Semi-supervised Learning from General Unlabeled Data

Kaizhu Huang
Department of Engineering Mathematics
University of Bristol
Bristol BS8 1TR, United Kingdom
K.Huang@bris.ac.uk

Zenglin Xu, Irwin King, Michael R. Lyu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
{zlxu, king, lyu}@cse.cuhk.edu.hk

Abstract

We consider the problem of Semi-supervised Learning (SSL) from general unlabeled data, which may contain irrelevant samples. Within the binary setting, our model manages to better utilize the information from unlabeled data by formulating them as a three-class ($-1, +1, 0$) mixture, where class 0 represents the irrelevant data. This distinguishes our work from the traditional SSL problem where unlabeled data are assumed to contain relevant samples only, either $+1$ or -1, which are forced to be the same as the given labeled samples. This work is also different from another family of popular models, universum learning (universum means "irrelevant" data), in that the universum need not to be specified beforehand. One significant contribution of our proposed framework is that such irrelevant samples can be automatically detected from the available unlabeled data, even though they are mixed with relevant data. This hence presents a general SSL framework that does not force "clean" unlabeled data. More importantly, we formulate this general learning framework as a Semidefinite Programming problem, making it solvable in polynomial time. A series of experiments demonstrate that the proposed framework can outperform the traditional SSL on both synthetic and real data.

1 Introduction

Learning classifiers from data has been a popular and important topic in machine learning and data mining. Given a sufficiently large quantity of labeled instances called training data, one can exploit the traditional Supervised Learning (SL) algorithms to handle this task [23, 8, 10]. However, in many real world applications, the labeled data may be very few due to the expensive cost of manual labeling. On the other hand, the number of unlabeled instances could be very large since they are generally much easier to obtain. SL, taking only advantages of the labeled data, might not work appropriately in these cases. In contrast, Semi-supervised Learning (SSL), making use of both labeled data and unlabeled data, proves to be an effective solution in addressing this problem [29, 4]. Undoubtedly, SSL has achieved a great success in many domains involving machine learning and data mining. To guarantee good performance, SSL usually assumes that the unlabeled data should share the same labels as the labeled training samples. Although this assumption can be well satisfied in some cases, it appears still strong in certain other domains. In fact, it is very common that unlabeled data are collected by using automatical tools. This is actually frequently seen in the earlier stages of data collection. It is usually inevitable that those collected unlabeled data contain irrelevant samples. Feeding such "corrupted" unlabeled data to SSL may significantly affect the overall performance and incur severe problems consequently.

To attack this problem, we aim to build up a general SSL framework capable of learning from general unlabeled data systematically, where the unlabeled data may contain irrelevant samples. Our model manages to better utilize the information from unlabeled data by formulating them as a three-class $(-1, +1, 0)$ mixture.[1] This hence distinguishes our work from the traditional SSL problem where unlabeled data are assumed to contain the same labels as the labeled training samples [28, 7].

The benefits of taking the irrelevant data into account can be seen in Figure 1 and Figure 2. In both Figures, all the filled points ($\bullet$'s and $\star$'s) are unlabeled data, while the $\circ$'s and $\Box$'s are the two classes of labeled training samples. Clearly, Figure 1(a) illustrates that SSL can outperform the boundary given by the Support Vector Machines (SVM) [3, 23], the current state-of-the-art SL algorithm. However, SSL may encounter problems if the unlabeled data contain the "irrelevant" data. This can be observed in Figure 1(b): The boundary of SSL is obviously unreasonable. A more reasonable decision plane should pull away

[1] In this paper, we only consider the binary cases while multi-way problems can be easily approached via standard techniques, e.g., the one vs others technique [9].

1550-4786/08 $25.00 © 2008 IEEE
DOI 10.1109/ICDM.2008.61

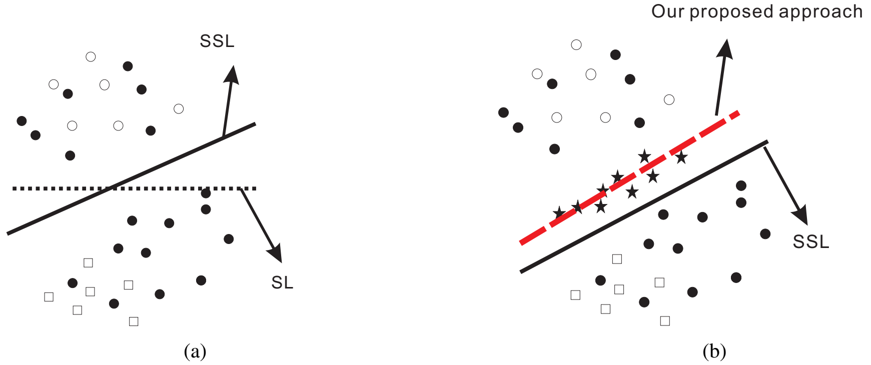


Figure 1. The “irrelevant” data $\star$’s can increase the performance of the SSL. The filled points ($\bullet$’s and $\star$’s) are unlabeled data, while the $\circ$’s and $\Box$’s are the two classes of labeled training samples. The filled $\star$’s describe the irrelevant unlabeled data. The decision planes of the SL and SSL are given by the SVMs.

the “relevant” data (maximizing the margin among the negative and positive data) while predicting the values of the “irrelevant” data as close to zero as possible (clustering the “0”-data around the decision line). Such a boundary (the dashed red line) can be observed in Figure 1(b).

Exploiting the unlabeled data neither positive nor negative can actually remedy the negative impact when both the unlabeled data and the labeled data are limited. Such a case can be seen in Figure 2. Assume the ground truth boundary is given as the dashed line in Figure 2(a). However, due to the limited training data (including both the labeled and relevant unlabeled data), the learned SSL boundary may be deviated from the actual one (as observed in Figure 2(a)). Sometimes, there are perhaps some “irrelevant” instances ($\star$’s in Figure 2(b)), being neither positive nor negative, mixed into the unlabeled data. By appropriately detecting and using these irrelevant data (trying to cluster such irrelevant unlabeled data around the decision plane), one can actually learn a more reasonable boundary as seen in Figure 2(b).

The idea of learning with the irrelevant data is similar to the work proposed in [19, 24], where the irrelevant data are called *universum*. However, they designed their system only within the Supervised Learning framework. In addition, these universum data need to be specified beforehand and are merely used as the labeled third class of samples. In other words, one needs to know which instances are universum data in advance so as to build a decision boundary. In comparison, we propose to exploit such irrelevant data in the semi-supervised context. More importantly, we do not need to specify which samples belong to the universum. Instead, we can learn from general unlabeled data, which means those relevant data or irrelevant data are mixed in the unlabeled data. Our novel model can output a more reasonable decision boundary, while simultaneously detecting the relevant data and irrelevant data automatically after the learning is finished.

Indeed, as far as we know, this work presents a novel study on how to perform learning from general unlabeled data consisting of both relevant and irrelevant instances. When the irrelevant data are known as prior knowledge by the user, this is the idea of “SSL with universum” proposed in [27]. In contrast, our work presents a more difficult and general SSL framework, where irrelevant data are mixed with the relevant unlabeled data, without any knowledge on which samples are relevant or irrelevant beforehand. As a major contribution, we successfully formulate such a difficult problem as a Semi-definite Programming (SDP) problem [13, 6, 21], making the framework solvable in polynomial time. Both theoretical analysis and empirical investigations demonstrate that the proposed framework outperforms the traditional SSL in many cases.

The rest of this paper is organized as follows. In the next section, we discuss the related work. In Section 3, we detail the proposed framework including the model definition, the theoretical analysis, and the practical solving method. In this section, we will demonstrate how the proposed model can be formulated in a Mixed Integer Programming (MIP) problem [16] and finally relaxed to be an SDP problem. In Section 4, we conduct a series of experiments to validate our novel approach. Finally, we set out the conclusion with final remarks.

2 Related Work

Researchers have devoted a lot of efforts on how to utilize unlabeled data via the effective semi-supervised learning [4, 28]. One assumption for SSL is that unlabeled data

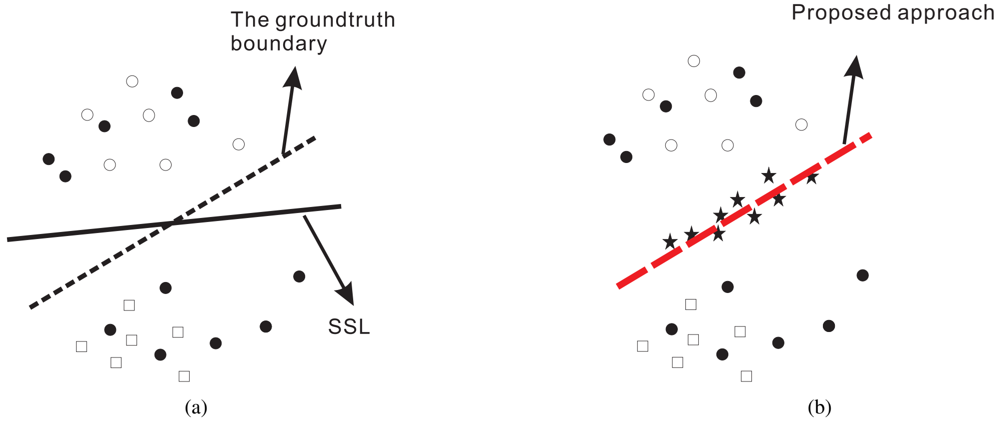


Figure 2. The "irrelevant" data $\star$'s can increase the performance when only a limited number of relevant unlabeled data is available. The filled points ($\bullet$'s and $\star$'s) are unlabeled data, while the $\circ$'s and $\square$'s are the two classes of labeled training samples. The filled $\star$'s describe the irrelevant unlabeled data. The decision plane of the SSL is given by the SVM.

are required to share the same distribution as the labeled data. The above assumption is relaxed in [2], where the distribution of training data is allowed to be arbitrarily different from the distribution of the unlabeled data. Unfortunately, although tolerating different distributions, it still requires the unlabeled data share the same class categories as the labeled training data. In fact, such requirement is enforced in most of SSL algorithms [28].

Alternatively, [24] studied the universum data, a special kind of unlabeled data that do not belong to any classes of the problem at hand, and showed that the universum data could boost the classification performance by encoding the prior knowledge of the domain. However, this interesting work is conducted in the context of Supervised Learning. The universum data need to be specified beforehand and are just used as the third class of samples. [17] proposed a Self-taught Learning (STL) and showed that weakly-related unlabeled data sharing a little structural information with the current task could also benefit the classification performance. The problem is that those weakly-related data are only exploited for extracting feature patterns and they are not involved in optimizing the decision boundary. Empirical study shows that STL sometimes extracts misleading patterns and hence might hurt the performance.

In addition, [11] and [27] studied the case that unlabeled data are a mixture of both relevant data, which are from the same domain of the current task, and irrelevant data, which are from a different task or the background. More specifically, [27] assumed that the prior knowledge about the composition of the mixture, i.e., the universum data and the data from the same distribution as the training data, is clear before learning a semi-supervised classification model. However, the application of the above methods requires the assumption that the prior knowledge of the composition of data should be known before learning. In contrast, this paper leverages the above requirement by learning from general unlabeled data which we do not know are relevant or not.

As a brief summary, our proposed framework presents a novel SSL framework that can learn from general unlabeled data. Such unlabeled data could consist of both relevant and irrelevant data. More importantly, we do not need to know which instances are relevant or irrelevant data. Based on solving an SDP problem, the proposed algorithm is able to automatically detect them, and consequently outputs a classification boundary that can exploit the unlabeled data more appropriately and more reasonably.

3 SSL from General Unlabeled Data

In this section, we first present the problem definition and the notation used in the paper. We then introduce the model definition, the theoretical analysis and the practical solving method in turn.

3.1 Problem Formalism

Given a training data set D, consisting of l labeled samples $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_l, y_l)\}$ drawn i.i.d. from a certain distribution S. Here $\mathbf{x}_i \in \mathcal{R}^n$ $(i = 1, 2, \ldots, l)$ describes an input feature vector, and $y_i \in \{-1, +1\}$ is the category label for $\mathbf{x}_i$. In addition, assume that m $(m \gg l)$ unlabeled data samples $\{\mathbf{x}_{l+1}, \mathbf{x}_{l+2}, \ldots, \mathbf{x}_{l+m}\}$ are also available (for brevity, we denote $n = l + m$). The unlabeled data contain both the relevant data sharing the same labels i.e., $\{-1,+1\}$ as the labeled data, and the irrelevant data which are different from the labeled data. Moreover, there are no prior knowledge on which instances are relevant or irrelevant.

The basic task here can be informally described as seeking a hypothesis $h : \mathcal{R}^n \rightarrow \{-1,+1\}$ that can predict the label $y \in \{-1,+1\}$ for the future input data sample $\mathbf{z} \in \mathcal{R}^n$ sampled from S by appropriately exploiting both the labeled data and the general unlabeled data. The hypothesis usually takes the linear form of $h = sign(f(\mathbf{z}))$, where $f(\mathbf{z}) = \mathbf{w} \cdot \mathbf{z} + b$ ($\mathbf{w} \in \mathcal{R}^n$, $b \in \mathcal{R}$). Note that the linear form can be easily extended to the non-linear form based on the standard kernelization trick [18].

3.2 Framework

The novel framework is introduced in the following. We first present the model definition followed by the theoretical analysis showing the inner justifications of our model. Finally, we show how to transform the problem to an SDP problem.

3.2.1 Model Definition

The novel model is formulated as the following Problem I:

Problem I:

$$\begin{aligned}
\min_{\mathbf{w},b,\xi,\eta,\mathbf{y}_{l+1:n}} \quad & \frac{1}{2}||\mathbf{w}||^2 + C_L\sum_{i=1}^{l}\xi_i + C_U\sum_{j=l+1}^{n}\min(\eta_j,\xi_j) \\
\text{s.t.} \quad & y_i(\mathbf{w}_i\cdot\mathbf{x}_i + b) \geq 1-\xi_i, i=1,\ldots,l, && (1)\\
& y_j(\mathbf{w}_j\cdot\mathbf{x}_j + b) \geq 1-\xi_j, && (2)\\
& |\mathbf{w}_j\cdot\mathbf{x}_j + b| \leq \varepsilon + \eta_j, && (3)\\
& \eta_j \geq 0, j = l+1,\ldots,n, \\
& \xi_k \geq 0, k = 1,\ldots,n,
\end{aligned}$$

where $\mathbf{x}_i$, $i = 1,\ldots,l$ are the labeled training samples. Namely, $y_i \in \{-1,+1\}$ $i = 1,\ldots,l$ is known beforehand. $\mathbf{x}_j$, $j = l+1,\ldots,n$ are the unlabeled data, where the associated labels are unknown, but restricted in the set of $\{-1,0,+1\}$. C_L and C_U are two positive penalty parameters used to trade-off the margin and the training loss. ε is a small positive parameter describing the insensitiveness level.

Constraint (1) describes the loss for the labeled data. Constraint (2) provides the loss if $\mathbf{x}_j$ is judged as the ±1 (i.e., the relevant data), while (3) presents the loss if $\mathbf{x}_j$ is judged as the class of 0 (i.e., the irrelevant class). The loss incurred by the unlabeled sample $\mathbf{x}_j$ is finally given by the minimum loss that it is judged as the class of ±1 or 0. This can be seen in the objective function of Problem I. Intuitively, the above model attempts to maximize the margin among the positive relevant data and negative relevant data, while predicting the values of the irrelevant data as close to zero as possible simultaneously. In addition, our model can automatically detect or assign the unlabeled samples to either ±1 (relevant classes) or 0 (irrelevant class) by choosing the smaller cost associated with the assigned label.

Note that two types of loss functions are adopted in Problem I. The loss function for the relevant data is the hinge loss $H_{-\varepsilon} = max\{0, t-\varepsilon\}$ as seen in (2), where $t = 1$. On the other hand, the loss function of the irrelevant data is defined as the ε-insensitive loss $U[t] = H_{-\varepsilon}[t] + H_{\varepsilon}[t]$. Both loss functions are plotted in Figure 3. When a data point is judged as a relevant instance, we should push it as faraway as possible from the margin $f(\mathbf{z}) = \pm1$. Hence a hinge loss is more appropriate for such a setting. When the data point belongs to the irrelevant class, it should be around the decision plane $f(\mathbf{z}) = 0$. In this sense, an ε-insensitive loss function is more suitable. An analogy can also be seen in choosing the loss functions for SVM (using hinge loss) and Support Vector Regression (using ε-insensitive loss) [20].

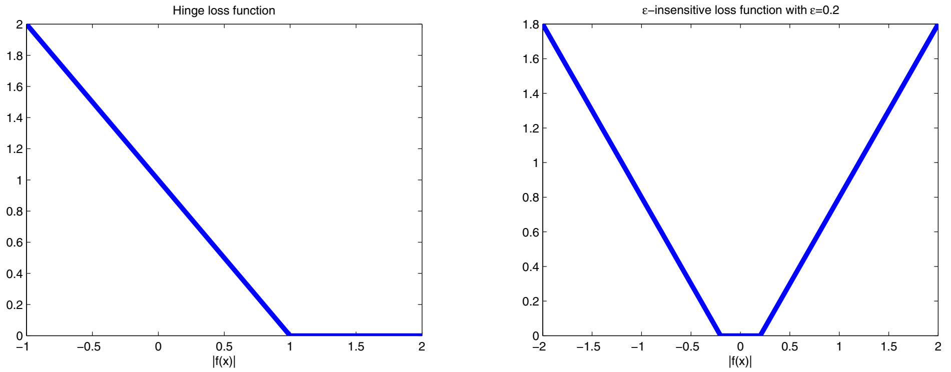
It is not easy to directly optimize Problem I because of the operator of min. However, by introducing an integer variables $d_j = \begin{cases} 0 & \text{if} \quad y_j = \pm1 \\ 1 & \text{if} \quad y_j = 0 \end{cases}$, $\forall j, l+1 \leq j \leq n$, we can transform Problem I to the following problem:

Problem II:

$$\begin{aligned}
\min_{\mathbf{w},b,\xi,\eta,\mathbf{y}_{l+1:n},\mathbf{d}} \quad & \frac{1}{2}||\mathbf{w}||^2 + C_L\sum_{i=1}^{l}\xi_i \\
+ \quad & C_U\sum_{j=l+1}^{n}(\eta_j+\xi_j), && (4)\\
\text{s.t.} \quad & \\
& y_i(\mathbf{w}_i\cdot\mathbf{x}_i + b) \geq 1-\xi_i, i=1,\ldots,l && (5)\\
& y_j(\mathbf{w}_j\cdot\mathbf{x}_j + b) + \xi_j + M(1-d_j) \geq 1, && (6)\\
& |\mathbf{w}_j\cdot\mathbf{x}_j + b| \leq \varepsilon + \eta_j + Md_j, && (7)\\
& d_j = \{0,1\} \quad j = l+1,\ldots,n, \\
& \eta_j \geq 0, j = l+1,\ldots,n, \\
& \xi_k \geq 0, k = 1,\ldots,n.
\end{aligned}$$

In the above, M is a large positive constant. When d_j is equal to 0, $M(1-d_j) = M$ is a big value. Hence (6) will naturally be satisfied, leading $\xi_i = 0$ and further $\min(\xi_j,\eta_j) = \xi_j + \eta_j$. A similar analysis can be obtained when $d_j = 1$. Therefore, we can know that Problem II is strictly equivalent to Problem I, provided that M is set to a sufficiently large value. Problem II is a Mixed Integer Programming problem [1, 16].

In the literature, there are a lot of proposals which can solve the MIP problem. In the following, we will first derive a theorem showing the justification of our proposed algorithm. We then revisit the optimization and propose our



(a) (b)

Figure 3. Hinge loss and ε-insensitive loss

practical solving method.

3.2.2 Analysis

In this section, we conduct some analysis showing that the utilization of irrelevant data has a nice theoretical justification. For clarity, we slightly modify Problem II to the following optimization problem. Based on the modified problem, we then derive the analysis. Problem II is changed as follows:

$$
\begin{aligned}
\min_{\mathbf{w},b,\xi,\eta,\mathbf{y}_{l+1:n},\mathbf{d}} \quad & \frac{1}{2}||\mathbf{w}||^2 + C_L\sum_{i=1}^{l}\xi_i + C_{rU}\sum_{j=l+1}^{n}\xi_j \\
& +C_{iU}\sum_{j=l+1}^{n}\eta_j \qquad (8) \\
\text{s.t.} \quad & y_i(\mathbf{w}_i\cdot\mathbf{x}_i + b) \geq 1-\xi_i, i=1,\ldots,l \\
& y_j(\mathbf{w}_j\cdot\mathbf{x}_j + b) + \xi_j + M(1-d_j) \geq 1, \\
& |\mathbf{w}_j\cdot\mathbf{x}_j + b| \leq \varepsilon + \eta_j + Md_j, \\
& d_j = \{0,1\} \quad j=l+1,\ldots,n. \\
& \eta_j \geq 0, j=l+1,\ldots,n, \\
& \xi_k \geq 0, k=1,\ldots,n.
\end{aligned}
$$

C_{rU} represents the penalty parameter for the relevant samples, while C_{iU} describes the penalty imposed on the irrelevant data points. We first present the following theory.

Theorem 1 *The above learning machine with $C_{iU} = \infty$ and $\varepsilon = 0$ is equivalent to training a standard Transductive SVM [5] with the training points projected onto the orthogonal complement of span $\{\mathbf{z}_j - \mathbf{z}_0, \mathbf{z}_j \in \mathcal{U}\}$, where $\mathbf{z}_0$ is an arbitrary element of the space spanned by the irrelevant samples denoted by $\mathcal{U}$.*

Sketch of Proof: $C_{iU} = \infty$ and $\varepsilon = 0$ implies that any $\mathbf{w}$ yielding the optimal solution of (8) satisfies $\mathbf{w}\cdot\mathbf{z} + b = 0$ for any $\mathbf{z}$ judged as irrelevant samples. Hence, we have $\mathbf{w}\cdot(\mathbf{z}-\mathbf{z}_0) = 0$, implying $\mathbf{w}$ is orthogonal to the subspace spanned by all the irrelevant samples. Hence the optimization of (8) intends to find a traditional transductive SVM in a subspace which contains only the relevant samples, while the irrelevant samples are suppressed. In addition, from the previous argument, the space $\mathcal{U}$ spanned by the irrelevant samples can also benefit the classification, since it is $\mathcal{U}$ that decides the optimization subspace. □

Theorem 1 shows that the optimization of our proposed algorithm actually tries to find the most suitable subspace in which the margin can be maximized while the overall error can be minimized. The irrelevant data do not contribute to the final accuracy directly. However, it determines the subspace where the resultant decision boundary is derived and will consequently affect the final performance. Theorem 1 clearly shows how the irrelevant data can affect and eventually improve the overall performance.

3.2.3 Practical Solving Method

We now revisit the optimization of Problem II. Although there are softwares that are able to deal with MIP involved in Problem II, the computational complexity is usually high. It is even difficult to perform optimization with more than 50 $\{0,1\}$ integer variables. Hence we would like to relax the problem to other solvable optimization forms. To achieve this purpose, we first reformulate Problem II to its dual form.

Problem III:

$$\begin{aligned}
\max_{\lambda,\mathbf{z}^+,\mathbf{z}^-}\ \min_{\mathbf{y}_{l+1:n},\mathbf{d}} \quad & -\beta^T\mathbf{K}\beta + 2\sum_{i=1}^{n}\lambda_i \\
& -2M\sum_{j=l+1}^{n}(1-d_j)\lambda_j \\
& -2M\sum_{j=l+1}^{n}d_j(z_j^- + z_j^+) \\
\text{s.t.} \quad & 0 \le \lambda_i \le C_L, i = 1,\ldots,l && (9)\\
& 0 \le \lambda_j \le C_U, && (10)\\
& z_j^- + z_j^+ \le C_U, && (11)\\
& z_j^-, z_j^+ \ge 0, && (12)\\
& d_j = \{0,1\}, j = 1+1,\ldots,n && (13)
\end{aligned}$$

In the above, β_j is defined as $\beta_j = \begin{cases} \lambda_j y_j & j \le l \\ \lambda_j y_j + (z_j^- - z_j^+) & l+1 \le j \le n \end{cases}$. λ_j is the Lagrangian multiplier for (5) and (6) associated with $\mathbf{x}_j$, and z_j^- and z_j^- correspond the Lagrangian multipliers for (7) when the abs operator is expanded. And $\mathbf{K}$ is the kernel matrix defined as $\mathbf{K}_{i,j} = \mathbf{x}_i \cdot \mathbf{x}_j$.

Before proceeding to re-organized Problem III, we present some notation first. We denote a new vector $\alpha = (\lambda; \mathbf{z}_-; \mathbf{z}_+)$. We further define $\mathbf{P}_1 = (X\text{Diag}(\mathbf{y}), X_{l+1:n}, -X_{l+1:n})^T$, where X represents the matrix $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, $X_{k_1:k_2}$ represents the matrix consisting of the columns of X from k_1 to k_2, and $X \circ \text{Diag}(\mathbf{y})$ represents the element-wise matrix multiplication of X and $\text{Diag}(\mathbf{y})$. We further define $\mathbf{a} = (\mathbf{1}_l; \mathbf{1}_m - M(\mathbf{1} - \mathbf{d}); -M\mathbf{d}; -M\mathbf{d})$, where $\mathbf{1}_k$ represents a $\mathbf{k}$-dimension column vector with all the elements as 1. We denote the matrix $B = \begin{pmatrix} \mathbf{I}_{n\times n}, & \mathbf{0}_{n\times 2m} \\ \mathbf{0}_{m\times n}, & Q_{m\times 2m} \end{pmatrix}$, $Q_{m\times 2m} = (\mathbf{I}_{m\times m}, \mathbf{I}_{m\times m})$, $C = (\mathbf{C}_{\mathbf{L}l}; \mathbf{C}_{\mathbf{U}2m})$. Here $\mathbf{I}_{n\times n}$ is an $n \times n$ unit matrix, $\mathbf{0}_{k_1\times k_2}$ describes a $k_1 \times k_2$ matrix with all the elements as 0, $\mathbf{C}_{\mathbf{L}l}$ defines an l-dimensional column vector with all the elements as C_L. Other symbols are similarly defined.

We can re-organized Problem III to the following problem by using the above notation.

$$\begin{aligned}
\max_{\alpha}\ \min_{\mathbf{y}_{l+1:n},\mathbf{d}} \quad & -\alpha^T\mathbf{P}_1\mathbf{P}_1^T\alpha + 2\mathbf{a}^T\alpha \\
\text{s.t.} \quad & \alpha \ge 0, \\
& B\alpha \le C, \\
& d_j \in \{0,1\}, \forall j, l+1 \le j \le n.
\end{aligned}$$

Once again, the dual form of the above optimization objective can be written to the following problem:

$$\begin{aligned}
\max_{\alpha}\ \min_{\mathbf{y}_{l+1:n},\mathbf{d},\nu,\delta} \quad & -\alpha^T\mathbf{P}_1\mathbf{P}_1^T\alpha + 2\mathbf{a}^T\alpha + 2\nu^T\alpha \\
& +2\delta^T(\mathbf{C} - \mathbf{B}\alpha), && (14)
\end{aligned}$$

where $\nu, \delta \ge 0$ are the Lagrangian multipliers.

We can easily obtain the optimal $\alpha = (\mathbf{P}_1\mathbf{P}_1^T)^{-1}(\mathbf{a} + \nu - B^T\delta)$. Substituting the optimum value of α into (14), we further get the optimization problem as follows:

$$\begin{aligned}
\max_{\alpha}\ \min_{\mathbf{y}_{l+1:n},\mathbf{d},\nu,\delta} \quad & (\mathbf{a} + \nu - B^T\delta)^T(\mathbf{P}_1\mathbf{P}_1^T)^{-1}(\mathbf{a} + \nu - B^T\delta) \\
& +2\delta^T\mathbf{C} \\
\text{s.t.} \quad & \nu \ge 0, \delta \ge 0, \\
& d_j \in \{0,1\}, \forall j, l+1 \le j \le n.
\end{aligned}$$

Finally, the above optimization problem can equivalently be transformed to a form similar to the Semi-definite Problem (SDP) by using Schur Complement Lemma [12, 13].
Problem IV:

$$\begin{aligned}
\min_{\mathbf{y}_{l+1:n},\mathbf{d},\nu,\delta,t} \quad & t \quad \text{s.t.} \\
& \begin{pmatrix} P & \mathbf{a} + \nu - B^T\delta \\ (\mathbf{a} + \nu - B^T\delta)^T & t - 2\delta^T\mathbf{C} \end{pmatrix} \succeq 0, \\
& d_j \in \{0,1\}, \\
& y_j \in \{-1,+1\}, \forall j, l+1 \le j \le n.
\end{aligned}$$

Here P is defined as $\begin{pmatrix} \mathbf{K} \circ (\mathbf{y}\mathbf{y}^T) & \text{Diag}(\mathbf{y})\mathbf{K}_{1:n,l:n} & -\text{Diag}(\mathbf{y})\mathbf{K}_{1:n,l:n} \\ \mathbf{K}_{1:n,l:n}^T\text{Diag}(\mathbf{y}) & \mathbf{K}_{l+1:n,l+1:n} & -\mathbf{K}_{l+1:n,l+1:n} \\ -\mathbf{K}_{1:n,l:n}^T\text{Diag}(\mathbf{y}) & -\mathbf{K}_{l+1:n,l+1:n} & \mathbf{K}_{l+1:n,l+1:n} \end{pmatrix}$ and a matrix $\mathbf{A} \succeq 0$ means that $\mathbf{A}$ is a Semi-definite matrix.

Similar to the work presented in [13], we relax $(\mathbf{y}\mathbf{y}^T)$ as rank-one matrix $\mathbf{M}$. We further relax $d_j \in \{0,1\}$ to $0 \le d_j \le 1$. We can finally write the optimization problem as Problem V:

Problem V:

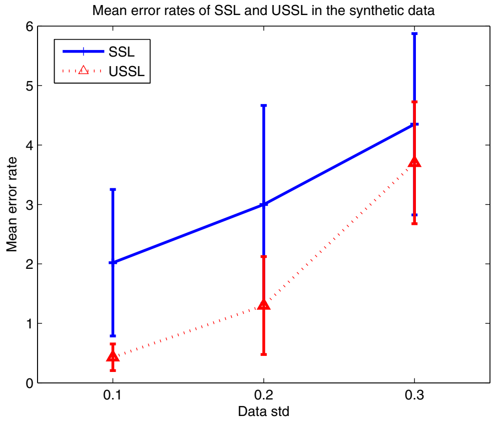
$$\begin{aligned}
\min_{\mathbf{M},\mathbf{d},\nu,\delta,t} \quad & t \quad \text{s.t.} \\
& \begin{pmatrix} P & \mathbf{a} + \nu - B^T\delta \\ (\mathbf{a} + \nu - B^T\delta)^T & t - 2\delta^T\mathbf{C} \end{pmatrix} \succeq 0, \\
& 0 \le d_j \le 1, \\
& rank(\mathbf{M}) = 1, \mathbf{M}_{1:l,1:l} = \mathbf{y}_{1:l}\mathbf{y}_{1:l}^T.
\end{aligned}$$

Following most optimization methods in SSL [25, 26, 5, 22], we further remove the rank-one constraint, the above problem is exactly an SDP problem. Note that $\text{Diag}(\mathbf{y})$ appearing in the matrix P can be represented by the elements of $\mathbf{M}$. For example, assume $y_1 = 1$, then $\text{Diag}(\mathbf{y})$ can be written as $\text{Diag}(M_{11}, M_{12}, ..., M_{1n})$. This SDP problem can be solved in polynomial time by some packages such as Sedumi[21].

4 Experiment

In this section, we evaluate our proposed framework on both synthetic and real data. A synthetic example will be firstly presented in order to illustrate the model clearly. We then compare our model with the traditional SSL and the Universum Support Vector Machine (USVM) [24] on benchmark data sets, USPS [2] and MNIST data[3]. For brevity, we name our model as Universum Semi-supervised Learning, in short, USSL from now on. However, we should keep in mind that it is significantly different from the work presented in [27] in that the universum must be known beforehand in their work, while we do not have such requirement. Hence our proposed model presents a more general SSL framework. We implement our model by using a generic convex programming solver CVX.[4] The traditional SSL and the universum SVM are solved based on the package UniverSVM.[5]

4.1 Evaluation on Synthetic Data



(a)

Figure 4. Comparison of SSL and the proposed USSL on the synthetic data

We generate three synthetic data sets to validate our proposed algorithm. In more details, we obtain the training data for all the three data sets from three two-dimensional Gaussian distributions, which are centered at -0.3, 0, and $+0.3$ respectively. The two types of relevant data are centered at ± 0.3 both with the standard deviations as 0.13 for each data set, while the irrelevant data are located around 0, but with standard deviations as 0.1, 0.2, and 0.3 respectively for three data sets. The number of training samples for the labeled data and the relevant unlabeled data is respectively set to 5 and 30 for each class in all the three sets. The number of irrelevant unlabeled data samples for all the three cases is also set to 30. The test data consists of 500 samples for each class, generated from the same distributions as the labeled data. We train our proposed model USSL in comparison with SSL on the training data consisting of both irrelevant and relevant data samples, and evaluate its performance on the test data sets. In both SSL and USSL, C_U and C_L are set to 100. ε is set to 0.2. Note that again, we do not know which data samples are irrelevant or irrelevant beforehand. They are merely input as the unlabeled data for training in both USSL and SSL. The above process is repeated for 20 times and the average accuracy is reported in Figure 4.

It is obvious that the proposed general framework USSL demonstrates much better performance than SSL. The mean error rates of USSL are significantly lower than those of SSL in all the three cases. On the other hand, when the standard deviation increases, USSL tends to approximate the SSL in terms of the error rate, since it is difficult to detect irrelevant data in such cases.

In order to have a closer examination on the proposed USSL, we also draw the training set including the labeled and unlabeled data, the test data, and the decision boundaries for one of 20 evaluations in Figure 5. Figure 5(a), (b), and (c) show the training samples for the three sets, where the labeled samples are plotted as $\circ$'s and $\square$'s for $+1$ and -1 class respectively, while **.**'s depict the unlabeled instances consisting of both relevant and irrelevant samples. Figure 5(d), (e), and (f) show the final class labels for the unlabeled data and the decision boundary given by the traditional SSL. The filled points represent the unlabeled data, but their shapes imply their class, i.e., the filled $\square$'s are judged as -1 class, while the filled $\circ$'s are classified as $+1$ class. Similarly, we show the decision boundary given by USSL and the associated final class labels of the unlabeled samples for the three cases in Figure 5(g), (h), and (i) respectively. We use the similar symbols to describe different points. The difference is that our proposed USSL is able to indicate which samples are irrelevant. Such irrelevant samples are finally marked as ▲. It is interesting that almost all the irrelevant samples can be correctly detected by our proposed USSL as observed in these three sub figures. Moreover, the decision boundaries given by USSL are actually more reasonable than the ones derived by the traditional SSL. This can be also observed in Figure 5(j), (k), and (l), which show the test results for the three cases respectively.

4.2 Evaluation on Real Data

In this section, we evaluate the proposed novel model in comparison with the traditional SSL and the USVM [24]

[2] The USPS data set can be downloaded from the web site http://www-stat-class.standford.edu / tibs/ElemStatLearn/data.html.

[3] The MNIST data set is available at http://yann.lecun.com/exdb/mnist.

[4] The matlab source codes of the CVX package can be downloaded from http://www.stanford.edu/ boyd/cvx/.

[5] The package of UniverSVM can be obtained from http://www.kyb.tuebingen.mpg.de/bs/people/fabee/universvm.html.

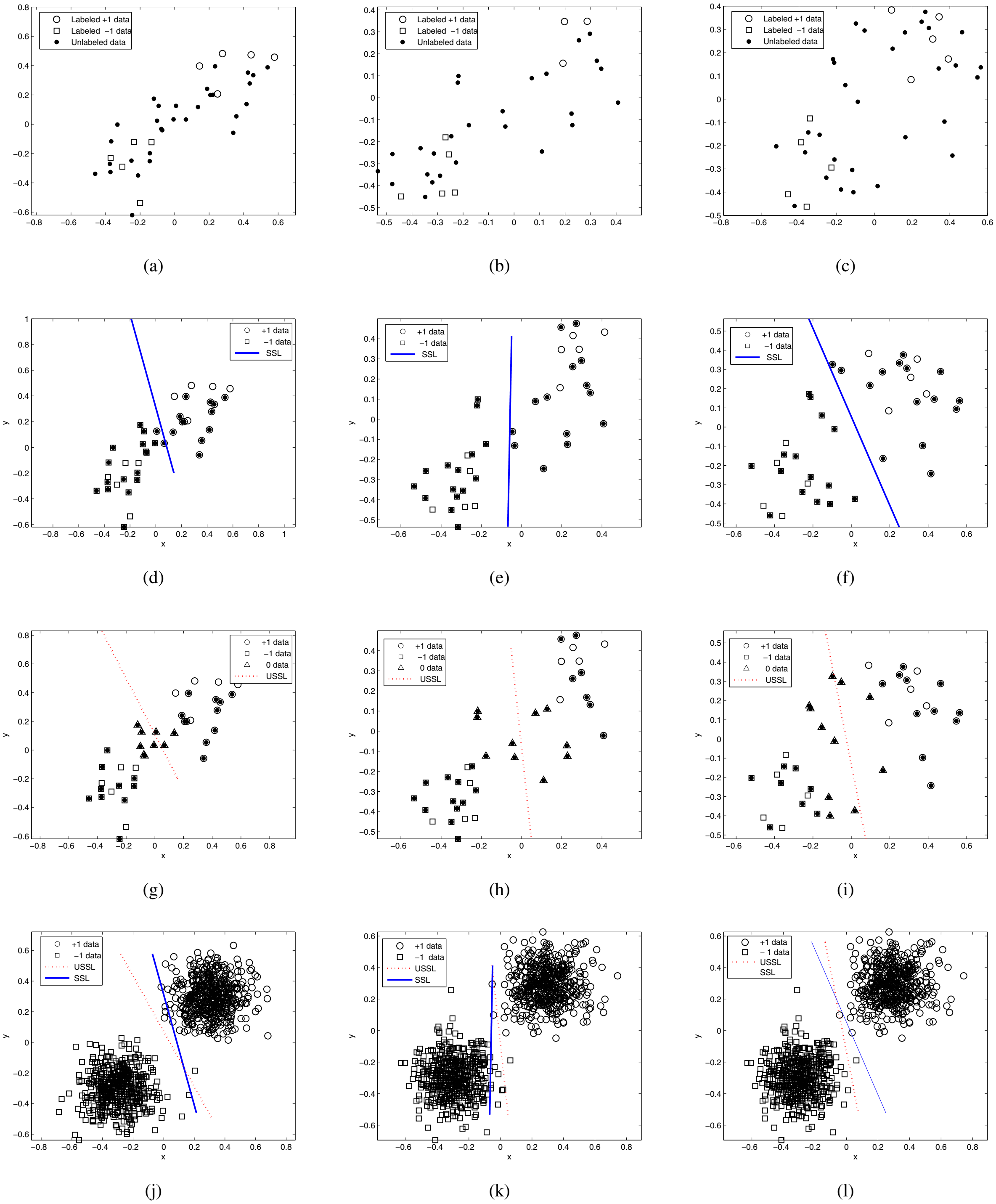


Figure 5. Comparison of SSL and the proposed model USSL on synthetic data. (a)-(c) plots the training data for the three data sets respectively. (d)-(f) plots the decision boundary given by SSL as well as the class label of the unlabeled data assigned by SSL. (g)-(i) plot the decision boundary given by USSL as well as the class label of the unlabeled data assigned by USSL. (j)-(l) show the results on test data. The proposed USSL generates more reasonable decision boundaries and outperforms the traditional SSL.

on real data, the USPS and the MNIST data. We follow [24, 19] and exploit the digits of 5 and 8 as the labeled data and use the remaining digits as the irrelevant data. Hence we have 8 data sets, depending on which category of digits is used as the irrelevant data. We randomly extract 20 labeled samples and 30 random data points as relevant unlabeled samples from 5 and 8 respectively. We further obtain 30 samples randomly extracted from a certain category of digits other than 5 and 8. The test data set contains 400 digits randomly extracted from the 5 and 8 digits. The parameters involved in SSL and USSL are searched via cross validation. More specifically, C_L and C_U are searched in $\{1, 10, 100, 1000\}$, while ε is searched in $\{0.1, 0.2, 0.3, 0.4\}$. The final test accuracy is given as the result averaged on the 10 random evaluations for both USPS and MNIST. In addition, as verified by many researches in Optical Character Recognition [14, 15], especially in handwritten numeral recognition, kernel based methods are just slightly better than the linear classifier, but with significantly heavier computational cost.[6] Hence, we only conduct the comparisons based on the linear version of USVM, USSL and SSL in the following.

The evaluation results are reported in Table 1 and Table 2 for USPS and MNIST respectively. Once again, our proposed USSL outperforms the traditional SSL and the USVM. More specifically, the proposed USSL demonstrates significantly better performance than SSL and USVM in the 0, 1, 2, 3, 6, and 7 data sets of USPS according to a t-test at the 5% significance level. Similarly, a t-test indicates that the result of USSL is also significantly different from those of SSL and USVM in the 0, 1, 3, 4, 6, 7, and 9 data sets of MNIST at the significant level of 5%. SSL simply regards all the unlabeled data as relevant data, while USVM considers all the unlabeled data as universum. Hence it is inappropriate for them to deal with the general unlabeled data containing both relevant and irrelevant data. In comparison, our proposed approach can automatically model the impact caused by the relevant and irrelevant data into the final decision boundary. It demonstrates superior performance and is more appropriate in handling Semi-supervising Learning from general data.

Table 1. Experimental results on USPS data

Data set	USVM	SSL	USSL
0	67.05 ± 2.31	85.05 ± 1.94	$\mathbf{89.85 \pm 1.47}$
1	71.45 ± 1.59	83.61 ± 2.52	$\mathbf{89.23 \pm 1.89}$
2	69.50 ± 4.29	84.44 ± 2.08	$\mathbf{89.81 \pm 2.34}$
3	70.43 ± 1.68	84.75 ± 1.86	$\mathbf{89.65 \pm 2.24}$
4	65.80 ± 3.04	85.12 ± 3.91	$\mathbf{86.69 \pm 2.01}$
6	64.80 ± 2.36	78.45 ± 2.21	$\mathbf{83.70 \pm 1.90}$
7	66.93 ± 3.75	87.37 ± 2.51	$\mathbf{90.42 \pm 1.75}$
9	72.37 ± 3.42	82.86 ± 2.39	$\mathbf{85.13 \pm 2.31}$

Table 2. Experimental results on MNIST data

Data Set	USVM	SSL	USSL
0	45.25 ± 2.19	53.25 ± 2.84	$\mathbf{58.25 \pm 2.11}$
1	52.77 ± 1.42	54.10 ± 2.78	$\mathbf{60.25 \pm 2.75}$
2	54.58 ± 2.67	56.92 ± 3.12	$\mathbf{57.67 \pm 2.97}$
3	55.14 ± 1.90	52.09 ± 2.30	$\mathbf{57.25 \pm 1.32}$
4	56.65 ± 1.22	57.12 ± 2.49	$\mathbf{59.25 \pm 2.10}$
6	52.75 ± 2.80	54.50 ± 2.12	$\mathbf{57.67 \pm 1.27}$
7	60.51 ± 2.12	58.09 ± 3.01	$\mathbf{68.50 \pm 2.26}$
9	59.25 ± 1.15	48.25 ± 2.64	$\mathbf{63.00 \pm 1.50}$

5 Conclusion

We have proposed a novel framework that can learn from general unlabeled data. In contrast to the traditional Semi-supervised Learning that requires unlabeled data to share the same category labels as the labeled data, the proposed framework is able to learn from unlabeled data with irrelevant samples. Moreover, we do not need the prior knowledge on which data samples are relevant or irrelevant. Consequently it is significantly different from the recent Semi-supervised Learning with universum or the Universum Support Vector Machines. As an important contribution, we have successfully formulated this new learning approach as a Semi-definite Programming problem, making it solvable in polynomial time. We have also presented theoretical analysis to justify our model. A series of experiments demonstrate that this novel framework has advantages over the Semi-supervised Learning on both synthetic and real data in many facets.

Acknowledgements

The work described in this paper was fully supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4150/07E and No. CUHK4125/07).

References

[1] K. Bennett and A. Demiriz. Semi-supervised support vector machines. In *Advances in Neural Information Processing Systems 11 (NIPS 11)*, pages 368–374. MIT Press, 1999.

[2] S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning for differing training and test distributions. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 81–88, New York, NY, USA, 2007. ACM Press.

[6] The performance of various methods on MNIST can be seen in the web site http://yann.lecun.com/exdb/mnist/.

[3] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

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

[5] R. Collobert, F. Sinz, J. Weston, and L. Bottou. Large scale transductive SVMs. *Journal of Machine Learning Reseaerch*, 7:1687–1712, 2006.

[6] T. De Bie and N. Cristianini. Semi-supervised learning using semi-definite programming. In O. Chapelle, B. Schölkopf, and A. Zien, editors, *Semi-Supervised Learning*, pages 120–135. MIT Press, Cambridge, MA, 2006.

[7] G. Druck, C. Pal, A. McCallum, and X. Zhu. Semi-supervised classification with hybrid generative/discriminative methods. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 280–289, New York, NY, USA, 2007. ACM.

[8] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[9] C. W. Hsu and C. J. Lin. A comparision of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13:415–425, 2002.

[10] K. Huang, H. Yang, I. King, and M. R. Lyu. Maxi-min margin machine: Learning large margin classifiers globally and locally. *IEEE Transactions on Neural Networks*, 19:260–272, 2008.

[11] S. Kaski and J. Peltonen. Learning from relevant tasks only. In *Machine Learning: ECML 2007*, pages 608–615, 2007.

[12] S. Kruk and H. Wolkowicz. General nonlinear programming. In H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors, *Handbook of Semidefinite Programming: Theory, Algorithms, and Applications*, pages 563–575. Kluwer Academic Publishers, Boston, 2000.

[13] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[14] C.-L. Liu and H. Fujisawa. Classification and learning methods for character recognition: Advances and remaining problems. In *Machine Learning in Document Analysis and Recognition*, pages 139–161, 2008.

[15] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa. Handwritten digit recognition: benchmarking of state-of-the-art techniques. *Pattern Recognition*, 36(10):2271–2285, 2003.

[16] H. Marchand, A. Martin, R. Weismantel, and L. Wolsey. Cutting planes in integer and mixed integer programming. *Discrete Appl. Math.*, 123(1-3):397–446, 2002.

[17] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 759–766, New York, NY, USA, 2007. ACM Press.

[18] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[19] F. H. Sinz, O. Chapelle, A. Agarwal, and B. Schölkopf. An analysis of inference with the universum. In *Advances in Neural Information Processing Systems (NIPS-07)*.

[20] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. Technical Report NC2-TR-1998-030, NeuroCOLT2, 1998.

[21] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999.

[22] H. Valizadegan and R. Jin. Generalized maximum margin clustering and unsupervised kernel learning. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*. MIT Press, Cambridge, MA, 2007.

[23] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 2nd edition, 1999.

[24] J. Weston, R. Collobert, F. Sinz, L. Bottou, and V. Vapnik. Inference with the universum. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 1009–1016, New York, NY, USA, 2006. ACM.

[25] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans. Maximum margin clustering. In *Advances in Neural Information Processing Systems 16 (NIPS 16)*, 2004.

[26] Z. Xu, R. Jin, J. Zhu, I. King, and M. Lyu. Efficient convex relaxation for transductive support vector machine. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1641–1648. MIT Press, Cambridge, MA, 2008.

[27] D. Zhang, J. Wang, F. Wang, and C. Zhang. Semi-supervised classification with universum. In *SDM*, pages 323–333, 2008.

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

[29] X. Zhu, Z. Ghahramani, and J. D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of Twentith International Conference on Machine Learning (ICML-2003)*, pages 912–919, 2003.