Classification with Partial Labels

Nam Nguyen, Rich Caruana
Cornell University
Department of Computer Science
Ithaca, New York 14853
{nhnguyen, caruana}@cs.cornell.edu

ABSTRACT

In this paper, we address the problem of learning when some cases are fully labeled while other cases are only partially labeled, in the form of partial labels. Partial labels are represented as a set of possible labels for each training example, one of which is the correct label. We introduce a discriminative learning approach that incorporates partial label information into the conventional margin-based learning framework. The partial label learning problem is formulated as a convex quadratic optimization minimizing the L2-norm regularized empirical risk using hinge loss. We also present an efficient algorithm for classification in the presence of partial labels. Experiments with different data sets show that partial label information improves the performance of classification when there is traditional fully-labeled data, and also yields reasonable performance in the absence of any fully labeled data.

Categories and Subject Descriptors

I.5 [**Pattern Recognition**]: Design Methodology

General Terms

Algorithms, Design

Keywords

Partial Labels, Support Vectors

1. INTRODUCTION

Partially labeled training data such as pairwise constraints have been shown to improve performance in both supervised [20, 21, 10, 11, 12, 14, 15, 18, 6] and semi-supervised [17, 3, 13, 19, 7, 2, 4] learning. While labeled data is usually expensive, time consuming to collect, and sometimes requires human domain experts to annotate, partially labeled data is often relatively easier to obtain. Much attention in the machine learning community has been focused on integrating partially labeled data that are complementary to the fully labeled training data into existing learning framework. However, in previous research partially labeled information is usually presented in the form of pairwise constraints which indicate whether a pair of examples belongs to the same class or not. In [20], the authors showed significant performance improvement in video object classification using a modified logistic regression algorithm which can learn the decision boundary with labeled data as well as additional pairwise constraints. Moreover, in [21] the authors proposed a discriminative method which can effectively utilize pairwise constraints to construct a sign-insensitive consistent estimator with respect to the optimal linear boundary.

In this paper, we investigate the usefulness of a different partially labeled information, *Partial Labels*[1]. Partial labels are presented as a set of possible labels for each training example, one of which is the correct label. Unlike fully labeled data that would require users to have prior knowledge or experience with a data set, partial labels relatively require often less effort from users. For example, in the task of predicting nationality based on facial images, it is relatively easier for users to determine if a face belongs to a group of countries such as Asian countries, African countries or Western countries than to identify the exact nationality.

In addition to using partially labeled data to improve the performance of classifiers, unlabeled data is the main focus of semi-supervised learning [22]. In this setting, a small amount of labeled data is augmented with a large amount of unlabeled data is used to learn better classifiers. Note that unlabeled data may not always help. For example, Cozman et al [8] showed that unlabeled data can degrade classification performance even in situations where additional labeled data would increase the performance. Partially labeled data is a perfect tradeoff between fully labeled data and unlabeled data. We will show that partially labeled data in form of partial labels helps producing better classifiers without too much labeling annotation from users.

In this work, we propose a discriminative learning approach which incorporates partial label information into the conventional margin-based learning framework. First, we review the margin-based learning framework for the multiclass classification problem [9]. Then we extend the learning framework to include partial label information. In our experiment with a variety of data sets, partial labels not only

KDD'08, August 24–27, 2008, Las Vegas, Nevada, USA.
Copyright 2008 ACM 978-1-60558-193-4/08/08 ...$5.00.

[1] Here we want to make a distinction between partially labeled data and partial labels. Partially labeled data indicates only partial information about examples is given instead of the actual correct labels. Both pairwise constraints and partial labels are subcategories of partially labeled data.

improve the performance of classification when there is traditional fully-labeled data, but also yields reasonable performance in the absence of any fully-labeled data. The paper is structured as follow: in section 2, we describe in detail the novel partial label classification algorithm; in section 3 we review related work on supervised and semi-supervised learning with partially labeled data; the experimental results and conclusion are given in section 4 and 5, respectively.

2. CLASSIFICATION WITH PARTIAL LABEL INFORMATION

In this section, we start with the margin-based multiclass classification problem. Then, we show how partial label information fits into the margin-based discriminative learning framework.

2.1 Margin-based Multiclass Classification

In the supervised setting, a learning algorithm typically takes a set of labeled training examples, $\mathbf{L} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ as input, where $x_i \in \mathcal{X}$ and the corresponding label y_i belongs to a finite set of classes denoted as $\mathcal{Y}$. The goal of classification is to form a hypothesis $h : \mathcal{X} \mapsto \mathcal{Y}$ which maps an input $x \in \mathcal{X}$ to an output $y \in \mathcal{Y}$. Many machine learning algorithms is formulated to minimize the regularized empirical risk

$$\min_{w} R_{reg}(w) := \lambda\Omega(w) + L(w) \tag{1}$$

$$\text{where } L(w) := \frac{1}{n}\sum_{i=1}^{n} l(x_i, y_i, w) \tag{2}$$

where $\Omega(\cdot)$ is a convex and monotonically increasing function which serves as a regularizer with a regularization constant $\lambda > 0$; and $l(x_i, y_i, w)$ is a nonnegative loss function of an example x_i measuring the amount of inconsistency between the correct label y_i and the predicted label arising from using the weight parameter w.

Consider a mapping $\Phi : \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{F}$ which projects each example-label pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$ to $\Phi(x, y)$ in a new space $\mathcal{F}$, is defined as

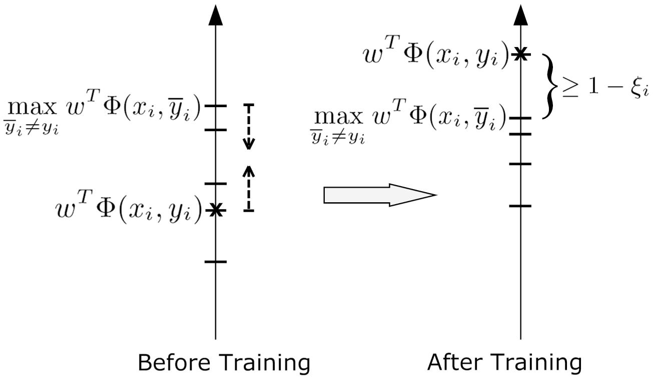
$$\Phi(x, y) = \begin{bmatrix} x \cdot \mathcal{I}(y = 1) \\ \ldots \\ x \cdot \mathcal{I}(y = i) \\ \ldots \\ x \cdot \mathcal{I}(y = |\mathcal{Y}|) \end{bmatrix},$$

where $\mathcal{I}(\cdot)$ is the indicator function. We can obtain the multiclass-SVM proposed by [9] by considering the situation where we use the L2-norm regularization,

$$\Omega(w) = \frac{1}{2}\|w\|^2,$$

and the loss function $l(x_i, y_i, w)$ is set to the hinge loss,

$$\max\left(0, 1 - \left[w^T\Phi(x_i, y_i) - \max_{\overline{y}_i \neq y_i} w^T\Phi(x_i, \overline{y}_i)\right]\right).$$


$w^T\Phi(x_i, y_i)$
$\geq 1 - \xi_i$
$\max_{\overline{y}_i \neq y_i} w^T\Phi(x_i, \overline{y}_i)$
$\max_{\overline{y}_i \neq y_i} w^T\Phi(x_i, \overline{y}_i)$
$w^T\Phi(x_i, y_i)$
Before Training
After Training

∗ : represents the correct label y_i of the example x_i
— : represents other labels $\overline{y}_i \neq y_i$ of the example x_i

Figure 1: Illustration of how the relative positions of the scores associated with example-label pairs $w^T\Phi(x_i, \cdot)$ change from before training to after training for a fully labeled example.

Specifically, the multiclass-SVM learns a weight vector w and slack variables ξ via the following quadratic optimization problem:

Optimization Problem I: Multiclass-SVM

$$\min_{w,\xi} : \frac{\lambda}{2}\|w\|^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i \tag{3}$$

subject to:

$$\forall (x_i, y_i) \in \mathbf{L} : w^T\Phi(x_i, y_i) - \max_{\overline{y}_i \neq y_i} w^T\Phi(x_i, \overline{y}_i) \geq 1 - \xi_i,\ \xi_i > 0.$$

After we have learned w and ξ, the classification of a test example x is done by

$$h(x) = \operatorname*{argmax}_{y \in \mathcal{Y}} w^T\Phi(x, y).$$

In this margin-based learning framework, we observed that for a training examples $(x_i, y_i) \in \mathbf{L}$ the score associated with the correct label y_i, $w^T\Phi(x_i, y_i)$, is greater than the score associated with any other labels $\overline{y}_i \neq y_i$, $w^T\Phi(x_i, \overline{y}_i)$, by at least the amount, $1 - \xi_i$. In Figure 1, we demonstrate how the relative positions of the scores associated with example-label pairs, $w^T\Phi(x_i, \cdot)$, change from before training to after training for a fully labeled example, (x_i, y_i).

2.2 Margin-based Partial Label Classification

In this section, we address the problem of learning when there are additional partially labeled data, in the form of partial labels, augmented with fully labeled data. Partial labels are presented as a set of possible labels for each training example, one of which is the correct label. Let $\mathbf{PL} = \{(x_1, Y_1), \ldots, (x_m, Y_m)\}$ be the set of partial label training data, where $x_i \in \mathcal{X}$ and the corresponding set of possible labels $Y_i \subset \mathcal{Y}$, one of which is the correct label.

The partial label learning problem is also formulated to minimize the regularized empirical risk as shown in Equation (1), where the loss function $L(w)$ is the addition of the empirical loss due to the fully labeled data and the partial label data. Formally, the loss function can be expressed as,

$$L(w) := \frac{1}{n+m}\left[\sum_{i=1}^{n} l(x_i, y_i, w) + \sum_{i=1}^{m} l(x_i, Y_i, w)\right]. \qquad (4)$$

In addition to utilizing the same L2-norm regularization and the hinge loss for the fully labeled data, we use the following hinge loss, $l(x_i, Y_i, w)$, for the partial label data:

$$\max\left(0, 1 - \left[\max_{y_i \in Y_i} w^T \Phi(x_i, y_i) - \max_{\overline{y}_i \notin Y_i} w^T \Phi(x_i, \overline{y}_i)\right]\right).$$

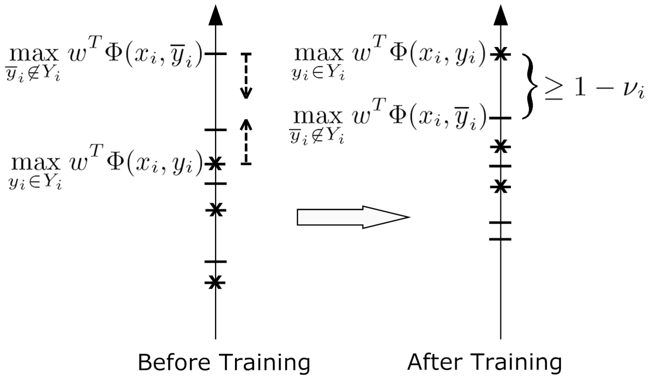
The justification of using the hinge loss for the partial label data is that for a partial label training example $(x_i, y_i) \in$ **PL** the maximum score associated with the partial labels $y_i \in Y_i$,

$$\max_{y_i \in Y_i} w^T \Phi(x_i, y_i),$$

is greater than the maximum score associated with any other labels $\overline{y}_i \notin Y_i$,

$$\max_{\overline{y}_i \notin Y_i} w^T \Phi(x_i, \overline{y}_i),$$

by at least the amount, $1 - \nu_i$. In Figure 2, we demonstrate how the relative positions of the scores associated with example-label pairs, $w^T \Phi(x_i, \cdot)$, change from before training to after training for a partial label example, (x_i, Y_i).


$\max_{\overline{y}_i \notin Y_i} w^T \Phi(x_i, \overline{y}_i)$
$\max_{y_i \in Y_i} w^T \Phi(x_i, y_i)$
$\max_{y_i \in Y_i} w^T \Phi(x_i, y_i)$
$\max_{\overline{y}_i \notin Y_i} w^T \Phi(x_i, \overline{y}_i)$
$\geq 1 - \nu_i$
Before Training
After Training

✱ : represents the partial labels $y_i \in Y_i$ of the example x_i
— : represents other labels $\overline{y}_i \notin Y_i$ of the example x_i

Figure 2: Illustration of how the relative positions of the scores associated with example-label pairs $w^T \Phi(x_i, \cdot)$ change from before training to after training for a partial label example.

In this learning setting, the average size of the partial labels,

$$\frac{1}{m}\sum_{i=1}^{m} |Y_i|,$$

of the partial label data indicates the amount of labeled information given to the learning algorithm. In the limit, if $|Y_i| = 1$ then we have the conventional supervised learning framework where each training example is given the correct label. Moreover, if $|Y_i| = |\mathcal{Y}|$ then we obtain the semi-supervised learning framework where there is additional unlabeled data augmented with the fully labeled data. We will show later in the experiment section, how the classification performance changes in according to the variation of the size of the partial labels.

Formally, the partial label SVM classification (PL-SVM) learns a weight vector w and slack variables ξ and ν via the following quadratic optimization problem:

Optimization Problem II: PartialLabel-SVM

$$\min_{w,\xi,\nu} : \frac{\lambda}{2}\|w\|^2 + \frac{1}{n+m}\left(\sum_{i=1}^{n} \xi_i + \sum_{i=1}^{m} \nu_i\right) \qquad (5)$$

subject to:

$\forall (x_i, y_i) \in \mathbf{L}$:

$$w^T \Phi(x_i, y_i) - \max_{\overline{y}_i \neq y_i} w^T \Phi(x_i, \overline{y}_i) \geq 1 - \xi_i,\ \xi_i > 0;$$

$\forall (x_i, Y_i) \in \mathbf{PL}$:

$$\max_{y_i \in Y_i} w^T \Phi(x_i, y_i) - \max_{\overline{y}_i \notin Y_i} w^T \Phi(x_i, \overline{y}_i) \geq 1 - \nu_i,\ \nu_i > 0.$$

The classification of test examples are done in the same manner as for the multiclass-SVM classification.

In order to solve the partial label SVM classification, we apply the partial label Pegasos (PL-Pegasos), a extended version of the Pegasos algorithm proposed by [16]. The PL-Pegasos is a simple and effective iterative algorithm for solving the above QP and does not require transforming to the dual formulation. The algorithm alternates between gradient descent steps and projection steps. In each iteration, the algorithm first computes a set of labeled examples $\mathbf{A_L} \subset \mathbf{L}$ and a set of partially labeled examples $\mathbf{A_{PL}} \subset \mathbf{PL}$ that contain violated examples. Then the weight vector w is updated according to the violated sets $\mathbf{A_L}$ and $\mathbf{A_{PL}}$. In the projection step, the weight vector w is projected to the sphere of radius $1/\sqrt{\lambda}$. The details of the PL-Pegasos are given in Algorithm 1.

In order to used the *kernel trick*, as pointed out in [16], we set $w_1 = 0$ then w_t can be written as

$$w_t = \sum_{x,y} \varphi_{xy} \Phi(x, y).$$

Hence, we can incorporate the usage of kernel when computing inner product operations, i.e.:

$$\begin{aligned} \langle w, \Phi(x', y')\rangle &= \sum_{x,y} \varphi_{xy} \mathbf{K}(x, y, x', y') \\ \|w\|^2 &= \sum_{x,y} \sum_{x',y'} \varphi_{xy} \varphi_{x'y'} \mathbf{K}(x, y, x', y') \end{aligned}$$

In our experiments, we use the polynomial kernel,

$$\mathbf{K}(x, y, x', y') = \langle \Phi(x, y), \Phi(x', y')\rangle^d$$

where the polynomial kernel degree d is chosen from the set $\{1, 2, 3, 4, 5\}$.

Algorithm 1 : Partial Label SVM Classification (PL-SVM)

```
Input: $\mathbf{L}$ - the labeled data, $\mathbf{PL}$ - the partial label data
       $\lambda$ and $T$ - parameters of the QP

Initialize: Choose $w_1$ such that $\|w_1\| \leq 1/\sqrt{\lambda}$

for $t = 1$ to $T$ do
    Set $\mathbf{A_L} = \left\{(x_i, y_i) \in \mathbf{L} \mid w_t^T \Phi(x_i, y_i) - \max_{\overline{y}_i \neq y_i} w_t^T \Phi(x_i, \overline{y}_i) < 1\right\}$
    Set $\mathbf{A_{PL}} = \left\{(x_i, Y_i) \in \mathbf{PL} \mid \max_{y_i \in Y_i} w_t^T \Phi(x_i, y_i) - \max_{\overline{y}_i \notin Y_i} w_t^T \Phi(x_i, \overline{y}_i) < 1\right\}$
    Set $\eta_t = \frac{1}{\lambda t}$
    Set $w_{t+\frac{1}{2}} = (1 - \eta_t \lambda) w_t + \frac{\eta_t}{n+m} \left\{ \sum_{(x_i, y_i) \in \mathbf{A_L}} [\Phi(x_i, y_i) - \Phi(x_i, \overline{y}_i)] + \sum_{(x_i, Y_i) \in \mathbf{A_{PL}}} \left[\Phi(x_i, y_i^{\mathrm{PL}}) - \Phi(x_i, \overline{y}_i^{\mathrm{PL}})\right] \right\}$
        where $\overline{y}_i = \operatorname{argmax}_{\overline{y}_i \neq y_i} w_t^T \Phi(x_i, \overline{y}_i)$ for $(x_i, y_i) \in \mathbf{A_L}$;
              $y_i^{\mathrm{PL}} = \operatorname{argmax}_{y_i \in Y_i} w_t^T \Phi(x_i, y_i)$,
              $\overline{y}_i^{\mathrm{PL}} = \operatorname{argmax}_{\overline{y}_i \notin Y_i} w_t^T \Phi(x_i, \overline{y}_i)$ for $(x_i, Y_i) \in \mathbf{A_{PL}}$

    Set $w_{t+1} = \min\left\{1, \frac{1/\sqrt{\lambda}}{\|w_{t+\frac{1}{2}}\|}\right\} w_{t+\frac{1}{2}}$
end for

Output: $w_{T+1}$
```

The efficiency and guaranteed performance of PL-SVM in solving the quadratic optimization problem is shown by the following theorem:

THEOREM 1. *Let*

$$R = 2 \max_{x,y} \|\Phi(x, y)\|$$

then the number of iterations for Algorithm 1 to achieve a solution of accuracy $\delta > 0$ *is* $\tilde{O}(R^2/(\lambda\delta))$.

The proof of Theorem 1 is omitted since it is similar to the one given in [16].

3. RELATED WORK

In supervised learning, partially labeled data in the form of pairwise constraints have been shown to improve the performance of classifiers. In [21, 20], the authors proposed a discriminative learning framework which can simultaneously learn the fully labeled data and pairwise constraints. In addition, the pairwise constraint information is also used to learn metric learning algorithms [10, 11, 12, 14, 15, 18, 6]. Metric learning algorithms first learn a Mahalanobis distance metric and then apply distance-based classifier such as K-nearest neighbor to the transformed data.

In semi-supervised learning, partially labeled data in the form of pairwise constraints is used as users' feedback to guide the clustering process [17, 3, 13, 19, 7, 2, 4]. In particular, CKmeans [17] is a semi-supervised variant of Kmeans. The objective function of CKmeans is reformulated to incorporate the cost incurred by violating any pairwise constraints specified by the user. In addition, [4] utilized both metric learning and pairwise constraints in the clustering process. In MPCKmeans (metric learning and constraints Kmeans), a separate weight matrix for each cluster is learned to minimize the distance between must-link instances and maximize the distance between cannot-link instances. Hence, the objective function of MPCKmeans minimizes cluster dispersion under the learned metrics while reducing constraint violations. However, most existing algorithms may get stuck at local-optimal solutions for the clustering problem with pairwise constraints as users' feedback.

4. EXPERIMENTS

We evaluate our proposed algorithm (PL-SVM) on six data set from the UCI repository [1] and the LIBSVM data [5]. A summary of the data sets is given in Table 1.

Table 1: A summary of the data sets.

Data sets	Classes	Train	Test	Features
leter	26	15000	5000	16
mnist	10	60000	10000	780
pendigits	10	8992	2000	16
satimage	6	4435	2000	36
segment	7	1960	350	19
usps	10	7291	2007	256

In our experiments, we compare the classification performance of the PL-SVM algorithm which utilizes the partial label information against the regular SVM which ignores the partial labels. As a upper bound for the performance of the PL-SVM algorithm, we train a regular SVM using the fully labeled data and the partial label data where the true labels are revealed to the algorithm. (We refer to this training

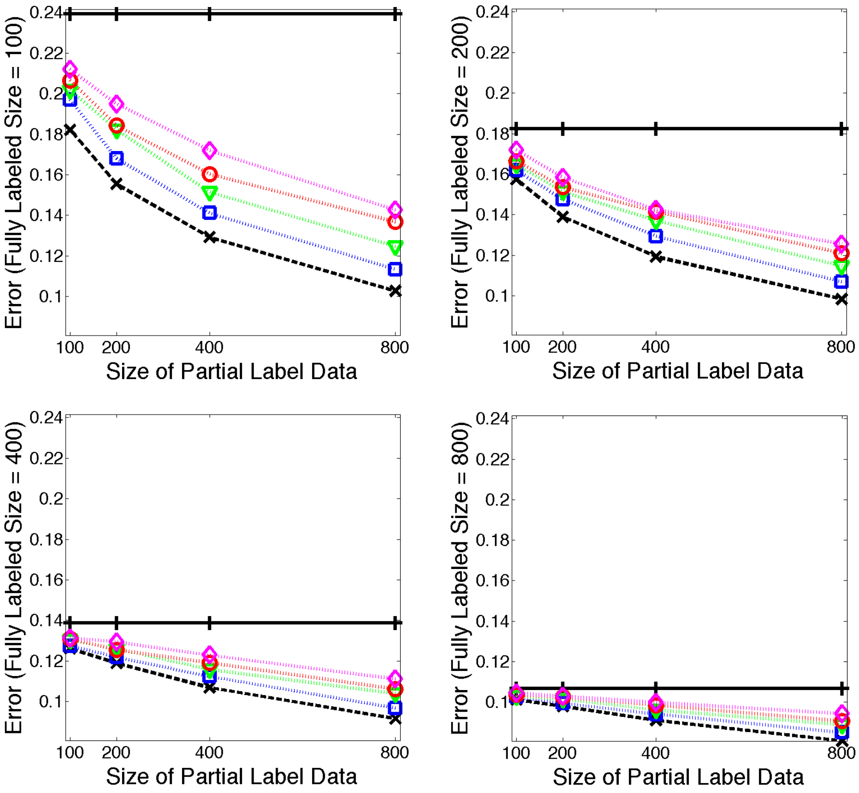


Figure 3: Average Classification Performance of SVM (+-black-solid), SVM All ($\times$-black-dashed), and PL-SVM ($\Box$-blue-dotted: Partial Labels=2, $\bigtriangledown$-green-dotted: Partial Labels=3, $\odot$-red-dotted: Partial Labels=4, $\diamond$-magenta-dotted: Partial Labels=5) versus the size of partial label data across six data sets.

procedure as SVM All.) For all the algorithms, we set the parameter values as follows:

- The regularization constant λ and the polynomial kernel degree d are chosen from the set $\{10^i\}_{i=-3}^{3}$ and $\{1, 2, 3, 4, 5\}$, respectively. Both the parameters λ and d are selected using two fold cross validation on the fully labeled training set.

- The sizes of the fully labeled training data and of the partial label training data, $|\mathbf{L}|$ and $|\mathbf{PL}|$, are selected from the set $\{100, 200, 400, 800\}$. Both the fully labeled training data and the partial label training data are randomly sampled from the training set.

- The size of the partial labels, $|Y_i|$, is chosen from the set $\{2, 3, 4, 5\}$. For each example, the labels in the partial labels (except the true label) are randomly selected.

In Figures 6 and 7, we plot the classification performance of SVM, SVM All and PL-SVM (one for each value of the partial label size) versus the size of the partial label data at different sizes of the fully labeled training data for six data sets. To summarize the information, Figure 3 shows the same information but averaging across the six data sets. For all six data sets, we observe that the performance of the PL-SVM is between the performance of SVM and SVM All. This behavior is what we should expect since partial label information helps to significantly improve the performance of PL-SVM over SVM which does not use this information; and fully labeled data should still provide more discriminative information to the SVM All than partial labels could to the PL-SVM. We also notice the expected learning curve for PL-SVM as the size of the partial label data is varied. For a fixed fully labeled training size, as we increase the amount of the partial label data the performance of PL-SVM is also increasing. Moreover, we also observed the inverse relation between the amount of performance improvement of PL-SVM over SVM and the size of the partial labels. For fixed sizes of the fully labeled data and the partial label data, as we increase the size of the partial labels the performance of PL-SVM is decreasing. This behavior is expected since the larger the size of the partial labels the less the discriminative power of each partial label example.

5. CONCLUSION

In this paper, we address the problem of learning when some cases are fully labeled while other cases are only partially labeled, in the form of partial labels. Partial labels are represented as a set of possible labels for each training example, one of which is the correct label. We formulate the partial label learning problem as a convex quadratic optimization minimizing the L2-norm regularized empirical risk using hinge loss and present an efficient algorithm for classification in the presence of partial labels.

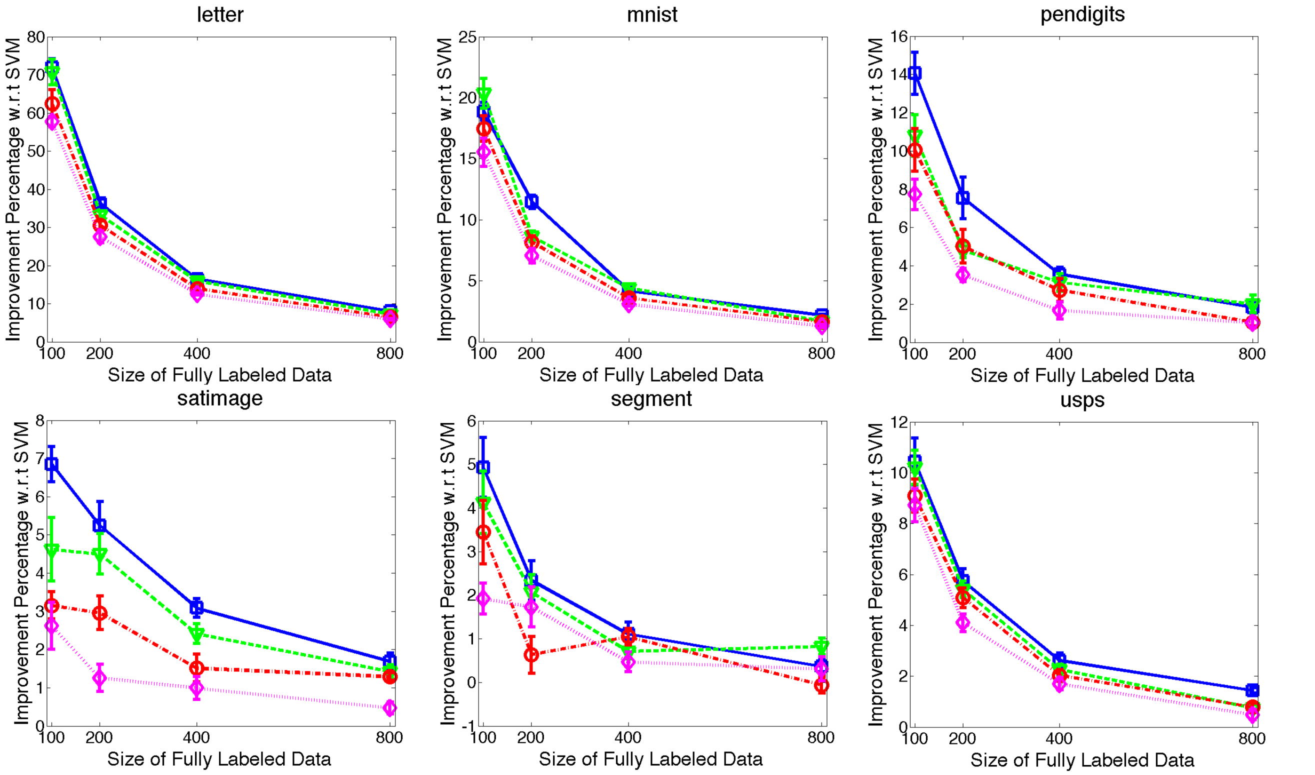


Figure 4: Percentage of Performance Improvement of PL-SVM (□-blue-solid: Partial Labels=2, ▽-green-dashed: Partial Labels=3, ⊙-red-dashdot: Partial Labels=4, ◇-magenta-dotted: Partial Labels=5) over SVM versus the size of fully labeled data where the size of the partial label data is fixed at 800.

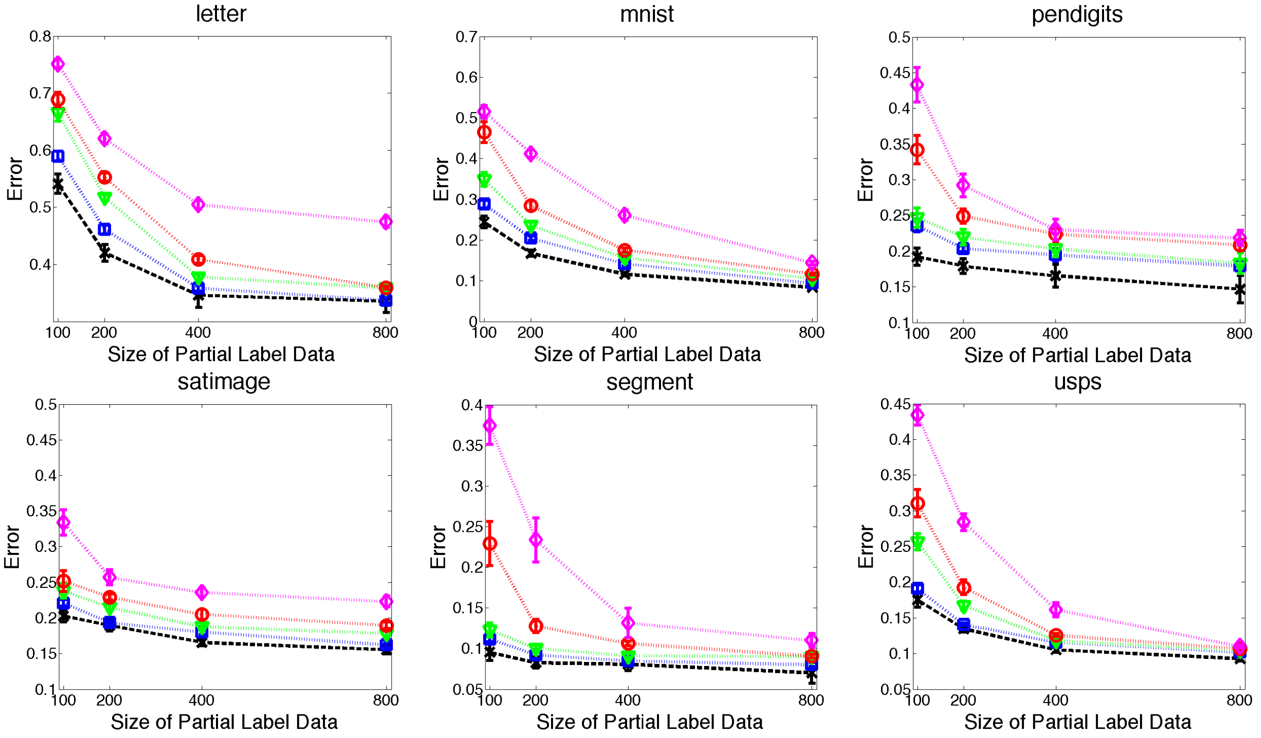


Figure 5: Classification Performance of SVM All (×-black-dashed), and PL-SVM (□-blue-dotted: Partial Labels=2, ▽-green-dotted: Partial Labels=3, ⊙-red-dotted: Partial Labels=4, ◇-magenta-dotted: Partial Labels=5) versus the size of partial label data in the absence of fully labeled data.

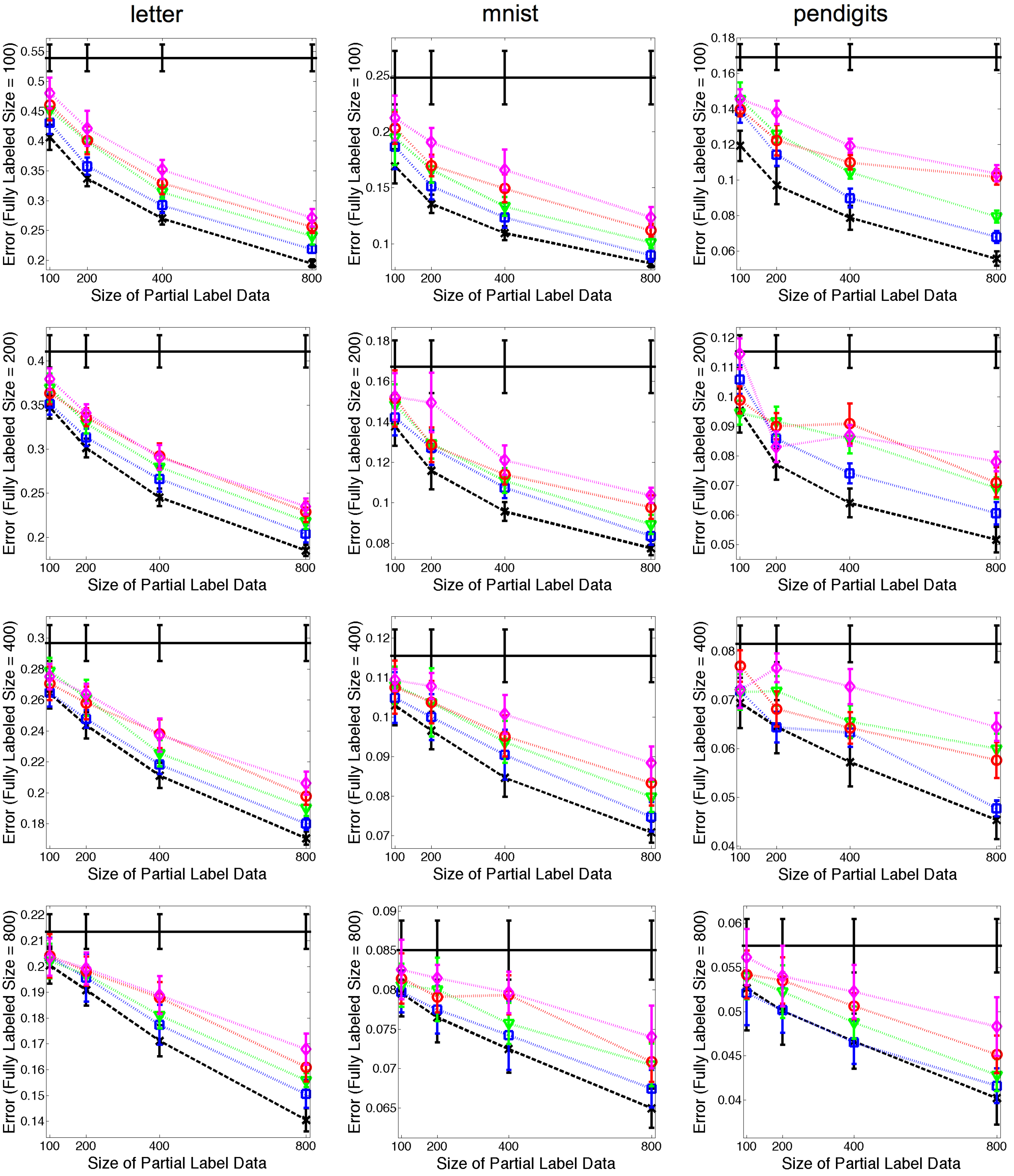


Figure 6: Classification Performance of SVM (+-black-solid), SVM All (×-black-dashed), and PL-SVM (□-blue-dotted: Partial Labels=2, ▽-green-dotted: Partial Labels=3, ⊙-red-dotted: Partial Labels=4, ◇-magenta-dotted: Partial Labels=5) versus the size of partial label data for three data sets: letter, mnist, and pendigits.

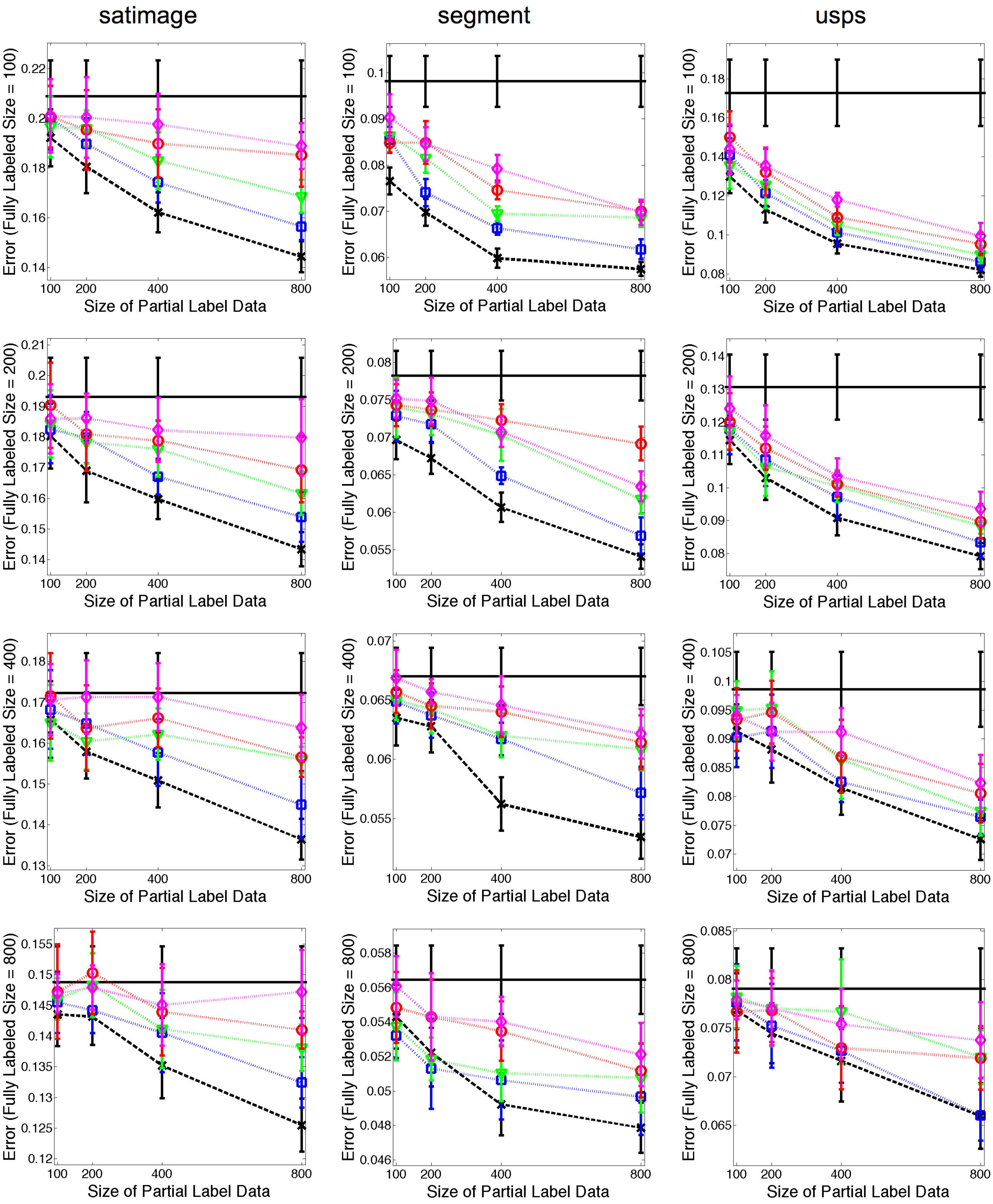


Figure 7: Classification Performance of SVM (+-black-solid), SVM All (×-black-dashed), and PL-SVM (□-blue-dotted: Partial Labels=2, ▽-green-dotted: Partial Labels=3, ⊙-red-dotted: Partial Labels=4, ◇-magenta-dotted: Partial Labels=5) versus the size of partial label data for three data sets: satimage, segment, and usps.

6. ACKNOWLEDGMENTS

This work was supported by NSF CAREER Grant # 0347318. We would like to thank anonymous reviewers for valuable comments and suggestions.

7. REFERENCES

[1] A. Asuncion and D. Newman. UCI machine learning repository. www.ics.uci.edu/∼mlearn/MLRepository.html, 2007.

[2] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In *Proceedings of 20th International Conference on Machine Learning*, 2003.

[3] M. Bilenko, S. Basu, and R. J. Mooney. Semi-supervised clustering by seeding. In *Proceedings of 19th International Conference on Machine Learning*, 2002.

[4] M. Bilenko, S. Basu, and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of 21th International Conference on Machine Learning*, 2004.

[5] C.-C. Chang and C.-J. Lin. Libsvm data. www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/, 2001.

[6] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[7] D. Cohn, R. Caruana, and A. McCallum. Semi-supervised clustering with user feedback. In *Cornell University Technical Report TR2003-1892*, 2003.

[8] F. Cozman, I. Cohen, and M. Cirelo. Semi-supervised learning of mixture models and bayesian networks. In *Proceedings of the Twentieth International Conference of Machine Learning*, 2003.

[9] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2001.

[10] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[11] A. Globerson and S. Roweis. Metric learning by collapsing classes. In *Advances in Neural Information Processing Systems (NIPS)*, 2005.

[12] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[13] D. Klein, S. D. Kamvar, and C. D. Manning. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *Proceedings of 19th International Conference on Machine Learning*, 2002.

[14] M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[15] S. Shalev-Shwartz, Y. Singer, and A. Y. Ng. Online and batch learning of pseudo-metrics. In *Proceedings of the 21st International Conference on Machine Learning*, 2004.

[16] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proceedings of the 24th International Conference on Machine Learning*, pages 807–814, New York, NY, USA, 2007. ACM.

[17] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. In *Proceedings of 18th International Conference on Machine Learning*, 2001.

[18] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances ing Neural Information Processing Systems (NIPS)*, 2006.

[19] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15*, 2003.

[20] R. Yan, J. Zhang, J. Yang, and A. G. Hauptmann. A discriminative learning framework with pairwise constraints for video object classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):578–593, 2006.

[21] J. Zhang and R. Yan. On the value of pairwise constraints in classification and consistency. In *Proceedings of the 24th International Conference on Machine Learning*, pages 1111–1118, 2007.

[22] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.