Int. J. Appl. Math. Comput. Sci., 2022, Vol. 32, No. 2, 311–322
DOI: 10.34768/amcs-2022-0023

JOINT FEATURE SELECTION AND CLASSIFICATION FOR POSITIVE UNLABELLED MULTI–LABEL DATA USING WEIGHTED PENALIZED EMPIRICAL RISK MINIMIZATION

PAWEŁ TEISSEYRE [a,b]

[a]Institute of Computer Science
Polish Academy of Sciences
Jana Kazimierza 5, 01-248 Warsaw, Poland
e-mail: Pawel.Teisseyre@ipipan.waw.pl

[b]Faculty of Mathematics and Information Science
Warsaw University of Technology
Koszykowa 75, 00-062 Warsaw, Poland

We consider the positive-unlabelled multi-label scenario in which multiple target variables are not observed directly. Instead, we observe surrogate variables indicating whether or not the target variables are labelled. The presence of a label means that the corresponding variable is positive. The absence of the label means that the variable can be either positive or negative. We analyze embedded feature selection methods based on two weighted penalized empirical risk minimization frameworks. In the first approach, we introduce weights of observations. The idea is to assign larger weights to observations for which there is a consistency between the values of the true target variable and the corresponding surrogate variable. In the second approach, we consider a weighted empirical risk function which corresponds to the risk function for the true unobserved target variables. The weights in both the methods depend on the unknown propensity score functions, whose estimation is a challenging problem. We propose to use very simple bounds for the propensity score, which leads to relatively simple forms of weights. In the experiments we analyze the predictive power of the methods considered for different labelling schemes.

Keywords: positive and unlabelled data, multi-label classification, feature selection, empirical risk minimization.

1. Introduction

1.1. Problem description. Multi-label classification (Dembczyński *et al*., 2012; Zhang and Zhou, 2013; Gibaja and Ventura, 2015) is a variant of the classification task in which many binary target variables $y_1, \ldots, y_K$ are considered simultaneously. The goal is to build a model using training data which predicts values of the target variables using feature vector $\mathbf{x} = (x_1, \ldots, x_p)$. In the positive-unlabelled multi-label (PU-ML) scenario, the true target variables are not observed directly in the training data. Instead, we observe surrogate target variables $s_1, \ldots, s_K$, which indicate whether the target variables are labelled or not. The presence of the label means that the target variable is positive, i.e., $s_k = 1$ implies $y_k = 1$. The absence of the label ($s_k = 0$) means that the true target variable can be either positive ($y_k = 1$) or negative ($y_k = 0$). The objective is to predict the true target variables using the model which is based on feature vector $\mathbf{x}$ and surrogate target variables $s_1, \ldots, s_K$.

Positive unlabelled multi-label (PU-ML) data appear naturally in many different fields. As an example, consider a problem of predicting multi-morbidity, i.e., co-occurrence of multiple diseases in one patient using patients characteristics, which is a typical multi-label task (Zufferey *et al*., 2015; Teisseyre, 2020). It may happen that some diseases are not diagnosed or are not reported in a given patient. However the absence of a diagnosis does not mean that the patient does not have the disease in question. Consequently, we can distinguish three groups of patients: those with the diagnosed disease ($s_k = 1$ and thus $y_k = 1$); patients without diagnosed disease who have the disease ($s_k = 0$ and $y_k = 1$) and, finally,

patients without the diagnosed disease who really do not have the disease ($s_k = 0$ and $y_k = 0$). The issue is important as many studies indicate that certain diseases, such as hypertension or diabetes, are often undiagnosed (Walley, 2018) and, in consequence, the model fitted on such incomplete labelled data may give misleading predictions. The positive unlabelled learning is also closely related to the problem of under-reporting (Sechidis *et al*., 2017). An important example are under-reporting adverse drug reactions (Hazell and Shakir, 2006). In this case the first group consists of the respondents who report adverse drug reactions ($s_k = 1$ and thus $y_k = 1$). The second group experiences adverse reactions but does not report it ($s_k = 0$ and $y_k = 1$). Finally, in the third group the adverse drug reactions do not occur and this group has nothing to report ($s_k = 0$ and $y_k = 0$).

1.2. Related work. PU-ML learning is a natural generalization of the standard PU scenario (Bekker and Davis, 2020) to the case of multiple target variables. There are many partial observability schemes related to PU learning. For example, in multi-label learning with missing labels (Zhu *et al*., 2018; He *et al*., 2019) some values of target variables are positive, some are negative and the others are unlabelled (can be either positive or negative). The PU-ML setting can be also seen as a special case of the more general problem of learning from noisy labels (Natarajan *et al*., 2013; Frenay and Verleysen, 2014) when labels are incorrectly assigned. In such general scenario, the value of the true class variable y_k can be flipped with some probabilities $\omega_1 := P(s_k = 0|y_k = 1)$ and $\omega_2 := P(s_k = 1|y_k = 0)$ and instead y_k we observe $s_k = 1 - y_k$. This problem reduces to the PU setting for $\omega_2 = 0$. The other related task is 'coarse data' analysis (Heitjan and Rubin, 1991; Couso *et al*., 2017), where instead of the exact value of y_k, only some subset of the possible values of y_k is given in the training data.

In this paper we consider a problem of feature selection for PU-ML data. Feature selection is an essential part when building classification models (Guyon and Elisseeff, 2003). First, it improves the prediction accuracy of the models. Fitting the models based on a large number of features includes estimation of a large number of parameters. It is well known that fitting models with many noisy features increases the variance of estimators and thus decreases the prediction accuracy of the model (see, e.g., Hastie *et al.*, 2009, Chapter 7). Finally, feature selection methods are used to discover a dependency structure in data and to have a model which can be interpreted (Biecek, 2018), which is particularly important in biological and medical applications.

Although both positive unlabelled multi-label learning (Sun *et al*., 2010; Bucak *et al*., 2011; Wei *et al*., 2018; Wu *et al*., 2013; Kanehira and Harada, 2016; Teisseyre, 2021) as well as the problem of feature selection in multi-label classification (Pereira *et al*., 2018; Kashef *et al*., 2018; Lee and Kim, 2017) have attracted close attention, a combination of these two problems, to the best of our knowledge, remains an unexplored area. In this paper, we focus on penalized empirical risk minimization frameworks. In all methods considered we use $\ell_{2,1}$ regularization which ensures that the selected features will be shared across the models corresponding to different target variables. Such regularization is very natural in multi-label classification. It has been effectively exploited in simultaneous multi-task learning problems (Argyriou *et al*., 2008) and also in multi-label classification (Ji *et al*., 2010; Naula *et al*., 2014). The $\ell_{2,1}$ regularization is useful when one wants to fit the model subject to a constraint on the number of features. Such constraints are important in domains where the acquisition of the feature values is costly or is associated with a certain risk. For example, in medical diagnosis, each diagnostic test is associated with its cost. Moreover, unnecessary diagnostic tests or treatments may cause negative effects and even increase the risk of death; examples include treatments under general anesthesia (Lagasse, 2002) or diagnostic X-rays (Hall and Brenner, 2008).

The key problem in the feature selection methods considered is the choice of the empirical risk function. The naive approach is to consider the risk function for the surrogate (observed) target variables. Such an approach is called 'biased' as the risk does not correspond to the risk for the true target variables and, therefore, the resulting estimates of the posterior probabilities will underestimate the true posterior probabilities $P(y_k = 1|\mathbf{x})$. This leads to poor predictive performance of the naive method, especially when the fraction of labelled examples is small (Teisseyre *et al*., 2020). On the other hand, in the previous studies (Sechidis *et al*., 2014; Sechidis and Brown, 2018) it was demonstrated that in the context of feature selection, the naive approach can rank the features correctly (i.e., it assigns the highest scores to the relevant features), provided that the SCAR (selected completely at random) assumption is met. In particular, it has been shown that the mutual information $I(s_k, \mathbf{x}) = 0$ if and only if $I(y_k, \mathbf{x}) = 0$. The above issues are confirmed by our experiments, namely the naive method ranks the features correctly, but its predictive power is significantly lower when compared with the proposed methods.

1.3. Contribution. In this work, we consider two weighted empirical risk minimization problems with $\ell_{2,1}$ regularization. The difference between the methods is that they use different risk functions. In the first proposed method, we modify the naive approach by introducing weights of observations. The idea is to assign larger weights to observations for which there is a consistency between the values of the true target variable and the

corresponding surrogate variable, i.e., $s_k = y_k$, and smaller weights to observations for which $s_k \neq y_k$. This is challenging as we only know that $y_k = s_k$, when $s_k = 1$, i.e., for labelled examples. In the second method, we consider another weighted empirical risk function which corresponds to the risk function for the true target variables. The weights in both the methods depend on the unknown propensity score function $e_k(\mathbf{x}) = P(s_k = 1|y_k = 1, \mathbf{x})$ which is difficult to estimate. We overcome this limitation by using simple, yet effective, bounds on the propensity score. Both the proposed methods can be treated as two-step procedures; in the first step we use the naive method to estimate the weights; in the second step we solve weighted empirical risk minimization problems with the weights obtained in the first step. In the case of the second method, we give a bound on the excess risk.

The article is structured as follows. In Section 2 we formally describe the problem and the naive approach. In Sections 3 and 4 we discuss the proposed methods. The experiments are described in Section 5 and Section 6 summarizes our research.

2. Multi-label positive unlabelled learning

2.1. Background. In multi-label classification, each instance is described by feature vector $\mathbf{x} = (x_1, \ldots, x_p) \in \mathcal{X}$ and label vector $\mathbf{y} = (y_1, \ldots, y_K) \in \{0, 1\}^K$. In the positive unlabelled (PU) setting we do not observe label vector $\mathbf{y}$ directly, we only observe a vector of surrogate variables $\mathbf{s} = (s_1, \ldots, s_K)$. The surrogate variable s_k indicates whether the k-th target is labelled (and thus positive, i.e. $s_k = 1 \implies y_k = 1$) or not ($s_k = 0$). Note that $s_k = 0$ does not imply that $y_k = 0$. Indeed, unlabelled examples ($s_k = 0$) can be either positive ($y_k = 1$) or negative ($y_k = 0$). In this paper we assume the so-called single data scenario (Elkan and Noto, 2008), according to which there is some unknown distribution $P(\mathbf{x}, \mathbf{y}, \mathbf{s})$ such that $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}, \mathbf{s}^{(i)})$, $i = 1, \ldots, n$ constitute an independent and identically distributed sample drawn from it and only data $D = \{(\mathbf{x}^{(i)}, \mathbf{s}^{(i)}), i = 1, \ldots, n\}$ are observed. The above setting is a generalization of the positive unlabelled scenario for single label classification to the case of multi-label classification. The goal of feature selection is to reduce the dimensionality of $\mathbf{x}$, i.e., to choose relevant features which affect labels $\mathbf{y}$ using only observed training data $(\mathbf{x}^{(i)}, \mathbf{s}^{(i)})$. We will focus on penalized empirical risk minimization methods.

Below we recall some basic quantities which play an important role in the PU setting. Let $q_k(\mathbf{x}) := P(y_k = 1|\mathbf{x})$ and $s_k(\mathbf{x}) := P(s_k = 1|\mathbf{x})$ be the posterior probabilities for the k-th target. In the PU setting, $q_k(\mathbf{x})$ cannot be directly estimated as we do not observe y_k, whereas it is possible to estimate $s_k(\mathbf{x})$ by learning a model using data $(\mathbf{x}^{(i)}, s_k^{(i)})$ (this approach is called 'biased learning' or 'naive learning', see the next subsection). Importantly, it follows from the Law of Total Probability that the above two quantities are related as

$$P(s_k = 1|\mathbf{x}) = e_k(\mathbf{x})P(y_k = 1|\mathbf{x}), \tag{1}$$

where $e_k(\mathbf{x}) := P(s_k = 1|y_k = 1, \mathbf{x})$ is the so-called propensity score function for the k-th target. The propensity score describes the labelling mechanism for the k-th target. It measures the likelihood of being labelled for the positive example described by feature vector $\mathbf{x}$. The estimation of $e_k(\mathbf{x})$ is a very challenging task and therefore recovering the posterior for the true target y_k is not straightforward in a general scenario. Most authors try to deal with this problem by imposing additional assumptions, e.g., the SCAR assumption according to which the probability of labelling for positive example does not depend on feature vector $\mathbf{x}$, i.e., $P(s_k = 1|y_k = 1, \mathbf{x}) = P(s_k = 1|y_k = 1) = c_k$, where c_k is a constant (label frequency) which has to be estimated (Elkan and Noto, 2008; Ramaswamy *et al*., 2016; Jain *et al*., 2016; Plessis *et al*., 2017; Bekker and Davis, 2018; Jaskie *et al*., 2020; Łazęcka *et al*., 2021). Since the SCAR is very often an unrealistic assumption (Bekker *et al*., 2019), in this paper, we try to avoid the SCAR by using simple bounds on the propensity score function.

2.2. Naive approach: Empirical risk minimization for surrogate target variables. Let $A \in \mathbb{R}^{K \times p}$ be a matrix of parameters, whose k-th row a_k is a parameter vector corresponding to the model for the k-th label. The linear predictor for the k-th label is $a_k^T \mathbf{x}$. The quality of the prediction for the k-th label is assessed by a loss function $l(\hat{y}, y)$. For example, the logistic loss is written as $-[y \log(\sigma(\hat{y})) + (1 - y) \log(1 - \sigma(\hat{y}))]$, where σ is a sigmoid logistic function. A natural approach is to consider the empirical risk for surrogate variables

$$\hat{R}_0(A) = \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} l(a_k^T \mathbf{x}^{(i)}, s_k^{(i)}). \tag{2}$$

This approach is called 'naive' (or biased) learning as it is based on observed target variables $\mathbf{s}$ and not the true target variables $\mathbf{y}$. We learn matrix A by minimizing the penalized empirical risk $\hat{A}^{(0)} = \arg\min_{A \in R^{K \times p}} [\hat{R}_0(A) + \lambda \sum_{j=1}^{p} ||A_j||_2]$, where A_j is the j-th column of matrix A. The $\ell_{2,1}$ penalty term ensures a common sparsity pattern in A and thus it allows us to select features which are relevant to predict all labels simultaneously.

3. Method 1: Weighted risk for surrogate target variables

The main limitation of the naive method is related to using the noisy target variables $\mathbf{s}$, i.e., for some observations

it may happen that $s_k^{(i)} \neq y_k^{(i)}$. The main goal of the proposed method is to modify the naive method by introducing weights of observations $w_k(\mathbf{x}^{(i)})$ which describe the consistency between s_k and y_k. In the proposed method, we consider the weighted empirical risk function

$$\hat{R}_1(A) = \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} w_k(\mathbf{x}^{(i)}) l(a_k^T \mathbf{x}^{(i)}, s_k^{(i)}) \quad (3)$$

and solve the related regularized problem $\hat{A}^{(1)} = \arg\min_{A \in \mathbb{R}^{K \times p}} [\hat{R}_1(A) + \lambda \sum_{j=1}^{p} ||A_j||_2]$. The crucial element in the above method is the choice of the weights $w_k(\mathbf{x}^{(i)})$. Note that in the PU scenario, one can distinguish three groups: (a) observations for which $y_k = 1, s_k = 1$ (labelled examples), (b) $y_k = 0, s_k = 0$ (unlabelled, negative examples) and (c) $y_k = 1, s_k = 0$ (unlabelled positive examples). For the first two groups, there is a consistency between s_k and y_k, whereas for group (c), the values of s_k and y_k do not match.

The main idea is to assign larger weights to observations from groups (a) and (b) and smaller weights to observations from group (c). In this way we can eliminate (or at least reduce) the influence of observations from (c). For the remaining observations, s_k can be replaced by y_k and then (3) will mimic the risk function for the true unobserved target variables.

This is a challenging task as y_k is not observable and therefore it is not possible to distinguish directly between groups (b) and (c).

A natural choice of weights for the k-th label is to assign $w_k(\mathbf{x}) = 1$ when $s_k = 1$ and

$$w_k(\mathbf{x}) = P(y_k = 0|\mathbf{x}) = \frac{1 - s_k(\mathbf{x})}{e_k(\mathbf{x})}$$

when $s_k = 0$. Such weights meet the above assumptions. As was mentioned in Section 2, estimation of the propensity score $e_k(\mathbf{x})$ is a challenging task. The existing methods, proposed for the single label case, such as the EM-type algorithm (Bekker *et al*., 2019), require multiple iterations and are too computationally demanding in the context of the multi-label case and applying $\ell_{2,1}$ regularization. To deal with this problem, we propose two simple yet effective solutions. In the first one, we use inequality $s_k(\mathbf{x}) \leq e_k(\mathbf{x}) \leq 1$ and simply replace $e_k(\mathbf{x})$ by $0.5(1 + s_k(\mathbf{x}))$, which is the average of the two ends of the interval $[s_k(\mathbf{x}), 1]$. The other possibility is to simply assume that $e_k(\mathbf{x}) = c$ and, using inequality $P(s_k = 1) \leq c \leq 1$, replace $e_k(\mathbf{x})$ by $0.5(1 + P(s_k = 1))$. This leads to the following estimators of weights. Since, $s_k(\mathbf{x})$ can be relatively easily estimated (using the naive approach), we express the weights in terms of $s_k(\mathbf{x})$. We have

$$w_k^{(1)}(\mathbf{x}) := \begin{cases} 1 - \dfrac{s_k(\mathbf{x})}{0.5(1 + s_k(\mathbf{x}))} = \dfrac{1 - s_k(\mathbf{x})}{1 + s_k(\mathbf{x})} & \text{if } s_k = 0, \\ 1 & \text{if } s_k = 1 \end{cases}$$

and

$$w_k^{(2)}(\mathbf{x}) := \begin{cases} \max(0, 1 - \dfrac{2s_k(\mathbf{x})}{1 + P(s_k = 1)}) & \text{if } s_k = 0, \\ 1 & \text{if } s_k = 1. \end{cases}$$

In the latter case we use the max function to ensure the positiveness of the weights. In the proposed method, we first estimate $s_k(\mathbf{x})$ by using the naive method (we compute $\hat{A}^{(0)}$) and determine the above weights. Probability $P(s_k = 1)$ is estimated as a fraction of observations for which $s_k = 1$. The whole procedure consists of two steps: in the first step we apply the naive approach to estimate the weights and then we compute $\hat{A}^{(1)}$.

The other natural approach is to consider

$$\begin{aligned} w_k(\mathbf{x}) &= P(y_k = 0|\mathbf{x}, s_k = 0) \\ &= 1 - P(y_k = 1|\mathbf{x}, s_k = 0) \quad (4) \\ &= 1 - \frac{P(s_k = 0|\mathbf{x}, y_k = 1) P(y_k = 1|\mathbf{x})}{P(s_k = 0|\mathbf{x})} \quad (5) \\ &= 1 - \frac{1 - e_k(\mathbf{x})}{e_k(\mathbf{x})} \frac{s_k(\mathbf{x})}{1 - s_k(\mathbf{x})}, \end{aligned}$$

where (4) follows from the Bayes theorem and (5) follows from (1). In much the name case as in the case of the above weights, we can replace $e_k(\mathbf{x})$ by $0.5(1 + s_k(\mathbf{x}))$ or by $0.5(1 + P(s_k = 1))$, which leads to another definition of weights,

$$w_k^{(3)}(\mathbf{x}) := \begin{cases} \dfrac{1}{1 + s_k(\mathbf{x})} & \text{if } s_k = 0, \\ 1 & \text{if } s_k = 1 \end{cases}$$

and

$$w_k^{(4)}(\mathbf{x}) := \begin{cases} 1 - \dfrac{1 - P(s_k = 1)}{1 + P(s_k = 1)} \dfrac{s_k(\mathbf{x})}{1 - s_k(\mathbf{x})} & \text{if } s_k = 0, \\ 1 & \text{if } s_k = 1. \end{cases}$$

Figure 1 shows weights as a function of $s_k(\mathbf{x})$. As expected, all weights decrease with $s_k(\mathbf{x})$. In the case of the weights of Methods 1 and 2 we usually assign smaller values than in the case of the weights of Methods 3 and 4. In addition, the weights of Methods 2 and 4 depend on $P(s_k = 1)$.

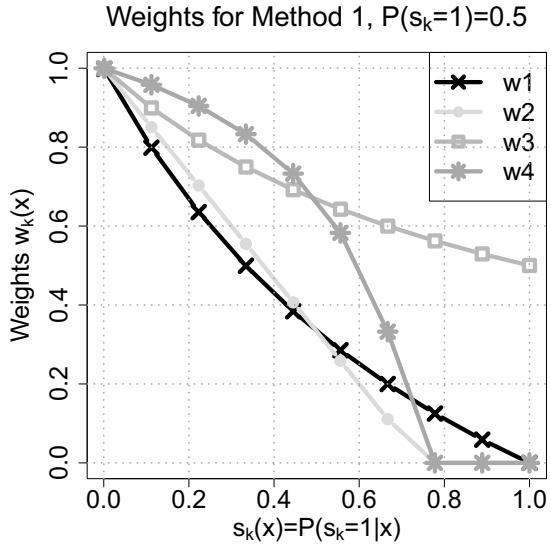

Fig. 1. Weights of observations for Method 1 with respect to $P(s_k = 1|\mathbf{x})$, for $P(s_k = 1) = 0.5$.

4. Method 2: Approximating the risk for true target variables

We consider the theoretical risk corresponding to the true unobserved target variables

$$R(A) = E_{\mathbf{x},\mathbf{y}} \sum_{k=1}^{K} l(a_k^T\mathbf{x}, y_k) = \sum_{k=1}^{K} E_{\mathbf{x},y_k} l(a_k^T\mathbf{x}, y_k). \tag{6}$$

The empirical version of the above risk function cannot be directly optimized as we do not observe $\mathbf{y}$. However, the following result shows that $R(A)$ can be written in an alternative form which will be used in our method. We consider loss functions that can be decomposed as

$$l(a_k^T\mathbf{x}, y_k) = \begin{cases} l^+(a_k^T\mathbf{x}) & \text{if } y_k = 1, \\ l^-(a_k^T\mathbf{x}) & \text{if } y_k = 0, \end{cases} \tag{7}$$

where $l^+(\cdot)$ and $l^-(\cdot)$ are losses for positive and negative examples, respectively.

Many popular loss functions can be represented as (7). For example, in the case of the log loss (cross entropy loss) $l(\hat{y}, y) = -[y\log(\sigma(\hat{y})) + (1-y)\log(1-\sigma(\hat{y}))]$ we have $l^+(\hat{y}) = -\log[\sigma(\hat{y})]$ and $l^-(\hat{y}) = -\log[1-\sigma(\hat{y})]$. Moreover, define an auxiliary loss

$$\tilde{l}(a_k^T\mathbf{x}, s_k) = \begin{cases} l^+(a_k^T\mathbf{x}) & \text{if } s_k = 1, \\ w_k(\mathbf{x})l^-(a_k^T\mathbf{x}) + (1-w_k(\mathbf{x}))l^+(a_k^T\mathbf{x}) & \text{if } s_k = 0, \end{cases}$$

where weights $w_k(\mathbf{x}) = P(y_k = 0|s_k = 0, \mathbf{x})$ were already considered in Method 1. The following result will be crucial for Method 2. We will prove that the theoretical risk for the k-th target variable can be expressed as the expected value of the auxiliary loss, where the expectation is taken with respect to $(\mathbf{x}, s_k)$.

Theorem 1. *The following equality holds:*

$$E_{\mathbf{x},y_k} l(a_k^T\mathbf{x}, y_k) = E_{\mathbf{x},s_k}\tilde{l}(a_k^T\mathbf{x}, s_k).$$

Proof. Using the Law of Total Probability and the facts that $P(s_k = 1, y_k = 0|\mathbf{x}) = 0$ and $P(s_k = 1, y_k = 1|\mathbf{x}) = P(s_k = 1|\mathbf{x})$, we have

$$\begin{aligned} &E_{y_k|\mathbf{x}} l(a_k^T\mathbf{x}, y_k) \\ &\quad = P(s_k = 1|\mathbf{x})l^+(a_k^T\mathbf{x}) \\ &\qquad + P(s_k = 0, y_k = 0|\mathbf{x})l^-(a_k^T\mathbf{x}) \\ &\qquad + P(s_k = 0, y_k = 1|\mathbf{x})l^+(a_k^T\mathbf{x}) \\ &\quad = P(s_k = 1|\mathbf{x})l^+(a_k^T\mathbf{x}) \\ &\qquad + w_k(\mathbf{x})P(s_k = 0|\mathbf{x})l^-(a_k^T\mathbf{x}) \\ &\qquad + (1 - w_k(\mathbf{x}))P(s_k = 0|\mathbf{x})l^+(a_k^T\mathbf{x}) \\ &\quad = E_{s_k|\mathbf{x}}\tilde{l}(a_k^T\mathbf{x}, s_k). \end{aligned}$$

Taking the expectation with respect to $\mathbf{x}$ yields the desired assertion. ∎

Importantly, the empirical version of the expression given in Theorem 1 can be directly optimized. Thus, using Theorem 1, the theoretical risk (6) and its empirical version can be written as

$$R(A) = \sum_{k=1}^{K} E_{\mathbf{x},s_k}\tilde{l}(a_k^T\mathbf{x}, s_k),$$

$$\hat{R}(A) = \sum_{k=1}^{K} \frac{1}{n}\sum_{i=1}^{n} \tilde{l}(a_k^T\mathbf{x}^{(i)}, s_k^{(i)}),$$

respectively.

Observe that $\tilde{l}(a_k^T\mathbf{x}^{(i)}, s_k^{(i)})$ depends on $e_k(\mathbf{x})$. Therefore, as in Method 1, we first estimate $s_k(\mathbf{x})$ by using the naive method and then estimate $e_k(\mathbf{x})$ as $0.5(1+s_k(\mathbf{x}))$. In the next step, we compute the matrix of parameters as

$$\hat{A} := \arg\min_A \left[\hat{R}(A) + \lambda\sum_{j=1}^{p} ||A_j||_2\right]. \tag{8}$$

Note that it is equivalent to solving $\hat{A} := \arg\min_{A\in\mathcal{A}} \hat{R}(A)$, where class $\mathcal{A} := \{A \in \mathbb{R}^{K\times p} : \sum_{j=1}^{p} ||A_j||_2 \le \Lambda\}$ and Λ is related to λ.

In the following we will bound the excess risk for $\hat{A}$ using the Rademacher complexity bounds. The multi-label Rademacher complexity is defined as

$$\mathcal{R}_D(\mathcal{A}) := E_\epsilon\left[\frac{1}{n}\sup_{A\in\mathcal{A}}\sum_{i=1}^{n}\sum_{k=1}^{K}\epsilon_i^k\left(a_k^T\mathbf{x}^{(i)}\right)\right].$$

Here, the expectation is over ϵ_i^k, which are i.i.d. Rademacher random variables, i.e., $P(\epsilon_i^k = 1) = P(\epsilon_i^k = -1) = 0.5$. It has been already shown (Kakade *et al*., 2012) that the multi-label Rademacher complexity for class $\mathcal{A}$ can be bounded by

$$\mathcal{R}_D(\mathcal{A}) \leq O\left(\Lambda B\sqrt{\frac{K\log(p)}{n}}\right), \tag{9}$$

where $D = \{(\mathbf{x}^{(i)}, \mathbf{s}^{(i)}) : i = 1, \ldots, n\}$ and $B > 0$ is a constant, such that $\max_i ||\mathbf{x}^{(i)}||_\infty < B$. To produce a bound to the excess risk, it is necessary to know that the loss function is Lipschitz with respect to its first argument. It is very easy to check that when both $l^+()$ and $l^-()$ are Lipschitz with some constant ρ then $\tilde{l}(\cdot,\cdot)$ is Lipschitz with constant 2ρ with respect to its first argument, as $w_k(\mathbf{x}) \leq 1$. For example, in the case of the log loss, it is easy to check that $l^+()$ and $l^-()$ are Lipschitz with constant 1. In the following result we bound the (expected) excess risk. Let $A^* := \arg\min_{A\in\mathcal{A}} R(A)$ be the optimal solution corresponding to the theoretical risk.

Theorem 2. *Assume that* $\max_i ||\mathbf{x}^{(i)}||_\infty < B$ *and* $l^+(\cdot)$ *and* $l^-(\cdot)$ *are* ρ*-Lipschitz. Then we have*

$$E_D[R(\hat{A}) - R(A^*)] \leq O\left(2\rho\Lambda B\sqrt{\frac{K\log(p)}{n}}\right).$$

Proof. Note that

$$\begin{aligned} &E_D[R(\hat{A}) - R(A^*)] \\ &\quad\leq E_D[R(\hat{A}) - \hat{R}(\hat{A})] \\ &\quad\leq E_D \sup_{A\in\mathcal{A}}[R(A) - \hat{R}(A)], \end{aligned}$$

where the first inequality follows from the fact that $R(A^*) = E_D\hat{R}(A^*) \geq E_D\hat{R}(\hat{A})$, as $\hat{A}$ minimizes the empirical risk $\hat{R}$ within class $\mathcal{A}$. Next, it follows from adaptation of Lemma 26.2 by Shalev-Shwartz and Ben-David (2013) to the multi-label case and the fact that $\tilde{l}(\cdot,\cdot)$ is Lipschitz with constant 2ρ that the last term is bounded by $2\rho E_D\mathcal{R}_D(\mathcal{A})$. This, combined with (9), yields the desired assertion. ■

5. Experiments

In the experiments we compare the methods described in the previous sections: the naive method, Method 1 and Method 2. We also consider the oracle method which assumes the full knowledge of target variables. The oracle method serves as a reference method and obviously it cannot be used in practise for PU-ML data. In the experiments we investigate how much we lose compared with the oracle method and how much the proposed methods improve the prediction accuracy of the naive approach. We also explore the impact of various labelling schemes and the choice of optimal weights in Method 1.

5.1. Data sets. We consider two artificial data sets (called 'Artificial 1' and 'Artificial 2'). The main advantage of using artificial data sets is that we know in advance which features are relevant and which are noisy. Therefore, in addition to the classification performance, we can also assess the quality of feature ranking. We use the following method of generating artificial data sets. We first generate the feature vector $\mathbf{x} \sim N(0, I)$, where I is the $p \times p$ identity matrix. Then we generate the true target variables from the Bernoulli distribution with probability of success

$$P(y_k = 1|\mathbf{x}) = \sigma(\mathbf{x}_{T_k}^T\beta_{T_k}),$$

where $\sigma(\cdot)$ is a sigmoid activation function $\sigma(s) = 1/(1+\exp(-s))$, $T_k \subseteq \{1, \ldots, p\}$ is a set of the indices corresponding to the relevant features for the k-th target variable, $\mathbf{x}_{T_k}$ is a subvector of $\mathbf{x}$ corresponding to features from T_k and β is a parameter vector.

Artificial data set 1. We consider $K = 5$, $n = 500$, $p = 50$ features and define sets of relevant variables as $T_k = \{1, \ldots, k\}$, where $k = 1, \ldots, K$. We consider $\beta_{T_k} = (1, \ldots, 1)^T$. Observe that in such scenario, y_1 is affected by only one feature, whereas y_K by K features. The first feature is relevant for all target variables considered, the second feature is relevant for all but one target variable, etc. The K-th feature is only relevant for the K-th target variable. Features $\{K+1, \ldots, p\}$ are not relevant they only serve as noisy features to make; the feature selection task more challenging.

Artificial data set 2. We consider $K = 5$, $n = 500$, $p = 50$ features and define sets of relevant variables as $T_k = \{1, \ldots, 5k\}$, where $k = 1, \ldots, K$ and $\beta_{T_k} = (0.5, \ldots, 0.5)^T$. Features $\{5K+1, \ldots, p\}$ are not relevant they only serve as noisy features. This data set contains 25 relevant variables.

Real multi-label data sets. We also consider 4 real data sets, representing different domains, from the MULAN repository (Tsoumakas *et al*., 2011): music, scene, yeast and genbase. The average label densities (fractions of active labels) are 31%, 17%, 39% and 10%, respectively.

We have chosen data sets for which there is a significant gap in performance between the oracle and the naive methods as for such data sets there is room for an improvement. As in other related studies (see, e.g., Kanehira and Harada, 2016; Bekker and Davis, 2018) we created PU data sets from the original completely labelled data sets. In this way, we can control the labelling mechanism. We used two labelling schemes described below.

5.2. Labelling schemes. We consider two methods of generating observed target variables $s_1, \ldots, s_K$ based on the true target variables $y_1, \ldots, y_K$. The first scheme

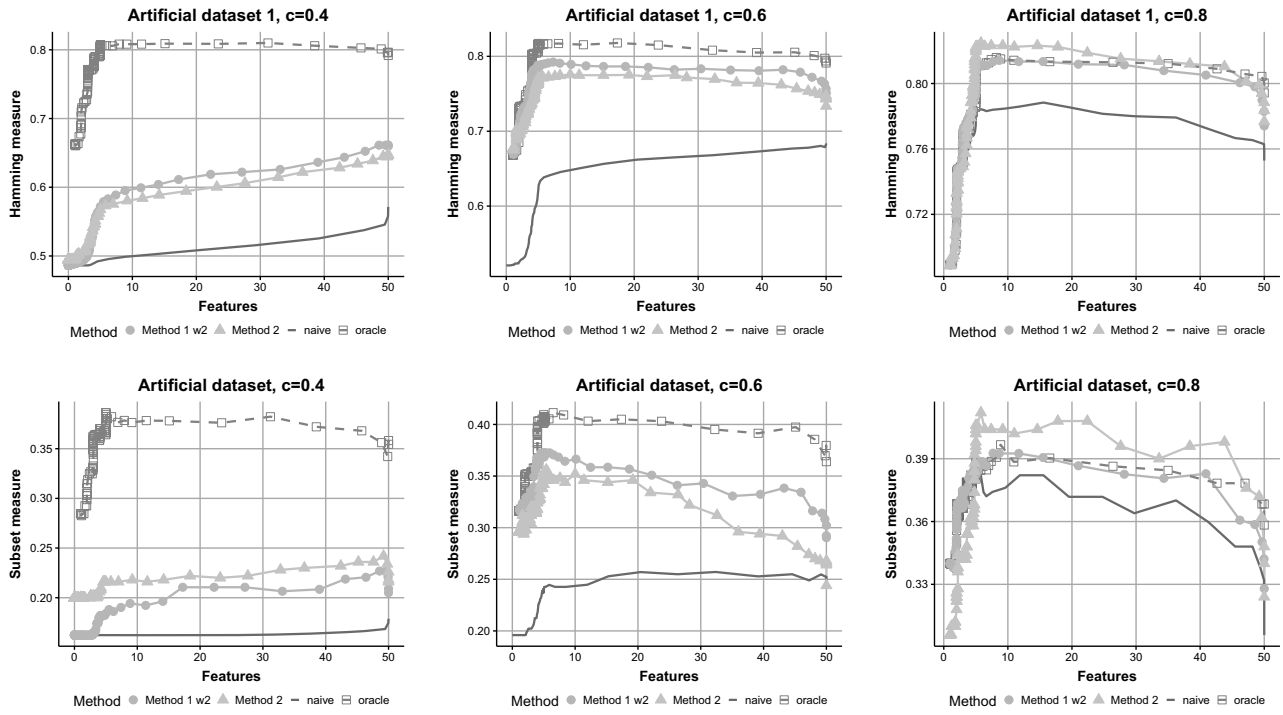

Fig. 2. Hamming and subset measures with respect to the number of selected features for the first artificial data set and different values of parameter c (labelling scheme 1).

corresponds to the SCAR assumption whereas for the second scheme the SCAR is not satisfied.

Scheme 1: If $y_k = 0$ then we set $s_k = 0$. When $y_k = 1$, we draw s_k from the Bernoulli distribution with probability of success $e_k(\mathbf{x}) = P(s_k = 1|y_k = 1, \mathbf{x}) = c$, where c is a parameter which varies in simulations. We consider $c = 0.4, 0.6, 0.8$.

Scheme 2: If $y_k = 0$ then we set $s_k = 0$. When $y_k = 1$, we draw s_k from the Bernoulli distribution with probability of success $e_k(\mathbf{x}) = P(s_k = 1|y_k = 1, \mathbf{x}) = \sigma(b\mathbf{x}^T\mathbf{e})$, where $\mathbf{e} = (1, \ldots, 1)^T$ and $b \in \mathbb{R}$ is a parameter which varies in simulations. For larger b, the dependence between $\mathbf{x}$ and $\mathbf{s}$ increases, whereas for $b = 0$, we have $P(s_k = 1|y_k = 1, \mathbf{x}) = 0.5$ which corresponds to Scheme 1. Note that $e_k(\mathbf{x})$ is a function of all the features considered.

5.3. Evaluation measures. We present the results for two popular evaluation measures: the subset accuracy and the Hamming measure. Importantly, the conclusions remain the same for other measures, e.g., the F-measure.[1] Below we recall definitions of the measures considered. Let $\mathbf{y} = (y_1, \ldots, y_K)$ be a vector of true labels and $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_K)$ be the vector of predicted labels for some instance $\mathbf{x}$. The subset accuracy for a pair $(\mathbf{y}, \hat{\mathbf{y}})$ is defined as $I(\mathbf{y} = \hat{\mathbf{y}})$, which is directly related to a subset loss. It measures the correctness of joint prediction for all labels and is restrictive, especially when the number of labels K is large. It may happen that the subset accuracy is zero whereas other measures achieve significantly higher values. The Hamming measure, defined as $\frac{1}{K}\sum_{k=1}^{K} I(y_k = \hat{y}_k)$, is the average number of correct predictions. Both the above measures are averaged over all instances in testing data.

[1]The results for other evaluation measures are available in the supplement: `https://github.com/teisseyrep/mlpu_erm`

5.4. Results. We first explore how the classification accuracy of the methods considered depends on the number of features included in the model. The number of selected features is controlled by the regularization parameter λ, which varies between $\lambda_{\max}$ (for which all parameters are zero) and $\lambda_{\min} = 0.0001\lambda_{\max}$. For larger λ, the $K \times p$ matrix of estimated parameters (e.g., $\hat{A}^{(1)}$ for Method 1 or $\hat{A}$ for Method 2) will be sparse, which means that most of its columns will be zero and only the columns corresponding to the most relevant features will contain non-zero values. For small λ we include much more features in the model.

Figures 2–5 show how the evaluation measures depend on the number of features included in the model for Scheme 1 (the results are averaged over 10

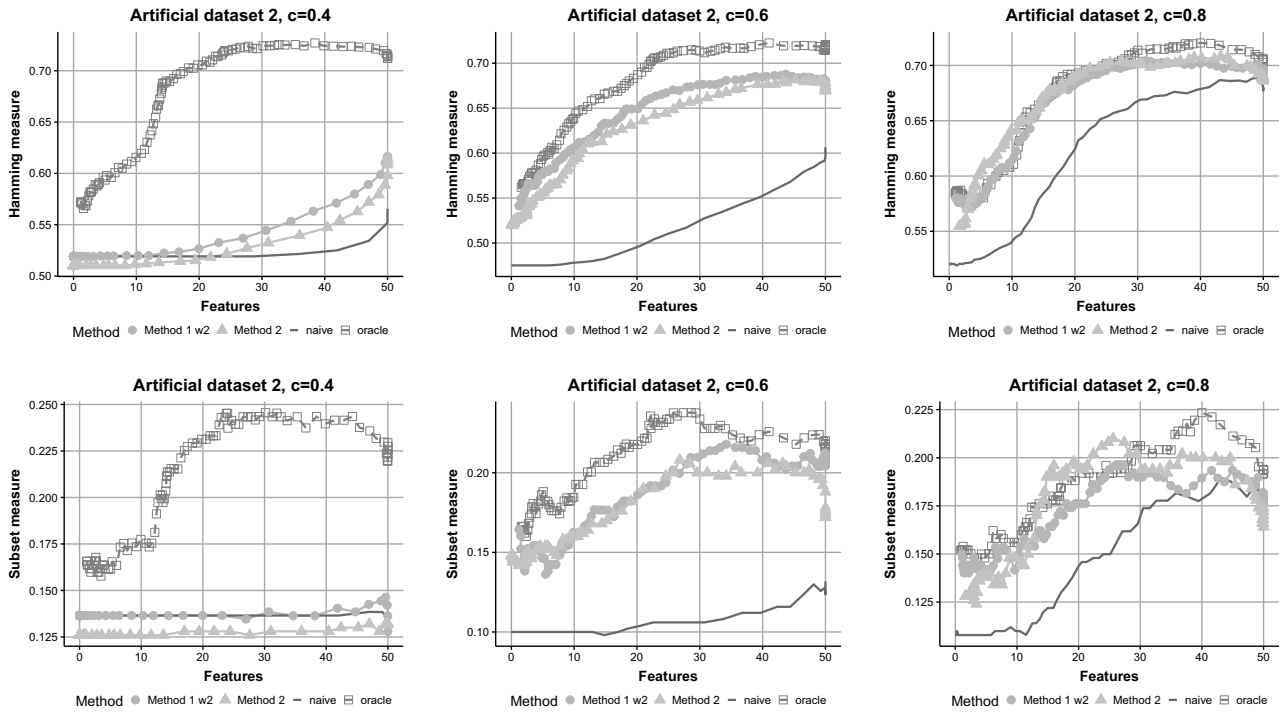

Fig. 3. Hamming and subset measures with respect to the number of selected features for the second artificial data set and different values of parameter c (labelling scheme 1).

cross-validation folds). For better clarity, in the case of Method 1, we only present the curves corresponding to weights $w_k^{(2)}$ as this variant usually outperforms the remaining ones. As expected, we observe the highest accuracy for the oracle method and the smallest for the naive method and the difference between these two methods is significant, especially for small c. Parameter c can be interpreted as a measure of difficulty of the PU-ML problem. Indeed, for small c, only for a small fraction of observations the target variables are assigned labels.

The poor performance of the naive method is associated with the fact that this method underestimates the posterior probability $P(y_k = 1|\mathbf{x})$ and the bias increases with decreasing c. The accuracy of the methods increases with c, as for larger c we are approaching the case of full data observability. For larger c, the differences between the methods considered and the oracle method become non significant, which suggests that the proposed methods can be successfully applied when the discrepancy between the true target variables and the observed ones is not very pronounced. In some cases, the proposed methods achieve even larger accuracies than the oracle method, see, e.g., Fig. 2 for $c = 0.8$.

Interestingly, for the artificial data, we can observe that all curves reach first plateau after including the first five features, which indicates that the methods rank the relevant features correctly as this data set contains five relevant features. The relevant features appear earlier on the regularization path than the noisy features. We observe this property even for the naive method when $c = 0.6$ or $c = 0.8$. Note, however, that although the naive method selects the relevant features before noisy ones, its accuracy is much lower when compared with the remaining methods, which is due to the biased estimation of the posterior probability. The above-mentioned effect is not so pronounced for the second artificial data set which contains much more relevant features and ranking them correctly is more challenging. In this case, only the oracle method clearly stabilizes after including top 25 features. For some real data sets (e.g., music), we observe that the curves decrease for a small value of λ which is related to over-fitting.

Tables 1–4 contain the values of the measures for Schemes 1 and 2. The results are averaged over $c = 0.4, 0.6, 0.8$ (for Scheme 1) and $b = 0.5, 1, 2$ (for Scheme 2), 100 values of λ and 10 CV folds. The winner method (among all methods considered except the oracle method) is in bold face. Although Method 1 has a deeper theoretical justification, Method 1 with weights $w_k^{(2)}$ usually outperforms other methods, whereas Method 2 is the second best. Importantly, the proposed methods work significantly better than the naive method (which is confirmed by the t-test), also for Scheme 2, when the SCAR assumption is not satisfied, which indicates

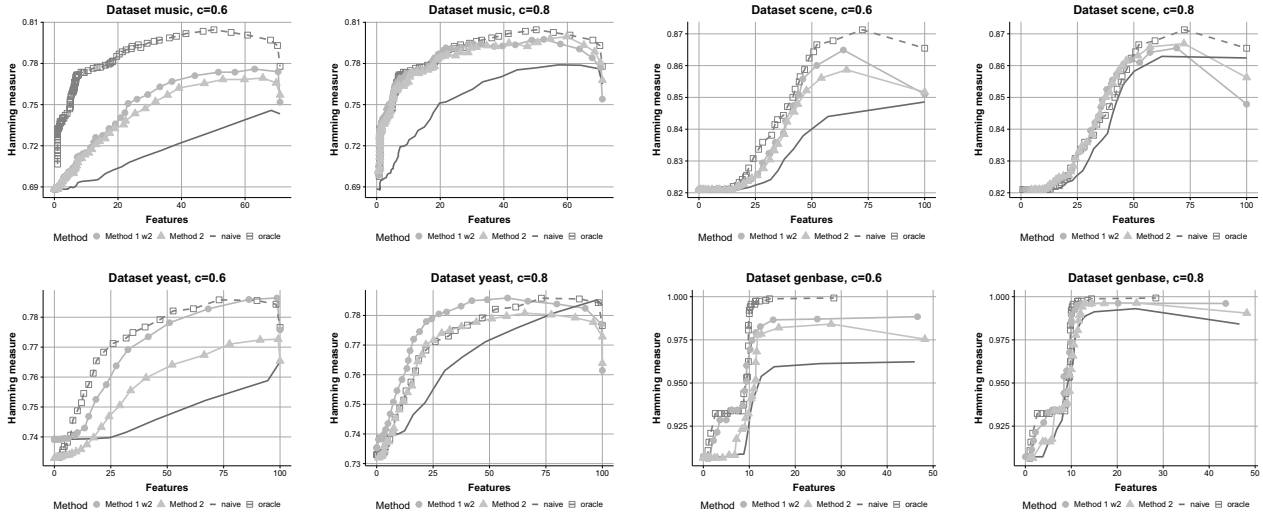

Fig. 4. Hamming measure with respect to the number of selected features for real data sets (labelling scheme 1).

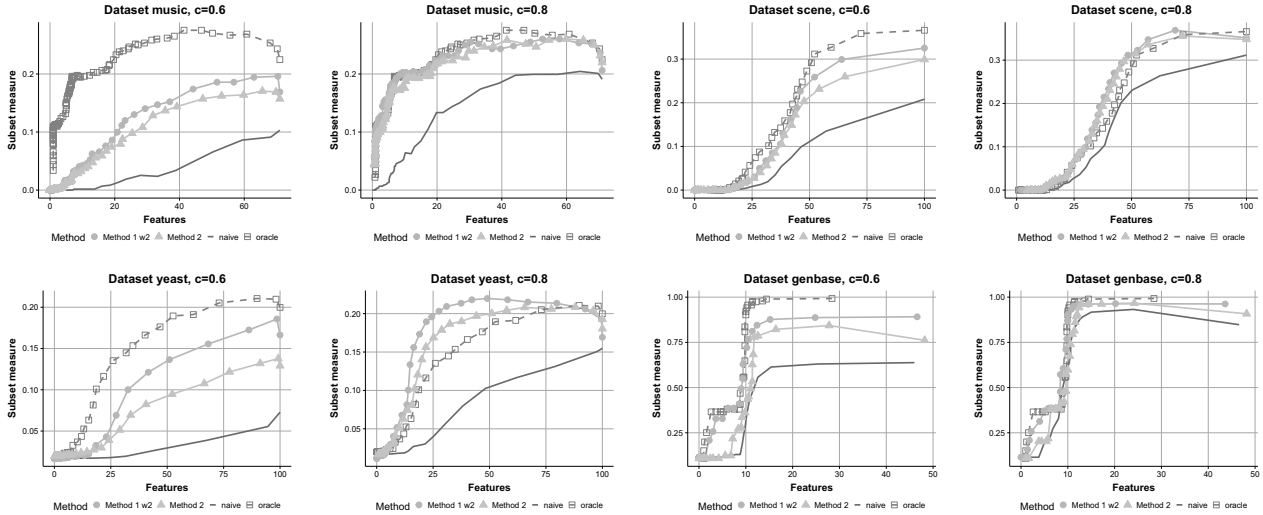

Fig. 5. Subset accuracy with respect to the number of selected features for real data sets (labelling scheme 1).

that they are quite robust against departures from the SCAR assumption.

6. Conclusions

We presented a study of two embedded feature selection procedures. In both the approaches we estimate the weights of observations using the naive method in the first step, and solve the weighted penalized risk minimization problem in the second step. We give a theoretical justification for Method 2, namely the empirical risk in Method 2 corresponds to the theoretical risk of the oracle method. Moreover, we bound the (expected) excess risk for Method 2. It follows from the experiments that Method 1 with weights $w_k^{(2)}$ and Method 2 have the largest predictive power among the methods considered. The methods work properly for various labelling schemes including the cases in which the propensity score function is not constant.

There are still interesting issues for future research. The problem of the propensity score estimation is important and worth further investigations. Although the proposed simple method of the propensity score estimation works effectively and is very fast, we believe that there is still room for improvements. Accurate estimation of $e_k(\mathbf{x})$ could improve the performance of both proposed methods. Combining the proposed empirical risk functions with other forms of regularization

Table 1. Hamming measure for labelling scheme 1 averaged over different $c = 0.4, 0.6, 0.8$, different values of λ and 10-cross-validation folds. The winner method (among all methods considered except the oracle method) is in boldface. The star * denotes that the method is significantly better than the naive method according to the t-test (for $\alpha = 0.05$).

	oracle	naive	Method 1 w1	Method 1 w2	Method 1 w3	Method 1 w4	Method 2
Art1	0.757 ± 0.055	0.598 ± 0.037	0.669 ± 0.049	$\mathbf{0.673 \pm 0.052}$ *	0.627 ± 0.048	0.656 ± 0.047	0.670 ± 0.048
Art2	0.658 ± 0.059	0.528 ± 0.036	0.596 ± 0.044	$\mathbf{0.600 \pm 0.044}$ *	0.562 ± 0.041	0.572 ± 0.049	0.586 ± 0.047
music	0.756 ± 0.03	0.696 ± 0.014	0.713 ± 0.02	$\mathbf{0.716 \pm 0.021}$ *	0.702 ± 0.018	0.708 ± 0.021	0.715 ± 0.02
scene	0.837 ± 0.017	0.825 ± 0.007	0.828 ± 0.011	0.829 ± 0.012	0.826 ± 0.009	0.828 ± 0.011	$\mathbf{0.829 \pm 0.011}$ *
yeast	0.752 ± 0.02	0.742 ± 0.007	0.748 ± 0.013	$\mathbf{0.749 \pm 0.014}$ *	0.745 ± 0.01	0.747 ± 0.012	0.736 ± 0.013
genbase	0.951 ± 0.036	0.918 ± 0.018	0.926 ± 0.025	$\mathbf{0.928 \pm 0.028}$ *	0.922 ± 0.021	0.927 ± 0.026	0.928 ± 0.025
avg. rank	1.0	6.8	3.8	**2.2**	5.8	4.7	3.7

Table 2. Subset measure for labelling scheme 1 averaged over different $c = 0.4, 0.6, 0.8$, different values of λ and 10-cross-validation folds. The winner method (among all methods considered except the oracle method) is in boldface. The star * denotes that the method is significantly better than the naive method according to the t-test (for $\alpha = 0.05$).

	oracle	naive	Method 1 w1	Method 1 w2	Method 1 w3	Method 1 w4	Method 2
Art1	0.360 ± 0.034	0.245 ± 0.017	0.289 ± 0.022	$\mathbf{0.292 \pm 0.025}$ *	0.260 ± 0.021	0.280 ± 0.021	0.289 ± 0.027
Art2	0.195 ± 0.028	0.123 ± 0.014	0.157 ± 0.017	$\mathbf{0.161 \pm 0.017}$ *	0.138 ± 0.014	0.143 ± 0.018	0.153 ± 0.018
music	0.243 ± 0.022	0.040 ± 0.044	0.123 ± 0.04	$\mathbf{0.136 \pm 0.037}$ *	0.075 ± 0.05	0.107 ± 0.047	0.125 ± 0.034
scene	0.272 ± 0.063	0.064 ± 0.069	0.145 ± 0.089	$\mathbf{0.166 \pm 0.091}$ *	0.096 ± 0.08	0.140 ± 0.093	0.157 ± 0.077
yeast	0.076 ± 0.072	0.024 ± 0.018	0.048 ± 0.046	$\mathbf{0.052 \pm 0.05}$ *	0.032 ± 0.03	0.040 ± 0.04	0.049 ± 0.042
genbase	0.539 ± 0.34	0.223 ± 0.168	0.294 ± 0.238	$\mathbf{0.319 \pm 0.267}$ *	0.255 ± 0.201	0.307 ± 0.247	0.314 ± 0.24
avg. rank	1.0	7.0	4.0	**2.0**	6.0	4.8	3.2

as well as with classical information criteria for feature selection (such as AIC, BIC or EBIC) is also worth studying.

References

Argyriou, A., Evgeniou, T. and Pontil, M. (2008). Convex multi-task feature learning, *Machine Learning* **73**(3): 243–272.

Bekker, J. and Davis, J. (2018). Estimating the class prior in positive and unlabeled data through decision tree induction, *Proceedings of the 32nd AAAI Conference on Artificial Intelligence, New Orleans, USA*, pp. 1–8.

Bekker, J. and Davis, J. (2020). Learning from positive and unlabeled data: A survey, *Machine Learning* **109**(4): 719–760.

Bekker, J., Robberechts, P. and Davis, J. (2019). Beyond the selected completely at random assumption for learning from positive and unlabeled data, *in* U. Brefeld *et al.* (Eds), *Proceedings of the 2019 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, Springer, Cham, pp. 71–85.

Biecek, P. (2018). DALEX: Explainers for complex predictive models in R, *Journal of Machine Learning Research* **19**(1): 3245–3249.

Bucak, S.S., Jin, R. and Jain, A.K. (2011). Multi-label learning with incomplete class assignments, *Proceedings of the Conference on Computer Vision and Pattern Recognition, Colorado Springs, USA*, pp. 2801–2808.

Couso, I., Dubois, D. and Hüllermeier, E. (2017). Maximum likelihood estimation and coarse data, *Proceedings of the International Conference on Scalable Uncertainty Management, Granada, Spain*, pp. 3–16.

Dembczyński, K., Waegeman, W., Cheng, W. and Hüllermeier, E. (2012). On label dependence and loss minimization in multi-label classification, *Machine Learning* **88**(1): 5–45.

Elkan, C. and Noto, K. (2008). Learning classifiers from only positive and unlabeled data, *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'08, Las Vegas, USA*, pp. 213–220.

Frenay, B. and Verleysen, M. (2014). Classification in the presence of label noise: A survey, *IEEE Transactions on Neural Networks and Learning Systems* **25**(5): 845–869.

Gibaja, E. and Ventura, S. (2015). A tutorial on multilabel learning, *ACM Computing Surveys* **47**(3): 1–38.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection, *Journal of Machine Learning Research* **3**(1): 1157–1182.

Hall, E.J. and Brenner, D.J. (2008). Cancer risks from diagnostic radiology, *British Journal of Radiology* **81**(965): 362–378.

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, New York.

Hazell, L. and Shakir, S. (2006). Under-reporting of adverse drug reactions: A systematic review, *Drug Safety* **29**(5): 385–396.

He, Z.-F., Yang, M., Gao, Y., Liu, H.-D. and Yin, Y. (2019). Joint multi-label classification and label correlations with missing labels and feature selection, *Knowledge-Based Systems* **163**(1): 145–158.

Heitjan, D.F. and Rubin, D.B. (1991). Ignorability and coarse data, *Annals of Statistics* **19**(4): 2244–2253.

Jain, S., White, M. and Radivojac, P. (2016). Estimating the class prior and posterior from noisy positives and unlabeled

Table 3. Hamming measure for labelling scheme 2 averaged over different $b = 0.5, 1, 2$, different values of λ and 10-cross-validation folds. The winner method (among all methods considered except the oracle method) is in boldface. The star * denotes that the method is significantly better than the naive method according to the t-test (for $\alpha = 0.05$).

	oracle	naive	Method 1 w1	Method 1 w2	Method 1 w3	Method 1 w4	Method 2
Art1	0.760 ± 0.053	0.589 ± 0.076	0.733 ± 0.034	$\mathbf{0.736 \pm 0.035}$ *	0.674 ± 0.047	0.711 ± 0.031	0.704 ± 0.036
Art2	0.665 ± 0.056	0.545 ± 0.059	0.614 ± 0.064	$\mathbf{0.618 \pm 0.063}$ *	0.567 ± 0.07	0.583 ± 0.073	0.615 ± 0.061
music	0.756 ± 0.03	0.692 ± 0.01	0.695 ± 0.013	0.695 ± 0.013	0.694 ± 0.012	0.695 ± 0.013	$\mathbf{0.696 \pm 0.013}$ *
scene	0.837 ± 0.017	0.826 ± 0.009	0.828 ± 0.011	0.828 ± 0.011	0.827 ± 0.01	0.827 ± 0.01	$\mathbf{0.829 \pm 0.011}$
yeast	0.745 ± 0.019	0.742 ± 0.007	0.743 ± 0.007	$\mathbf{0.743 \pm 0.007}$ *	0.743 ± 0.007	0.743 ± 0.007	0.729 ± 0.007
genbase	0.952 ± 0.036	0.914 ± 0.011	0.914 ± 0.012	0.914 ± 0.012	0.914 ± 0.011	0.914 ± 0.012	$\mathbf{0.916 \pm 0.013}$
avg. rank	1.0	6.8	3.8	**2.5**	5.8	4.5	3.5

Table 4. Subset measure for labelling scheme 2 averaged over different $b = 0.5, 1, 2$, different values of λ and 10-cross-validation folds. The winner method (among all methods considered except the oracle method) is in boldface. The star * denotes that the method is significantly better than the naive method according to the t-test (for $\alpha = 0.05$).

	oracle	naive	Method 1 w1	Method 1 w2	Method 1 w3	Method 1 w4	Method 2
Art1	0.369 ± 0.033	0.215 ± 0.038	0.338 ± 0.026	$\mathbf{0.340 \pm 0.027}$ *	0.279 ± 0.024	0.315 ± 0.02	0.307 ± 0.025
Art2	0.215 ± 0.029	0.141 ± 0.024	0.178 ± 0.032	$\mathbf{0.181 \pm 0.031}$ *	0.151 ± 0.032	0.158 ± 0.035	0.173 ± 0.028
music	0.244 ± 0.022	0.032 ± 0.026	0.054 ± 0.026	$\mathbf{0.057 \pm 0.025}$ *	0.042 ± 0.029	0.050 ± 0.028	0.057 ± 0.026
scene	0.237 ± 0.082	0.078 ± 0.067	0.104 ± 0.068	0.108 ± 0.066	0.090 ± 0.069	0.101 ± 0.07	$\mathbf{0.119 \pm 0.062}$*
yeast	0.056 ± 0.063	0.028 ± 0.027	0.033 ± 0.033	0.034 ± 0.033	0.030 ± 0.03	0.032 ± 0.032	$\mathbf{0.036 \pm 0.031}$ *
genbase	0.543 ± 0.339	0.167 ± 0.091	0.175 ± 0.101	0.176 ± 0.102	0.170 ± 0.095	0.175 ± 0.1	$\mathbf{0.187 \pm 0.112}$
avg. rank	1.0	7.0	3.8	**2.5**	6.0	4.7	3.0

data, *Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain*, pp. 2693–2701.

Jaskie, K., Elkan, C. and Spanias, A. (2020). A modified logistic regression for positive and unlabeld learning, *53rd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, USA*, pp. 2007–2011.

Ji, S., Tang, L., Yu, S. and Ye, J. (2010). A shared-subspace learning framework for multi-label classification, *ACM Transactions on Knowledge Discovery from Data* **4**(2): 1–29.

Kakade, S.M., Shalev-Shwartz, S. and Tewari, A. (2012). Regularization techniques for learning with matrices, *Journal of Machine Learning Research* **13**(1): 1865–1890.

Kanehira, A. and Harada, T. (2016). Multi-label ranking from positive and unlabeled data, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA*, pp. 5138–5146.

Kashef, S., Nezamabadi-pour, H. and Nikpour, B. (2018). Multilabel feature selection: A comprehensive review and guiding experiments, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **8**(2): 1–29.

Lagasse, R.S. (2002). Anesthesia safety: Model or myth?: A review of the published literature and analysis of current original data, *Anesthesiology: The Journal of the American Society of Anesthesiologists* **97**(6): 1609–1617.

Łazęcka, M., Mielniczuk, J. and Teisseyre, P. (2021). Estimating the class prior for positive and unlabelled data via logistic regression, *Advances in Data Analysis and Classification* **15**(4): 1039–1068.

Lee, J. and Kim, D.-W. (2017). SCLS: Multi-label feature selection based on scalable criterion for large label set, *Pattern Recognition* **66**(1): 342–352.

Natarajan, N., Dhillon, I.S., Ravikumar, P. and Tewari, A. (2013). Learning with noisy labels, *Proceedings of the 26th International Conference on Neural Information Processing Systems, NIPS'13, Lake Tahoe, USA*, pp. 1196–1204.

Naula, P., Airola, A., Salakoski, T. and Pahikkala, T. (2014). Multi-label learning under feature extraction budgets, *Pattern Recognition Letters* **40**(1): 56–65.

Pereira, R.B., Plastino, A., Zadrozny, B. and Merschmann, L. H. C. (2018). Categorizing feature selection methods for multi-label classification, *Artificial Intelligence Review* **49**(1): 1–22.

Plessis, M.C., Niu, G. and Sugiyama, M. (2017). Class-prior estimation for learning from positive and unlabeled data, *Machine Learning* **106**(4): 463–492.

Ramaswamy, H., Scott, C. and Tewari, A. (2016). Mixture proportion estimation via kernel embeddings of distributions, *Proceedings of the 33rd International Conference on Machine Learning, New York, USA*, pp. 2052–2060.

Sechidis, K. and Brown, G. (2018). Simple strategies for semi-supervised feature selection, *Machine Learning* **107**(2): 357–395.

Sechidis, K., Calvo, B., and Brown, G. (2014). Statistical hypothesis testing in positive unlabelled data, *Machine Learning and Knowledge Discovery in Databases, Nancy, France*, pp. 66–81.

Sechidis, K., Sperrin, M., Petherick, E.S., Lujan, M. and Brown, G. (2017). Dealing with under-reported variables: An information theoretic solution, *International Journal of Approximate Reasoning* **85**(1): 159–177.

Shalev-Shwartz, S. and Ben-David, S. (2013). *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press, Cambridge.

Sun, Y.-Y., Zhang, Y. and Zhou, Z.-H. (2010). Multi-label learning with weak label, *Proceedings of the 24th AAAI Conference on Artificial Intelligence, AAAI'10, Atlanta, USA*, pp. 593–598.

Teisseyre, P. (2020). Learning classifier chains using matrix regularization: application to multimorbidity prediction, *Proceedings of the European Conference on Artificial Intelligence, ECAI 2020, Santiago de Compostela, Spain*, pp. 1–8.

Teisseyre, P. (2021). Classifier chains for positive unlabelled multi-label learning, *Knowledge-Based Systems* **213**(1): 1–16.

Teisseyre, P., Mielniczuk, J. and Łazecka, M. (2020). Different strategies of fitting logistic regression for positive and unlabelled data, *Proceedings of the International Conference on Computational Science, ICCS 2020, Amsterdam, The Netherlands*, pp. 3–17.

Tsoumakas, G., Spyromitros-Xioufis, E., Vilcek, J. and Vlahavas, I. (2011). Mulan: A Java library for multi-label learning, *Journal of Machine Learning Research* **12**(1): 2411–2414.

Walley, N.M. *et al.* (2018). Characteristics of undiagnosed diseases network applicants: Implications for referring providers, *BMC Health Services Research* **18**(1): 1–8.

Wei, T., Guo, L.-Z., Li, Y.-F. and Gao, W. (2018). Learning safe multi-label prediction for weakly labeled data, *Machine Learning* **107**(4): 703–725.

Wu, L., Jin, R. and Jain, A.K. (2013). Tag completion for image retrieval, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **35**(3): 716–727.

Zhang, M. and Zhou, Z. (2013). A review on multi-label learning algorithms, *IEEE Transactions on Knowledge and Data Engineering* **26**(8): 1819–1837.

Zhu, P., Xu, Q., Hu, Q., Zhang, C. and Zhao, H. (2018). Multi-label feature selection with missing labels, *Pattern Recognition* **74**(1): 488–502.

Zufferey, D., Hofer, T., Hennebert, J., Schumacher, M., Ingold, R. and Bromuri, S. (2015). Performance comparison of multi-label learning algorithms on clinical data for chronic diseases, *Computers in Biology and Medicine* **65**(1): 34–43.

Paweł Teisseyre works as an assistant professor in the Institute of Computer Science, Polish Academy of Sciences, and at the Faculty of Mathematics and Information Science, Warsaw University of Technology. His research interests include feature selection in high-dimensional supervised problems, multi-label classification, learning from partially labelled data and applications of machine learning methods in medicine and genetics.

Received: 14 October 2021
Revised: 20 January 2022
Accepted: 25 January 2022