Wine(2)

Figure 2: Comparison of embedding based label propagation methods: LELP(C), LTSALP(C), LLELP(C).

	GWEM	GF	LGC	LNP	TIKREG	CK	LTSALP
Ionosphere	56.45	48.9	**36.21**	**36.21**	42.85	**36.21**	44.17
Hepatitis	24.96	23.20	22.24	27.36	22.40	**20.48**	30.40
Cancer	9.48	12.37	13.75	11.52	12.55	13.75	**9.20**
Pima	34.99	33.65	36.02	33.37	33.26	35.04	**33.23**
Wine	37.02	34.55	35.89	34.67	35.04	41.14	**33.69**
USPS	6.83	6.47	10.58	6.96	5.22	**2.03**	2.25
Letter	10.09	9.46	9.74	9.83	9.75	50.45	**9.15**
Dif100	13.48	**10.83**	14.77	13.86	12.50	7.27	12.58
Dif1000	16.91	21.64	17.11	15.33	**10.12**	36.62	43.66
Sam100	52.38	52.23	45.89	47.16	**46.34**	64.62	53.73
Sam1000	**42.50**	53.36	45.24	45.53	44.34	53.22	61.00
Sim100	40.07	**35.04**	33.78	34.59	34.14	58.14	42.01

Figure 3: Performance comparisons with 30 labeled points on 12 datasets and 6 methods. LTSALP and CK perform well on UCI datasets, TIKREG and GF perform well on text datasets. 5.

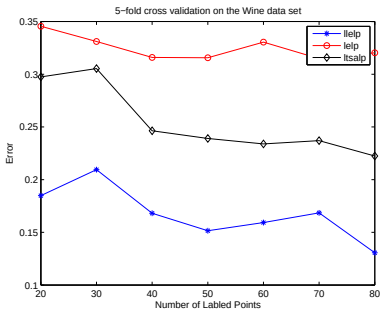
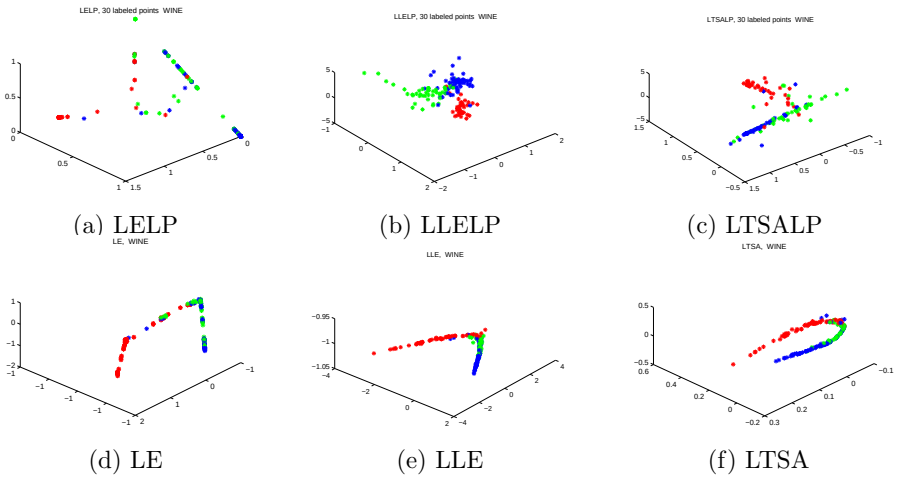


Figure 4: Embedding based LP methods on Wine corresponding to the Figure 5.



(a) LELP (b) LLELP (c) LTSALP

(d) LE (e) LLE (f) LTSA

Figure 5: Unsupervised and unconstrained semi-supervised embedding on Wine. The prediction performance (top) is better if the unsupervised embedding (bottom) keeps the classes separate.