Int. J. Appl. Math. Comput. Sci., 2022, Vol. 32, No. 2, 299–309
DOI: 10.34768/amcs-2022-0022

REVISITING STRATEGIES FOR FITTING LOGISTIC REGRESSION FOR POSITIVE AND UNLABELED DATA

ADAM WAWRZEŃCZYK [a], JAN MIELNICZUK [a,b,*]

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

[b]Faculty of Mathematics and Information Sciences
Warsaw University of Technology
Koszykowa 5, 00-662 Warsaw, Poland

Positive unlabeled (PU) learning is an important problem motivated by the occurrence of this type of partial observability in many applications. The present paper reconsiders recent advances in parametric modeling of PU data based on empirical likelihood maximization and argues that they can be significantly improved. The proposed approach is based on the fact that the likelihood for the logistic fit and an unknown labeling frequency can be expressed as the sum of a convex and a concave function, which is explicitly given. This allows methods such as the concave-convex procedure (CCCP) or its variant, the disciplined convex-concave procedure (DCCP), to be applied. We show by analyzing real data sets that, by using the DCCP to solve the optimization problem, we obtain significant improvements in the posterior probability and the label frequency estimation over the best available competitors.

Keywords: positive and unlabeled learning, empirical risk, logistic regression, concave-convex optimization.

1. Introduction

In the paper, we consider a supervised classification setting when the data are subjected to a certain type of censoring, which makes class indicators assigned to objects (positive or negative in the case of a binary classification) only partially available. In the positive and unlabeled (PU) scenario considered here, it is assumed that some observations from the positive class are labeled, whereas the remaining observations (either positive or negative) are unlabeled. Thus in the PU setting the true binary class indicator Y is not observed directly but only through a binary label S. One knows that if $S = 1$ (labeled case), Y has to be 1, but for $S = 0$ (unlabeled case) Y may be either 1 or 0. Besides, each object is described by a vector of features x. This setup encompasses a plethora of practical classification problems, which explains why it is so intensively investigated (see, e.g., the work of Bekker and Davis (2020) for a recent review). Many examples include disease data (diagnosed patients with a specific disease detected ($S = 1$), and patients yet to be diagnosed who may be ill or not ($S = 0$)), web pages preferences of specific users (pages bookmarked as 'of interest' and pages not yet viewed) and ecological examples when habitats are labeled if a specific species of interest lives there, and unlabeled if this species has not been spotted yet. For representative examples of such applications, see, e.g., the works of Bahorik *et al*. (2014), Ward *et al*. (2009), Liu *et al*. (2003) and Yang *et al*. (2014). Another group of important applications from a different domain is under-reporting in survey data when some respondents fail to give a truthful answer to a sensitive question. Under-reporting might occur when the question asked concerns, e.g., reckless behavior such as texting while driving. In such cases, $S = 0$ occurs in two situations, the first one, when a driver abstains from texting during the drive ($Y = 0$), and the second, when such behavior takes place but is not reported ($Y = 1$) (see Sechidis

*Corresponding author

et al., 2017).

One of the popular approaches to learn from PU data is to impose certain parametric assumptions on the distribution of (X, Y), as it is commonly done in the traditional classification task, together with some additional assumptions on labeling mechanism S. This is necessary as in a general situation the posterior distribution of Y given x as well as prior probability $P(Y = 1)$ are not identifiable for the PU scenario (Łazęcka *et al*., 2021). It is common to consider the logistic type of dependence for the posterior probability $P(Y = 1|X = x)$ and assume that the censoring mechanism acts indiscriminately of x and thus is described only by the label frequency $c = P(S = 1|Y = 1)$ (the SCAR assumption discussed below). The majority of current learning approaches have been developed under the SCAR assumption (see, e.g., Elkan and Noto, 2008).

In particular, for the task of the posterior distribution estimation, Teisseyre *et al*. (2020) proposed the JOINT method, which consisted in minimization of the empirical risk for the observed data $(X_i, S_i), i = 1, \ldots, n$ with respect to the parameter of the logistic distribution and the label frequency. The JOINT method can be considered as a generic method with various variants possible depending on the optimization technique used. The optimization issue is a subtle one as it turns out that the empirical risk to be minimized is *not* a convex function of its parameters. In particular, Teisseyre *et al*. (2020) used the BFGS algorithm, whereas the approach by Łazęcka *et al*. (2021) in the context of prior estimation has been based on the minorization-maximization (MM) technique. In the present contribution, we argue that the superior estimators can be obtained by exploiting the specific structure of the empirical risk which turns out to be the sum of a convex and a concave function.

2. Notions and auxiliary results

We first introduce some basic notation. Let $X \in \mathbb{R}^p$ be a random variable corresponding to a feature vector, $Y \in \{0, 1\}$ be a true class label, and $S \in \{0, 1\}$ an indicator of an example being labeled ($S = 1$) or not ($S = 0$). We assume that there is some unknown distribution $P_{Y,X,S}$ such that $(Y_i, X_i, S_i), i = 1, \ldots, n$ is an independent and identically distributed (i.i.d.) sample drawn from it. Observed data consist of $(X_i, S_i), i = 1, \ldots, n$. This is the single sample scenario as opposed to the case-control scenario when the samples from the positive class and the general population are given (see Bekker and Davis (2020) for a thorough discussion of differences between both the scenarios). Only positive examples ($Y = 1$) can be labeled, i.e., $P(S = 1|X, Y = 0) = 0$. Thus we know that $Y = 1$ when $S = 1$ but when $S = 0$, Y can be either 1 or 0. Our primary aim is to learn the binary posterior distribution of Y given $X = x$, i.e., $y(x) = P(Y = 1|X = x)$ when we only observe samples from the distribution of (X, S), where $S = Y$ with a certain probability. We refer to the work of Scott *et al*. (2013) for a possible generalization of the censoring scenario considered.

To this end, we define the binary posterior distribution of S given x equal $s(x) = P(S = 1|x)$ and the propensity score function $e(x) = P(S = 1|Y = 1, x)$. We note that by conditioning on Y we have

$$\begin{aligned} s(x) &= P(S = 1|x) \\ &= P(S = 1|Y = 1, x)P(Y = 1|x) \\ &\quad + P(S = 1|Y = 0, x)P(Y = 0|x) \\ &= e(x)y(x), \end{aligned} \tag{1}$$

as $P(S = 1|Y = 0, x) = 0$.

In this paper we adopt the fundamental selected-completely-at-random (SCAR) assumption, which stipulates that $e(x)$ does not depend on x; thus,

$$e(x) = P(S = 1|Y = 1) := c^*,$$

where c^* stands for the labeling frequency. For approaches relaxing this condition, we refer to the works of Bekker *et al*. (2019) and Na *et al*. (2020). SCAR is frequently assumed and is equivalent to the condition that S and X are conditionally independent given Y. Another way of viewing it is to say that $S = \varepsilon \times Y$, where ε is a $\{0, 1\}$-valued Bernoulli variable independent of (X, Y) and such that $P(\varepsilon = 1) = c^*$. We stress, however, that it is valid only when the label value is assigned regardless of characteristics of an item and thus solely depends on the value of Y. Under this assumption $s(x) = c^* \times y(x)$ and it is easy to see that $P_{X|S=1} = P_{X|Y=1}$, whereas $P_{X|S=0}$ is a mixture

$$P_{X|S=0} = \frac{\alpha - \alpha c^*}{1 - \alpha c^*} P_{X|Y=1} + \frac{1 - \alpha}{1 - \alpha c^*} P_{X|Y=0}, \tag{2}$$

and $\alpha = P(Y = 1)$ is a prior probability of $Y = 1$. We also note that $c^* = P(S = 1|Y = 1) = P(S = 1)/P(Y = 1) = P(S = 1)/\alpha$. We do not assume any previous knowledge of c^* (although it is frequently imposed (see, e.g., Elkan and Noto, 2008)) and thus we only know that $0 \leq c^* \leq 1$. Estimation of c^* will be also investigated here as besides being of independent interest, its quality has a major impact on the accuracy of the posterior probability estimation.

We will adopt a parametric model for the posterior probability $y(x)$ assuming that it is governed by the logistic function

$$y(x) = \frac{\exp(b^{*T} x)}{1 + \exp(b^{*T} x)} = \sigma(b^{*T} x), \tag{3}$$

where $\sigma(s) = \exp(x)/(1 + \exp(s))$ is a logistic function and b^* is an unknown parameter (note that all vectors here

are considered as column vectors). This is commonly assumed and the previous research suggests that the empirical risk minimization described below is robust to misspecification of $y(x)$. Thus, in the setting considered, our problem boils down to adequate estimation of (c^*, b^*).

Define the logistic risk $R(b,c) = -E\big[S\log(c\sigma(b^Tx) + (1-S)\log(1-c\sigma(b^Tx))\big]$. The maximum likelihood method is based on the observation that if (3) holds for b^* and the SCAR is valid then

$$\arg\min_{b,c} R(b,c) = (b^*, c^*), \tag{4}$$

which follows from the information inequality (see, e.g., Cover and Thomas, 1991, Theorem 2.6.3). Thus one considers the empirical risk of observed data $(s_i, x_i)_{i=1}^n$ (the negative of the log-likelihood function divided by the sample size) equal to

$$\begin{aligned} R_n(b,c) = -\frac{1}{n}\sum_{i=1}^{n}\big(&S_i\log(c\sigma(b^TX_i)) \\ &+ (1-S_i)\log(1-c\sigma(b^TX_i))\big), \end{aligned} \tag{5}$$

and defines the maximum likelihood estimator $(\hat{c}^*, \hat{b}^{*T})$ of (c^*, b^*) as a minimizer of $R_n(b,c)$. The rationale here is that since in the view of the law of large numbers $R_n(b,c)$ approaches $R(b,c)$ almost surely, the minimizer of the former function should be close to that of the latter. Once $\hat{b}^*$ is obtained, we define an estimator of the posterior probability as $\hat{y}(x) = \sigma(\hat{b}^{*T}x)$

This problem seems rather easy to solve, however, it turns out that the introduction of the additional parameter c into the empirical risk destroys its convexity, i.e., $R_n(b,c)$ is not a convex function of b in contrast to $R_n(b,1)$, i.e., the logistic risk. There are two existing approaches to find the minimizer of (5): the first one called the JOINT method (introduced by Teisseyre *et al.* (2020)) which relies on the BFGS algorithm, and the other is its variant based on the majorization-minimization (MM) algorithm, introduced by Łazęcka *et al.* (2021) and relying on iterative minimization of a suitably constructed convex bound to $R_n(b,c)$. Here we propose a substantial modification of the generic JOINT method which uses an explicit representation of the log-likelihood as the difference of two convex functions and applies the concave-convex procedure (CCCP) tailored for such a case, as well as its disciplined version—DCCP.

3. Convex-concave representation of the empirical risk

Previous research (Teisseyre *et al.*, 2020) indicates that the JOINT method yields excellent results in PU tasks. It achieved state-of-the-art results both on multiple real data sets and synthetic examples. We note, however, the that general-purpose BFGS method was applied in the original JOINT method, and in the context of c^* estimation it was replaced by the MM algorithm (which optimizes a convex minorant of the risk) by Łazęcka *et al.* (2021). Thus the natural question arises whether the generic JOINT method can be improved by using the knowledge about the specific form of the risk in the PU problem. We show that the answer is positive and is based on the following important lemma.

Lemma 1. *The empirical risk function $R_n(b,c)$ in (5) can be represented as the sum of convex and concave functions of b (referred to as a convex-concave function). Moreover, it is a convex function of c.*

Proof. We note that the empirical risk in (5) may be written as $-n^{-1}\sum_{i=1}^n K_i(b,c)$, where

$$\begin{aligned} K_i(b,c) = s_i\log(c\sigma(b^Tx_i)) \\ + (1-s_i)\log(1-c\sigma(b^Tx_i)). \end{aligned} \tag{6}$$

Elementary transformations yield (we omit index i in $K_i(b,c)$ to avoid notional clutter)

$$\begin{aligned} K(b,c) &= s\log\left(\frac{ce^{b^Tx}}{1+e^{b^Tx}}\right) \\ &\quad + (1-s)\log\left(1-\frac{ce^{b^Tx}}{1+e^{b^Tx}}\right) \\ &= s\log c + s\log\left(\frac{e^{b^Tx}}{1+e^{b^Tx}}\right) \\ &\quad + (1-s)\log\left(\frac{1}{1+e^{b^Tx}}\right) \\ &\quad + (1-s)\log\left(1+(1-c)e^{b^Tx}\right) \\ &= s\log c + s\log\sigma(b^Tx) \\ &\quad + (1-s)\log\left(1-\sigma(b^Tx)\right) \\ &\quad + (1-s)\log\left(1+(1-c)e^{b^Tx}\right) \\ &=: f_1(c) + f_2(b) + f_3(b,c), \end{aligned} \tag{7}$$

where

$$\begin{aligned} f_1(c) &= s\log c, \\ f_2(b) &= s\log\sigma\left(b^Tx\right) + (1-s)\log\left(1-\sigma(b^Tx)\right), \\ f_3(b,c) &= (1-s)\log\left(1+(1-c)e^{b^Tx}\right). \end{aligned}$$

We note that $f_2(b)$ corresponds to the log-likelihood in the logistic model; thus, it is a concave function of b and, moreover,

$$\begin{aligned} &\frac{\partial^2}{\partial b^2}f_3(b,c) \\ &\qquad = (1-s)(1-c)\frac{e^{b^Tx}}{\left(1+(1-c)e^{b^Tx}\right)^2}xx^T. \end{aligned} \tag{8}$$

As the matrix xx^T is obviously nonnegative definite, we obtain that $f_3(b,c)$ is a convex function of b. Thus $K(b,c)$ is a sum of a concave and a convex function of b. This is obviously true also for $R_n(b,c)$ which is the average of the functions $-K_i(b,c)$.

The concavity of $K(b,c)$ with respect to c follows from the observation that

$$\frac{\partial^2}{\partial c^2}K(b,c) = -\frac{s}{c^2} - (1-s)\frac{e^{2b^Tx}}{\left(1+(1-c)e^{b^Tx}\right)^2} \leq 0, \quad (9)$$

and thus $-K_i(b,c)$ is convex with respect to c. ■

The algorithms proposed in the next section are motivated by Lemma 1.

4. JOINT method risk revisited: Proposed algorithms

The proposed approach is inspired by the variant of the JOINT method variant described by Łazęcka *et al.* (2021). A novel CD+MM procedure, combining cyclic coordinate descent (CD) and majorization-minimization (MM) algorithms, was proposed there and shown to outperform the JOINT method for the problem of class prior estimation. In a nutshell, the algorithm consists in minimization of a convex function, which majorizes the given non-convex function and the convex majorant is changed in an appropriate way at each step. Further on, we will refer to that approach as the MM method.

In this section we will introduce two new JOINT method variants, reutilizing the CD procedure (Algorithm 1). The procedure treats the value of one optimization variable as fixed while performing the optimization of the other variable. The variables are then swapped, and the procedure works in the loop until convergence. Thus in Step 4 minimization with respect to c is performed while the current value of b is held fixed, whereas in Step 5 optimization is with respect to b with c held fixed. Cyclic coordinate descent allows us to separate the optimization variables; as shown in Lemma 1, the summands of the JOINT risk function have different curvatures with respect to b and c. It allows us to utilize the optimization methods designed for both of them.

It is important to note that while Step 4 of Algorithm 1 is a routine convex optimization task, Step 5 is much harder to perform effectively. Convex-concave functions require different optimization algorithms, which will be the subject of the discussion which follows. More specifically, we propose variants of two algorithms, CCCP and DCCP, which are applied to perform the task in Step 5 above.

Algorithm 1. Cyclic coordinate descent.

Require: $b^{(0)}, c^{(0)}$: initial guesses for parameter vector and label frequency, $k_{\max}$: maximal number of iterations, ϵ: tolerance

1: $k := 0$
2: **repeat**
3: *Next iteration.* $k := k+1$.
4: *Minimize w.r.t. c.* Minimize convex function

$$\hat{R}_{joint}(c) := -\frac{1}{n}\sum_{i=1}^{n}\left[s_i \log\left(c\sigma(x_i^T b_{est}^{(k-1)})\right) + (1-s_i)\log\left(1-c\sigma(x_i^T b_{est}^{(k-1)})\right)\right]$$

$$c_{est}^{(k)} := \arg\min(\hat{R}_{joint}(c)).$$

5: *Minimize w.r.t. b.* Minimize convex-concave function

$$\hat{R}_{joint}(b) := -\frac{1}{n}\sum_{i=1}^{n}\left[s_i \log\left(c_{est}^{(k)}\sigma(x_i^T b)\right) + (1-s_i)\log\left(1-c_{est}^{(k)}\sigma(x_i^T b)\right)\right]$$

$$b_{est}^{(k)} := \arg\min(\hat{R}_{joint}(b)).$$

6: *Next iteration.* $k := k+1$.
7: **until** $\left|c_{est}^{(k)} - c_{est}^{(k-1)}\right| < \epsilon$ or $k = k_{max}$
8: **return** $b_{est} := b^{(k)}, c_{est} := c^{(k)}$ {returns optimal parameter vector and label frequency.}

The key difference here with the previous research is an explicit use of the concave-convex nature of the JOINT risk function. The MM method, used previously in this context, is a popular tool to optimize any non-convex function. The methods used in our paper, which will be introduced in Sections 4.1 and 4.2, are designed to minimize the convex-concave functions, and therefore might significantly improve the performance of the procedure.

4.1. CCCP variant. The convex-concave procedure (CCCP) was proposed by Yuille and Rangarajan (2003). For a review of the present state of the ART, see the work of Lipp and Boyd (2016). The paper stems from the observation that although any sufficiently smooth function can be represented in a convex-concave form, substantial gains can be obtained from the usage of the explicit form of the representation. This resulted in the proposal of the CCCP procedure itself, which is the iterative algorithm that can be used to minimize convex-concave functions. In its basic form, the algorithm requires the inverse function of the derivative of the convex part in each step, which is

not feasible for some functions. An alternative version is provided for such cases. The procedure is stated in Theorem 1 for an easy reference.

Theorem 1. (Yuille and Rangarajan, 2003) *Let* $E(b) = E_{vex}(b) + E_{cave}(b)$ *be a convex-concave function. Then the CCCP update rule for the* $E(b)$ *minimization* $b^t \to b^{t+1}$ *can be stated as*

$$b^{t+1} = \arg\min_b E_{t+1}(b),$$

where for $b = (b_1, \ldots, b_p)^T \in \mathbb{R}^p$

$$E_{t+1}(b) = E_{vex}(b) + \sum_{j=1}^{p} b_j \frac{\partial}{\partial b_j} E_{cave}(b^t). \qquad (10)$$

Note that for each t function $E_{t+1}(b)$ is convex as the sum of convex and linear functions.

In the case of $R_n(b, c)$ (see the proof of Lemma 1)

$$E_{cave}(b) = -\frac{1}{n} \sum_i \left(f_{1,i}(b) + f_{3,i}(b)\right)$$

and

$$E_{vex}(x) = -\frac{1}{n} \sum_i f_{2,i}(b).$$

Thus, according to the CCCP procedure, the step $b^t \to b^{t+1}$ consists in minimizing the convex function $E_{t+1}(b)$ which equals

$$-\frac{1}{n}\left[\sum_i \Big(s_i \log\sigma(x_i^T b) + (1-s_i)\log\big(1-\sigma(x_i^T b)\big)\Big) + \sum_j b_j \sum_i (1-s_i)(1-c)\frac{x_{i,j} e^{x_i^T b^t}}{1+(1-c)e^{x_i^T b^t}}\right],$$

where $x_i = (x_{i,1}, \ldots, x_{i,p})^T$.

The final form of the algorithm is presented as Algorithm 2. Note that Step 4 of the algorithm (minimization with respect to to b) might be achieved using traditional convex optimization algorithms, such as CGD or BFGS. We recall the the algorithm below is used to perform the task in Step 5 of Algorithm 1.

4.2. DCCP variant. Disciplined convex-concave programming (cf. Shen *et al.*, 2016) combines the ideas of the CCCP and disciplined convex programming (DCP). Problems expressed using DCP can be automatically converted to a standard form and solved by a generic solver. In order to do that, an objective function to be minimized must be expressed as a result of compositions of atomic functions, where applied operations belong to a family of composition rules which ensure that the curvature (convexity or concavity) of the resulting

Algorithm 2. JOINT risk minimization—the CCCP method.

Require: $b^{(0)}$: initial parameter values, $k_{\max}$: maximal number of iterations, ϵ: tolerance, c: label frequency

1: $k := 0$
2: **repeat**
3: *Next iteration.* $k := k + 1$.
4: *Perform CCCP step.* Minimize convex function $E(b)$ equal to

$$-\frac{1}{n}\left[\sum_i \Big(s_i \log\sigma(x_i^T b) + (1-s_i)\log\big(1-\sigma(x_i^T b)\big)\Big) + \sum_j b_j \sum_i (1-s_i)(1-c)\frac{x_{i,j} e^{x_i^T b^{(k-1)}}}{1+(1-c)x_{i,j} e^{x_i^T b^{(k-1)}}}\right]$$

5: *Next iteration.* $k := k + 1$.
6: **until** $\max_i \left|b_i^{(k)} - b_i^{(k-1)}\right| < \epsilon$ or $k = k_{max}$
7: **return** b_{est} {returns optimal parameter vector.}

expression is known. Functions that conform to those requirements will be referred to as DCP-compliant.

Any convex-concave problem of the form $E_{vex}(b) + E_{cave}(b)$ can obviously equivalently be expressed as difference of two convex functions $f(b) - g(b)$, where both $f(b) = E_{vex}(b)$ and $g(b) = -E_{cave}(b)$ are convex. We can formally formulate this as the DCCP problem:

$$\begin{aligned} &\text{minimize} && f(x) - t \\ &\text{subject to} && t = g(x), \end{aligned} \qquad (11)$$

where t is a new optimization variable. Note that when $f(b)$ has a DCP-verified curvature, $f(b) - t$ has it as well, in contrast to $f(b) - g(b)$. Contrary to standard DCP problems, which restrict curvatures of the objective function and the constraints to convex functions, DCCP problems can use expressions of arbitrary curvatures, as long as they are DCP-compliant.

To solve such a problem, a penalized version is used. First, curvatures of all expressions are checked. Then equality constraints are replaced by a pair of inequality constraints and slack variables are introduced to cope with the issue of infeasibility (we refer to Shen *et al.* (2016) for details). Note that the penalized CCCP defined in this way is a convex problem.

In our case (see the proof of Lemma 1) $f(b) = E_{cave}(x) = -\sum_i \left(f_{1,i}(b) + f_{3,i}(b)\right)/n$ and $g(b) = -E_{vex}(b) = \sum_i f_{2,i}(b)/n$. In order to use the DCCP method, however, we need to ensure that curvatures of those expressions can be DCP-verified. The main issue are the expressions involving the logarithmic function appearing in the definitions of f_2 and f_3. For example,

$\log(1 + \exp(x))$ is a concave function of its convex argument $1 + \exp(x)$. Curvatures of such expressions are impossible to check using solely DCP rules.

In order to support such scenarios (similar expressions are present, e.g., when optimizing the traditional logistic classifier), the logistic function (defined as $\text{logistic}(x) = \log(1 + \exp(x))$) has to be introduced as an atomic function of a known curvature (in this case, convex). Therefore, using elementary transformations, we can obtain the following DCP-compliant form of risk components:

$$\begin{aligned} f_2(b) &= sx^T b - \text{logistic}(x^T b) \\ f_3(b) &= (1-s)\,\text{logistic}\left(\log(1-c) + x^T b\right). \end{aligned} \tag{12}$$

Equations (12) are used to define a valid DCCP problem. The final form of the algorithm is presented as Algorithm 3. The DCCP problem present in Step 2 of the algorithm might be solved using existing DCCP solvers, such as the `DCCP` extension of the `CVXPY` Python package. As in the case of CCCP, the present algorithm is applied to solve the task in Step 5 of Algorithm 1.

5. Experiments: Analysis of real data sets

We tested all of the approaches proposed in the preceding section on 12 popular benchmark classification data sets from the UCI repository,[1] presented in Table 1. Their sizes range from a few hundred to tens of thousands of observations, with different dimensionalities and class priors. Is it thus apparent that selected data sets pose diverse classification problems, varying in size and complexity. A common approach used in the literature (Teisseyre *et al*., 2020; Bekker and Davis, 2018), which we employed as well, is to use common classification data sets to create artificial PU data sets by applying random labeling with a given label frequency c. The advantage of such a data set construction is the knowledge of the true class labels of each sample, which allows for better performance assessment, which in general constitutes a formidable challenge in PU learning (Bekker and Davis, 2020).

A general preprocessing procedure was used to prepare data for learning. Missing values were imputed using the mean feature value, and then the 5 best features were selected according to the mutual information filter. The filter relies on calculating plug-in estimators $\hat{I}(X_k, Y)$ of the mutual information (Cover and Thomas, 1991) between individual features X_k and response Y and then selecting features corresponding to 5 largest values of $\hat{I}(X_k, Y)$. Finally, the 80:20 train/test split was applied and the resulting data sets were standardized. Python

[1] `https://archive.ics.uci.edu/ml/datasets.php`.

Algorithm 3. JOINT risk minimization—the DCCP method.

Require: c: label frequency

1: *Determine DCP-compliant form of functions*

$$f(b) := -\frac{1}{n}\sum_i \left(s\log c + sx_i^T b - \text{logistic}(x_i^T b)\right),$$

$$g(b) := \frac{1}{n}\sum_i \left((1-s)\text{logistic}\left(\log(1-c) + x^T b\right)\right).$$

2: *Solve DCCP problem:* b_{est} solves

$$\begin{aligned} &\text{minimize} && f(b) - t \\ &\text{subject to} && t = g(b). \end{aligned}$$

3: **return** b_{est} {Returns optimal parameter vector.}

Table 1. Analysed datasets and their statistics.

Name	Size	Features	Class prior α
Adult	32561	57	0.24
BreastCancer	699	9	0.34
ILPD	583	10	0.28
credit-a	690	38	0.44
credit-g	1000	48	0.30
diabetes	768	8	0.35
heart-c	303	19	0.46
ionosphere	351	34	0.64
madelon	4400	500	0.5
spambase	4601	57	0.39
vote	435	32	0.39
wdbc	569	31	0.37

procedures used to obtain results presented in this paper are publicly available on GitHub.[2]

The proposed CCCP and DCCP PU algorithms were compared with the basic JOINT method and its modification using the MM algorithm, which was used by Łazęcka *et al*. (2021) to estimate the label frequency. Additionally, we tested the performance of the weighted logistic regression method introduced by Bekker *et al*. (2019). It is important to note that this needs an accurate label frequency estimate in order to approximate the posterior probability. We provided it using two algorithms, the classic Elkan–Noto approach (Elkan and Noto, 2008) (EN), and the state-of-the-art TIcE (Bekker and Davis, 2018) method. We focused on the weighted logistic regression and the basic JOINT for comparison purposes as it was shown by Teisseyre *et al*. (2020) that the basic JOINT outperforms other competitors.

The results were evaluated using two metrics.

[2] `https://github.com/adamw00000/PU-joint-CCCP-DCCP-2021`.

The first one is motivated by the fact that an ideal PU classifier should behave like a traditional logistic regression classifier. We will refer to such a method as the oracle method, since it describes an idealized situation where we have access to true class labels for all samples. The key metric will therefore be the approximation error of the posterior probability (AE). It is defined as

$$AE = n^{-1} \sum_{i=1}^{n} \left| \hat{y}_{\text{oracle}}(x_i) - \hat{y}_{\text{method}}(x_i) \right|,$$

where "method" stands for any of the tested methods. It measures how well the model trained on PU data can approximate the oracle estimator of the posterior probability. The other metric that we used was the label frequency estimation error, defined as $LFE = |\hat{c} - c|$.

Label frequency is the other parameter estimated by the proposed methods, and the quality of its estimation will greatly influence the prediction performance of the models. Moreover, we also evaluated the computing time. The compared methods all require different learning procedures, and thus using a standard measure (e.g., the number of function evaluations or iterations) yields inappropriate results. It led us to use the training time, measured in seconds, as the performance metric. All of the results presented below were averaged over 100 runs. The initial parameter vector was a zero vector, while the initial the label frequency estimate utilized a fact that label frequency c satisfies $P(S = 1) \leq c \leq 1$ and we used a midpoint $(\hat{P}(s = 1) + 1)/2$ as $c^{(0)}$.

6. Results

First, we evaluated the AE metric for each data set and the label frequency value (Fig. 1), and assessed the variability of the estimates. The main conclusion is that although no clear winner is performing the best on all the data sets considered, the three JOINT-based methods (MM, CCCP, and DCCP) perform overall very well and are superior to its competitors (the original JOINT and the weighted logistic regression). The performance of the DCCP is the most variable among the three modified versions: some data sets have proven to be difficult for this method (e.g., Adult, wdbc), but on the other hand, it was by far the best method on multiple other data sets (e.g., credit-a, credit-g, diabetes, heart-c, madelon). The difference is most noticeable for low label frequency values. This is expected in the PU setting; as the c value grows, more information on Y is available, which makes the task easier. It is the best method when averaged performance with respect to c is considered (see Table 1).

MM and CCCP variants proved more stable than the DCCP method. Their performance is usually comparable, but overall they both substantially outperform the basic JOINT method. Both variants of the weighted method underperform on most of the data sets, which is hardly surprising, as the previous study (Teisseyre *et al.*, 2020) has shown their inferiority to the original JOINT method.

Table 2 presents AE results for each data set averaged over c (the best results are in boldface). We see that overall the DCCP method clearly outperforms its competitors. Nevertheless, there are a few data sets where this classifier achieves inferior results (see, e.g. the Adult data set). On the other hand, CCCP and MM usually perform very similarly, with MM being slightly superior on most of the tested datasets. A notable exception is once again the Adult data set, where CCCP is the clear winner. The MM method performed satisfactorily overall, but its results are rarely the best. All of the JOINT-based approaches substantially outperform the weighted method, which yields poor results outside of the BreastCancer data set.

Figure 2 shows the quality of label frequency estimates on each data set versus the true label frequency value. Note that while for the weighted regression approach using both label frequency estimation methods, the label frequency error increases as c gets larger, for the JOINT-based methods its behavior is mostly stable. Interestingly, there are also a few data sets where the label frequency error starts decreasing for large c values. The relative stability of the estimation quality of c with regard to its value is a remarkable property of the methods considered based on the minimization of the empirical risk.

We can clearly see that the proposed CCCP and DCCP variants, along with the MM method, improve the label frequency error for difficult data sets (e.g., credit-a, credit-g, diabetes, heart-c, madelon). Estimation performance depends on the data set; for example, the DCCP method is unrivaled on the diabetes and madelon data set, whereas it is slightly poorer when used on the Adult data set.

The overall results gathered in Table 3 indicate that DCCP achieves the lowest averaged error value; it proved better than the alternatives for 5 data sets. CCCP also managed to outperform the MM algorithm for the label frequency estimation. Even though the difference in the mean results is not large, CCCP consistently estimates the label frequency better than the MM method, except for the spambase data set.

While the results presented above seem very promising, it is also important to highlight the drawbacks of the proposed algorithms. Table 4 shows that, similarly to the MM method, using DCCP and CCCP significantly increases the training time. This behavior is expected, as the cyclic coordinate descent requires multiple classification model fits internally, and all three of those algorithms are iterative by nature, as well. The internal minimization problem for both the proposed algorithms is also more complex than for the MM method and thus both of those approaches are even more computationally expensive.

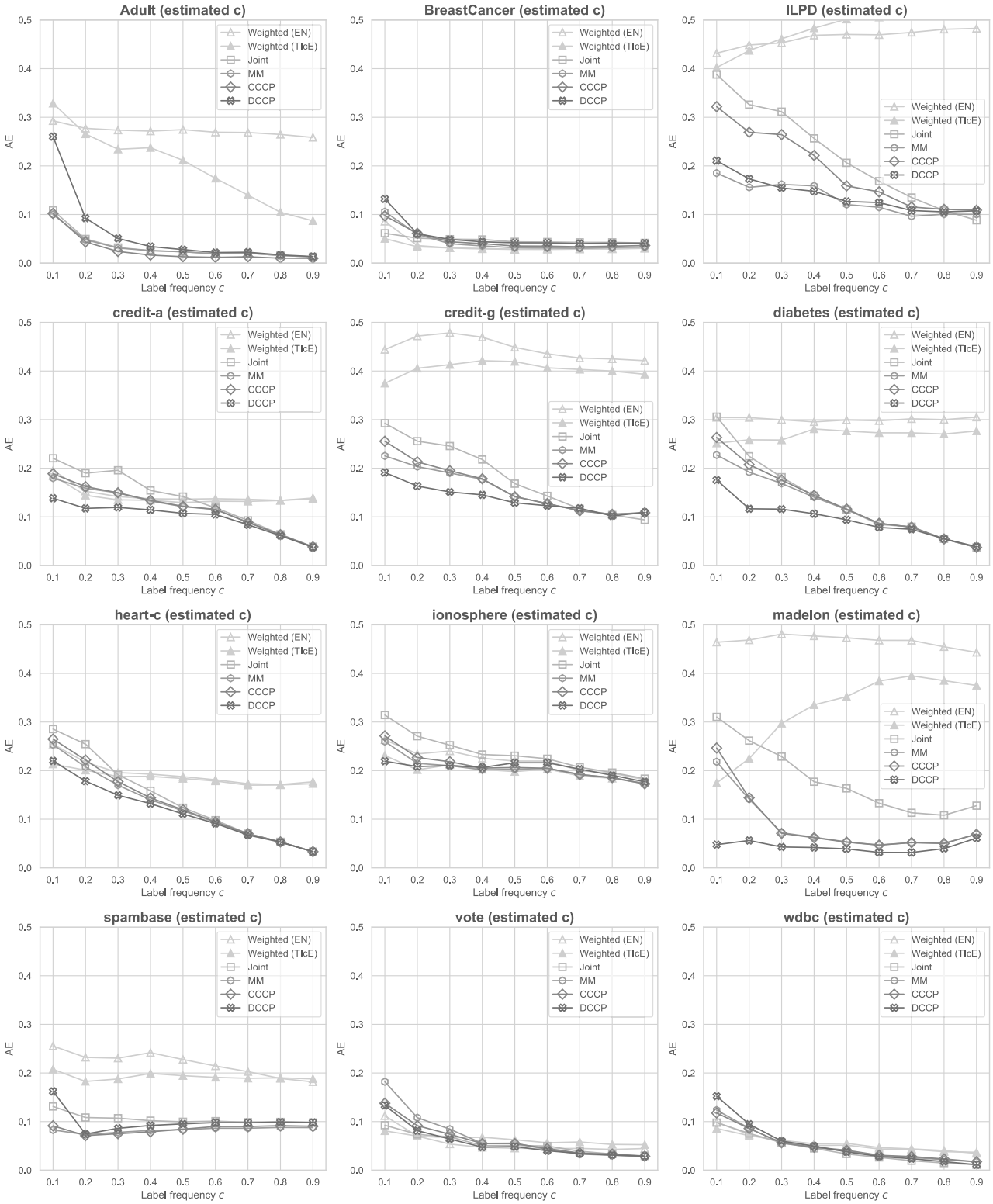

Fig. 1. Approximation error of the posterior vs. the label frequency c.

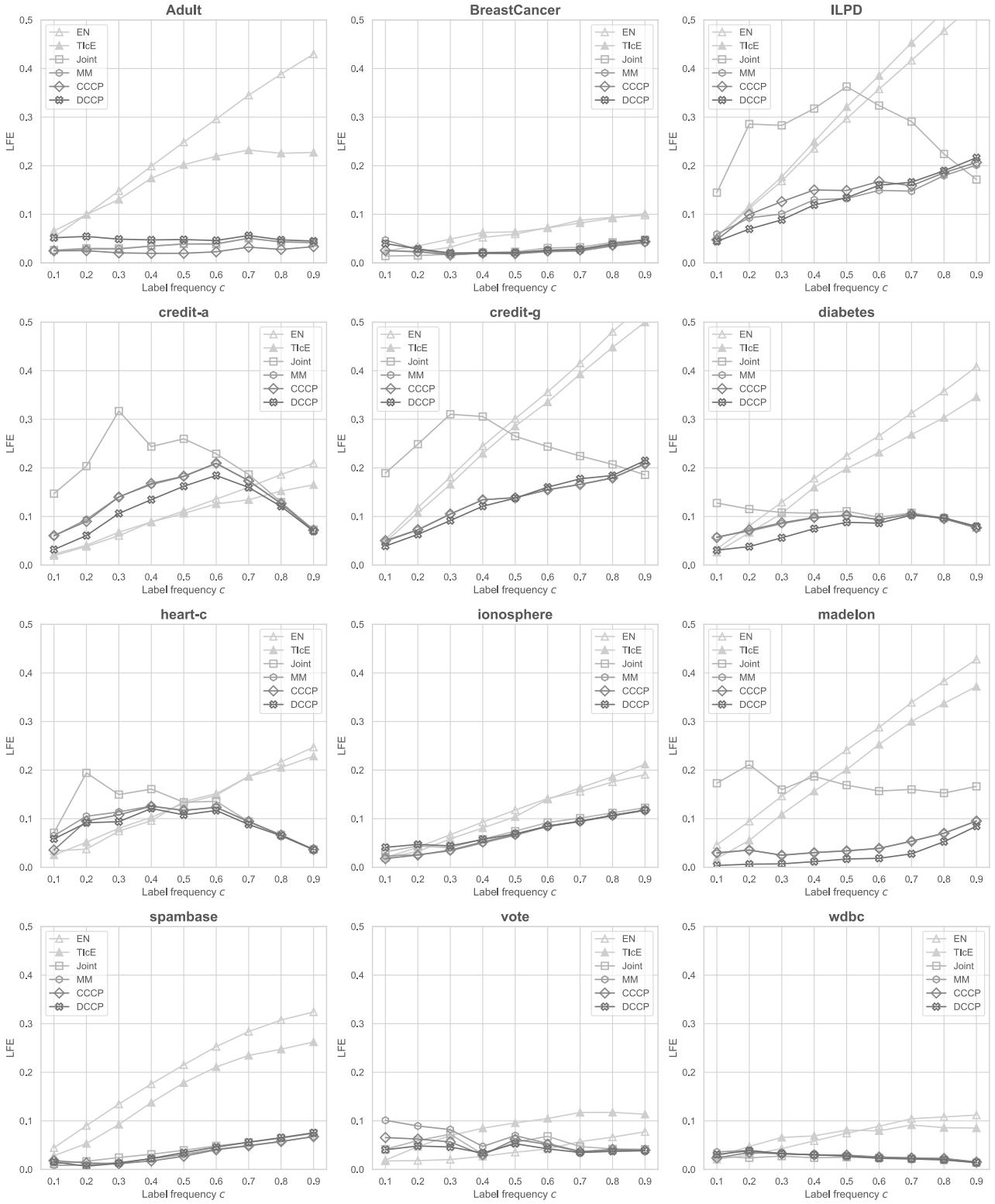

Fig. 2. Label frequency error vs. the true label frequency.

Table 2. Approximation error of the posterior per data set averaged over the labeling frequency c. Standard errors are given in brackets.

Dataset	Weighted (EN)	Weighted (TIcE)	Joint	MM	CCCP	DCCP
Adult	.272 (.003)	.198 (.010)	.034 (.011)	.033 (.010)	**.027** (.011)	.060 (.004)
BreastCancer	.036 (.007)	**.033** (.005)	.047 (.006)	.044 (.008)	.046 (.006)	.055 (.012)
ILPD	.464 (.017)	.484 (.023)	.221 (.041)	**.133** (.033)	.191 (.044)	.140 (.025)
credit-a	.145 (.013)	.140 (.013)	.135 (.023)	.116 (.017)	.118 (.017)	**.098** (.016)
credit-g	.447 (.024)	.404 (.030)	.182 (.037)	.155 (.036)	.160 (.038)	**.137** (.020)
diabetes	.301 (.014)	.269 (.016)	.136 (.028)	.122 (.022)	.129 (.024)	**.095** (.016)
heart-c	.194 (.014)	.185 (.015)	.141 (.022)	.126 (.018)	.131 (.018)	**.115** (.019)
ionosphere	.220 (.017)	**.200** (.018)	.235 (.021)	.205 (.018)	.209 (.016)	.205 (.019)
madelon	.466 (.008)	.325 (.025)	.180 (.048)	.085 (.025)	.088 (.028)	**.044** (.012)
spambase	.219 (.017)	.192 (.016)	.105 (.007)	**.083** (.005)	.085 (.005)	.100 (.009)
vote	.056 (.010)	.063 (.006)	**.054** (.014)	.069 (.015)	.062 (.014)	.057 (.013)
wdbc	.055 (.009)	.055 (.008)	**.042** (.008)	.050 (.010)	.050 (.009)	.053 (.012)
Mean	.238	.213	.132	.106	.113	**.099**

7. Conclusions

In this paper we introduce two new variants of the JOINT method, CCCP and DCCP. Both of the methods achieve approximation errors better or on par with the newly proposed MM method. The DCCP method seems especially promising, outperforming the competitors on multiple data sets. Both the methods also achieve very low label frequency errors, without the need of using external estimation procedures. Moreover, for the task of label frequency estimation, they outperform existing TIcE and EN methods handily and achieve better results than the MM variant. The proposed algorithms are recommended in practice for the smaller tasks; for large data sets, the high computation costs highlighted in this paper might make their usage problematic.

References

Bahorik, A.L., Newhill, C.E., Queen, C.C. and Eack, S.M. (2014). Under-reporting of drug use among individuals with schizophrenia: Prevalence and predictors, *Psychological Medicine* **44**(12): 61–69, DOI: 10.1017/S0033291713000548.

Bekker, J. and Davis, J. (2018). Estimating the class prior in positive and unlabeled data through decision tree induction, *Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, USA* **32**(1): 2712–2719.

Bekker, J. and Davis, J. (2020). Learning from positive and unlabeled data: A survey, *Machine Learning* **109**(4): 719–760, DOI: 10.1007/s10994-020-05877-5.

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, DOI: 10.1007/978-3-030-46147-8_5.

Cover, T. and Thomas, J. (1991). *Elements of Information Theory*, Wiley, New York, DOI: 10.1002/047174882X.

Elkan, C. and Noto, K. (2008). Learning classifiers from only positive and unlabeled data, *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, USA*, pp. 213–220, DOI: 10.1145/1401890.1401920.

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, DOI: 10.1007/s11634-021-00444-9.

Lipp, T. and Boyd, S. (2016). Variations and extension of the convex-concave procedure, *Optimization and Engineering* **17**(2): 263–287, DOI: 10.1007/s11081-015-9294-x.

Liu, B., Dai, Y., Li, X., Lee, W.S. and Yu, P.S. (2003). Building text classifiers using positive and unlabeled examples, *Proceedings of the 3rd IEEE International Conference on Data Mining, ICDM'03, Melbourne, USA*, pp. 179–186, DOI: 10.1109/ICDM.2003.1250918.

Na, B., Kim, H., Song, K., Joo, W., Kim, Y.-Y. and Moon, I.-C. (2020). Deep generative positive-unlabeled learning under selection bias, *Proceedings of the 29th ACM International Conference on Information and Knowledge Management, CIKM'20, Ireland,* pp. 1155–1164, DOI: 10.1145/3340531.3411971, (virtual event).

Scott, B., Blanchard, G. and Handy, G. (2013). Classification with asymetric label noise: Consistency and maximal denoising, *Proceedings of Machine Learning Research* **30**(2013): 1–23.

Sechidis, K., Sperrin, M., Petherick, E.S., Luján, M. and Brown, G. (2017). Dealing with under-reported variables: An information theoretic solution, *International Journal of Approximate Reasoning* **85**(1): 159–177, DOI: 10.1016/j.ijar.2017.04.002.

Shen, X., Diamond, S., Gu, Y. and Boyd, S. (2016). Disciplined convex-concave programming, *Proceedings of 2016 IEEE 55th Conference on Decision and Control (CDC), Las Vegas, USA*, pp. 1009–1014, DOI: 10.1109/CDC.2016.7798400.

Table 3. Labeling frequency error per data set averaged over the labeling frequency c. Standard errors are given in brackets.

Dataset	EN	TIcE	Joint	MM	CCCP	DCCP
Adult	.245 (.001)	.175 (.004)	.037 (.005)	.037 (.005)	**.025** (.005)	.049 (.005)
BreastCancer	.060 (.008)	.065 (.010)	.027 (.005)	.028 (.013)	**.026** (.006)	.030 (.011)
ILPD	.295 (.008)	.318 (.010)	.267 (.063)	**.132** (.033)	.143 (.034)	**.132** (.027)
credit-a	.112 (.007)	**.100** (.011)	.199 (.068)	.136 (.036)	.136 (.037)	.114 (.032)
credit-g	.299 (.008)	.279 (.012)	.242 (.067)	.134 (.029)	.134 (.029)	**.132** (.024)
diabetes	.221 (.008)	.190 (.011)	.105 (.042)	.087 (.023)	.087 (.023)	**.073** (.018)
heart-c	.131 (.012)	.129 (.015)	.116 (.056)	.094 (.040)	.089 (.037)	**.086** (.037)
ionosphere	.111 (.008)	.111 (.012)	.075 (.021)	**.067** (.010)	**.067** (.009)	.073 (.010)
madelon	.240 (.003)	.200 (.008)	.171 (.069)	.046 (.017)	.046 (.017)	**.025** (.014)
spambase	.203 (.004)	.161 (.007)	.041 (.004)	**.033** (.004)	.034 (.004)	.037 (.003)
vote	**.040** (.006)	.086 (.008)	.051 (.036)	.062 (.028)	.049 (.026)	.042 (.019)
wdbc	.071 (.009)	.070 (.011)	**.023** (.013)	.028 (.012)	.026 (.008)	.026 (.009)
Mean	.164	.156	.118	.076	.075	**.07**

Table 4. Mean training time (in seconds) per data set. Standard errors are given in brackets.

Dataset	Weighted (EN)	Weighted (TIcE)	Joint	MM	CCCP	DCCP
Adult	.80 (.08)	1.66 (.16)	**.35** (.04)	273.47 (20.97)	825.05 (83.33)	2698.24 (103.07)
BreastCancer	.04 (.00)	.04 (.00)	**.02** (.00)	6.35 (1.10)	19.95 (5.74)	106.88 (25.65)
ILPD	.08 (.01)	.06 (.01)	**.04** (.01)	77.67 (8.79)	104.24 (20.49)	55.98 (9.83)
credit-a	.03 (.00)	.03 (.00)	**.01** (.00)	16.99 (3.54)	23.99 (6.49)	42.66 (7.30)
credit-g	.03 (.00)	.04 (.00)	**.02** (.00)	15.29 (3.75)	35.58 (8.62)	45.28 (11.31)
diabetes	.03 (.00)	.03 (.00)	**.01** (.00)	9.93 (2.06)	23.14 (5.79)	40.09 (10.91)
heart-c	.02 (.00)	.02 (.00)	**.01** (.00)	6.84 (1.98)	12.84 (3.46)	25.20 (6.20)
ionosphere	.03 (.00)	.03 (.00)	**.02** (.00)	8.72 (1.51)	24.08 (5.82)	98.69 (25.51)
madelon	.05 (.01)	.08 (.01)	**.03** (.01)	9.81 (2.84)	30.14 (14.19)	82.29 (31.77)
spambase	.08 (.01)	.12 (.01)	**.06** (.01)	13.80 (.91)	123.64 (22.21)	195.18 (25.51)
vote	.04 (.01)	.03 (.01)	**.02** (.00)	16.71 (4.62)	43.37 (13.13)	92.47 (25.75)
wdbc	.03 (.00)	.03 (.00)	**.02** (.00)	6.32 (1.07)	37.24 (9.49)	24.61 (3.80)
Mean	.06	.082	**.031**	22.827	60.843	131.903

Teisseyre, P., Mielniczuk, J. and Łazęcka, M. (2020). Different strategies of fitting logistic regression for positive and unlabelled data, *in* V.V. Krzhizhanovskaya *et al.* (Eds), *Proceedings of the International Conference on Computational Science ICCS'20*, Springer International Publishing, Cham, pp. 3–17, DOI: 10.1007/978-3-030-50423-6_1.

Ward, G., Hastie, T., Barry, S., Elith, J. and Leathwick, J. (2009). Presence-only data and the EM algorithm, *Biometrics* **65**(2): 554–563, DOI: 10.1111/j.1541-0420.2008.01116.x.

Yang, P., Li, X., Chua, H., Kwoh, C. and Ng, S. (2014). Ensemble positive unlabeled learning for disease gene identification, *PLOS ONE* **9**(5): 1–11, DOI: 10.1371/journal.pone.0097079.

Yuille, A. and Rangarajan, A. (2003). The concave-convex procedure, *Neural Computation* **15**(4): 915–936, DOI: 10.1162/08997660360581958.

Adam Wawrzeńczyk holds an MSc degree in computer science from the Faculty of Mathematics and Information Sciences of the Warsaw University of Technology and is currently a PhD student at the Doctoral School of Information and Biomedical Technologies at the Polish Academy of Sciences. His research interests include recent advances in machine learning and deep learning, in particular inference from partially observable data.

Jan Mielniczuk is a full professor at the Institute of Computer Science, Polish Academy of Sciences, and a professor at the Faculty of Mathematics and Information Sciences of the Warsaw University of Technology. His main research contributions concern computational statistics and data mining, in particular time series modeling and prediction, inference for high dimensional and misspecified data, model selection, computer-intensive methods, asymptotic analysis, and quantification of dependence. He is an author and a co-author of two books and over eighty articles.

Received: 5 November 2021
Revised: 28 January 2022
Accepted: 10 February 2022