Mach Learn (2017) 106:573–593
DOI 10.1007/s10994-016-5606-4

Maximum margin partial label learning

Fei Yu[1,2] · **Min-Ling Zhang**[1,2]

Received: 15 February 2016 / Accepted: 24 October 2016 / Published online: 23 December 2016

Abstract Partial label learning aims to learn from training examples each associated with a set of *candidate* labels, among which only one label is valid for the training example. The basic strategy to learn from partial label examples is disambiguation, i.e. by trying to recover the ground-truth labeling information from the candidate label set. As one of the popular machine learning paradigms, maximum margin techniques have been employed to solve the partial label learning problem. Existing attempts perform disambiguation by optimizing the margin between the maximum modeling output from candidate labels and that from non-candidate ones. Nonetheless, this formulation ignores considering the margin between the ground-truth label and other candidate labels. In this paper, a new maximum margin formulation for partial label learning is proposed which directly optimizes the margin between the ground-truth label and all other labels. Specifically, the predictive model is learned via an alternating optimization procedure which coordinates the task of *ground-truth label identification* and *margin maximization* iteratively. Extensive experiments on artificial as well as real-world datasets show that the proposed approach is highly competitive to other well-established partial label learning approaches.

Keywords Partial label learning · Candidate label · Disambiguation · Maximum margin

Editors: Geoff Holmes, Tie-Yan Liu, Hang Li, Irwin King and Zhi-Hua Zhou.

✉ Min-Ling Zhang
zhangml@seu.edu.cn

Fei Yu
yuf@seu.edu.cn

[1] School of Computer Science and Engineering, Southeast University, Nanjing 210096, China

[2] Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, Nanjing, China

1 Introduction

Partial label learning deals with the problem where each training example is associated with a set of *candidate* labels, among which only one label is valid (Cour et al. 2011; Zhang 2014). In recent years, partial label learning techniques have been found useful in solving many real-world scenarios such as web mining (Jie and Orabona 2010), multimedia content analysis (Cour et al. 2009; Zeng et al. 2013), ecoinformatics (Liu and Dietterich 2012), etc.

Formally speaking, let $\mathcal{X} = \mathbb{R}^d$ be the d-dimensional instance space and $\mathcal{Y} = \{1, 2, \ldots, q\}$ be the label space with q class labels. Given the partial label training set $\mathcal{D} = \{(\mathbf{x}_i, S_i) \mid 1 \leq i \leq m\}$, the task of partial label learning is to induce a multi-class classifier $f : \mathcal{X} \mapsto \mathcal{Y}$ from $\mathcal{D}$. Here, $\mathbf{x}_i \in \mathcal{X}$ is a d-dimensional feature vector $(x_{i1}, x_{i2}, \ldots, x_{id})^\top$ and $S_i \subseteq \mathcal{Y}$ is the associated candidate label set. Partial label learning takes the core assumption that the ground-truth label y_i of $\mathbf{x}_i$ resides in its candidate label set S_i and not directly accessible to the learning algorithm.[1]

Intuitively, the basic strategy for handling partial label learning problem is *disambiguation*, i.e. trying to identify the ground-truth label from the candidate label set associated with each training example. As one of the popular machine learning techniques, maximum margin criterion has been applied to learn from partial label examples. Specifically, existing attempts disambiguate the partial label training example by optimizing the margin between the maximum modeling output from its candidate labels and that from its non-candidate labels (Nguyen and Caruana 2008). In other words, given the parametric model with parameters $\boldsymbol{\Theta}$ and $\boldsymbol{x}_i$'s modeling output $F(\mathbf{x}_i, y; \boldsymbol{\Theta})$ on each class label $y \in \mathcal{Y}$, the existing formulation works by maximizing the following predictive difference over $\mathbf{x}_i$: $\max_{y_j \in S_i} F(\mathbf{x}_i, y_j; \boldsymbol{\Theta}) - \max_{y_k \notin S_i} F(\mathbf{x}_i, y_k; \boldsymbol{\Theta})$. Nonetheless, this formulation fails to consider the predictive difference between the ground-truth label (i.e. y_i) and other labels in the candidate label set (i.e. $S_i \backslash \{y_i\}$). Due to the ignorance of such discriminative properties, the generalization performance of the resulting maximum margin partial label learning approach might be suboptimal.

Essentially, the task of partial label learning is to induce a multi-class classifier $f : \mathcal{X} \mapsto \mathcal{Y}$. Therefore, the canonical *multi-class margin*, i.e. $F(\mathbf{x}_i, y_i; \boldsymbol{\Theta}) - \max_{\tilde{y}_i \neq y_i} F(\mathbf{x}_i, \tilde{y}_i; \boldsymbol{\Theta})$, should be a natural choice to learn from partial label examples. In this way, the modeling output from the ground-truth label is distinguished with those from all the other labels. In view of this observation, a new maximum margin partial label learning approach named M3PL, i.e. *MaxiMum Margin Partial Label learning*, is proposed in this paper. Evidently, the major challenge in making use of the multi-class margin for partial label training examples lies in that the ground-truth labeling information is not accessible to the learning algorithm. To overcome this difficulty, an iterative optimization procedure is employed by M3PL which alternates between the task of identifying the ground-truth label and maximizing the multi-class margin. Comprehensive comparative studies against state-of-the-art partial label learning approaches clearly validate the effectiveness of the proposed formulation.

The remainder of this paper is organized as follows. Section 2 briefly discusses related work on partial label learning. Section 3 introduces technical details of the proposed M3PL

[1] In some of the literature the framework of *partial label learning* is also termed as *ambiguous label learning* (Hüllermeier and Beringer 2006; Chen et al. 2014), *soft label learning* (Côme et al. 2008), or *superset label learning* (Liu and Dietterich 2014). In addition, there are studies which admit cases where the ground-truth label is not confined with the candidate label set (Cid-Sueiro 2012).

approach. Section 4 reports experimental results across a broad range of datasets. Finally, Sect. 5 summarizes the paper and indicates several future research issues.

2 Related work

As the labeling information conveyed by each partial label training example is ambiguous, partial label learning can be regarded as one of the *weakly-supervised* learning frameworks. Conceptually speaking, it situates between the two ends of the supervision spectrum, i.e. standard supervised learning with explicit supervision and unsupervised learning with blind supervision. Learning with weak supervision has found wide application in solving various learning tasks as it is generally hard to obtain explicit and sufficient supervision information in real-world scenarios (Pfahringer 2012). In particular, partial label learning is related to several well-studied weakly-supervised learning frameworks, including *semi-supervised learning*, *multi-instance learning* and *multi-label learning*, while the weak supervision scenario considered by partial label learning is different to those counterpart frameworks.

Semi-supervised learning (Chapelle et al. 2006; Zhu and Goldberg 2009) aims to induce a classifier $f : \mathcal{X} \mapsto \mathcal{Y}$ from few labeled training examples along with abundant unlabeled training examples. For an unlabeled example the ground-truth label assumes the whole label space, while for a partial label example the ground-truth label is confined within its candidate label set. Multi-instance learning (Dietterich et al. 1997; Amores 2013) aims to induce a classifier $f : 2^{\mathcal{X}} \mapsto \mathcal{Y}$ from training examples each represented as a labeled bag of instances. For a multi-instance example the label is assigned to bag of instances, while for a partial label example the label is assigned to single instance. Multi-label learning (Zhang and Zhou 2014; Gibaja and Ventura 2015) aims to learn a classifier $f : \mathcal{X} \mapsto 2^{\mathcal{Y}}$ from training examples each associated with multiple labels. For a multi-label example the associated labels are all valid ones, while for a partial label example the associated labels are only candidate ones.

In recent years, a number of partial label learning approaches have been proposed by adapting major machine learning techniques. Maximum likelihood techniques are introduced to learn from partial label examples by maximizing the likelihood function $\sum_{i=1}^{m} \log(\sum_{y \in S_i} F(\mathbf{x}_i, y; \boldsymbol{\Theta}))$, where EM-based optimization is performed by treating the ground-truth label as a latent variable (Jin and Ghahramani 2003; Liu and Dietterich 2012). To enable convex optimization for partial label learning, a relaxed formulation is proposed by discriminating the average output from all candidate labels, i.e. $\frac{1}{|S_i|} \sum_{y \in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\Theta})$, against the outputs from non-candidate labels, i.e. $F(\boldsymbol{x}_i, y; \boldsymbol{\Theta})$ $(y \notin S_i)$ (Cour et al. 2011). For instanced-based approaches, the labeling information from neighboring training examples are combined by weighted voting to make predictions for unseen instances (Hüllermeier and Beringer 2006; Zhang and Yu 2015). There are also some approaches which transform the problem of partial label learning into the problem of binary classification by error-correcting output codes (ECOC) (Zhang 2014), the problem of sparse coding by dictionary learning (Chen et al. 2014), or the problem of multioutput regression by manifold analysis (Zhang et al. 2016).

Specifically, maximum margin techniques have also been employed to design partial label learning approaches (Nguyen and Caruana 2008). Given the parametric model $\boldsymbol{\Theta} = \{(\mathbf{w}_p, b_p) \mid 1 \leq p \leq q\}$ with one linear classifier $(\mathbf{w}_p, b_p)$ for each class label, the existing maximum margin partial label formulation aims to solve the following optimization problem (OP):

OP 1: Existing Maximum Margin Formulation

$$\begin{aligned}\min_{\boldsymbol{\Theta},\boldsymbol{\xi}} \quad & \frac{1}{2}\sum_{p=1}^{q}||\mathbf{w}_p||^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.}: & \max_{y_j\in S_i}(\mathbf{w}_{y_j}^\top\cdot\mathbf{x}_i + b_{y_j}) - \max_{y_k\notin S_i}(\mathbf{w}_{y_k}^\top\cdot\mathbf{x}_i + b_{y_k}) \geq 1-\xi_i \\ & \xi_i \geq 0 \qquad \forall i\in\{1,2,\ldots,m\}\end{aligned}$$

Here, $\boldsymbol{\xi} = \{\xi_1, \xi_2, \ldots, \xi_m\}$ represents the set of slack variables and C is the regularization parameter. As shown in **OP 1**, the existing formulation focuses on distinguishing the maximum output from candidate labels, i.e. $\max_{y_j\in S_i}(\mathbf{w}_{y_j}^\top\cdot\mathbf{x}_i + b_{y_j})$, with the maximum output from non-candidate labels, i.e. $\max_{y_k\notin S_i}(\mathbf{w}_{y_k}^\top\cdot\mathbf{x}_i + b_{y_k})$. One potential drawback of this formulation lies in the fact that the predictive difference between the ground-truth label and other candidate labels is not taken into account, which may lead to suboptimal performance for the resulting partial label learning approach.

In the next section, a new maximum margin formulation towards partial label learning is proposed, which aims to maximize the canonical multi-class margin between the ground-truth label and all other labels in the label space.

3 The M3PL approach

3.1 Proposed formulation

Based on the notation given in Sect. 1, the training set $\mathcal{D}$ is composed of m partial label examples $(\boldsymbol{x}_i, S_i)$ $(1\leq i\leq m)$ with $\mathbf{x}_i\in\mathcal{X}$ and $S_i\subseteq\mathcal{Y}$. In addition, let $\mathbf{y} = (y_1, y_2, \ldots, y_m)$ be the (unknown) ground-truth label assignments for the training examples. Following partial label learning assumption, the ground-truth label of each instance $\mathbf{x}_i$ should reside in its candidate label set S_i. Therefore, the feasible solution space of $\mathbf{y}$ corresponds to $\mathcal{S} = S_1\times S_2\times\cdots\times S_m$.

As in common practice, M3PL assumes a maximum margin learning system with q linear classifiers $\boldsymbol{\Theta} = \{(\mathbf{w}_p, b_p)\mid 1\leq p\leq q\}$, one for each class label. Once the ground-truth label assignments $\mathbf{y} = (y_1, y_2, \ldots, y_m)$ are fixed, M3PL proceeds to maximize the canonical multi-class margin over each instance $\mathbf{x}_i$, i.e.: $(\mathbf{w}_{y_i}^\top\cdot\mathbf{x}_i + b_{y_i}) - \max_{\tilde{y}_i\neq y_i}(\mathbf{w}_{\tilde{y}_i}^\top\cdot\mathbf{x}_i + b_{\tilde{y}_i})$. By introducing slack variables $\boldsymbol{\xi} = \{\xi_1, \xi_2, \ldots, \xi_m\}$ to accommodate margin relaxations, the maximum margin problem considered by M3PL can be formulated as follows:

OP 2: Proposed Maximum Margin Formulation

$$\begin{aligned}\min_{\mathbf{y},\boldsymbol{\Theta},\boldsymbol{\xi}} \quad & \frac{1}{2}\sum_{p=1}^{q}||\mathbf{w}_p||^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.}: & (\mathbf{w}_{y_i}^\top\cdot\mathbf{x}_i + b_{y_i}) - \max_{\tilde{y}_i\neq y_i}(\mathbf{w}_{\tilde{y}_i}^\top\cdot\mathbf{x}_i + b_{\tilde{y}_i}) \geq 1-\xi_i \\ & \xi_i\geq 0 \qquad \forall i\in\{1,2,\ldots,m\} \\ & \mathbf{y}\in\mathcal{S} \\ & \sum_{i=1}^{m}\mathbb{I}(y_i = p) = n_p \qquad \forall p\in\{1,2,\ldots,q\}\end{aligned}$$

As shown in **OP 2**, the first two constraints enforce the maximum margin criterion over each training example. In addition, the third constraint enforces that the ground-truth label assignment $\mathbf{y}$ should take values within the feasible solution space $\mathcal{S}$. The fourth constraint, i.e. $\sum_{i=1}^{m} \mathbb{I}(y_i = p) = n_p$, serves as an extra enforcement on $\mathbf{y}$ reflecting its compatibility with the *prior* class distribution.[2] Intuitively, n_p represents the prior number of examples which take the p-th class label in $\mathcal{Y}$ as their ground-truth label.

By sharing equal labeling confidence $\frac{1}{|S_i|}$ among each candidate label in S_i, the prior number can be roughly estimated as:

$$\hat{n}_p = \sum_{i=1}^{m} \mathbb{I}(p \in S_i) \cdot \frac{1}{|S_i|} \tag{1}$$

Obviously, $\sum_{p=1}^{q} \hat{n}_p = m$ holds. Furthermore, let $\lfloor \hat{n}_p \rfloor$ be the integer part of $\hat{n}_p$ and $r = m - \sum_{p=1}^{q} \lfloor \hat{n}_p \rfloor$ be the corresponding residual number w.r.t. the rounding operation. Then, the integer value n_p for the fourth constraint is set as:

$$n_p = \begin{cases} \lfloor \hat{n}_p \rfloor + 1 & if\ p\ is\ among\ the\ r\ class\ labels\ with\ least\ \hat{n}_p\ values \\ \lfloor \hat{n}_p \rfloor & otherwise \end{cases} \tag{2}$$

Accordingly, $\sum_{p=1}^{q} n_p = m$ still holds.

Note that **OP 2** corresponds to an optimization problem involving mixed-type variables (i.e. integer variables $\mathbf{y}$ and real-valued variables $\boldsymbol{\Theta}$), whose values are difficult to be optimized simultaneously. In the following subsection, an alternating optimization procedure is employed to update $\mathbf{y}$ and $\boldsymbol{\Theta}$ in an iterative manner.

3.2 Alternating optimization

3.2.1 Fix $\mathbf{y}$*, update* $\boldsymbol{\Theta}$

By fixing the ground-truth label assignments $\mathbf{y} = (y_1, y_2, \ldots, y_m)$, **OP 2** turns out to be the following optimization problem:

OP 3: Classification Model Optimization

$$\begin{aligned} \min_{\boldsymbol{\Theta}, \boldsymbol{\xi}} \quad & \frac{1}{2} \sum_{p=1}^{q} ||\mathbf{w}_p||^2 + C \sum_{i=1}^{m} \xi_i \\ \text{s.t. } : & (\mathbf{w}_{y_i}^{\top} \cdot \mathbf{x}_i + b_{y_i}) - \max_{\tilde{y}_i \neq y_i} (\mathbf{w}_{\tilde{y}_i}^{\top} \cdot \mathbf{x}_i + b_{\tilde{y}_i}) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad \forall i \in \{1, 2, \ldots, m\} \end{aligned}$$

As shown in **OP 3**, the resulting optimization problem coincides with the well-studied single-label multi-class maximum margin formulation (Crammer and Singer 2001; Hsu and Lin 2002). Therefore, **OP 3** can be readily solved by utilizing any off-the-shelf implementation on multi-class SVM (Fan et al. 2008).

[2] $\mathbb{I}(a)$ is an indicator function which returns 1 if predicate a is true, and 0 otherwise.

3.2.2 Fix $\boldsymbol{\Theta}$*, update* $\mathbf{y}$

By fixing the classification model $\boldsymbol{\Theta} = \{(\mathbf{w}_p, b_p) \mid 1 \leq p \leq q\}$, **OP 2** turns out to be the following optimization problem:

OP 4: Ground-truth Label Assignment Optimization (Version 1)

$$\begin{aligned} \min_{\mathbf{y},\boldsymbol{\xi}} \quad & \sum_{i=1}^{m} \xi_i \\ \text{s.t. :} \quad & \xi_i \geq 1 - \eta_i^{y_i} \\ & \xi_i \geq 0 \qquad \forall i \in \{1, 2, \ldots, m\} \\ & \mathbf{y} \in \mathcal{S} \\ & \sum_{i=1}^{m} \mathbb{I}(y_i = p) = n_p \qquad \forall p \in \{1, 2, \ldots, q\} \end{aligned}$$

Here, $\eta_i^{y_i}$ represents the multi-class margin on $\mathbf{x}_i$ by taking y_i as its ground-truth label, i.e.:

$$\eta_i^{y_i} = (\mathbf{w}_{y_i}^{\top} \cdot \mathbf{x}_i + b_{y_i}) - \max_{\tilde{y}_i \neq y_i} (\mathbf{w}_{\tilde{y}_i}^{\top} \cdot \mathbf{x}_i + b_{\tilde{y}_i}) \tag{3}$$

By setting $\xi_i = \max(0, 1 - \eta_i^{y_i})$ according to the first two constraints, **OP 4** can be re-written in the following form:

OP 5: Ground-truth Label Assignment Optimization (Version 2)

$$\begin{aligned} \min_{\mathbf{y}} \quad & \sum_{i=1}^{m} \max(0, 1 - \eta_i^{y_i}) \\ \text{s.t. :} \quad & \mathbf{y} \in \mathcal{S} \\ & \sum_{i=1}^{m} \mathbb{I}(y_i = p) = n_p \qquad \forall p \in \{1, 2, \ldots, q\} \end{aligned}$$

Let $\mathbf{Z} = [z_{pi}]_{q \times m}$ be the *binary-valued* labeling matrix for training examples, where $z_{pi} = 1$ indicates that the p-th class label in $\mathcal{Y}$ is the ground-truth label for $\mathbf{x}_i$. Accordingly, set the coefficient matrix $\mathbf{C} = [c_{pi}]_{q \times m}$ as follows:

$$\forall 1 \leq p \leq q,\ 1 \leq i \leq m: \quad c_{pi} = \begin{cases} \max(0, 1 - \eta_i^p) & if \quad p \in S_i \\ M & otherwise \end{cases} \tag{4}$$

Here, M is a user-specified *large constant* so that the learning algorithm can refrain from assigning ground-truth labels outside the candidate label set.[3] Based on the above definitions, **OP 5** can be re-written in the following form:

OP 6: Ground-truth Label Assignment Optimization (Version 3)

$$\min_{\mathbf{Z}} \quad \sum_{p=1}^{q} \sum_{i=1}^{m} c_{pi} \cdot z_{pi}$$

[3] In this paper, M is set to be 10^5.

$$\text{s.t.}: \sum_{p=1}^{q} z_{pi} = 1 \qquad \forall i \in \{1, 2, \ldots, m\}$$
$$\sum_{i=1}^{m} z_{pi} = n_p \qquad \forall p \in \{1, 2, \ldots, q\}$$
$$z_{pi} \in \{0, 1\}$$

Here, the first constraint $\sum_{p=1}^{q} z_{pi} = 1$ ensures that each training example has a unique ground-truth label. In addition, the second constraint $\sum_{i=1}^{m} z_{pi} = n_p$ enforces the constraint w.r.t. the prior class distribution.

Note that **OP 6** corresponds to a binary integer programming (BIP) problem, which is generally NP-hard to solve. Nonetheless, it is interesting that **OP 6** actually falls into a special case of BIP where the constraint matrix is *totally unimodular* (TU) and the right-hand sides of the constraints are integers. To show this, let $\mathbf{z} = [z_{11}, \ldots, z_{q1}, \ldots, z_{1m}, \ldots, z_{qm}]^\top$ denote the vector formed by sequentially concatenating each column of $\mathbf{Z}$. According to **OP 6**, the set of constraints $\sum_{p=1}^{q} z_{pi} = 1$ ($\forall i \in \{1, 2, \ldots, m\}$) and $\sum_{i=1}^{m} z_{pi} = n_p$ ($\forall p \in \{1, 2, \ldots, q\}$) can be expressed in the following form:

$$\mathbf{A}\mathbf{z} = \mathbf{s} \tag{5}$$

Here, $\mathbf{A} \in \mathbb{R}^{(m+q)\times mq}$ is the constraint matrix which corresponds to the concatenation of two matrices $\mathbf{B} \in \mathbb{R}^{m\times mq}$ and $\mathbf{C} \in \mathbb{R}^{q\times mq}$, i.e. $\mathbf{A} = [\mathbf{B}^\top, \mathbf{C}^\top]^\top$. Specifically, entries of the matrices $\mathbf{B} = [b_{ij}]_{m\times mq}$, $\mathbf{C} = [c_{ij}]_{q\times mq}$ and the right-hand side vector $\mathbf{s} = [s_1, s_2, \ldots, s_{m+q}]^\top$ are set as:

$$\forall 1 \le i \le m,\ 1 \le j \le mq:\ b_{ij} = \begin{cases} 1, & \text{if } j \in [(i-1)\cdot m + 1, i \cdot m] \\ 0, & \text{otherwise} \end{cases} \tag{6}$$
$$\forall 1 \le i \le q,\ 1 \le j \le mq:\ c_{ij} = \begin{cases} 1, & \text{if } j \mod m = i - 1 \\ 0, & \text{otherwise} \end{cases}$$
$$\forall 1 \le i \le m + q:\ s_i = \begin{cases} 1, & \text{if } i \in [1, m] \\ n_{i-m}, & \text{if } i \in [m+1, m+q] \end{cases}$$

To show that the constraint matrix $\mathbf{A}$ is TU, it suffices to show that $\mathbf{A}$ satisfies the following four conditions (Heller and Tompkins 1956):

1. Each column of $\mathbf{A}$ contains at most two non-zero entries;
2. Every entry in $\mathbf{A}$ takes value of 0, 1 or -1;
3. If two non-zero entries in a column of $\mathbf{A}$ have the same sign, then the row of one entry is in $\mathbf{B}$ and the row of another entry is in $\mathbf{C}$;
4. If two non-zero entries in a column of $\mathbf{A}$ have opposite sign, then the rows of both entries are either in $\mathbf{B}$ or in $\mathbf{C}$.

As defined in Equation (6), for both matrices $\mathbf{B}$ and $\mathbf{C}$, every entry takes a value of 0 or 1 and each column contains a unique non-zero entry. Therefore, it is not difficult to show that all four TU conditions hold for the constraint matrix $\mathbf{A}$. Furthermore, the right-hand-side vector of Eq. (5) contain integer entries according to the definition of Eq. (6).

Based on the properties of $\mathbf{A}$ being TU and $\mathbf{s}$ being integer-valued, the original BIP problem of **OP 6** can be equivalently solved in its linear programming (LP) relaxation form by replacing the integer constraint $z_{pi} \in \{0, 1\}$ with the weaker interval constraint $z_{pi} \in [0, 1]$ (Papadimitriou and Steiglitz 1998):

OP 7: Ground-truth Label Assignment Optimization (Version 4)

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \sum_{p=1}^{q}\sum_{i=1}^{m} c_{pi} \cdot z_{pi} \\ \text{s.t.}: & \sum_{p=1}^{q} z_{pi} = 1 \qquad \forall i \in \{1, 2, \ldots, m\} \\ & \sum_{i=1}^{m} z_{pi} = n_p \qquad \forall p \in \{1, 2, \ldots, q\} \\ & 0 \leq z_{pi} \leq 1 \end{aligned}$$

Thereafter, a solution to the relaxation problem **OP 7** can be efficiently found by employing standard LP solvers such as the simplex algorithm or the interior point algorithm (Boyd and Vandenberghe 2004).

3.3 Iterative implementation

To initialize the alternating optimization procedure, M3PL sets the initial coefficient matrix $\mathbf{C}$ by consulting the candidate label sets:

$$\forall 1 \leq p \leq q,\ 1 \leq i \leq m: \quad c_{pi} = \begin{cases} \frac{1}{|S_i|} & if \quad p \in S_i \\ M & otherwise \end{cases} \tag{7}$$

By solving **OP 7** based on initialized coefficients, the ground-truth label assignment $\mathbf{y} = (y_1, y_2, \ldots, y_m)$ would be $y_i = \arg\max_{1 \leq p \leq q} z_{pi}$. Then, the classification model $\boldsymbol{\Theta}$ is updated by solving **OP 3** and the alternating optimization procedure iterates. After every round of alternating update, the iteration procedure will terminate once the objective function value in **OP 2** decreases l.

Other than fixing the value for the regularization parameter C, M3PL chooses to gradually increase the value of C within an outer annealing loop. A similar strategy has been used in solving other weakly-supervised learning problems (Joachims 1999; Chapelle et al. 2008) to reduce the risk of getting stuck with a local minimum solution.

Algorithm 1 summarizes the complete procedure of M3PL.[4] Given the partial label training set, M3PL firstly initializes the regularization parameter C and the ground-truth label assignment (Steps 1-3). After that, the classification model and ground-truth label assignment are alternatively optimized until convergence (Steps 7-13). An outer loop is used to gradually increase the value of C by a factor of $1 + \Delta$ (Step 5). Finally, the unseen instance is classified based on the learned classification model (Step 15).[5] In Step 9, by introducing the kernel trick to solve the multi-class maximum margin problem **OP 3** (Crammer and Singer 2001), the resulting kernelized version of M3PL is denoted as M3PL-kernel.

Within each outer loop, it is not difficult to show that the objective function in **OP 2** converges as the inner alternating optimization procedure proceeds (Steps 7–13). Let $f(\boldsymbol{\Theta}^{(t)}, \mathbf{y}^{(t)})$ denote the value of the objective function at the t-th iteration, it suffices

[4] Code package of the M3PL algorithm is publicly-available at http://cse.seu.edu.cn/PersonalPage/zhangml/files/M3PL.zip.

[5] In this paper, **OP 3** (Step 9) and **OP 7** (Step 11) are implemented with the LibLinear toolbox (Fan et al. 2008) and the CVX toolbox (Grant and Boyd 2014) respectively. Furthermore, Δ is set to be 0.5 following (Chapelle et al. 2008) and δ is set to be 10^{-4}.

Algorithm 1 The M3PL Approach

Inputs:

$\mathcal{D}$: the partial label training set $\{(\mathbf{x}_i, S_i) \mid 1 \leq i \leq m\}$ ($\mathbf{x}_i \in \mathcal{X}, S_i \subseteq \mathcal{Y}$)

$C_{\max}$: the maximum value for regularization parameter

$\mathbf{x}^*$: the unseen instance

Outputs:

y^*: the predicted class label for $\mathbf{x}^*$

Process:

1: Initialize the regularization parameter: $C = 10^{-5} \cdot C_{\max}$;
2: Initialize the coefficient matrix $\mathbf{C}$ according to Eq.(7);
3: Solve the LP problem **OP 7**, and then initialize the ground-truth label assignment $\mathbf{y}$ with $y_i = \arg\max_{1 \leq p \leq q} z_{pi}$ $(1 \leq i \leq m)$;
4: **while** $C < C_{\max}$ **do**
5: $C = \min\{(1 + \Delta)C,\ C_{\max}\}$;
6: Initialize the objective function value in **OP 2**: $Obj = -\infty$;
7: **repeat**
8: $Obj_{\text{old}} = Obj$;
9: Solve the multi-class maximum margin problem **OP 3**, and then update the classification model Θ;
10: Set the coefficient matrix $\mathbf{C}$ according to Eq.(4);
11: Solve the LP problem **OP 7**, and then update the ground-truth label assignment $\mathbf{y}$ with $y_i = \arg\max_{1 \leq p \leq q} z_{pi}$ $(1 \leq i \leq m)$;
12: Calculate the new objective function value in **OP 2**: $Obj = \frac{1}{2}\sum_{p=1}^{q} ||\mathbf{w}_p||^2 + C\sum_{i=1}^{m} \max(0, 1 - \eta_i^{y_i})$;
13: **until** $Obj_{\text{old}} - Obj < \delta$
14: **end while**
15: **return** $y^* = \arg\max_{p \in \mathcal{Y}} \mathbf{w}_p^\top \cdot \mathbf{x}^* + b_p$;

to prove the convergence of the objective function if $f(\cdot, \cdot)$ is *bounded below* and *non-increasing* as t increases. On the one hand, as shown in **OP 2**, $f(\mathbf{\Theta}, \mathbf{y}) = \frac{1}{2}\sum_{p=1}^{q} ||\mathbf{w}_p||^2 + C\sum_{i=1}^{m} \max(0, 1 - \eta_i^{y_i})$ with $\eta_i^{y_i} = (\mathbf{w}_{y_i}^\top \cdot \mathbf{x}_i + b_{y_i}) - \max_{\tilde{y}_i \neq y_i}(\mathbf{w}_{\tilde{y}_i}^\top \cdot \mathbf{x}_i + b_{\tilde{y}_i})$. Therefore, the property of being bounded below naturally holds with $f(\mathbf{\Theta}, \mathbf{y}) \geq 0$. On the other hand, solving the first alternating optimization problem (**OP 3**; Step 9) leads to $f(\mathbf{\Theta}^{(t)}, \mathbf{y}^{(t)}) \geq f(\mathbf{\Theta}^{(t+1)}, \mathbf{y}^{(t)})$, and solving the second alternating optimization problem (**OP 7**; Steps 10-11) leads to $f(\mathbf{\Theta}^{(t+1)}, \mathbf{y}^{(t)}) \geq f(\mathbf{\Theta}^{(t+1)}, \mathbf{y}^{(t+1)})$. Therefore, the property of being non-increasing naturally holds with $f(\mathbf{\Theta}^{(t)}, \mathbf{y}^{(t)}) \geq f(\mathbf{\Theta}^{(t+1)}, \mathbf{y}^{(t)}) \geq f(\mathbf{\Theta}^{(t+1)}, \mathbf{y}^{(t+1)})$.

It is obvious that the proposed M3PL approach coincides with the existing maximum margin formulation (Nguyen and Caruana 2008) if the size of the candidate label set shrinks to 1. Correspondingly, iterative optimization has also been employed by a number of partial label learning approaches for disambiguating the candidate label set (Jin and Ghahramani 2003; Nguyen and Caruana 2008; Liu and Dietterich 2012; Chen et al. 2014). As shown in Eq. (4), a parameter M is utilized such that **OP 6** (or equivalently **OP 5**) can be solved by restricting the assigned label only in the candidate label set. This restriction ensures the validity of the ground-truth label assignments $\mathbf{y}$ which will be fixed as constants for solving **OP 3**.

Table 1 Characteristics of the experimental datasets

Controlled UCI datasets				Configurations
Dataset	#Examples	#Features	#Class labels	
Glass	214	10	5	(I) $r = 1$, $p \in \{0.1, 0.2, \ldots 0.7\}$
Ecoli	336	7	8	(II) $r = 2$, $p \in \{0.1, 0.2, \ldots 0.7\}$
Dermatology	364	23	6	(III) $r = 3$, $p \in \{0.1, 0.2, \ldots 0.7\}$
Vehicle	846	18	4	(IV) $p = 1, r = 1, \epsilon \in \{0.1, 0.2, \ldots 0.7\}$
Segment	2310	18	7	
Satimage	6435	36	7	

Real-world datasets					
Dataset	# Examples	# Features	# Class labels	Avg. #CLs	Domain
FG-NET	1002	262	78	7.48	*Facial age estimation*
Lost	1122	108	16	2.23	*Automatic face naming*
MSRCv2	1758	48	23	3.16	*Object classification*
BirdSong	4998	38	13	2.18	*Bird song classification*
Soccer Player	17472	279	171	2.09	*Automatic face naming*
Yahoo! News	22991	163	219	1.91	*Automatic face naming*

4 Experiment

4.1 Experimental setup

In this section, two series of experiments are conducted to evaluate the performance of M3PL, with one series on controlled UCI datasets (Bache and Lichman 2013) and the other on real-world partial label datasets. Table 1 summarizes characteristics of the employed datasets.

Following the widely-used controlling protocol over multi-class UCI datasets (Cour et al. 2011; Chen et al. 2014; Liu and Dietterich 2012; Zhang 2014), an artificial partial label dataset can be generated under different configurations of three controlling parameters p, r and ϵ. Here, p controls the proportion of examples which are partially labeled (i.e. $|S_i| > 1$), r controls the number of false positive labels in the candidate label set (i.e. $|S_i| = r + 1$), and ϵ controls the co-occurring probability between one coupling candidate label and the ground-truth label. As shown in Table 1, a total of 28 (4×7) configurations are considered for each of the six UCI datasets.

The real-world partial label datasets are collected from several task domains, such as *facial age estimation* including `FG-NET` (Panis and Lanitis 2015), *automatic face naming* including `Lost` (Cour et al. 2011), `Soccer Player` (Zeng et al. 2013), `Yahoo! News` (Guillaumin et al. 2010), *bird song classification* including `BirdSong` (Briggs et al. 2012), and *object classification* including `MSRCv2` (Liu and Dietterich 2012).[6] For the task of facial age estimation, human faces with landmarks are represented as instances while ages annotated by ten crowdsourced labelers together with the ground-truth age are regarded as candidate labels. For the task of automatic face naming, faces cropped from an image or

[6] These datasets are publicly-available at: http://cse.seu.edu.cn/PersonalPage/zhangml/Resources.htm#partial_data.

video frame are represented as instances while names extracted from the associated captions or subtitles are regarded as candidate labels. For the task of bird song classification, singing syllables of the birds are represented as instances while bird species jointly singing during a 10-second period are regarded as candidate labels. For the task of object classification, image segmentations are represented as instances while objects appearing within the same image are regarded as candidate labels. As shown in Table 1, the average number of candidate labels (avg. #CLs) for each real-world partial label dataset is also recorded.

Four well-established partial label learning approaches are employed for comparative studies, each implemented with parameter setup as suggested in the respective literature:

- An existing maximum margin partial label learning approach named PL-SVM (Nguyen and Caruana 2008) [suggested setup: regularization parameter pool with $\{10^{-3}, \ldots, 10^{3}\}$] as well as its kernelized version named PL-SVM-kernel [suggested setup: polynomial kernel and degree pool with $\{1, \ldots, 5\}$].
- The k-nearest neighbor partial label learning approach named PL-KNN (Hüllermeier and Beringer 2006) [suggested setup: k=10].
- The convex optimization partial label learning approach named CLPL (Cour et al. 2011) [suggested setup: SVM with squared hinge loss].
- The maximum likelihood partial label learning algorithm named LSB-CMM (Liu and Dietterich 2012) [suggested setup: q mixture components].

For PL-SVM and LSB-CMM, both algorithms conduct disambiguation by treating the ground-truth label as a latent variable to be iteratively refined. Specifically, PL-SVM (and its kernelized version) works by maximizing the margin between the largest output from candidate labels and that from non-candidate labels, while LSB-CMM works by maximizing the likelihood function over partial label training examples with EM-based optimization over a conditional multinomial model. For PL-KNN and CLPL, both algorithms conduct disambiguation by treating each candidate label equally to be further aggregated. Specifically, PL-KNN works by voting among the candidate labels of each neighboring example whose voting weight is inversely proportional to its distance from the test instance, while CLPL works by transforming the original partial label learning problem into a binary learning problem which is then solved by conventional SVM classification.

For M3PL, the parameter $C_{\max}$ is chosen among $\{10^{-2}, \ldots, 10^{2}\}$ via cross-validation. In addition, a Gaussian kernel with width parameter $\frac{1}{d}$ is used to instantiate the M3PL-kernel. In this paper, ten-fold cross-validation is performed on each artificial as well as real-world dataset where the mean predictive accuracies as well as standard deviations are recorded for all comparing approaches.

4.2 Experimental result

4.2.1 Controlled UCI datasets

Figures 1, 2 and 3 illustrate the classification accuracy of each comparing algorithm as p increases from 0.1 to 0.7 with step-size 0.1 ($r = 1, 2, 3$). For any partial label example, its candidate label set contains the ground-truth label along with r additional labels randomly chosen from $\mathcal{Y}$. Figure 4 illustrates the classification accuracy of each comparing algorithm as ϵ increases from 0.1 to 0.7 with step-size 0.1 ($p = 1, r = 1$). For any label $y \in \mathcal{Y}$, one extra label $y' \in \mathcal{Y}$ is designated as the coupling label which co-occurs with y in the candidate label set with probability ϵ. Otherwise, any other class label would be randomly chosen to co-occur with y.

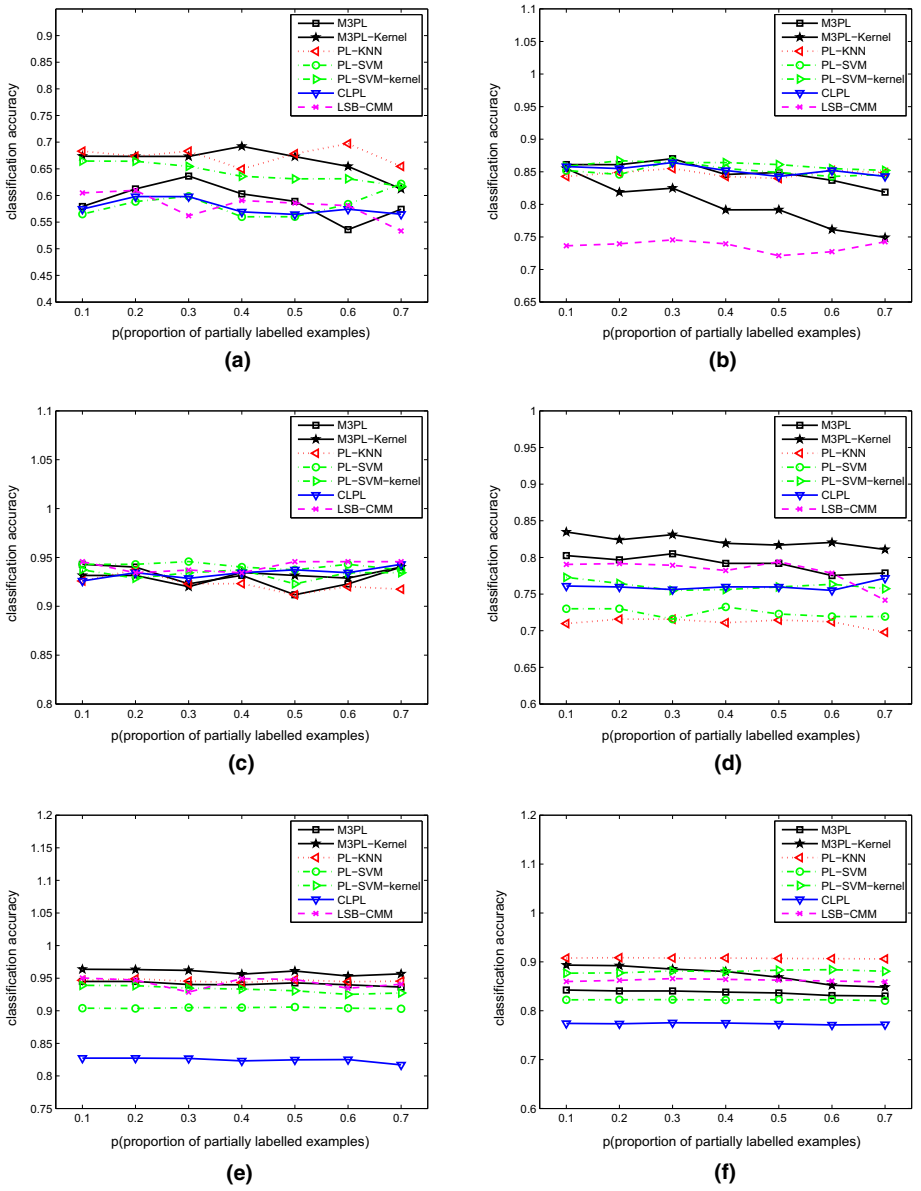

Fig. 1 Classification accuracy of each comparing algorithm changes as p (proportion of partially labeled example) increases from 0.1 to 0.7 ($r = 1$). **a** Glass, **b** ecoli, **c** dermatology, **d** vehicle, **e** segment, **f** satimage

As shown in Figs. 1, 2, 3 and 4, in most cases, M3PL and its kernelized version achieve competitive performance against the comparing algorithms. Based on pairwise t test at 0.05 significance level, Tables 2 and 3 summarize the win/tie/loss counts of M3PL and M3PL-kernel against the comparing algorithms respectively. Out of the 168 statistical comparisons (28 configurations $\times$ 6 datasets), the following observations can be made:

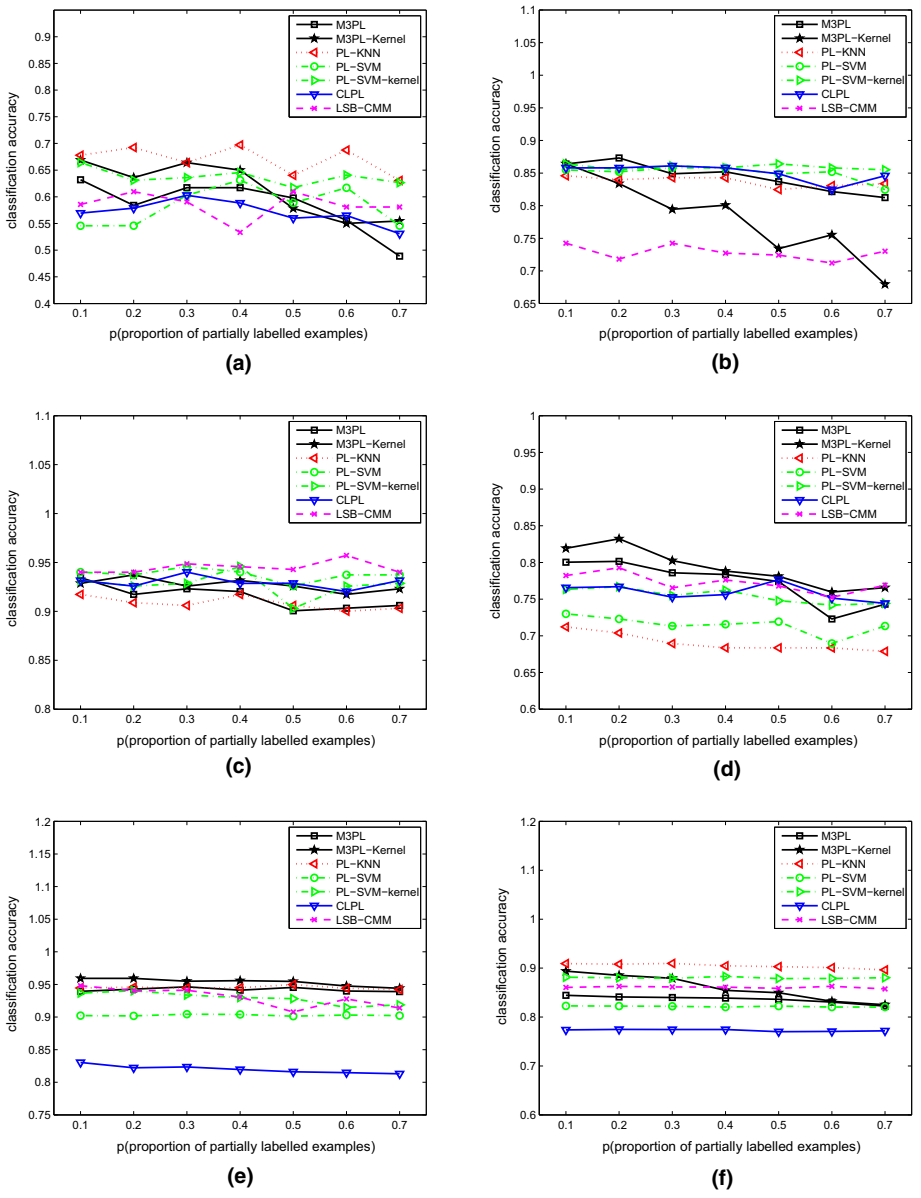

Fig. 2 Classification accuracy of each comparing algorithm changes as p (proportion of partially labeled example) increases from 0.1 to 0.7 ($r = 2$). **a** Glass, **b** ecoli, **c** dermatology, **d** vehicle, **e** segment, **f** satimage

– Compared to the existing maximum margin counterpart PL-SVM (Nguyen and Caruana 2008), M3PL achieves superior or at least comparable performance in 77.9% cases. Although M3PL does not perform favorably against PL-SVM-kernel, its kernelized version M3PL-kernel achieves superior or at least comparable performance against PL-SVM and PL-SVM-kernel in 74.4 and 85.7% cases respectively. These results indicate the

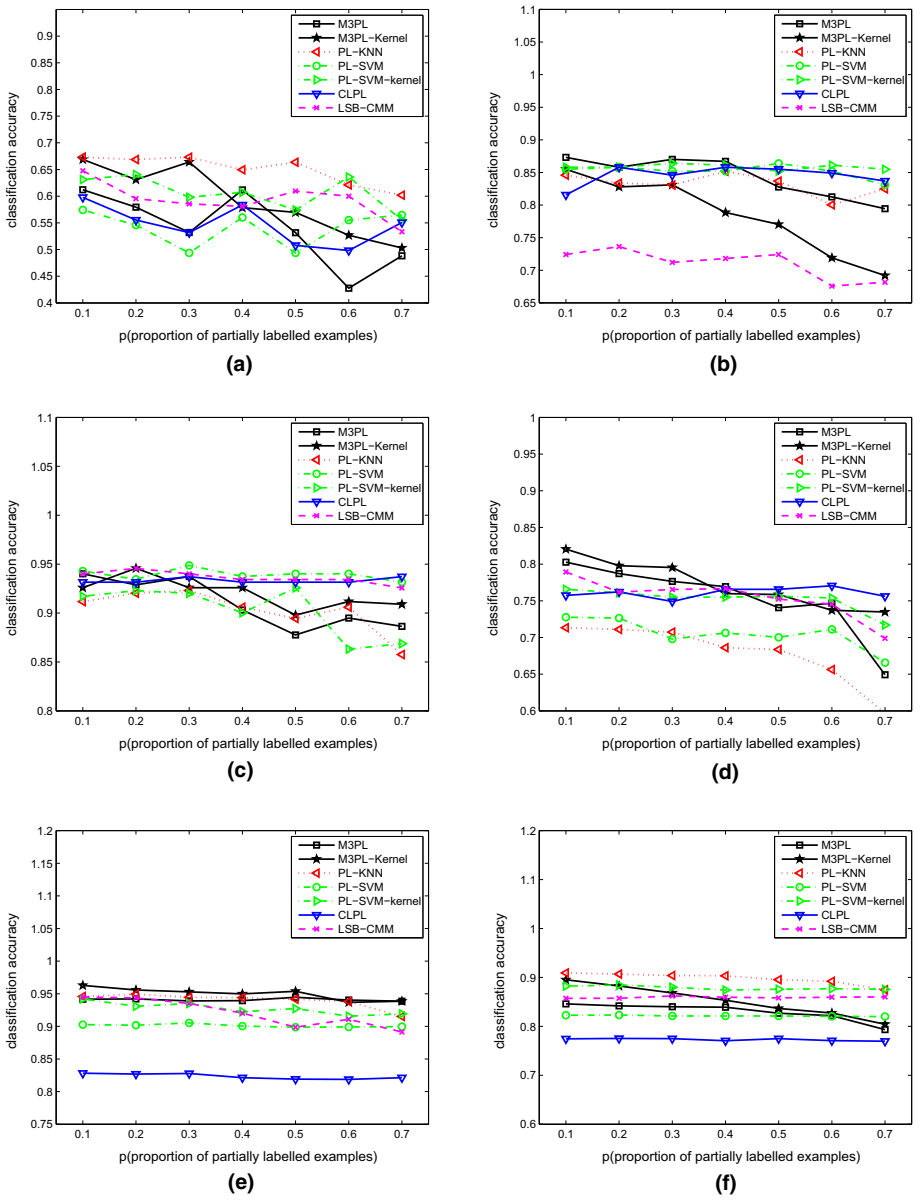

Fig. 3 Classification accuracy of each comparing algorithm changes as p (proportion of partially labeled example) increases from 0.1 to 0.7 ($r = 3$). **a** Glass, **b** ecoli, **c** dermatology, **d** vehicle, **e** segment, **f** satimage

advantage of the proposed formulation against existing maximum margin partial label formulations;

– Compared to PL-KNN (Hüllermeier and Beringer 2006), CLPL (Cour et al. 2011) and LSB-CMM (Liu and Dietterich 2012), M3PL achieves superior or at least comparable performance in 79.8, 85.1 and 61.3% cases respectively, and M3PL-kernel achieves superior or at least comparable performance in 65.4, 81.5 and 82.1% cases respectively.

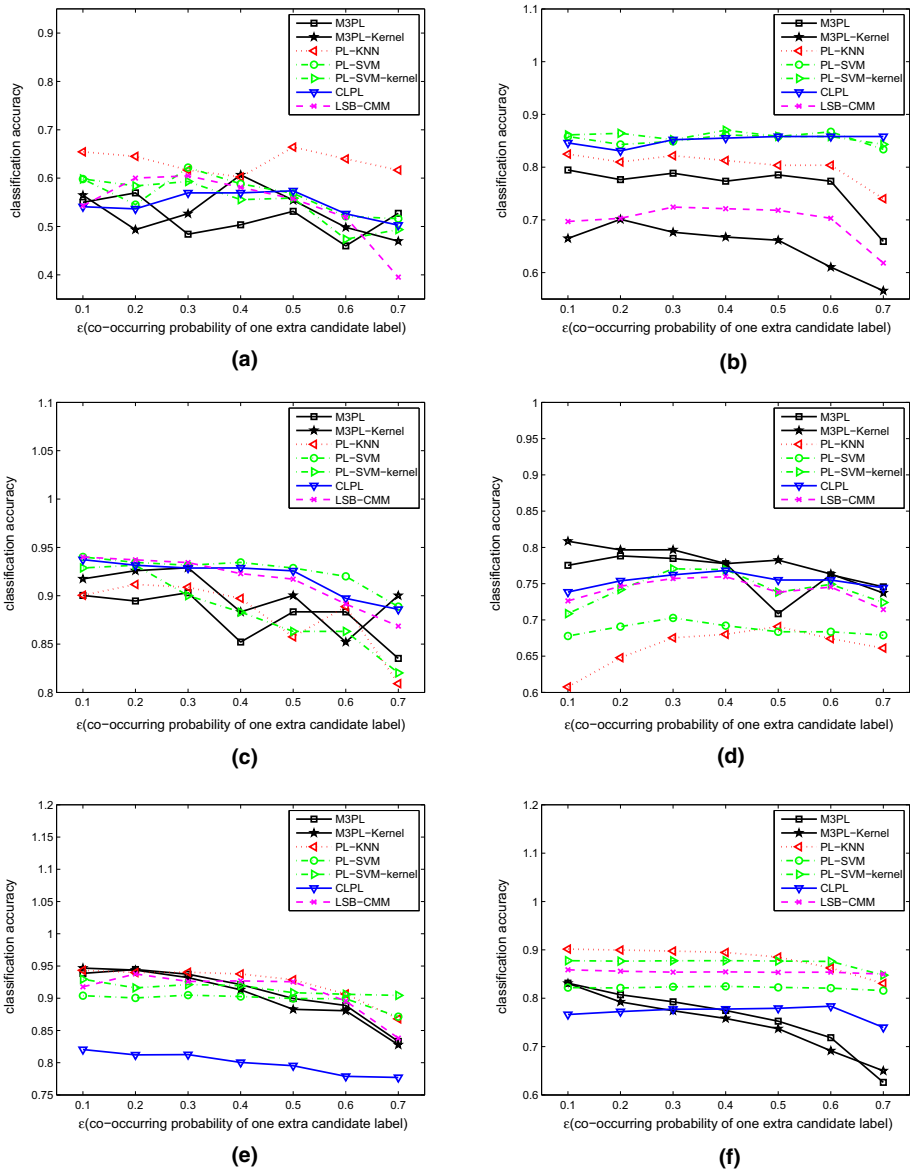

Fig. 4 Classification accuracy of each comparing algorithm changes as ϵ (co-occurring probability of the coupling label) increases from 0.1 to 0.7 ($p = 1, r = 1$). **a** Glass, **b** ecoli, **c** dermatology, **d** vehicle, **e** segment, **f** satimage

These results validate the ability of M3PL in achieving state-of-the-art generalization performance for the partial label learning problem.

It is worth noting that on some controlled UCI datasets (e.g.: Fig. 1, *segment* and *satimage*), the performance of CLPL is much inferior to the comparing algorithms. One potential reason might lie in the procedure employed by CLPL to transform the partial label learning problem

Table 2 Win/tie/loss counts (pairwise t test at 0.05 significance level) on the classification performance of M3PL against each comparing algorithm

	M3PL against				
	PL-SVM	PL-SVM-kernel	PL-KNN	CLPL	LSB-CMM
r=1, varying p	14/25/3	13/15/14	7/26/9	14/27/1	6/24/12
r=2, varying p	16/22/4	16/16/10	7/34/1	14/27/1	7/19/16
r=3, varying p	17/18/7	14/18/10	7/31/4	18/19/5	7/20/15
r=1, p=1, varying ϵ	5/14/23	4/20/18	3/19/20	6/18/18	4/16/22
In total	52/79/37	47/69/52	24/110/34	52/91/25	24/79/65
	M3PL against				
	MSVM	MSVM-kernel			
$r = 1$, varying p	0/24/18	0/13/29			
$r = 2$, varying p	0/20/22	0/9/33			
$r = 3$, varying p	0/17/25	0/9/33			
$r = 1$, p=1, varying ϵ	0/6/36	0/0/42			
In total	0/67/101	0/31/137			

In addition, multi-class SVM (MSVM) trained with ground-truth label and its kernelized version (MSVM-kernel) are also utilized as the upper-bound baselines for reference purposes

Table 3 Win/tie/loss counts (pairwise t test at 0.05 significance level) on the classification performance of M3PL-kernel against each comparing algorithm

	M3PL-kernel against				
	PL-SVM	PL-SVM-kernel	PL-KNN	CLPL	LSB-CMM
r=1, varying p	25/11/6	23/17/2	11/21/10	26/10/6	26/13/3
r=2, varying p	21/12/9	14/22/6	8/23/11	19/17/6	19/17/6
r=3, varying p	22/10/10	13/24/5	10/19/13	21/15/6	14/15/13
r=1, p=1, varying ϵ	10/14/18	5/26/11	7/11/24	13/16/13	10/24/8
In total	78/47/43	55/89/24	36/74/58	79/58/31	69/69/30
	M3PL-kernel against				
	MSVM	MSVM-kernel			
r=1, varying p	11/24/7	0/20/22			
r=2, varying p	6/21/15	0/8/34			
r=3, varying p	4/17/21	0/6/36			
r=1, p=1, varying ϵ	0/11/31	0/1/41			
In total	21/73/74	0/35/133			

In addition, multi-class SVM (MSVM) trained with ground-truth label and its kernelized version (MSVM-kernel) are also utilized as the upper-bound baselines for reference purposes

into the binary learning problem. Specifically, each partial label training example $(\mathbf{x}_i, S_i) \in \mathcal{D}$ is transformed into *one positive example* by aggregating all candidate labels, and $q - |S_i|$ *negative examples* each for one non-candidate label. For the resulting binary training set, the ratio between the number of negative examples and positive examples would be $q - q'$, where $q' = \frac{\sum_{i=1}^{m} |S_i|}{m}$ corresponds to the average number of candidate labels in $\mathcal{D}$. The corresponding binary learning problem would be highly *class-imbalanced* when q is much larger than q', which can lead to performance deterioration for the binary learning algorithm.

In addition, multi-class SVM (MSVM) trained with ground-truth labels and its kernelized version (MSVM-kernel) are also employed for comparative studies, which serve as the upper-bound baseline for partial label learning algorithms.[7] As shown in Table 2, the performance of M3PL is comparable to MSVM and MSVM-kernel in 39.8 and 22.6% cases, and inferior to them in the rest of the cases. As shown in Table 3, it is interesting that the performance of M3PL-kernel is even superior to MSVM in a few (12.5%) cases, and is comparable or inferior to MSVM and MSVM-kernel in the rest of the cases.

4.2.2 Real-world datasets

Table 4 reports the performance of each comparing algorithm on the real-world partial label datasets. Based on the results of ten-fold cross-validation, pairwise t tests at 0.05 significance level between M3PL and the comparing algorithms are recorded as well. Note that the average number of candidate labels (avg. #CLs) for the `FG-NET` dataset (i.e. 7.48 as shown in Table 1) is quite large, which makes the task of facial age estimation (based on training examples with partial labels) rather challenging. Furthermore, the state-of-the-art performance on this dataset (based on training examples with ground-truth labels) corresponds to more than 3 years of mean average error (MAE) between the predicted age and the ground-truth age (Panis and Lanitis 2015). In Table 4, one extra classification accuracy is reported on the `FG-NET` dataset where an unseen example is regarded to be correctly classified if the difference between the predicted age and the ground-truth age is less than 3 years (MAE3).

As shown in Table 4, it is impressive to observe that:

- M3PL significantly outperforms its maximum margin counterpart on all real-world datasets except `FG-NET` and its MAE3 variant, on which both algorithms achieve comparable performance. In terms of the kernelized version, M3PL-kernel significantly outperforms PL-SVM-kernel on the `MSRCv2`, `BirdSong`, `Soccer Player` and `Yahoo! News` datasets, and performs comparably on the rest of the datasets;
- Both M3PL and its kernelized version significantly outperforms PL-KNN on all datasets, except on `Soccer Player` where the performance of M3PL is inferior to PL-KNN;
- For M3PL, its performance is only inferior to CLPL and LSB-CMM on `FG-NET (MAE3)` and `Soccer Player` respectively. For M3PL-kernel, its performance is only inferior to CLPL on `FG-NET (MAE3)`. On the other cases, both M3PL and M3PL-kernel achieve superior or at least comparable performance against CLPL and LSB-CMM.

In addition, both M3PL and its kernelized version achieve comparable performance to MSVM and MSVM-kernel on `FG-NET` and its MAE3 variant, and are inferior to MSVM and MSVM-kernel in the rest of the cases. It is worth noting that for either M3PL or PL-SVM,

[7] For the sake of illustration clarity, detailed results of MSVM and MSVM-kernel haven't been depicted in Figs. 1, 2, 3 and 4.

Table 4 Classification accuracy (mean±std) of each comparing algorithm on the real-world partial label datasets

	FG-NET	FG-NET (MAE3)	Lost	MSRCv2
M3PL	0.061±0.020	0.344±0.032	0.767±0.043	0.521±0.030
M3PL-kernel	0.065±0.023	0.362±0.065	0.764±0.031	0.559±0.036
PL-SVM	0.065±0.024	0.386±0.050	0.729±0.040•▲	0.482±0.043•
PL-SVM-kernel	0.062±0.026	0.372±0.031	0.791±0.030	0.443±0.042•▲
PL-KNN	0.038±0.025•▲	0.299±0.025•▲	0.424±0.041•▲	0.448±0.037•▲
CLPL	0.063±0.027	0.456±0.038○ △	0.742±0.038	0.413±0.039•▲
LSB-CMM	0.052±0.012	0.380±0.035	0.707±0.055•▲	0.456±0.031•▲
MSVM	0.071±0.019	0.333±0.052	0.838±0.038○ △	0.623±0.028○ △
MSVM-kernel	0.083±0.016	0.361±0.041	0.865±0.036○ △	0.693±0.026○ △
	BirdSong	Soccer Player	Yahoo! News	
M3PL	0.709±0.010	0.446±0.013	0.655±0.009	
M3PL-kernel	0.716±0.022	0.534±0.018	0.654±0.010	
PL-SVM	0.663±0.032•▲	0.443±0.014•▲	0.636±0.010•▲	
PL-SVM-kernel	0.692±0.015 ▲	0.498±0.012○▲	0.630±0.010•▲	
PL-KNN	0.614±0.024•▲	0.497±0.014○▲	0.457±0.010•▲	
CLPL	0.632±0.017•▲	0.368±0.010•▲	0.462±0.009•▲	
LSB-CMM	0.717±0.024	0.525±0.015○	0.648±0.007 ▲	
MSVM	0.758±0.011○ △	0.597±0.009○ △	0.670±0.008○ △	
MSVM-kernel	0.770±0.012○ △	0.567±0.014○ △	0.688±0.012○ △	

In addition, •/○ indicates whether M3PL is statistically superior/inferior to the comparing algorithm on each dataset (pairwise t test at 0.05 significate level). Similarly, ▲/△ indicates whether M3PL-kernel is statistically superior/inferior to the comparing algorithm. The performance of MSVM and MSVM-kernel are also shown for reference purposes

although their performance is expected to be improved by employing the kernel trick, there are still some cases where the kernelized version achieves lower classification accuracy. These observations indicate the necessity of kernel function selection for learning from partial label examples.

In addition to inductive performance on unseen examples, it is also interesting to study the *transductive* performance of each comparing algorithm on classifying training examples (Cour et al. 2011). Here, for each training example $(\mathbf{x}_i, S_i)$, its ground-truth label is predicted by consulting the candidate label set, i.e.: $y_i = \arg\max_{y \in S_i} F(\mathbf{x}_i, y; \boldsymbol{\Theta})$. In other words, transductive performance of the partial label learning algorithm reflects its disambiguation ability in recovering ground-truth labeling information from the candidate label set. Similar to Table 4, Table 5 reports the transductive accuracy of each comparing algorithm together with the outcomes of pairwise t tests at 0.05 significance level. Furthermore, as the training procedure of M3PL terminates, the identified ground-truth label assignment $\mathbf{y}$ can be also used as the disambiguation predictions on the training examples. The resulting transductive performance is reported in Table 5 as well (denoted as M3PL† and M3PL-kernel†) for reference purposes.

As shown in Table 5, M3PL significantly outperforms all the other comparing algorithms on the `MSRCv2`, `BirdSong` and `Soccer Player` datasets, and achieves superior or at

Table 5 Transductive accuracy (mean±std) of each comparing algorithm on the real-world partial label datasets

	FG-NET	FG-NET (MAE3)	Lost	MSRCv2
M3PL	0.134±0.020	0.505±0.016	0.860±0.006	0.732±0.025
M3PL-kernel	0.144±0.006	0.580±0.020	0.803±0.015	0.699±0.031
M3PL†	0.134±0.020	0.510±0.015	0.860±0.006	0.732±0.025
M3PL-kernel†	0.144±0.006	0.580±0.020	0.794±0.011	0.681±0.031
PL-SVM	0.145±0.006	0.534±0.015○▲	0.887±0.012 △	0.653±0.024●
PL-SVM-kernel	0.173±0.012○ △	0.595±0.022○	0.901±0.020 △	0.666±0.030●
PL-KNN	0.109±0.005●▲	0.580±0.008○	0.615±0.036●▲	0.616±0.006●▲
CLPL	0.173±0.012○ △	0.589±0.008○	0.894±0.005 △	0.656±0.010●
LSB-CMM	0.162±0.006○ △	0.564±0.012○	0.721±0.010●▲	0.524±0.007●▲
	BirdSong	Soccer Player	Yahoo! News	
M3PL	0.855±0.030	0.761±0.010	0.870±0.002	
M3PL-kernel	0.816±0.010	0.705±0.010	0.841±0.002● △	
M3PL†	0.861±0.048	0.766±0.009	0.881±0.002	
M3PL-kernel†	0.816±0.010	0.701±0.010	0.840±0.002● △	
PL-SVM	0.825±0.012●	0.688±0.014●	0.871±0.002 △	
PL-SVM-kernel	0.826±0.021●	0.709±0.020●	0.845±0.002●	
PL-KNN	0.772±0.021●▲	0.492±0.015●▲	0.692±0.010●	
CLPL	0.822±0.004●	0.680±0.010●	0.834±0.002●	
LSB-CMM	0.716±0.014●▲	0.704±0.002● △	0.872±0.001 △	

In addition, ●/○ indicates whether M3PL is statistically superior/inferior to the comparing algorithm on each dataset (pairwise t test at 0.05 significate level). Similarly, ▲/△ indicates whether M3PL-kernel is statistically superior/inferior to the comparing algorithm

least comparable performance against other comparing algorithms on the `Lost` and `Yahoo! News` dataset. Although the transductive performance of M3PL is not satisfactory on FG-NET and its MAE3 variant, its inductive performance on them is competitive to the comparing algorithms in most cases. On the other hand, out of the 35 statistical tests (7 datasets × 5 comparing algorithms), the transductive performance of M3PL-kernel† is superior to the comparing algorithms in 9 cases, comparable in 17 cases, and inferior in 9 cases. As expected, M3PL and M3PL† (also for M3PL-kernel and M3PL-kernel†) show similar transductive performance over each real-world dataset.

5 Conclusion

This paper extends our earlier research on maximum margin partial label learning (Yu and Zhang 2015), where a new formulation of the maximum margin criterion is proposed to learning from partial label examples. Specifically, the canonical multi-class margin is directly optimized by the proposed M3PL approach with an alternating optimization procedure. Comprehensive comparative studies on artificial as well as real-world partial label datasets clearly validate the effectiveness of M3PL.

In the future, it is interesting to investigate other ways to solve the proposed maximum margin formulation **OP 2** other than utilizing alternating optimization. Furthermore, domain knowledge (e.g. the ordinal information among class labels) could be incorporated into partial label learning algorithms to improve their performance on specific tasks such as facial age estimation.

Acknowledgements The authors wish to thank the editors and anonymous reviewers for their insightful comments and suggestions. This work was supported by the National Science Foundation of China (61222309, 61573104), the MOE Program for New Century Excellent Talents in University (NCET-13-0130), and the Collaborative Innovation Center of Wireless Communications Technology.

References

Amores, J. (2013). Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*, *201*, 81–105.

Bache, K., & Lichman, M. (2013). *UCI machine learning repository*. School of Information and Computer Sciences, University of California, Irvine. http://archive.ics.uci.edu/ml.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge: Cambridge University Press.

Briggs, F., Fern, X. Z., & Raich, R. (2012). Rank-loss support instance machines for MIML instance annotation. In *Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining, Beijing, China* (pp. 534–542).

Chapelle, O., Schölkopf, B., Zien, A., et al. (2006). *Semi-supervised learning*. Cambridge, MA: MIT Press.

Chapelle, O., Sindhwani, V., & Keerthi, S. S. (2008). Optimization techniques for semi-supervised support vector machines. *Journal of Machine Learning Research*, *9*, 203–233.

Chen, Y. C., Patel, V. M., Chellappa, R., & Phillips, P. J. (2014). Ambiguously labeled learning using dictionaries. *IEEE Transactions on Information Forensics and Security*, *9*(12), 2076–2088.

Cid-Sueiro, J. (2012). Proper losses for learning from partial labels. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Weinberger (Eds.), *Advances in neural information processing systems* (Vol. 25, pp. 1574–1582). Cambridge, MA: MIT Press.

Côme, E., Oukhellou, L., Denœux, T., & Aknin, P. (2008). Mixture model estimation with soft labels. In D. Dubois, M. A. Lubiano, H. Prade, M. A. Gil, P. Grzegorzewski, & O. Hryniewicz (Eds.), *Advances in soft computing* (Vol. 48, pp. 165–174). Berlin: Springer.

Cour, T., Sapp, B., Jordan, C., & Taskar, B. (2009). Learning from ambiguously labeled images. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition, Miami, FL* (pp. 919–926).

Cour, T., Sapp, B., & Taskar, B. (2011). Learning from partial labels. *Journal of Machine Learning Research*, *12*, 1501–1536.

Crammer, K., & Singer, Y. (2001). On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, *2*, 265–292.

Dietterich, T. G., Lathrop, R. H., & Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, *89*(1), 31–71.

Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, *9*, 1871–1874.

Gibaja, E., & Ventura, S. (2015). A tutorial on multilabel learning. *ACM Computing Surveys*. doi:10.1145/2716262.

Grant, M., & Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx.

Guillaumin, M., Verbeek, J., & Schmid, C. (2010). Multiple instance metric learning from automatically labeled bags of faces. In K. Daniilidis, P. Maragos, & N. Paragios (Eds.), *Lecture notes in computer science* (Vol. 6311, pp. 634–647). Berlin: Springer.

Heller, I., & Tompkins, C. B. (1956). An extension of a theorem of dantzig's. In H. W. Kuhn & A. W. Tucker (Eds.), *Linear inequalities and related systems* (pp. 247–254). Princeton, NJ: Princeton University Press.

Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, *13*(2), 415–425.

Hüllermeier, E., & Beringer, J. (2006). Learning from ambiguously labeled examples. *Intelligent Data Analysis*, *10*(5), 419–439.

Jie, L., & Orabona, F. (2010). Learning from candidate labeling sets. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, & A. Culotta (Eds.), *Advances in neural information processing systems* (Vol. 23, pp. 1504–1512). Cambridge, MA: MIT Press.

Jin, R., & Ghahramani, Z. (2003). Learning with multiple labels. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in neural information processing systems* (Vol. 15, pp. 921–928). Cambridge, MA: MIT Press.

Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *Proceedings of the 16th international conference on machine learning, Bled, Slovenia* (pp. 200–209).

Liu, L., & Dietterich, T. G. (2012). A conditional multinomial mixture model for superset label learning. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Weinberger (Eds.), *Advances in neural information processing systems* (Vol. 25, pp. 557–565). Cambridge, MA: MIT Press.

Liu, L., & Dietterich, T. G. (2014). Learnability of the superset label learning problem. In *Proceedings of the 31st international conference on machine learning, Beijing, China* (pp. 1629–1637).

Nguyen, N., & Caruana, R. (2008). Classification with partial labels. In *Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining, Las Vegas, NV* (pp. 551–559).

Panis, G., & Lanitis, A. (2015). An overview of research activities in facial age estimation using the FG-NET aging database. In L. Agapito, M. M. Bronstein, & C. Rother (Eds.), *Lecture notes in computer science* (Vol. 8926, pp. 737–750). Berlin: Springer.

Papadimitriou, C. H., & Steiglitz, K. (1998). *Combinatorial optimization: Algorithms and complexity*. Mineola, NY: Dover Publications.

Pfahringer, B. (2012). Learning with weak supervision: Charting the territory. In *Keynote at the 1st international workshop on learning with weak supervision, Singapore*.

Yu, F., & Zhang, M. L. (2015). Maximum margin partial label learning. In *Proceedings of the 7th Asian conference on machine learning, Hong Kong, China* (pp. 96–111).

Zeng, Z., Xiao, S., Jia, K., Chan, T. H., Gao, S., Xu, D., & Ma, Y. (2013). Learning by associating ambiguously labeled images. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition, Portland, OR* (pp. 708–715).

Zhang, M. L. (2014). Disambiguation-free partial label learning. In *Proceedings of the 14th SIAM international conference on data mining, Philadelphia, PA* (pp. 37–45).

Zhang, M. L., & Yu, F. (2015). Solving the partial label learning problem: An instance-based approach. In *Proceedings of the 24th international joint conference on artificial intelligence, Buenos Aires, Argentina* (pp. 4048–4054).

Zhang, M. L., & Zhou, Z. H. (2014). A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, *26*(8), 1819–1937.

Zhang, M. L., Zhou, B. B., & Liu, X. Y. (2016). Partial label learning via feature-aware disambiguation. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, San Francisco, CA* (pp. 1335–1344).

Zhu, X., & Goldberg, A. B. (2009). Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *3*(1), 1–130.