Laplacian Margin Distribution Boosting for Learning from Sparsely Labeled Data

Tao Wang[1,2], Xuming He[1,2], Chunhua Shen[2,3], and Nick Barnes[1,2]

[1]School of Engineering, Australian National University, ACT 2600, Australia
[2]NICTA*, Canberra Research Laboratory, ACT 2600, Australia
[3]Australian Center for Visual Technologies, University of Adelaide, SA 5005, Australia
Email: {*tao.wang, xuming.he, chunhua.shen, nick.barnes*}@*nicta.com.au*

Abstract

Boosting algorithms attract much attention in computer vision and image processing because of their strong performance in a variety of applications. Recent progress on theory of boosting algorithms suggests a close link between good generalization and the margin distrubtion of the classifier w.r.t. a dataset. In this paper, we propose a novel data-dependent margin distribution learning criterion for boosting, termed Laplacian MDBoost, which utilizes the intrinsic geometric structure of dataset. One key aspect of our method is that it can seamlessly incorporate unlabeled data by including a graph Laplacian regularizer. We derive a dual formulation of the learning problem that can be efficiently solved by column generation. Experiments on various datasets validate the effectiveness of the new graph Laplacian based learning criterion on both supervised and unsupervised learning settings. We also show that the performance of our algorithm outperforms the state-of-the-art semi-supervised learning algorithms on a variety of inductive inference tasks, including real world video segmentation.

1. Introduction

Boosting algorithms have achieved great popularity in a spectrum of computer vision problems due to their good generalization, robust performance, and intrinsic feature selection mechanism. Despite their success, the classic AdaBoost and its variants suffer from two disadvantages in real world applications. First, the exponential loss and greedy nature of its learning algorithm tend to generate a classifier with many weaker learners, which can be inefficient and prone to overfitting. Also, boosting usually requires a large number of training examples to achieve high accuracy. However, ground truth labeling is scarce and difficult to obtain in practice.

Our work aims to address those issues within a unified framework based on the margin distribution theory of boosting [17, 15, 19]. One key observation is that the appealing properties of boosting are closely related to the *margin distribution* (MD) instead of solely the minimum margin [15] – which are commonly used in margin-based classification. It has been shown that the margin distribution seems to play more important role in attaining better overall performance empirically and provides a tighter generalization bound in theory [7, 15]. Therefore, several papers advocate optimizing MD-based criteria to improve the test accuracy of boosting-like algorithms [11, 7, 18]. Notably, [18] proposed a totally corrective boosting, termed MDBoost, to maximize the average margin while minimizing margin variance. The new boosting method achieves competitive performance and faster convergence (*i.e*., fewer weak learners) on several classification tasks.

However, while the additional margin variance provides a better measure of the margin distribution, the overall criterion is based on the second-order statistics only, and thus lacks capacity of capturing finer-scale structure of the distribution. Inspired by manifold learning, we propose to improve MDBoost by incorporating a local representation of margin variance, in which only neighboring points on the data manifold contribute to the variance computation. Intuitively, the data-dependent margin variance may give a better description of the margin distribution. Due to its resemblance to the Laplacian Eigenmap [1], we refer to this new boosting approach as *Laplacian MDBoost*.

*NICTA is funded by the Australian Government as represented by the Department of Boardband, Communications and the Digital Economy and the Australian Research Council through the ICT Center of Excellence program.

More importantly, our learning criterion can be naturally generalized to semi-supervised learning scenario. Given both labeled and unlabeled data, we augment the supervised learning criterion with a graph Laplacian-based regularization term, which encourages the classifier outputs on unlabeled data to satisfy the data manifold constraint. This combined learning criterion provides a coherent framework and admits a simple convex quadratic dual formulation such as MDBoost. We employ a column-generation (CG) based optimization procedure to incrementally add informative weak learners, yielding a boosting-like algorithm.

We empirically demonstrate that the supervised Laplacian MDBoost is better than or comparable to AdaBoost(-CG) [19], LPBoost [6] and MDBoost in terms of classification performance on most UCI datasets [14]. In addition, we design a set of semi-supervised learning tasks based on UCI datasets and YouTube Celebrities Face datasets [9], and compare the semi-supervised Laplacian MDBoost with two recent approaches to learning from partially labeled data: LLGC [23] and SemiBoost [12]. The results show the semi-supervised Laplacian MDBoost outperforms the baseline methods on most of datasets.

We organize the rest of our paper as follows. In next section, we discuss the background and related work. Section 3 derive the supervised and semi-supervised Laplacian MDBoost based on the dual formulation of optimizing a novel margin distribution cost. We demonstrate the performance of our approach by comparing with several recent (semi-)supervised boosting methods on UCI and video segmentation tasks in Section 4. Finally, Section 5 summarizes our conclusion and discusses the future work.

2. Related Work

Boosting has attracted increasing attention in the machine learning community in the last ten years due to its performance and efficiency in classification. One way of deciphering the success of boosting lies in margin theory [17]. Several recent papers, such as LPBoost [6], adopt the minimum margin as an alternative learning criterion for boosting. Ryyzin and Schapire [15] point out that the generalization performance of boosting algorithms may depend more on the margin distribution instead of the minimum margin. Based on this observation, Shen *et al*. proposed MDBoost (margin distribution boosting) and achieved promising classification performance by directly maximizing the average margin and minimizing the margin variance [18]. Our work extends the MDBoost so that higher-order statistics and unlabeled data can be utilized to learn a better boosted classifier.

There has been a large amount of literature in semi-supervised learning and we refer the readers to the recent book [4] for a comprehensive review. Generally, semi-supervised learning methods can be categorized into either transductive or inductive based on the nature of inference. Transductive algorithms can only predict the labels of the data seen during training. Typical approaches include label propagation [24] and LLGC [23]. Inductive methods, on the other hand, can be used to predict the labels of data that are unseen during training, which includes co-training [2] and SemiBoost [12] as examples. Our approach belongs to the inductive category and is based on the "manifold assumption" in Laplacian Eigenmaps [1].

Several work have extended supervised boosting algorithms to semi-supervised setting. Semi-supervised MarginBoost [3] generalizes the margin concept to unlabeled data, and minimizes a margin-based loss by functional gradient descent. Chen and Wang also minimize the margin-based loss and introduce additional local smoothness into regularization in the Regularized Boost [5]. SERBoost [16] aims at scaling up to large dataset by using expectation regularization. In SemiBoost[12], Mallapragada *et al*. boost any supervised classifier by iteratively relabeling the unlabeled data. Unlike those existing approaches, our algorithm optimizes the margin distribution directly within a totally corrective framework, while incorporating manifold regularization on both labeled and unlabeled data coherently.

3. Our Approach

3.1. Margin Distribution and Laplacian MDBoost

We first review the key ideas of the margin distribution boost (MDBoost) in [18] and introduce some notations for formulating our Laplacian MDBoost. Let $\mathcal{D}_l = \{(\boldsymbol{x}_i, y_i)\}_{i=1,\cdots,M}$ be the training data set, where $\boldsymbol{x}_i \in \mathcal{X}$ is the input feature vector and $y_i \in \{-1, +1\}$ is the output label. Given the training data, our goal is to train a classifier to assign binary label to any input vector $\boldsymbol{x}$. In the setting of boosting methods, the classifier consists of a weighted combination of weak learners.

More specifically, denote $h(\cdot) \in \mathcal{H}$ as a weak learner that maps an input vector $\boldsymbol{x}$ into binary output. We assume we choose K weak learners from the set $\mathcal{H}$ in our boosted classifier, and define the matrix $H \in \mathbb{Z}^{M\times K}$ to be all the possible predictions of the training data using weak classifiers. That is, $H_{ij} = h_j(\boldsymbol{x}_i)$ is the label ($\{+1, -1\}$) given by weak classifier $h_j(\cdot)$ on the training example $\boldsymbol{x}_i$. We also use $H_{i:} = [H_{i1}\ H_{i2} \cdots H_{iK}]$ to denote the i-th row of H, which constitutes the output of all the weak classifiers on the training example $\boldsymbol{x}_i$. Let $\boldsymbol{\alpha}$ be the weight vector for the weak learners. We can write the output of the final classifier on any training data $\boldsymbol{x}_i$ as $H_{i:}\boldsymbol{\alpha}$, and the so-called (unnormalized) *margin* at data $\boldsymbol{x}_i$ is defined as $y_i H_{i:}\boldsymbol{\alpha}$.

Based on the margin distribution theory of boosting, MDBoost directly maximizes the average margin and minimizes the margin variance. Specifically, let ρ_i denote the unnormalized margin for the ith example datum, *i.e.*,

$\rho_i = y_i H_{i:}\boldsymbol{\alpha}$, $\forall i = 1, \cdots, M$. The cost function and the learning problem in MDBoost can be written as follows:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2(M-1)} \sum_{i>j} (\rho_i - \rho_j)^2 - \sum_{i=1}^{M} \rho_i$$
$$\text{s.t. } \boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \mathbf{1}^\top \boldsymbol{\alpha} = D, \tag{1}$$

where D is a regularization parameter. By defining a matrix $A \in \mathbb{R}^{M \times M}$, where

$$A = \begin{bmatrix} 1 & -\frac{1}{M-1} & \cdots & -\frac{1}{M-1} \\ -\frac{1}{M-1} & 1 & \cdots & -\frac{1}{M-1} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{M-1} & -\frac{1}{M-1} & \cdots & 1 \end{bmatrix},$$

the optimization problem can be rewritten into the following form:

$$\begin{aligned} &\min_{\boldsymbol{\alpha}} \ \tfrac{1}{2}\boldsymbol{\rho}^\top A \boldsymbol{\rho} - \mathbf{1}^\top \boldsymbol{\rho}, \\ &\text{s.t. } \boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \mathbf{1}^\top \boldsymbol{\alpha} = D, \\ &\qquad \rho_i = y_i H_{i:}\boldsymbol{\alpha}, \forall i = 1, \cdots, M. \end{aligned} \tag{2}$$

It has been shown [19] the problem in (2) can be efficiently solved by considering its dual form, *i.e.*,

$$\begin{aligned} &\min_{r, \boldsymbol{u}} \ r + \tfrac{1}{2D}(\boldsymbol{u} - \mathbf{1})^\top A^{-1} (\boldsymbol{u} - \mathbf{1}), \\ &\qquad \text{s.t. } \sum_{i=1}^{M} u_i y_i H_{i:} \preccurlyeq r \mathbf{1}^\top. \end{aligned} \tag{3}$$

The form of the dual problem allows us to incrementally search the solution space by the column generation technique. At each iteration, we obtain a new weak classifier through searching the most violated constraint:

$$h'(\cdot) = \operatorname*{argmax}_{h(\cdot)} \ \textstyle\sum_{i=1}^{M} u_i y_i h(\boldsymbol{x}_i). \tag{4}$$

While the MDBoost learning cost incorporates the margin variance information, the global variance can be restrictive and cannot describe the finer structure of the distribution beyond the second order statistics. We propose to use the "local" version of variance that considers the geometric property of the data manifold. Specifically, we adapt the concept of graph Laplacian of data manifold [1], and use a data-dependent margin variance in the MDBoost learning criterion:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2(M-1)} \sum_{i>j} w_{ij} (\rho_i - \rho_j)^2 - \sum_{i=1}^{M} \rho_i$$
$$\text{s.t. } \boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \mathbf{1}^\top \boldsymbol{\alpha} = D, \tag{5}$$

where $w_{ij} = \exp(\frac{-||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{t})$ is defined on a neighborhood graph. We refer to the new learning problem in (5) as Laplacian MDBoost.

Note that if we define the matrix $A = \{A_{ij}\}$ by the following terms,

$$A_{ij} = \begin{cases} w_{ij}, & \text{if } i \neq j, \\ \sum_{k=1, k \neq i}^{M} w_{ik}, & \text{if } i = j, \end{cases} \tag{6}$$

then we can derive new primal and dual problems with the same form as in (2) and (3). The dual problem can be solved with a column generation method such as in MDBoost. We notice that both MDBoost and Laplacian MDBoost in their dual form are regularized hard-margin LPBoost, but have different types of regularizer.

3.2. Semi-supervised Laplacian MDBoost

The main idea in Laplacian MDBoost, which makes use of the geometric property of data distribution, can be naturally extended to semi-supervised learning setting. Assume we have an additional unlabeled data set $\mathcal{D}_u = \{\boldsymbol{x}_i, i = M+1, \cdots, N\}$ and would like to use it to help improve the classification performance. Similar to [1], we incorporate a graph Laplacian-based regularization term into our objective function, which imposes a smoothness constraint over the class output on the unlabeled data w.r.t. the empirical estimate of data manifold structure.

Given a neighborhood graph defined on the dataset, we can define the graph Laplacian as $L = D - W$ where W is a $N \times N$ matrix and $w_{ij} = \exp(\frac{-||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{t})$, if x_i and x_j are adjacent and zero otherwise. D is a diagonal degree matrix given by $D_{ii} = \sum_i w_{ij}$. A smoothness regularization term on the class output $f(\boldsymbol{x})$ can be written as $\boldsymbol{f^t} L \boldsymbol{f} = \sum_{i,j=1}^{n} (f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j))^2 w_{ij}$.

In Laplacian MDBoost, the class prediction $f(\boldsymbol{x}_i)$, denoted by f_i, is the combined prediction of all weak classifiers for the ith example datum, *i.e.* $f_i = H_{i:}\boldsymbol{\alpha}$, $\forall i = 1, \cdots, M$. By adding the smoothness penalty as a regularization term into the primal objective function, we derive the following learning criterion for semi-supervised Laplacian MDBoost:

$$\min_{\boldsymbol{\alpha}} \frac{\sum_{i>j} w_{ij} (\rho_i - \rho_j)^2}{2(M-1)} + C \sum_{i>j} w_{ij} (f_i - f_j)^2 - \sum_{i=1}^{M} \rho_i$$
$$\text{s.t. } \boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \mathbf{1}^\top \boldsymbol{\alpha} = D, \tag{7}$$

where D is also a regularization parameter as in (1). Here we have two quadratic terms: the first one corresponds to the margin variance of labeled data, while the second is the smoothness penalty on all data (including the labeled and unlabeled). C is the tradeoff parameter between the two terms.

Denote A_1 as the matrix defined in (6) on all the data points (including labeled and unlabeled), and A_2 as the $M \times M$ upper left corner of A_1 (suppose the data is sorted that the labeled data are the first M elements when defining the graph Laplacian), our optimization problem can be rewritten into a concise form:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \;& \tfrac{C'}{2}\boldsymbol{f}^\top A_1 \boldsymbol{f} + \tfrac{1}{2}\boldsymbol{\rho}^\top A_2 \boldsymbol{\rho} - \mathbf{1}^\top \boldsymbol{\rho}, \\ \text{s.t. } & \boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \mathbf{1}^\top \boldsymbol{\alpha} = D, \\ & \rho_i = y_i H_{i:}\boldsymbol{\alpha}, \forall i = 1, \cdots, M, \\ & f_i = H_{i:}\boldsymbol{\alpha}, \forall i = 1, \cdots, N. \end{aligned} \tag{8}$$

where M refers to the number of labeled examples, while N is the number of all (labeled and unlabeled) examples. C' is equivalent to C up to a constant.

Notice that the new semi-supervised Laplacian MDBoost objective has a similar form to the supervised version, thus we can derive its dual formulation as follows. The Lagrangian of the convex optimization problem in (8) is written as

$$\begin{aligned} & L(\boldsymbol{\alpha}, \boldsymbol{\rho}, \boldsymbol{f}, \boldsymbol{u}, \boldsymbol{v}, r, \boldsymbol{q}) \\ & = \tfrac{C'}{2}\boldsymbol{f}^\top A_1 \boldsymbol{f} + \tfrac{1}{2}\boldsymbol{\rho}^\top A_2 \boldsymbol{\rho} - \mathbf{1}^\top \boldsymbol{\rho} + r(\mathbf{1}^\top \boldsymbol{\alpha} - D) - \boldsymbol{q}^\top \boldsymbol{\alpha} \\ & + \textstyle\sum_{i=1}^M u_i(\rho_i - y_i H_{i:}\boldsymbol{\alpha}) + \sum_{i=1}^N v_i(f_i - H_{i:}\boldsymbol{\alpha}), \end{aligned} \tag{9}$$

with $\boldsymbol{q} \succcurlyeq \mathbf{0}$. The infimum of L w.r.t. to the primal variable can be computed as

$$\begin{aligned} \inf_{\boldsymbol{\rho}, \boldsymbol{f}, \boldsymbol{\alpha}} L = & \inf_{\boldsymbol{f}} \left[\tfrac{C'}{2}\boldsymbol{f}^\top A_1 \boldsymbol{f} + \boldsymbol{v}^\top \boldsymbol{f}\right] \\ & + \inf_{\boldsymbol{\rho}} \left[\tfrac{1}{2}\boldsymbol{\rho}^\top A_2 \boldsymbol{\rho} + (\boldsymbol{u} - \mathbf{1})^\top \boldsymbol{\rho}\right] - Dr \\ & + \inf_{\boldsymbol{\alpha}} \left[(r\mathbf{1}^\top - \boldsymbol{q}^\top - \textstyle\sum_{i=1}^M u_i y_i H_{i:} - \sum_{i=1}^N v_i H_{i:})\boldsymbol{\alpha}\right]. \end{aligned} \tag{10}$$

Clearly, $r\mathbf{1}^\top - \boldsymbol{q}^\top - \sum_{i=1}^M u_i y_i H_{i:} - \sum_{i=1}^N v_i H_{i:} = \mathbf{0}$ must hold in order to have a finite infimum. Therefore, we have

$$\textstyle\sum_{i=1}^M u_i y_i H_{i:} + \sum_{i=1}^N v_i H_{i:} \preccurlyeq r\mathbf{1}^\top. \tag{11}$$

For the first and second term in (10), the gradient must vanish at the optimum:

$$\frac{\partial \left[\frac{C'}{2}\boldsymbol{f}^\top A_1 \boldsymbol{f} + \boldsymbol{v}^\top \boldsymbol{f}\right]}{\partial f_i} = 0, \forall i = 1, \cdots, N. \tag{12}$$

$$\frac{\partial \left[\frac{1}{2}\boldsymbol{\rho}^\top A_2 \boldsymbol{\rho} + (\boldsymbol{u} - \mathbf{1})^\top \boldsymbol{\rho}\right]}{\partial \rho_i} = 0, \forall i = 1, \cdots, M. \tag{13}$$

This leads to $\boldsymbol{f} = -A_1^{-1}\boldsymbol{v}$; and $\boldsymbol{\rho} = -A_2^{-1}(\boldsymbol{u} - \mathbf{1})$ and the infimum is $-\frac{C'}{2}\boldsymbol{v}^\top A_1^{-1}\boldsymbol{v} - \frac{1}{2}(\boldsymbol{u} - \mathbf{1})^\top A_2^{-1}(\boldsymbol{u} - \mathbf{1})$.

By substituting the results back to (10), we can write the dual problem as:

$$\begin{aligned} \max_{r, \boldsymbol{u}, \boldsymbol{v}} \; & -r - \tfrac{1}{2D}(\boldsymbol{u} - \mathbf{1})^\top A_2^{-1}(\boldsymbol{u} - \mathbf{1}) - \tfrac{C'}{2}\boldsymbol{v}^\top A_1^{-1}\boldsymbol{v}, \\ \text{s.t. } & (11). \end{aligned} \tag{14}$$

We employ a similar column generation strategy to induce weak learners incrementally. At each iteration, we choose a weak learner that violates the constraint most:

$$h'(\cdot) = \operatorname*{argmax}_{h(\cdot)} \sum_{i=1}^M u_i y_i h(\boldsymbol{x}_i) + \sum_{i=1}^N v_i h(\boldsymbol{x}_i). \tag{15}$$

We summarize the proposed algorithm in Algorithm 1.

Algorithm 1: Column generation based Semi-supervised Laplacian MDBoost.

Input: labeled training data $(\boldsymbol{x}_i, y_i), i = 1 \cdots M$; unlabeled training data $\boldsymbol{x}_i, i = M+1 \cdots N$; termination threshold $\varepsilon > 0$; regularization parameter D; maximum number of iterations $T_{\max}$.

Initialization: $N = 0$; $\boldsymbol{\alpha} = \mathbf{0}$; $u_i = \frac{1}{M}, i = 1 \cdots M$; and $v_i = \frac{1}{N}, i = 1 \cdots N$.

for iteration $= 1 : T_{\max}$ **do**

1. Obtain a new base $h'(\cdot)$ by solving (15);
2. Check for optimal solution: **if** $\sum_{i=1}^M u_i y_i h'(\boldsymbol{x}_i) + \sum_{i=1}^N v_i h'(\boldsymbol{x}_i) < r + \varepsilon$, **then** break and the problem is solved;
3. Add $h'(\cdot)$ to the restricted master problem, which corresponds to a new constraint in the dual problem;
4. Solve the dual problem (14) and update r, u_i ($i = 1 \cdots M$) and v_i ($i = 1 \cdots N$).
5. Count weak classifiers $T = T + 1$.

end

Output:

1. Compute the primal variable $\boldsymbol{\alpha}$ from the optimality conditions and the last solved dual problem (primal-dual interior point methods produce $\boldsymbol{\alpha}$ in the meantime);
2. The final strong classifier is $H(\boldsymbol{x}) = \mathbf{sign}\left(\sum_{j=1}^N \alpha_j h_j(\boldsymbol{x})\right)$.

4. Experimental Evaluation

In this section, we evaluate the performance of Laplacian MDBoost and Semi-supervised Laplacian MDBoost by conducting a set of experiments on synthetic and real world datasets. We first present a comparison between the proposed Laplacian MDBoost and several most widely-used supervised boosting algorithms. Following that, we

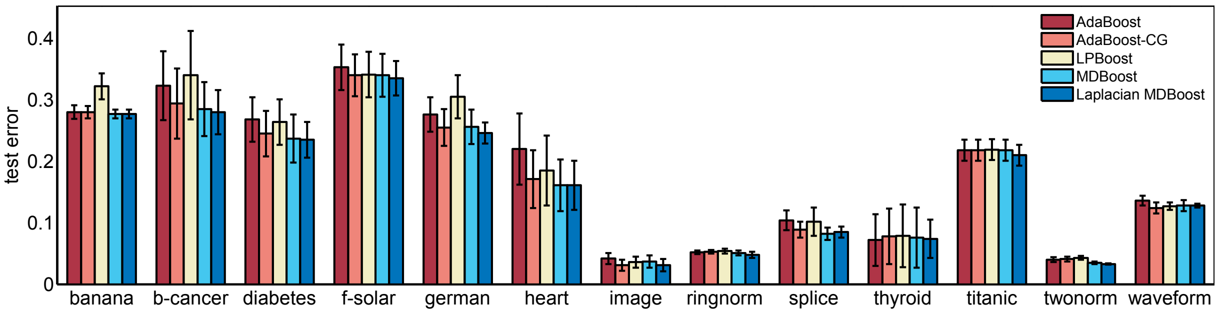

Figure 1. Average test error (with standard deviation) of AdaBoost, AdaBoost-CG, LPBoost, MDBoost and Laplacian MDBoost on 13 UCI benchmark datasets.

design a benchmark of semi-supervised inductive inference tasks by removing certain ratio of training data labels in UCI datasets. We test the proposed Semi-supervised Laplacian MDBoost against two baseline approaches, including LLGC[23] combined with MDBoost and SemiBoost[12]. Finally, we apply our semi-supervised method and two baselines to an object segmentation in video task.

4.1. Datasets and Setup

The first set of our experiments is based on the 13 UCI benchmark datasets from [14][1]. For supervised learning setting, we randomly split each of the UCI datasets into 3 subsets. 60% of the samples are used for training; 20% for cross validation and the rest for testing. For the larger datasets (**ringnorm**, **twonorm** and **waveform**), we randomly select 10% for training, 30% for cross validation and 60% for testing. All experiments are run 30 times for accuracy.

We choose the model hyperparameters by cross validation. The parameter D for AdaBoost-CG and all algorithms in the MDBoost family are chosen from $\{2, 5, 10, 20, 40, 70, 100, 150\}$. The search range of coefficient C for Semi-supervised Laplacian MDBoost and combining LLGC with MDBoost are set to $\{-3, -2, -1, -0.75, -0.5, -0.25, 0, 0.25, 0.5, 0.75, 1, 2, 3\}$ in negative log scale. The trade-off parameter C for LPBoost[6] are chosen similarly. For the graph Laplacian, we let t be proportional to the variance of data and normalize all feature values to $[-10, +10]$. We set parameters of LLGC and SemiBoost to their respective optimal values given by [23] and [12]. For simplicity, we use decision stumps as weak learners in all tests and limit the maximum number of iterations $T_{\max}$ to 1000 (note that all totally-corrective boosting algorithms converge earlier than 100 iterations). The convergence threshold ε are uniformly set to 10^{-5}.

To evaluate the performance of Semi-supervised Laplacian MDBoost on real-world applications, we also choose a subset of the YouTube Celebrities Face Tracking and Recognition Dataset[9], which includes 6 sequences, and apply our method to a semi-supervised object segmentation task.

[1] http://ida.first.fraunhofer.de/projects/bench/

4.2. Laplacian MDBoost for Supervised Learning

To demonstrate the effectiveness of the new Laplacian MDBoost learning criterion, we first test our algorithm in a fully-supervised learning setting. The performance of Laplacian MDBoost is compared with four other boosting algorithms, namely AdaBoost, AdaBoost-CG, LPBoost and MDBoost. The experiments are run on 13 UCI benchmark datasets for 30 times, and average test error with standard deviation are reported in Fig. 1. As we can see, Laplacian MDBoost outperforms its opponents in most cases. This result confirms our intuition and show that local variance is effective in representing the margin distribution.

4.3. Semi-supervised Laplacian MDBoost

We first evaluate the Semi-supervised Laplacian MDBoost on a set of partially labeled datasets derived from UCI benchmark. In this experiment, we followed the setup in Sec. 4.1 and choose randomly 10% of the original training data to keep their labels, while manually removing the labels of the other 90%. Our approach is compared with two other state-of-the-art semi-supervised algorithms: LLGC and SemiBoost. LLGC is widely used in different applications as a transductive algorithm [22, 13]. In contrast, SemiBoost is an inductive yet effective alternative [8, 10]. Note that LLGC is transductive so it does not by default offer the capability for predicting labels unseen during training. Therefore we combine it with MDBoost, by using LLGC first to predict the “fill-in” labels of unlabeled training data, then cascading with MDBoost as if all training data are labeled. For data with “fill-in” labels, we use a cross-validated coefficient during reweight sampling to limit their impact. This method effectively uses LLGC as a mean of manifold regularization while Laplacian MDBoost uses a Laplacian Eigenmap instead.

The results are summarized in Table 1. In 9 out of 13 datasets, utilizing unlabeled data helps to improve test per-

Table 1. Test error and standard deviation (in percentage %) of Laplacian MDBoost (using only labeled data), Semi-supervised Laplacian MDBoost (SemiLap-MDBoost), Learning with Local and Global Consistency combined with MDBoost (LLGC+MDBoost), and SemiBoost on UCI datasets.

	Laplacian MDBoost	**SemiLap-MDBoost**	**LLGC+ MDBoost**	**SemiBoost**
banana	57.1 ± 4.8	$\mathbf{41.6 \pm 3.2}$	51.5 ± 7.4	41.7 ± 2.3
b-cancer	38.5 ± 14.2	$\mathbf{31.4 \pm 9.1}$	34.7 ± 9.2	33.3 ± 9.4
diabetes	36.7 ± 14.6	$\mathbf{30.1 \pm 4.8}$	30.7 ± 4.5	32.9 ± 11.7
f-solar	46.3 ± 9.3	44.5 ± 7.9	49.0 ± 9.6	$\mathbf{43.9 \pm 8.6}$
german	39.5 ± 16.1	31.6 ± 3.4	$\mathbf{31.4 \pm 3.4}$	32.4 ± 3.3
heart	$\mathbf{29.5 \pm 8.7}$	32.5 ± 8.1	35.6 ± 8.8	40.4 ± 9.1
image	34.2 ± 10.4	$\mathbf{28.5 \pm 1.9}$	35.7 ± 2.7	34.0 ± 3.4
ringnorm	51.9 ± 10.0	38.0 ± 1.7	38.6 ± 2.3	$\mathbf{40.1 \pm 5.3}$
splice	36.5 ± 28.1	$\mathbf{25.8 \pm 3.7}$	26.4 ± 3.9	26.2 ± 5.8
thyroid	$\mathbf{22.8 \pm 7.3}$	23.5 ± 5.1	25.3 ± 5.4	25.0 ± 7.4
titanic	52.0 ± 12.2	$\mathbf{49.7 \pm 13.3}$	53.3 ± 14.0	50.7 ± 16.4
twonorm	$\mathbf{18.1 \pm 5.1}$	29.8 ± 5.7	30.0 ± 5.5	33.4 ± 5.3
waveform	$\mathbf{19.7 \pm 2.6}$	23.4 ± 3.5	25.1 ± 3.7	25.8 ± 3.7

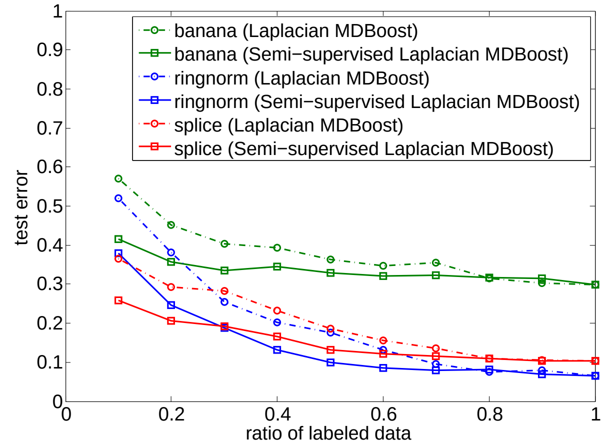

Figure 2. Performance of Laplacian MDBoost (dash-dot line) and Semi-supervised Laplacian MDBoost (solid line) on UCI datasets **banana** (green), **ringnorm** (blue) and **splice** (red).

formance, among which Semi-supervised Laplacian MDBoost is leading in 6 cases, showing the superior inductive inference performance.

Another interesting problem which will naturally arise is the performance gain under different ratios of labeled data. We present the results in Fig. 2, where the labeled data ratio changes from 10% to 100% with a step of 10%. We can see from the figure that, with limited labeled data and abundant unlabeled data, Semi-supervised Laplacian MDBoost significantly outperforms Laplacian MDBoost. However, with more unlabeled data turn into labeled, the performance gain decreases and the error rates converge at a same level. This is reasonable if we look at the objective function in Eq. 7. When there are little (or no) unlabeled data, the value of the second term will approach (or equal to) zero, making it close (or equal) to Eq. 5.

4.4. Video Segmentation with Semi-supervised Laplacian MDBoost

In this section, we apply our semi-supervised Laplacian MDBoost to an object segmentation in video problem. We randomly choose 6 video sequences from the YouTube Celebrities Face Tracking and Recognition Datasets[9]. For each sequence, we extract 15 consecutive frames. The first 10 frames are used for training and the last 5 frame for testing. The overall task is to accurately detect and label human face in each frame in a pixel-wise manner.

To facilitate the labeling task, we first apply a frontal face detector [21] to find a bounding box for human face as in Fig. 3. This would approximately guarantee that the face is in the center of the box while non-face located at the edges. Within the box we perform a segmentation [20] for superpixels. Each superpixel is then considered a basic input vector (datum) for the semi-supervised algorithms.

Next, an automated training strategy were adopted to train the semi-supervised algorithms. The superpixels in the center of the bounding box (within a 20 pixel range) are labeled positive (face) while the superpixels on the brim labeled negative (non-face). Two examples are shown in Fig. 3. The green areas are labeled positive in training while the blue ones are negative. All other superpixels in between are treated as unlabeled training data. This automated training process eliminates the need for manually labeling the ground-truth (which can be a tedious task in real world applications), while also generates a more challenging task for classification. We use color and position histograms as feature vectors.

Fig. 3 visualizes the test results of Semi-supervised Laplacian MDBoost, LLGC+MDBoost and SemiBoost on the two datasets. The performance difference is greater in the second case because the test frames involve a pose change which is likely to cause failure to the baseline classifiers. In both examples, Semi-supervised Laplacian MDBoost presents the best labeling performance visually. Full test results are reported in Table 2. In all 6 video sequences, Semi-supervised Laplacian MDBoost is the best in 5 cases in terms of test error, although SemiBoost is better at training error. This may imply that the baseline is prone to overfitting on these datasets.

5. Conclusion

In this paper, we have proposed a novel semi-supervised boosting algorithm based on the high performance margin distribution boosting. Inspired by Laplacian Eigenmaps, we use the graph Laplacian as an effective means of manifold regularization on both labeled and unlabeled data. Like MDBoost, the algorithm is totally-corrective and a column

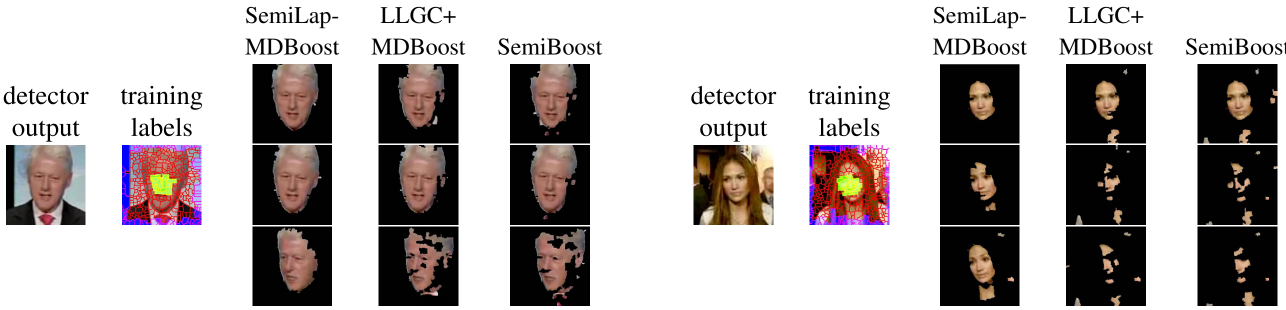

Figure 3. An illustration for video segmentation with three different semi-supervised algorithms: Semi-supervised Laplacian MDBoost (SemiLap-MDBoost), Learning with local and global consistency combined with MDBoost (LLGC+MDBoost) and SemiBoost. The video data are sequences 0370 and 0950 from the Youtube Celebrity Face Tracking and Recognition datasets[9].

Table 2. Average test and training error (in percentage %) of Semi-supervised Laplacian MDBoost (SemiLap-MDBoost), Learning with Local and Global Consistency combined with MDBoost (LLGC+MDBoost), and SemiBoost on the YouTube Celebrities Face Tracking and Recognition Datasets over 10 tests.

		test error	training error
0146 Al Pacino	SemiLap-MDBoost	$\mathbf{13.7 \pm 2.1}$	5.9 ± 1.2
	LLGC+MDBoost	15.4 ± 2.4	5.5 ± 1.1
	SemiBoost	19.8 ± 3.2	$\mathbf{4.2 \pm 0.6}$
0370 Bill Clinton	SemiLap-MDBoost	$\mathbf{11.1 \pm 1.6}$	10.5 ± 1.7
	LLGC+MDBoost	16.8 ± 2.0	$\mathbf{8.5 \pm 1.0}$
	SemiBoost	22.5 ± 2.2	10.7 ± 1.3
0564 Donald Trump	SemiLap-MDBoost	$\mathbf{7.2 \pm 2.1}$	4.3 ± 0.6
	LLGC+MDBoost	16.8 ± 3.2	3.9 ± 0.7
	SemiBoost	18.5 ± 4.7	$\mathbf{3.5 \pm 0.3}$
0727 Harrison Ford	SemiLap-MDBoost	12.6 ± 2.4	$\mathbf{4.9 \pm 0.3}$
	LLGC+MDBoost	15.3 ± 2.3	6.1 ± 0.5
	SemiBoost	$\mathbf{11.5 \pm 1.9}$	5.5 ± 0.3
0935 Jennifer Lopez	SemiLap-MDBoost	$\mathbf{16.5 \pm 3.2}$	11.5 ± 2.4
	LLGC+MDBoost	16.9 ± 2.9	10.2 ± 1.7
	SemiBoost	20.2 ± 4.1	$\mathbf{6.4 \pm 1.0}$
0950 Jennifer Lopez	SemiLap-MDBoost	$\mathbf{19.8 \pm 2.1}$	9.8 ± 2.2
	LLGC+MDBoost	29.1 ± 3.8	14.2 ± 2.9
	SemiBoost	28.3 ± 3.5	$\mathbf{7.9 \pm 1.2}$

generation based optimization technique is used to facilitate minimizing the objective function.

The proposed Semi-supervised Laplacian MDBoost, along with its supervised version, exhibits promising inductive performance in a variety of tasks including classification on real data and video segmentation. Our experiments show that Semi-supervised Laplacian MDBoost outperforms LLGC and SemiBoost in terms of classification accuracy.

Like almost all other semi-supervised classification algorithms, Semi-supervised Laplacian MDBoost is currently a two-class algorithm. We are exploring the possibility to a multiple class extension by introducing new similarity measures. We also want to test our algorithm on more practical applications to further explore the strength of graph Laplacian on different intrinsic geometric structures. One possible extension is to add more methods for manifold regularization to adapt to different manifold assumptions.

Acknowledgement

We thank the anonymous reviewers for their detailed comments and suggestions which resulted in the improvement of this paper.

References

[1] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003. 1, 2, 3

[2] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, 1998. 2

[3] F. Buc, Y. Grandvalet, and C. Ambroise. Semi-supervised marginboost. *NIPS*, 14:553–560, 2002. 2

[4] O. Chapelle, B. Scholkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, 2006. 2

[5] K. Chen and S. Wang. Regularized boost for semi-supervised learning. In *NIPS*, 2007. 2

[6] A. Demiriz, K. Bennett, and J. Shawe-Taylor. Linear programming boosting via column generation. *Machine Learning*, 46(1):225–254, 2002. 2, 5

[7] A. Garg and D. Roth. Margin distribution and learning algorithms. In *ICML*, 2003. 1

[8] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. *ECCV*, 2008. 5

[9] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley. Face tracking and recognition with visual constraints in real-world videos. In *CVPR*, 2008. 2, 5, 6, 7

[10] C. Leistner, H. Grabner, and H. Bischof. Semi-supervised boosting using visual similarity learning. In *CVPR*, 2008. 5

[11] H. Lodhi, G. Karakoulas, and J. Shawe-Taylor. Boosting the margin distribution. In *Intelligent Data Engineering and Automated Learning IDEAL 2000.*, 2000. 1

[12] P. Mallapragada, R. Jin, A. Jain, and Y. Liu. Semiboost: Boosting for semi-supervised learning. *IEEE TPAMI*, pages 2000–2014, 2008. 2, 5

[13] B. Pfahringer, C. Leschi, and P. Reutemann. Scaling up semi-supervised learning: An efficient and effective llgc variant. *KDD*, 2007. 5

[14] G. Ratsch, T. Onoda, and K. Muller. Soft margins for AdaBoost. *Machine Learning*, 42(3):287–320, 2001. 2, 5

[15] L. Reyzin and R. E. Schapire. How boosting the margin can also boost classifier complexity. In *ICML*, 2006. 1, 2

[16] A. Saffari, H. Grabner, and H. Bischof. Serboost: Semi-supervised boosting with expectation regularization. In *ECCV '08*, 2008. 2

[17] R. E. Schapire and Y. Freund. Boosting the margin: a new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26:322–330, 1998. 1, 2

[18] C. Shen and H. Li. Boosting through optimization of margin distributions. *IEEE Transactions on Neural Networks*, 21(4):659–666, April 2010. 1, 2

[19] C. Shen and H. Li. On the dual formulation of boosting algorithms. *IEEE TPAMI*, 2010. 1, 2, 3

[20] A. Stein and M. Hebert. Occlusion boundaries from motion: low-level detection and mid-level reasoning. *International journal of computer vision*, 82(3):325–357, 2009. 6

[21] P. Viola and M. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004. 6

[22] M. Wang, T. Mei, X. Yuan, Y. Song, and L. Dai. Video annotation by graph-based learning with neighborhood similarity. In *ACM Multimedia*, 2007. 5

[23] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Scholkopf. Learning with local and global consistency. In *NIPS*, pages 595–602, 2004. 2, 5

[24] X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. *Carnegie Mellon Univ., CS Tech. Rep. CMUCALD-02-107*, 2002. 2