Online Algorithms for Multiclass Classification Using Partial Labels

Rajarshi Bhattacharjee[1](✉) and Naresh Manwani[2]

[1] IIT Madras, Chennai, India
brajarshi91@gmail.com
[2] IIIT Hyderabad, Hyderabad, India
naresh.manwani@iiit.ac.in

Abstract. In this paper, we propose online algorithms for multiclass classification using partial labels. We propose two variants of Perceptron called Avg Perceptron and Max Perceptron to deal with the partially labeled data. We also propose Avg Pegasos and Max Pegasos, which are extensions of the Pegasos algorithm. We also provide mistake bounds for Avg Perceptron and regret bound for Avg Pegasos. We show the effectiveness of the proposed approaches by experimenting on various datasets and comparing them with the standard Perceptron and Pegasos.

Keywords: Online learning · Pegasos · Perceptron

1 Introduction

Multiclass classification is a well-studied problem in machine learning. However, we assume that we know the true label for every example in the training data. In many applications, we don't have access to the true class label as labeling data is an expensive and time-consuming process. Instead, we get a set of candidate labels for every example. This setting is called multiclass learning with partial labels. The true or ground-truth label is assumed to be one of the instances in the partial label set. Partially labeled data is relatively easier to obtain and thus provides a cheap alternative to learning with exact labels.

Learning with partial labels is referred to as superset label learning [13], ambiguous label learning [2], and by other names in different papers. Many proposed models try to *disambiguate* the correct labels from the incorrect ones. One popular approach is to treat the unknown correct label in the candidate set as a latent variable and then use an Expectation-Maximization type algorithm to estimate the correct label as well the model parameters iteratively [2,9,11,13,18].

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-47426-3_20) contains supplementary material, which is available to authorized users.

H. W. Lauw et al. (Eds.): PAKDD 2020, LNAI 12084, pp. 249–260, 2020.
https://doi.org/10.1007/978-3-030-47426-3_20

Other approaches to label disambiguation include using a maximum margin formulation [20] which alternates between ground truth identification and maximizing the margin from the ground-truth label to all other labels. Regularization based approaches [8] for partial label learning have also been proposed. Another model assumes that the ground truth label is the one to which the maximum score is assigned in the candidate label set by the model [14]. Then the margin between this ground-truth label and all other labels not in the candidate set is maximized.

Some approaches try to predict the label of an unseen instance by averaging the candidate labeling information of its nearest neighbors in the training set [10,21]. Some formulations combine the partial label learning framework with other frameworks like multi-label learning [19]. There are also specific approaches that do not try to disambiguate the label set directly. For example, Zhang et al. [22] introduced an algorithm that works to utilize the entire candidate label set using a method involving error-correcting codes.

A general risk minimization framework for learning with partial labels is discussed in Cour et al. [3,4]. In this framework, any standard convex loss function can be modified to be used in the partial label setting. For a single instance, since the ground-truth label is not available, an average over the scores in the candidate label set is taken as a proxy to calculate the loss. Nguyen and Caruana [14] propose a risk minimization approach based on a non-convex max-margin loss for a partial label setting.

In this paper, we propose online algorithms for multiclass classification using partially labeled data. Perceptron [15] algorithm is one of the earliest online learning algorithms. Perceptron for multiclass classification is proposed in [7]. A unified framework for designing online update rules for multiclass classification was provided in [5]. An online variant of the support vector machine [17] called Pegasos is proposed in [16]. This algorithm is shown to achieve $O(\log T)$ *regret* (where T is the number of rounds). Once again, all these online approaches assume that we know the true label for each example.

Online multiclass learning with partial labels remained an unaddressed problem. In this paper, we propose several online multiclass algorithms using partial labels. Our key contributions in this paper are as follows.

1. We propose Avg Perceptron and Max Perceptron, which extensions of Perceptron to handle the partial labels. Similarly, we propose Avg Pagasos and Max Pegasos, which are extensions of the Pegasos algorithm.
2. We derive mistake bounds for Avg Perceptron in both separable and general cases. Similarly, we provide $\log(T)$ regret bound for Avg Pegasos.
3. We also provide thorough experimental validation of our algorithms using datasets of different dimensions and compare the performance of the proposed algorithms with standard multiclass Perceptron and Pegasos.

2 Multiclass Classification Using Partially Labeled Data

We now formally discuss the problem of multiclass classification given partially labeled training set. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be the feature space from which the instances are drawn and let $\mathcal{Y} = \{1, \ldots, K\}$ be the output label space. Every instance $\mathbf{x} \in \mathcal{X}$ is associated with a candidate label set $Y \subseteq \mathcal{Y}$. The set of labels not present in the candidate label set is denoted by $\overline{Y}$. Obviously, $Y \cup \overline{Y} = [K]$.[1] The ground-truth label associated with $\mathbf{x}$ is denoted by lowercase y. It is assumed that the actual label lies within the set Y (i.e., $y \in Y$). The goal is to learn a classifier $h : \mathcal{X} \rightarrow \mathcal{Y}$. Let us assume that $h(\mathbf{x})$ is a linear classifier. Thus, $h(\mathbf{x})$ is parameterized by a matrix of weights $W \in \mathbb{R}^{d \times K}$ and is defined as $h(\mathbf{x}) = \arg\max_{i \in [K]} \ \mathbf{w}_i.\mathbf{x}$ where $\mathbf{w}_i$ (ith column vector of W) denotes the parameter vector corresponding to the i^{th} class. Discrepancy between the true label and the predicted label is captured using 0–1 loss as $L_{0-1}(h(\mathbf{x}), y) = \mathbb{I}_{\{h(\mathbf{x}) \neq y\}}$. Here, $\mathbb{I}$ is the 0–1 indicator function, which evaluates to true when the condition mentioned is true and 0 otherwise. However, in the case of partial labels, we use partial (ambiguous) 0–1 loss [3] as follows.

$$L_A(h(\mathbf{x}), Y) = \mathbb{I}_{\{h(\mathbf{x}) \notin Y\}} \tag{1}$$

Minimizing L_A is difficult as it is not continuous. Thus, we use continuous surrogates for L_A. A convex surrogate of L_A is the *average prediction* hinge loss (APH) [3] which is defined as follows.

$$L_{APH}(h(\mathbf{x}), Y) = \left[1 - \frac{1}{|Y|} \sum_{i \in Y} \mathbf{w}_i.\mathbf{x} + \max_{j \notin Y} \mathbf{w}_j.\mathbf{x}\right]_+ \tag{2}$$

where $|Y|$ is the size of the candidate label set and $[a]_+ = \max(a, 0)$. L_{APH} is shown to be a convex surrogate of L_A in [4]. There is another non-convex surrogate loss function called the *max prediction* hinge loss (MPH) [14] that can be used for partial labels which is defined as follows:

$$L_{MPH}(h(\mathbf{x}), Y) = \left[1 - \max_{i \in Y} \mathbf{w}_i.\mathbf{x} + \max_{j \notin Y} \mathbf{w}_j.\mathbf{x}\right]_+ \tag{3}$$

In this paper, we present online algorithms based on stochastic gradient descent on L_{APH} and L_{MPH}.

3 Multiclass Perceptron Using Partial Labels

In this section, we propose two variants of multiclass Perceptron using partial labels. Let the instance observed at time t be $\mathbf{x}^t$ and its corresponding label set be Y^t. The weight matrix at time t is W^t and the ith column of W^t is denoted by $\mathbf{w}_i^t$. To update the weights, we propose two different schemes: (a) Avg Perceptron (using stochastic gradient descent on L_{APH}) and (b) Max Perceptron (using stochastic gradient descent on L_{MPH}). We use following sub-gradients of the L_{APH} and L_{MPH}.

[1] We denote the set $\{1, \ldots, K\}$ using $[K]$.

$$\nabla_{\mathbf{w}_k} L_{APH} = \begin{cases} 0, & \text{if } \frac{1}{|Y|}\sum_{i\in Y}\mathbf{w}_i.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} \geq 1 \\ -\frac{\mathbf{x}}{|Y|}, & \text{if } \frac{1}{|Y|}\sum_{i\in Y}\mathbf{w}_i.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} < 1 \\ & \text{ and } k \in Y \\ \mathbf{x}, & \text{if } \frac{1}{|Y|}\sum_{i\in Y}\mathbf{w}_i.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} < 1 \\ & \text{ and } k = \arg\max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} \\ 0, & \text{if } \frac{1}{|Y|}\sum_{i\in Y}\mathbf{w}_i.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} < 1 \\ & , k \in \overline{Y} \text{ and } k \neq \arg\max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} \end{cases} \tag{4}$$

$$\nabla_{\mathbf{w}_k} L_{MPH} = \begin{cases} 0, & \text{if } \max_{j\in Y}\mathbf{w}_j.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} \geq 1 \\ -\mathbf{x}, & \text{if } \max_{j\in Y}\mathbf{w}_j.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} < 1 \\ & \text{ and } k = \arg\max_{i\in Y}\mathbf{w}_i.\mathbf{x} \\ \mathbf{x}, & \text{if } \max_{j\in Y}\mathbf{w}_j.\mathbf{x} - \max_{j\in\overline{Y}}\mathbf{w}_j.\mathbf{x} < 1 \\ & \text{and } k = \arg\max_{i\in\overline{Y}}\mathbf{w}_i.\mathbf{x} \end{cases} \tag{5}$$

We initialize the weight matrix as a matrix of zeros. At trial t, the update rule for $\mathbf{w}_i$ can be written as:

$$\mathbf{w}_i^{t+1} = \mathbf{w}_i^t - \eta\nabla_{\mathbf{w}_i} L(h^t(\mathbf{x}^t), Y^t)$$

where $\eta > 0$ is the step size and $\nabla_{\mathbf{w}_i} L(h^t(\mathbf{x}^t), Y^t)$ is found using Eq. (4) and (5). The complete description of Avg Perceptron and Max Perceptron is provided in Algorithm 1 and 2 respectively.

3.1 Mistake Bound Analysis

In the partial label setting, we say that mistake happens when the predicted class label for an example does not belong to its partial label set. We first define two variants of linear separability in a partial label setting as follows.

Definition 1 (Average Linear Separability in Partial Label Setting). *Let* $\{(\mathbf{x}^1, Y^1), \ldots, (\mathbf{x}^T, Y^T)\}$ *be the training set for multiclass classification with partial labels. We say that the data is average linearly separable if there exist* $\mathbf{w}_1, \ldots, \mathbf{w}_K \in \mathbb{R}^d$ *such that*

$$\frac{1}{|Y^t|}\sum_{i\in Y^t}\mathbf{w}_i.\mathbf{x}^t - \max_{j\in\overline{Y}^t}\mathbf{w}_j.\mathbf{x}^t \geq \gamma,\ \forall t \in [T].$$

Thus, average linear separability implies that $L_{APH}(h(\mathbf{x}^t), Y^t) = 0,\ \forall t \in [T]$.

Definition 2 (Max Linear Separability in Partial Label Setting). *Let* $\{(\mathbf{x}^1, Y^1), \ldots, (\mathbf{x}^T, Y^T)\}$ *be the training set for multiclass classification with partial labels. We say that the data is max linearly separable if there exist* $\mathbf{w}_1, \ldots, \mathbf{w}_K \in \mathbb{R}^d$ *such that*

$$\max_{i\in Y^t}\mathbf{w}_i.\mathbf{x}^t - \max_{j\in\overline{Y}^t}\mathbf{w}_j.\mathbf{x}^t \geq \gamma,\ \forall t \in [T].$$

Thus, max linear separability implies that $L_{MPH}(h(\mathbf{x}^t), Y^t) = 0,\ \forall t \in [T]$.

Algorithm 1. Avg Perceptron

```
Initialize $W^1 = 0$
for $t = 1$ to T do
    Get $\mathbf{x}^t$
    Predict $\hat{y}^t$ as $\hat{y}^t = \arg\max_{i\in[K]} \mathbf{w}_i^t.\mathbf{x}^t$
    Get the partial label set $Y^t$ of $\mathbf{x}^t$
    Calculate loss $L_{APH}(h^t(\mathbf{x}^t), Y^t)$ using Eq. (2)
    if $L_{APH}(h^t(\mathbf{x}^t, Y^t) > 0$ then
        $\mathbf{w}_i^{t+1} = \mathbf{w}_i^t + \eta\tau_i^t\mathbf{x}^t,\ i \in [K]$ where
            $\tau_i^t = \begin{cases} \frac{1}{|Y^t|}, & i \in Y^t \\ -1, & i = \arg\max_{j\in\overline{Y}^t} \mathbf{w}_j^t.\mathbf{x}^t \\ 0, & \forall i \in \overline{Y}^t,\ i \neq \arg\max_{j\in\overline{Y}^t} \end{cases}$
    else
        $\mathbf{w}_i^{t+1} = \mathbf{w}_i^t,\ \forall i \in [K]$
    end if
end for
```

We bound the number of mistakes made by Avg Perceptron (Algorithm 1) as follows.

Theorem 1 (Mistake Bound for Avg Perceptron Under Average Linear Separability). *Let* $(\mathbf{x}^1, Y^1), \ldots, (\mathbf{x}^T, Y^T)$ *be the examples presented to Avg Perceptron, where* $\mathbf{x}^t \in \mathbb{R}^d$ *and* $Y^t \subseteq [K]$*. Let* $W^* \in \mathbb{R}^{d\times K}$ $(\|W^*\| = 1)$ *be such that* $\frac{1}{|Y^t|}\sum_{i\in Y^t} \mathbf{w}_i^*.\mathbf{x}^t - \max_{j\in\overline{Y}^t} \mathbf{w}_j^*.\mathbf{x}^t \geq \gamma,\ \forall t \in [T]$*. Then we get the following mistake bound for Avg Perceptron Algorithm.*

$$\sum_{t=1}^{T} L_A(h^t(\mathbf{x}^t), Y^t) \leq \frac{2}{\gamma^2} + \left[\frac{1}{c} + 1\right]\frac{R^2}{\gamma^2}$$

where $c = \min_t |Y^t|$*,* $R = \max_t ||\mathbf{x}^t||$ *and* $\gamma \geq 0$ *is the margin of separation.*

The proof is given in Appendix A of [1]. We first notice that the bound is inversely proportional to the minimum label set size. This is intuitively obvious as the smaller the candidate label set size, the larger the chance of having a non-zero loss. When $c = 1$, the number of updates reduces to the normal multiclass Perceptron mistake bound for linearly separable data as given in [5]. Also, the number of mistakes is inversely proportional to γ^2. Linear separability (Definition 1) may not always hold for the training data. Thus, it is important to see how does the algorithm Avg Perceptron performs in such cases. We now bound the number of updates in T rounds for partially labeled data, which is linearly non-separable under L_{APH}.

Theorem 2 (Mistake Bound for Avg Perceptron in Non-Separable Case). *Let* $(\mathbf{x}^1, Y^1), \ldots, (\mathbf{x}^T, Y^T)$ *be an input sequence presented to Avg Perceptron. Let* W $(\|W\| = 1)$ *be weight matrix corresponding to a multiclass*

Algorithm 2. Max Perceptron

```
Initialize $W^1 = 0$
for $t = 1$ to T do
    Get $\mathbf{x}^t$
    Predict $\hat{y}^t$ as $\hat{y}^t = \arg\max_{i \in [K]} \mathbf{w}_i^t.\mathbf{x}^t$
    Get the partial label set $Y^t$ of $\mathbf{x}^t$
    Calculate loss $L_{MPH}(h^t(\mathbf{x}^t), Y^t)$ using Eq. (3)
    if $L_{MPH}(h^t(\mathbf{x}^t, Y^t) > 0$ then
        $\mathbf{w}_i^{t+1} = \mathbf{w}_i^t + \eta \tau_i^t \mathbf{x}^t,\ i \in [K]$ where
            $$\tau_i^t = \begin{cases} 1, & \text{if } \max_{j \in Y} \mathbf{w}_j.\mathbf{x} - \max_{j \in \overline{Y}} \mathbf{w}_j.\mathbf{x} < 1 \\ & \text{ and } i = \arg\max_{j \in Y} \mathbf{w}_j.\mathbf{x} \\ -1, & \text{if } \max_{j \in Y} \mathbf{w}_j.\mathbf{x} - \max_{j \in \overline{Y}} \mathbf{w}_j.\mathbf{x} < 1 \\ & \text{and } i = \arg\max_{j \in \overline{Y}} \mathbf{w}_j.\mathbf{x} \end{cases}$$
    else
        $\mathbf{w}_i^{t+1} = \mathbf{w}_i^t,\ \forall i \in [K]$
    end if
end for
```

classifier. Then for a fixed $\gamma > 0$, *let* $d^t = \max\left\{0, \gamma - [\frac{1}{|Y^t|}\sum_{i \in Y^t} \mathbf{w}_i.\mathbf{x}^t - \max_{j \in \overline{Y}^t} \mathbf{w}_j.\mathbf{x}^t]\right\}$. *Let* $D^2 = \sum_{t=1}^{T}(|Y^t| d^t)^2$ *and* $R = \max_{t \in [T]} ||\mathbf{x}^t||$ *and* $c = \min_{t \in [T]} |Y^t|$. *Then, mistakes bound for Avg Perceptron is as follows.*

$$\sum_{t=1}^{T} L_A(h^t(\mathbf{x}^t), Y^t) \leq 2\frac{Z^2}{\gamma^2} + 2K\frac{R^2 + \Delta^2}{(\frac{\gamma}{Z})^2}$$

where $Z = \sqrt{1 + \frac{D^2}{\Delta^2}}$, $\Delta = \left[\frac{D^2 + KD^2R^2}{K}\right]^{\frac{1}{4}}$ *and* $K = \left[\frac{1}{c} + 1\right]$.

The proof is provided in the Appendix B of [1].

4 Online Multiclass Pegasos Using Partial Labels

Pegasos [16] is an online algorithm originally proposed for an exact label setting. In Pegasos, L_2 regularizer of the weights is minimized along with the hinge loss, making the overall objective function strongly convex. The strong convexity enables the algorithm to achieve a $O(\log T)$ regret in T trials. The objective function of the Pegasos at trial t is the following.

$$f(W, \mathbf{x}^t, Y^t) = \frac{\lambda}{2}||W||^2 + L(h(\mathbf{x}^t), Y^t)$$

Here, λ is a regularization constant and $||W||$ is Frobenius norm of the weight matrix. Let W^t be the weight matrix at the beginning of trial t. Then, W^{t+1} is found as $W^{t+1} = \Pi_B(W^t - \eta_t \nabla^t)$. Here $\nabla^t = \nabla_{W^t} f(W^t, \mathbf{x}^t, Y^t)$, η_t is the step size at trial t and Π_B is a projection operation onto the set B which is defined as $B = \{W : ||W|| \leq \frac{1}{\sqrt{\lambda}}\}$. Thus, $\Pi_B(W) = \min\{1, \frac{1}{(\lambda||W||)}\}W$.

We now propose extension of Pegasos [16] for online multiclass learning using partially labeled data. We again propose two variants of Pegasos: (a) Avg Pegasos (using average prediction hinge loss (Eq. 2)) and (b) Max Pegasos (using max prediction hinge loss (Eq. (3)). We first note that ∇^t can be written as:

$$\nabla^t = \lambda W^t + \nabla_{W^t} L \tag{6}$$

where $\nabla_{W^t} L$ is given by Eq. (4) (for L_{APH}) and Eq. (5) (for L_{MPH}). Complete description of Avg Pegasos and Max Pegasos are given in Algorithm 3 and Algorithm 4 respectively.

Algorithm 3. Avg Pegasos

```
Input: $\lambda, T$
Initialize: $W_1$ s.t. $||W^1|| \leq \frac{1}{\sqrt{\lambda}}$
for $t = 1$ to $T$ do
    Get $\mathbf{x}^t, Y^t$
    Set $\eta_t = \frac{1}{\lambda t}$
    Calculate loss $L_{APH}(h^t(\mathbf{x}^t), Y^t)$ using Eq. (2)
    if $L_{APH} > 0$ then
        $W^{t+\frac{1}{2}} = (1 - \eta_t \lambda) W^t - \eta_t \nabla_W L_{APH}$ where $\nabla_W L_{APH}$ is given by Eq. (4)
        $W^{t+1} = \min\{1, \frac{1/\sqrt{\lambda}}{||W^{t+\frac{1}{2}}||}\} W^{t+\frac{1}{2}}$
    else
        $W^{t+1} = W^t$
    end if
end for
Output: $W^T$
```

4.1 Regret Bound Analysis of Avg Pegasos

We now derive the regret bound for Avg Pegasos.

Theorem 3. *Let* $(\mathbf{x}^1, Y^1), (\mathbf{x}^2, Y^1), \ldots, (\mathbf{x}^T, Y^T)$ *be an input sequence where* $\mathbf{x}^t \in \mathbb{R}^d$ *and* $Y^t \subseteq [K]$. *Let* $R = \max_t ||\mathbf{x}^t||$. *Then the regret of Avg Pegasos is given as:*

$$\frac{1}{T}\sum_{t=1}^{T} f(W^t, \mathbf{x}^t, Y^t) - \min_W \frac{1}{T}\sum_{t=1}^{T} f(W, \mathbf{x}^t, Y^t) \leq \frac{G^2 lnT}{\lambda T}$$

where $G = \sqrt{\lambda} + \sqrt{1 + \frac{1}{c}} R$ *and* $c = \min_t |Y^t|$

The proof is given in Appendix C of [1]. We again see the regret is inversely proportional to the size of the minimum candidate label set.

Algorithm 4. Max Pegasos

```
Input: $\lambda, T$
Initialize: $W_1$ s.t. $||W^1|| \leq \frac{1}{\sqrt{\lambda}}$
for $t = 1$ to $T$ do
    Get $\mathbf{x}^t, Y^t$
    Set $\eta_t = \frac{1}{\lambda t}$
    Calculate loss $L_{MPH}(h^t(\mathbf{x}^t), Y^t)$ using Eq. (3)
    if $L_{APH} > 0$ then
        $W^{t+\frac{1}{2}} = (1 - \eta_t \lambda) W^t - \eta_t \nabla_W L_{MPH}$ where $\nabla_W L_{MPH}$ is given by Eq. (5)
        $W^{t+1} = \min\{1, \frac{1/\sqrt{\lambda}}{||W^{t+\frac{1}{2}}||}\} W^{t+\frac{1}{2}}$
    else
        $W^{t+1} = W^t$
    end if
end for
Output: $W^T$
```

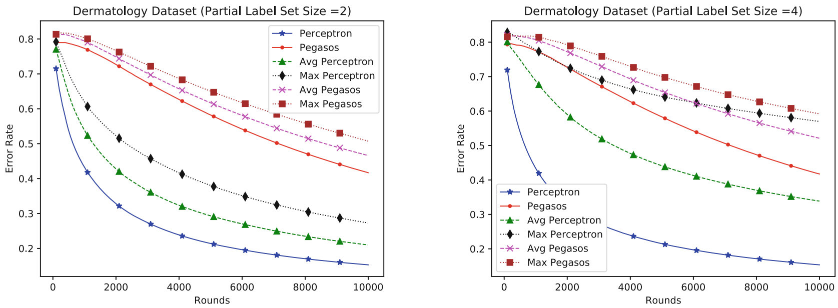

Fig. 1. Dermatology dataset results

5 Experiments

We now describe the experimental results. We perform experiments on Ecoli, Satimage, Dermatology, and USPS datasets (available on UCI repository [6]) and MNIST dataset [12]. We perform experiments using the proposed algorithms Avg Perceptron, Max Perceptron, Avg Pegasos, and Max Pegasos. For benchmarking, we use Perceptron and Pegasos based on exact labels.

For all the datasets, the candidate or partial label set for each instance contains the true label and some labels selected uniformly at random from the remaining labels. After every trial, we find the average mis-classification rate (average of L_{0-1} loss over examples seen till that trial) is calculated with respect to the true label. This sets a hard evaluation criteria for the algorithms. The number of rounds for each dataset is selected by observing when the error curves start to converge. For every dataset, we repeat the process of generating partial label sets and plotting the error curves 100 times and average the instantaneous error rates across the 100 runs. The final plots for each dataset have the average instantaneous error rate on the Y-axis and the number of rounds on the X-axis.

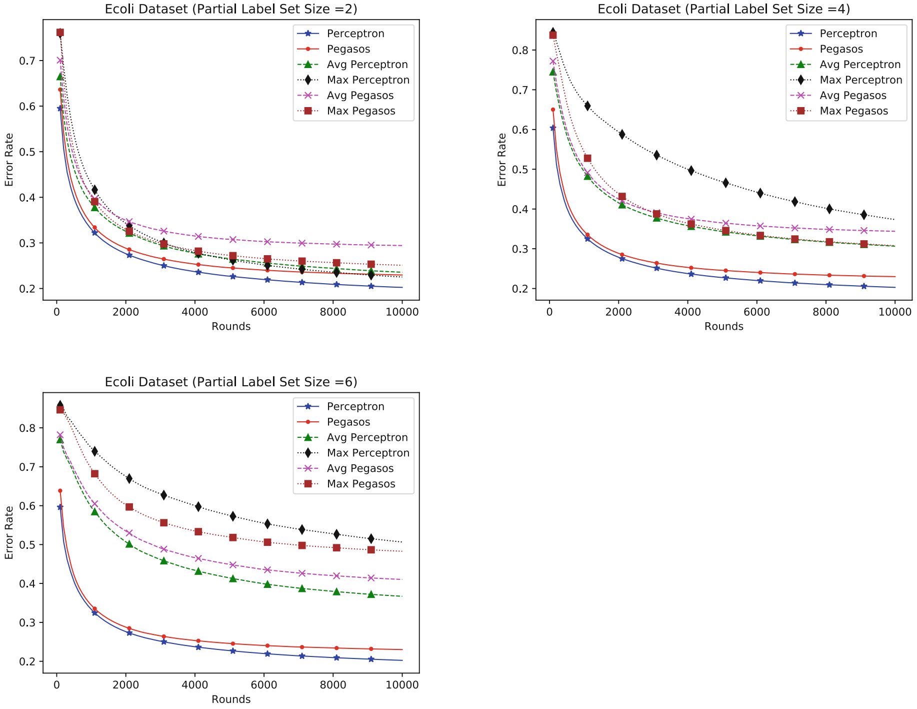

Fig. 2. Ecoli dataset results

For every dataset, we plot the error rate curves for all the algorithms for different candidate label set sizes. This helps us in understanding how the online algorithms behave as the candidate label set size increases. For the Dermatology dataset, which contains six classes, we take candidate labels sets of sizes 2 and 4, respectively, as shown in Fig. 1. We see that the average prediction loss based algorithms perform the better in both cases. The results for the Ecoli dataset for candidate label sets of size 2, 4 and 6 are shown in Fig. 2. Here, we find that the Max Pegasos algorithm performs comparably to the algorithms based on the Average Prediction Loss for candidate labels set sizes 2 and 4. But for candidate label set size 8, the Max Prediction Loss performs significantly worse than the Average Prediction Loss based algorithm. The results for Satimage and USPS datasets are shown in Fig. 3 and 4 respectively. For Satimage, the Max Pegasos performs the best for label set of size 2. But for label set size 4, the Average Prediction Loss based algorithms perform much better. For USPS, we see that though for candidate labels set sizes 2 and 4, the Max Perceptron and Max Pegasos perform better than our algorithms, for label set sizes 6 and 8, the Average Prediction Loss based algorithms perform much better. The results for MNIST are provided in Fig. 5. Here we observe the Max Perceptron and Max Pegasos performs much better than the other algorithms for label set sizes 2 and 4. However, for label set sizes 6 and 8, the Average Pegasos performs best.

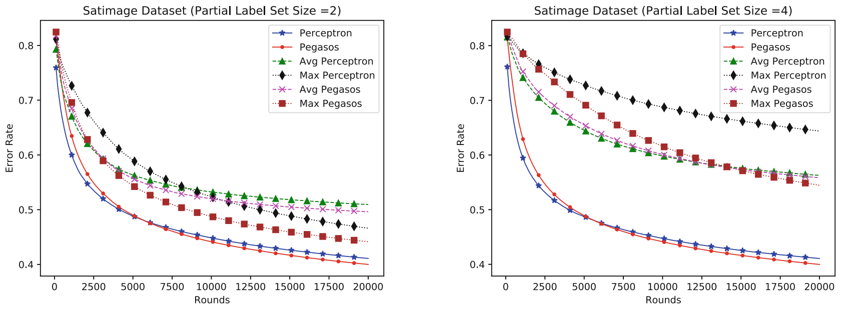

Fig. 3. Satimage dataset results

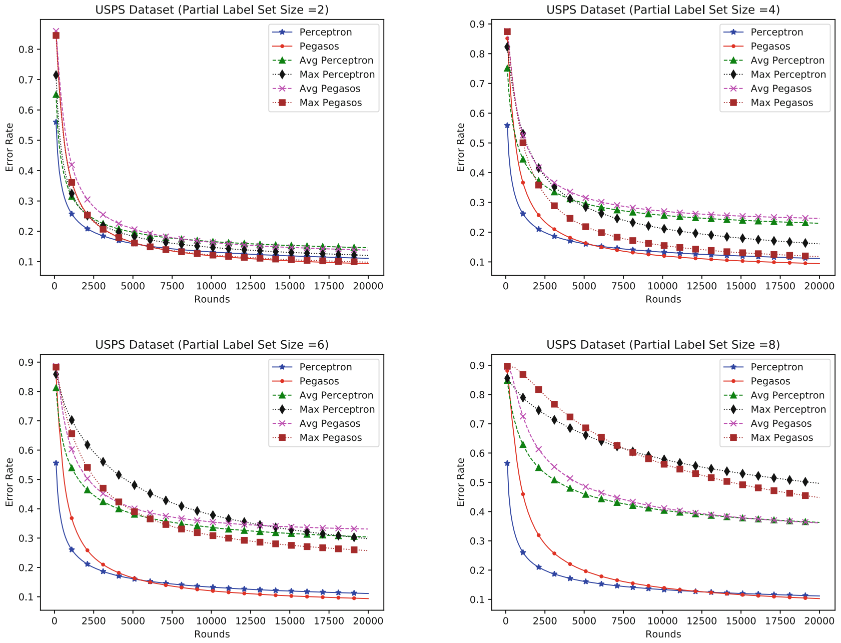

Fig. 4. USPS dataset results

Overall, we see that for smaller labels set sizes, the Max Prediction Loss performs quite well. However, the Average Prediction Loss shows the best for larger candidate label set sizes. Studying the convergence and theoretical properties of the non-convex Max Prediction Loss can be an exciting future direction for exploration.

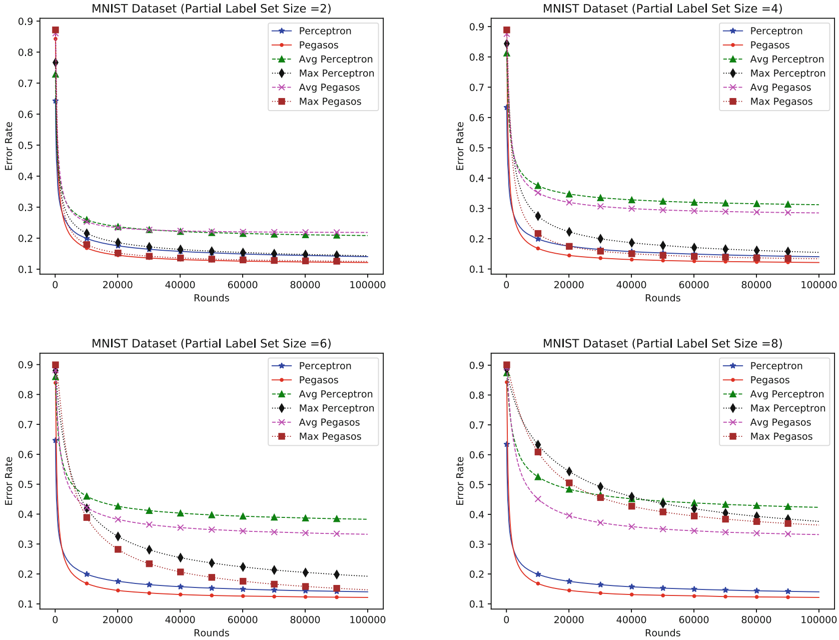

Fig. 5. MNIST dataset results

6 Conclusion

In this paper, we proposed online algorithms for classifying partially labeled data. This is very useful in real-life scenarios when multiple annotators give different labels for the same instance. We presented algorithms based on the Perceptron and Pegasos. We also provide mistake bounds for the Perceptron based algorithm and the regret bound for the Pegasos based algorithm. We also provide an experimental comparison of all the algorithms on various datasets. The results show that though the Average Prediction Loss is convex, the non-convex Max Prediction Loss can also be useful for small labels set sizes. Providing a theoretical analysis for the Max Prediction Loss can be a useful endeavor in the future.

References

1. Bhattacharjee, R., Manwani, N.: Online algorithms for multiclass classification using partial labels. arXiv e-prints arXiv:1912.11367, December 2019
2. Chen, Y., Patel, V.M., Chellappa, R., Phillips, P.J.: Ambiguously labeled learning using dictionaries. IEEE Trans. Inf. Forensics Secur. **9**(12), 2076–2088 (2014)
3. Cour, T., Sapp, B., Taskar, B.: Learning from partial labels. J. Mach. Learn. Res. **12**, 1501–1536 (2011)

4. Cour, T., Sapp, B., Jordan, C., Taskar, B.: Learning from ambiguously labeled images. In: Proceedings of the IEEE Computer Society Conference on Computing Vision and Pattern Recognition, pp. 919–926 (2009)
5. Crammer, K., Singer, Y.: Ultraconservative online algorithms for multiclass problems. J. Mach. Learn. Res. **3**, 951–991 (2003)
6. Dua, D., Graff, C.: UCI machine learning repository (2017)
7. Duda, R., Hart, P.: Pattern Classification and Scene Analysis. Wiley, New York (1973)
8. Feng, L., An, B.: Partial label learning with self-guided retraining. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, pp. 3542–3549. AAAI Press (2019)
9. Grandvalet, Y., Bengio, Y.: Learning from partial labels with minimum entropy. Center for Interuniversity Research and Analysis of Organizations (2004)
10. Hüllermeier, E., Beringer, J.: Learning from ambiguously labeled examples. Intell. Data Anal. **10**(5), 419–439 (2006)
11. Jin, R., Ghahramani, Z.: Learning with multiple labels. In: Becker, S., Thrun, S., Obermayer, K. (eds.) Advances in Neural Information Processing Systems, pp. 921–928. MIT Press, Cambridge (2003)
12. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
13. Liu, L., Dietterich, T.: A conditional multinomial mixture model for superset label learning. In: Bartlett, P., Pereira, F.C.N., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, pp. 557–565. MIT Press, Cambridge (2012)
14. Nguyen, N., Caruana, R.: Classification with partial labels. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery Data Mining, pp. 551–559 (2008)
15. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. Psychol. Rev. **65**, 386–407 (1958)
16. Shalev-Shwartz, S., Singer, Y., Srebro, N.: Pegasos: primal estimated sub-gradient solver for SVM. In: Proceedings of the International Conference on Machine Learning (ICML) (2007)
17. Smola, A.J., Schölkopf, B.: A tutorial on support vector regression. Stat. Comput. **14**(3), 199–222 (2004)
18. Vannoorenberghe, P., Smets, P.: Partially supervised learning by a **Cr**edal **EM** approach. In: Godo, L. (ed.) ECSQARU 2005. LNCS (LNAI), vol. 3571, pp. 956–967. Springer, Heidelberg (2005). https://doi.org/10.1007/11518655_80
19. Xie, M.K., Huang, S.J.: Partial multi-label learning. In: Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018), pp. 1–8 (2018)
20. Yu, F., Zhang, M.-L.: Maximum margin partial label learning. Mach. Learn. **106**(4), 573–593 (2016). https://doi.org/10.1007/s10994-016-5606-4
21. Zhang, M.L., Yu, F.: Solving the partial label learning problem: an instance-based approach. In: Yang, Q., Wooldridge, M. (eds.) Proceedings of the 24th International Conference on Artificial Intelligence, pp. 4048–4054. AAAI Press (2015)
22. Zhang, M.L., Yu, F., Tang, C.Z.: Disambiguation-free partial label learning. IEEE Trans. Knowl. Data Eng. **29**(10), 2155–2167 (2017)