Augmented prediction of a true class for Positive Unlabeled data under selection bias

Jan Mielniczuk[a,b,*,1] **and Adam Wawrzeńczyk**[a,1]

[a]Institute of Computer Science, Polish Academy of Sciences
[b]Faculty of Mathematics and Information Science, Warsaw University of Technology
ORCID (Jan Mielniczuk): https://orcid.org/0000-0003-2621-2303, ORCID (Adam Wawrzeńczyk): https://orcid.org/0000-0002-6202-7829

Abstract.
We introduce a new observational setting for Positive Unlabeled (PU) data where the observations at prediction time are also labeled. This occurs commonly in practice – we argue that the additional information is important for prediction, and call this task "augmented PU prediction". We allow for labeling to be feature dependent. In such scenario, Bayes classifier and its risk is established and compared with a risk of a classifier which for unlabeled data is based only on predictors. We introduce several variants of the empirical Bayes rule in such scenario and investigate their performance. We put a special focus on dangers (and ease) of applying classical classification rule in the augmented PU scenario – due to no preexisting studies, an unaware researcher is prone to skewing the obtained predictions. We conclude that the variant based on recently proposed variational autoencoder designed for PU scenario works on par or better than other considered variants and yields advantage over feature-only based methods in terms of accuracy for unlabeled samples.

1 Introduction

We consider Positive Unlabeled (PU) learning, that is a binary classification task in which information about class indicators is only partially observable. More specifically, in the PU scenario some observations from a positive class are assigned labels, whereas the remaining observations from this class, as well as all negative observations, are unlabeled. PU data is collected in many practical situations, usually when obtaining reliable negatives is difficult or costly. Under such scenario the most common Machine Learning task is construction of a classification rule based *only* on predictors, which will assign a new observation to a positive or a negative class. In genetics, one can find some genes influencing a particular disease via costly experiments, but it cannot be assumed that the other genes in GenBank are not relevant for this disease [25]. Other typical uses include e.g. ecology [23], survey analysis [1, 21], recommendation systems [20, 22, 5, 18] and fraud detection [14]. PU framework can be considered as a special case of data with noisy or partial labels [16, 4, 13, 10, 3]. However, there are PU learning problems where the data partial observability can be slightly loosened – in many applications we would like to perform classification on *new* PU observations for which along with predictors, *the labeling status is given*.

Such situations commonly happen. A typical example is occurrence of hypertension. People who check their blood pressure regularly and when it is abnormal report that to a doctor, are treated for hypertension. In such a case positive labels are assigned to them. However, the remaining (unlabeled) group consists of those who have abnormal blood pressure level but do not contact a doctor, and those who are healthy. Another example is reporting episodes of certain illness (i.e. migraine) using dedicated software, see e.g. [19]. Here, some patients who experience such episodes, fail to report them and thus they can not be distinguished from patients who do not have them, whence both groups fall into unlabeled category. Note that in the examples labeling may depend on characteristics of the patients: e.g. in the first one the better educated and thus more aware of consequences of untreated hypertension are more likely to consult a doctor, in the second, the age, influencing dexterity of using a dedicated smartphone application, may be an important factor. This is called selection bias (or instance-dependent labeling) and correspond to the fact that the distribution of selected (i.e. labeled) observations is different from that of a positive class.

Note that in the considered examples labeling of new observations occur naturally. In the first example above, in a new batch of patients, for those who fail to report hypertension, one would like to detect those who are likely to be positive. In the "migraine" example it is of interest to detect patients who likely have failed to report migraine episodes, in order to contact them. Of course, in such a case, assignment is an issue for unlabeled observations only, as for the labeled ones we know for sure that they belong to a positive class. For the sake of distinguishing such task from the usual classification based on predictors alone, we will call this problem prediction for augmented PU observations or, in short, *augmented PU prediction*.To the best of our knowledge the paper is the first approach discussing this problem in the literature.

In the paper we establish the form of Bayes selection rule for detection of positive observations among unlabeled ones and show that it is more conservative than Bayes classification rule based solely on predictors. The fact that we are less likely to classify items to a positive class *when they are unlabeled* is understandable when one realises that unlabeled class contains relatively *less* positive observations than the general population.

We calculate the Bayes risk for such scenario and bound the excess risk for an classification based solely on predictors, what sheds light on advantage of using labeling information. Also we introduce

[*] Corresponding Author. Email: jan.mielniczuk@ipipan.waw.pl.
[1] Equal contribution.

empirical Bayes classifiers taking advantage of recent proposals for posterior probability estimators in this context. We show that the variant based on variational autoencoder designed for PU data works promisingly when accuracy relative to unlabeled data is considered as an evaluation metric.

2 Preliminaries

In the PU scenario, one considers a random vector (X, Y, S) with a distribution $P_{X,Y,S}$ such that $X \in \mathbb{R}^p$ and Y, S are binary with values 0 or 1. Y is a class indicator, with $Y = 1$ denoting a positive class and $Y = 0$, a negative one, whereas $S = 1$ and $S = 0$ mean that observation is either labeled or unlabeled, respectively. The considered setting stipulates that only some positive observations are labeled, whereas the remaining positive observations and negative ones are unlabeled. We adopt Selected At Random (SAR) assumption, which states that probability of labeling positive observation depends on observed values of predictors corresponding to it. Note that it is less stringent, and, as mentioned in the Introduction, more realistic than assumption that labeling is random but independent of an observation's features (Selected Completely At Random or SCAR assumption).

In single training-sample scenario adopted here it is also assumed that random iid vectors $(X_i, Y_i, S_i), i = 1, \dots, n$ are generated according to $P_{X,Y,S}$, but the observable data is $\mathcal{T} = \{(X_i, S_i), i = 1, \dots, n\}$. This is in contrast to case-control case when it is assumed that two X samples are available, one pertaining to the positive class (i.e. sampled from $P_{X|Y=1}$) and the second corresponding to the general population (that is, sampled from P_X).

The basic numerical quantities partially describing distribution $P_{X,Y,S}$ are (unobservable) posterior probability of a positive class $y(x) = P(Y = 1|X = x)$ and (observable) posterior probability of being labeled $s(x) = P(S = 1|X = x)$. Note that since the considered mechanism ensures that $P(S = 1|Y = 0, X = x) = 0$, the law of total probability implies the following relation between them

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

where $e(x)$, probability of being labeled given that it is positive and $X = x$, is called propensity score. Note that under SAR assumption this is, usually not constant, function of vector of predictors x. Risk bounds when $e(x)$ is known are given in [6]. Under SCAR $s(x)$ is constant and equals probability of a positive element being labeled $P(S = 1|Y = 1)$, which will be denoted by c.

We briefly discuss PU research in SAR setting and single-training-sample (or censoring) scenario which gains momentum recently due to its less stringent assumptions on labeling mechanism. The main approach to model posterior probability of positive outcome $Y = 1$ and propensity score as parametric functions. Furthermore, treating Y as a hidden random variable one employs Expectation-Maximization (EM) algorithm to estimate them [8]. It is also possible to alternately optimize estimates of their Fisher consistent expressions [2]. Another approach avoids estimation of propensity function and uses Empirical Risk Minimization method along with modelling of posterior by deep NN to find a solution [17, 24]. Other methods use additional assumptions such as co-monotonicity of posterior probabilities for Y and for S or some form of functional relation between posterior and propensity score [7]. We will use modified version of variational autoencoder proposed in [24] to solve augmented PU prediction problem discussed here. We also mention case-control scenario, in which a selection bias is recently incorporated [11, 12].

3 Augmented PU prediction method and its properties

We consider now augmented prediction for PU observations (augmented PU prediction) scenario when a new observation (X, S) is given and we want to predict the corresponding value of Y. Obviously, when $S = 1$ under assumed scenario we have $Y = 1$ and thus we need to consider only the case $S = 0$. We introduce the following prediction rule

$$d_B^{PU}(x, s) = \begin{cases} 1, & \text{if } s = 1 \\ \begin{cases} 1, & \text{if } y(x) > \frac{1+s(x)}{2} \\ 0, & \text{otherwise,} \end{cases} & \text{if } s = 0 \end{cases} \tag{2}$$

where $y(x)$ is posterior probability of positive class defined above (1). We will investigate the loss of efficiency when label S, which carries information about Y, is not available for classification. To this end we consider Bayes rule $d_B(x)$ based solely on x:

$$d_B(x) = \begin{cases} 1, & \text{if } y(x) > \frac{1}{2} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Directly from the above definitions we have that $d_B^{PU}(X, S)$ is more conservative on class $S = 0$ than $d_B(X)$ i.e. it less likely assigns objects to the positive class:

$$P(d_B^{PU}(X, S) = 1|S = 0) \le P(d_B(X) = 1|S = 0).$$

Below we show that the rule d_B^{PU} is optimal for 0-1 loss and calculate its risk and the excess risk of $d_B(x)$. The fact that the optimal rule is less likely to assign positive class to unlabeled observations than d_B is due to the fact that positive observations occur less frequently among unlabeled ones than in general population. Note that as d_B^{PU} is more conservative, classification changes might occur for both positive and negative examples – though in the expectation, the precision gain should outweigh the lost recall. Also, it has practical consequences for recommendations on the thresholds applied for classification in the follow-up studies involving PU data; see Section 5. The introduced approach is based on a simple observation that the considered problem can be regarded as a problem of determining the Bayes risk in the case when the vector of predictors is augmented by an additional predictor S. This also motivates the name of the problem. We let

$$\tilde{y}(x, s) = P(Y = 1|(X, S) = (x, s)) \tag{4}$$

be posterior probability of $Y = 1$ given the augmented vector of predictors. We also define the excess risk (or regret) of any augmented PU prediction rule $d(x, s)$ as (see e.g. [15]):

$$\Delta(d) = P\left(d(X, S) \neq Y\right) - P\left(d_B^{PU}(X, S) \neq Y\right).$$

Theorem 1. *(i) $d_B^{PU}(X, S)$ defined in (2) is the Bayes rule for Y under $P_{X,Y,S}$ i.e. it is the classification rule yielding the smallest misclassification error $P(d(X, S) \neq Y)$. Moreover, $d_B^{PU}(X, 0)$ is the Bayes rule for Y under $P_{X,Y|S=0}$ yielding the smallest classification error $P(d(X) \neq Y|S = 0)$.*

(ii) Define $w(x) = 1 + s(x) - 2y(x)$. Then Bayes risk of $d_B^{PU}(x, s)$ equals

$$\begin{aligned} L_{PU}^* &= \frac{1}{2}\Big(P(S = 0) - \mathbb{E}_{X,S=0}\,|2\tilde{y}(X, 0) - 1|\Big) \\ &= \frac{1}{2}\Big(P(S = 0) - \mathbb{E}_X\,|w(X)|\Big) \end{aligned} \tag{5}$$

(iii) We have for excess risk of $d_B(x)$:

$$\mathbb{E}_X\left(s(X)\mathbb{I}\left\{y(X)<\frac{1}{2}\right\}\right)\le \Delta(d_B)\le P(S=1). \tag{6}$$

The inequalities above are tight when $P\left(y(X)<\frac{1}{2}\right)=1$.

(iv) Odds ratio $OR(x)$ for odds of $Y=1$ in class $\{S=0\}$ and odds of $Y=1$ in a general population equals

$$\begin{aligned} OR(x) &= \frac{P(Y=1|S=0,X=x)}{P(Y=0|S=0,X=x)}\bigg/\frac{P(Y=1|X=x)}{P(Y=0|X=x)}\\ &= 1-e(x). \end{aligned} \tag{7}$$

Proof. (i) As we have

$$\begin{aligned} P\Big(d_B^{PU}(X,S)\neq Y\Big) &= P\Big(d_B^{PU}(X,S)\neq Y\big|S=1\Big)P(S=1)\\ &+P\Big(d_B^{PU}(X,S)\neq Y\big|S=0\Big)P(S=0) \end{aligned}$$

and the first term on RHS equals 0, it is enough to prove that the second and the third line in (2) define Bayes rule on the strata $\{S=0\}$. The Bayes rule for this problem is given by assigning $Y=1$ when the following condition holds:

$$\frac{P(Y=1|S=0,X=x)}{P(Y=0|S=0,X=x)}>1.$$

Denoting by $f(x)$ either the density of X or its probability mass function at x we have, inverting conditional probabilities, that that the ratio above equals

$$\begin{aligned} \frac{P(S=0,Y=1,X=x)}{P(S=0,Y=0,X=x)} &= \frac{f(x)(y(x)-s(x))}{f(x)(1-y(x))}\\ &= \frac{y(x)-s(x)}{1-y(x)}. \end{aligned} \tag{8}$$

Then it is enough to note that

$$\frac{y(x)-s(x)}{1-y(x)}>1 \quad\equiv\quad y(x)>\frac{1+s(x)}{2}. \tag{9}$$

(ii) As $d_B^{PU}(x,s)$ is Bayes classifier its risk equals

$$L_{PU}^{*}=\mathbb{E}_{X,S}\min\big(\tilde{y}(X,S),1-\tilde{y}(X,S)\big), \tag{10}$$

where $\tilde{y}(x,s)$ is defined in (4). This is easily justified by noting that if $\tilde{y}(x,s)>\frac{1}{2}$ and thus (x,s) is assigned to a positive class by the Bayes classifier, it commits an error with probability $1-\tilde{y}(x,s)=\min(\tilde{y}(x,s),1-\tilde{y}(x,s))$. Moreover, we have that $\tilde{y}(x,1)=1$ and reasoning as in (8) we obtain

$$\begin{aligned} \tilde{y}(x,0) &= P(Y=1|(X,S)=(x,0))\\ &= \big(y(x)-s(x)\big)\,/\,\big(1-s(x)\big). \end{aligned}$$

In view of $\min(a,b)=(a+b-|b-a|)/2$ we have $\min(a,1-a)=(1-|2a-1|)/2$ and whence (10) implies that

$$\begin{aligned} L_{PU}^{*} &= \frac{1}{2}-\frac{1}{2}\mathbb{E}_{X,S}\left|2\tilde{y}(X,S)-1\right|\\ &= \frac{1}{2}-\frac{1}{2}\mathbb{E}_{X,S=1}\left|2\tilde{y}(X,1)-1\right|\\ &\quad-\frac{1}{2}\mathbb{E}_{X,S=0}\left|2\tilde{y}(X,0)-1\right|\\ &= \frac{1}{2}-\frac{1}{2}P(S=1)-\frac{1}{2}\mathbb{E}_{X,S=0}\left|2\tilde{y}(X,0)-1\right|. \end{aligned} \tag{11}$$

Thus we established the first equality in (5). Noting that

$$\begin{aligned} &\mathbb{E}_{X,S=0}\left|2\tilde{y}(X,0)-1\right|\\ &=\int\frac{\left|2y(x)-s(x)-1\right|}{1-s(x)}f(x)(1-s(x))\,\mathrm{d}x\\ &=\mathbb{E}_X\left|w(X)\right| \end{aligned}$$

we establish the second one. We note that from the proof above it follows that $d_B^{PU}(x,0)$ is the Bayes classifier on the strata $\{S=0\}$ and its Bayes risk L_{PU}^{*0} equals

$$\begin{aligned} L_{PU}^{*0}=\frac{L_{PU}^{*}}{P(S=0)} &= \frac{1}{2}-\frac{1}{2}\mathbb{E}_{X|S=0}\left|2\tilde{y}(X,0)-1\right|\\ &=\frac{1}{2}-\frac{\mathbb{E}_X\left|w(x)\right|}{P(S=0)}. \end{aligned} \tag{12}$$

(iii) Reasoning as above we have

$$L^{*}=P(d_B(X)\neq Y)=\frac{1}{2}-\frac{1}{2}\mathbb{E}_X\left|2y(X)-1\right|$$

and in view of (11) we obtain

$$\begin{aligned} L^{*}-L_{PU}^{*} &= \frac{1}{2}P(S=1)\\ &+\frac{1}{2}\mathbb{E}_X\Big\{\left|2y(X)-s(X)-1\right|-\left|2y(X)-1\right|\Big\}. \end{aligned}$$

RHS of (6) is obtained by using triangle inequality $\left|2y(X)-s(X)-1\right|-\left|2y(X)-1\right|\le s(x)$. To prove LHS of (6) we note that we have the following refinement of triangle inequality for $b\ge 0$

$$|a-b|\ge|a|-b+2b\times\mathbb{I}\{a<0\}$$

Applying this to $a:=2y(X)-1$ and $b:=s(X)$ we have that

$$\left|2y(X)-s(X)-1\right|\ge\left|2y(X)-1\right|-s(x)+2s(X)\mathbb{I}\left\{y(X)<\frac{1}{2}\right\}$$

and this implies the conclusion. Note that the lower bound equals the upper bound when for all x we have $y(x)<\frac{1}{2}$. In this case we note that $P\big(d_B(X)\neq Y\big)=P(Y=1)$ whereas $P\big(d_B^{PU}(X,S)\neq Y\big)=P(S=0,Y=1)$ and the excess risk is thus $P(Y=1)-P(S=0,Y=1)=P(S=1)$. The result in (iii) is intuitive: d_B^{PU} does not err on $S=1$, whereas d_B commits an error on this stratum if $y(x)<1/2$.

(iv) This follows by noting that in view of above derivations

$$OR(x)=\frac{y(x)-s(x)}{1-y(x)}\bigg/\frac{y(x)}{1-y(x)}=\frac{y(x)-s(x)}{y(x)}=1-e(x).$$

□

Remark 1. *(i) The threshold in (2) can be expressed as*

$$y(x)>\frac{1+s(x)}{2}\equiv y(x)>\frac{1}{2-e(x)}.$$

When $e(x)$ is large, then unlabeled element is less likely to be positive and the threshold becomes larger.

(ii) We note that when labeling is independent of an object in a positive class (SCAR assumption) and thus propensity score $e(x)\equiv c$, we have (cf. (i)):

$$d_B^{PU}(x,0)=1\iff y(x)>\frac{1}{2-c}.$$

For situation of complete lack of labeling ($c = 0$) unlabeled class is distributed according to P_X and $d_B^{PU}(x, 0)$ coincides with $d_B(x)$ in agreement with the last inequality. Note that since under SCAR positive observations are labeled or not, regardless of the predictors' values, the threshold $(2 - c)^{-1}$ above is due solely to the changed proportion of positives among unlabeled ones compared with the general population.

(iii) *The result can be generalised to strictly proper composite losses $\ell(s, y)$ such that corresponding Bayes classification function equals $\phi(OR(x))$ and ϕ is strictly increasing as e.g. for logistic loss $\ell_{logistic}(s, y) = \log(1 + \exp(-sy))$ for which $\phi(s) = s$. Then the Bayes rule on the class $S = 0$ assigns x to class $Y = 1$ when $y(x) > (\phi^{-1}(1) + s(x))/(1 + \phi^{-1}(1))$. In particular it is equal to $d_B^{PU}(x)$ for logistic loss.*

Below we calculate excess risk in (6) for a specific model.

Example 1. *Let $y(x) = \Phi(x), X \sim N(0, 1)$, and $x \in \mathbb{R}$ (univariate probit model with standard normal predictor), and let propensity score $e_a(x) = \mathbb{I}\{x > a\}$ i.e. above threshold $a \in \mathbb{R}$ all positive observations are labeled. In this case the excess risk of $d_B(x)$ defined in (3) for $a > 0$ equals (refer to appendix A for full derivation[2])*

$$\begin{aligned}\Delta(d_B) &= \mathbb{E}_X \left[\min \left(y(X), 1 - y(X) \right) \right] \\ &\quad - \mathbb{E}_{X,S} \left[\min \left(\tilde{y}(X, S), 1 - \tilde{y}(X, S) \right) \right] \\ &= \frac{1}{2} - \Phi(a) + \frac{\Phi^2(a)}{2} = \frac{1}{2} \left(\Phi(a) - 1 \right)^2 \geq 0,\end{aligned}$$

and for $a < 0$ equals $\frac{1}{4} - \frac{\Phi^2(a)}{2} \geq 0$. Note that for $a \to \infty$ excess risk tends to 0 as $P_{X,S=0}$ approaches P_X in this case and $d_B^{PU}(x, 0)$ tends to $d_B(x)$. For $a \to -\infty$ the excess risk tends to 1/4 (risk of $d_B(x)$) as the risk of $d_B^{PU}(x, s)$ tends to 0.

Example 2. *Consider the situation when $y(x) = \sigma(\alpha x)$ and $e(x) = \sigma(\beta x)$ for $x \in \mathbb{R}$ and $\alpha, \beta \geq 0$. Then we have for $\tilde{y}(x, 0)$ defined in (4)*

$$\begin{aligned}\tilde{y}_{\alpha,\beta}(x, 0) &= \frac{y(x) - s(x)}{1 - s(x)} = \frac{\sigma(\alpha x) - \sigma(\alpha x)\sigma(\beta x)}{1 - \sigma(\alpha x)\sigma(\beta x)} \\ &= \frac{\frac{1}{\sigma(\beta x)} - 1}{\frac{1}{\sigma(\alpha x)\sigma(\beta x)} - 1} = \frac{1}{1 + e^{-(\alpha - \beta)x} + e^{-\alpha x}}.\end{aligned} \tag{13}$$

The plot of $\tilde{y}_{\alpha,\beta}(x, 0)$ for $\alpha = 1$ and various βs is shown on Figure 1. Note that for $\alpha = \beta$ we have $\tilde{y}_{\alpha,\alpha}(x, 0) = (2 + \exp(-\alpha x))^{-1}$ which tends to $\frac{1}{2}$ when $x \to +\infty$, indicating the most difficult situation when $\tilde{y}(x, 0)$ is in a vicinity of $\frac{1}{2}$.

4 d_B^{PU} applications – VAE-PU-Bayes

The proposed d_B^{PU} rule uses are not limited to the direct applications to the augmented PU prediction style data (where the observation label is available for the test data). As a motivational example we consider first a typical PU problem, with only predictors available at the test time.

VAE-PU [17] classifier is a classifier based on variational autoencoder designed for PU data. It proved to be one of the most effective recent contributions to modern PU learning due to usage of generated PU examples to offset scarcity of labeled examples for low label frequencies. VAE-PU+OCC [24] model improves on VAE-PU via more refined choice of artificial PU sample. The modification consists of selecting most likely positive samples from unlabeled dataset instead of using artificially generated examples directly as it is done in VAE-PU [17]. For the selection task, usage of one-class-classification (OCC) methods trained on labeled sample was proposed. The modifications significantly improved of the baseline VAE-PU especially in the middle label frequency area, in particular in the case of the two recommended variants – combining VAE-PU with the A^3 and Isolation Forest models, respectively.

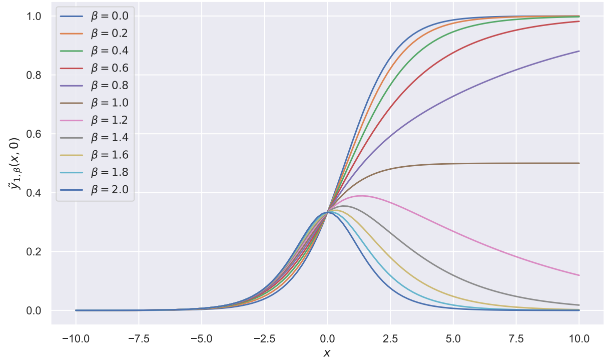

Figure 1. Values of $\tilde{y}_{1,\beta}(x, 0)$ depending on β.

VAE-PU-Bayes aims to further improve upon VAE-PU+OCC performance. Inner selection of the predicted positive examples is a crucial part of the VAE-PU+OCC, but general purpose one-class classifiers are – on the whole – a relatively low power methods, as they work with very limited information – only using the inlier sample distribution. Note that even in the standard PU problem, we can use more information than that, as we have access label information for all of the training examples. This allows for training a classifier which can be used for $s(x)$ estimation. VAE-PU-Bayes combines such a classifier with the VAE-PU $y(x)$ estimation in order to apply d_B^{PU} rule. As here the aim is to only filter the unlabeled set, we have $S = 0$ on it and can use the relevant part of the d_B^{PU} rule. Note that for the unlabeled stratum, we can rewrite it as follows (see Eq. (9)):

$$y(x) > \frac{1 + s(x)}{2} \equiv \frac{y(x) - s(x)}{1 - y(x)} > 1.$$

An important consideration is that for numerical reasons, the proportions in the training dataset are crucial to VAE-PU training (refer to [24] for details) – due to that, the number of selected likely positives should reflect the true portion on unlabeled positives present in the dataset. Thus, instead of choosing all unlabeled elements satisfying $\frac{y(x) - s(x)}{1 - y(x)} > 1$ as likely positives, we calculate an example's score as $\frac{y(x) - s(x)}{1 - y(x)}$ and select the appropriate number of examples with the highest score as an approximation of internal true PU sample. This approach to PU sample generation is consistent with the decision rule proposed in the paper, and can significantly outperform OCC-based models due to more powerful classification approach.

5 Numerical experiments

To check the effectiveness of the proposed approach, we prepared an extensive suite of experiments. We considered 4 synthetic and 6 real-world datasets:

- All synthetic datasets are generated using a mixture of two 20D Gaussian distributions (with different means 0 and μ and unit covariance I, except Variant 3) as a feature vector. This implies that

[2] Appendices are available at: https://arxiv.org/abs/2407.10309 and https://github.com/wawrzenczyka/VP-Bayes-S

indicator Y of an element of a mixture is drawn from the logistic distribution with parameter β (β is equal to direction of LDA boundary between feature clusters; we use intercept value which ensures $\pi = 0.5$). The following variants were used:

- **Variant 1.** Propensity score for each positive example $e_1(x)$ equals $\sigma(\gamma^T x + r)$, $\sigma(\cdot)$ being the logistic function, parameter vector $\gamma = [\gamma_1, \gamma_2, ..., \gamma_p] = [0.5, 0.5, ..., 0.5]$) and intercept r is tuned to ensure correct label frequency. Intercept tuning uses the assumed label frequency error as the objective, which is minimized using differential evolution algorithm.This allows us to construct synthetic datasets with both required labeling probabilities and label frequencies.
- **Variant 2.** Propensity score: $e_2(x) = e_1(x)^{10}$ which approximates step-wise function and has been considered in [9].
- **Variant 3.** In this variant, covariance matrix is diagonal, non-unit matrix in order to obtain non-logistic data (the diagonal vector equals: $[1, 2, 1, 2, ..., 1, 2]$), $e_3(x) = e_1(x)$,
- **Variant 4 (SCAR).** Constant propensity score, equal to label frequency: $e_4(x) = c$ (equivalent to the SCAR assumption).

- Real-world (characteristics of the data sets are given in the Appendix B, their prior probabilities π range from 0.4 to 0.53):
  - **MNIST**[3] – two different tasks, 3 versus 5 (images of digit *3* are positive, *5* – negative, abbreviated to 3v5) and OvE (images of *odd* digits are positive, *even* – negative),
  - **CIFAR-10**[4] – two different tasks, CT (*automobile* (car) images are positive, *truck* – negative) and VA (vehicles (*airplane*, *automobile*, *ship* and *truck*) images are positive; animals (*bird*, *cat*, *deer*, *dog*, *frog* and *horse*) – negative),
  - **STL-10**[5] – identical classes (but more complex images) as in CIFAR-10, only VA (Vehicle-Animal) split is considered,
  - **CDC-Diabetes**[6] – original class split (rebalanced).

We performed experiments for multiple label frequencies ($c \in \{0.02, 0.1, 0.3, 0.5, 0.7, 0.9\}$) in order to account for various PU task difficulties and labeling scenarios. To obtain such datasets, we synthetically generated label vectors S corresponding to each label frequency. For synthetic datasets, we use a labeling described above; for real-world datasets, we used feature-based labeling based on examples' properties. For MNIST datasets, examples are labeled depending on digit "boldness" – a portion of most bold (measured by average pixel value) examples are labeled; for CIFAR-10, a "redness" measure is used – the most red (according to the measure $r(x) = (R(x) - G(x)) + (R(x) - B(x))$, where $R(\cdot), G(\cdot), B(\cdot)$ correspond to mean R, G and B channel pixel values of input image x) examples are labeled; STL-10 uses labeling identical to CIFAR-10. "Maximal value" labeling (taking a portion of the dataset with the highest measure values) as opposed to probabilistic sampling (with probabilities based on those measures) aims to obtain a maximally difficult problem – note that in that case, labeled positive examples are maximally different from the unlabeled positive examples according to the labeling metric. While this deviates from the probabilistic, propensity score based labeling assumed by the methods, it also helps to measure robustness of the method against assumption violations. CDC-Diabetes aims to simulate a more practical, real-world PU scenario – there, labeling (diagnosis) probability scales with age (quadratically) and education level (linearly with subsequent stages of education) to model health awareness increase for senior citizens and more educated people.

We propose the following variants of the three popular no-SCAR PU methods:

- **LBE+S**. LBE [9] method is a natural candidate due to explicit modeling of both posterior probability $y(x)$ of $Y = 1$ and propensity score $e(x)$ (recall that we can obtain posterior probability of $S = 1$ by using $s(x) = e(x)y(x)$). After training the LBE classifier, we use both fitted components as plug-in estimators of $y(x)$ and $s(x)$ values in d_B^{PU} rule.
- **VAE-PU+S** (abbrev. **VP+S**). We use VAE-PU [17] classifier (described in section 4) as the base. As this model does not natively use the notion of propensity score in contrast to LBE, we introduce a separate feed-forward neural network for $s(x)$ estimation, trained separately from VAE-PU in the additional training step. Its predictions are then fed (together with VAE-PU's $y(x)$ estimations) to the proposed decision rule.
- **VAE-PU-Bayes+S** (abbrev. **VP-B+S**). We use a newly introduced VAE-PU-Bayes classifier (described in section 4) as the base. Similarly to VAE-PU, $s(x)$ estimator is trained and provided externally using available $(X_i, S_i)_{i=1}^n$ sample.

Note that for synthetic datasets, we can obtain accurate values of both $y(x)$ and $s(x)$; for those datasets we will additionally show results of the following two pseudo-methods:

- **S-Prophet**. Corresponds to the application of d_B^{PU} rule (2) with exact $y(x)$ and $s(x)$.
- **Y-Prophet**. Corresponds to a "naive" approach, where a researcher infers $Y = 1$ for test labeled examples with $S = 1$; but then (as one would in the standard PU task) blindly applies (3) to all other examples. Note that we assume knowledge of $y(x)$.

We also define a "naive" versions of LBE+S, VAE-PU+S and VAE-PU-Bayes+S in a similar way (as LBE, VAE-PU and VAE-PU-Bayes) – by assuming $Y = 1$ for labeled test examples, and using the simple d_B rule for the unlabeled examples.

In order to evaluate the performance, we focus on the "U-metrics", that is metrics calculated for unlabeled stratum. As prediction for labeled test examples is trivial, omitting them in the evaluation results paints clearer picture of the true, underlying decision performance. As an example, U-Accuracy is an Accuracy calculated only on the $S = 0$ stratum: $\textit{U-ACC} = {n_U}^{-1} \sum_{x_U \in U} \mathbb{I}\{d(x_U, s) = y_U\}$.

We prove the effectiveness of the proposed modification in two steps. First, we show which of the proposed variants (relying on d_B^{PU} rule) performs the best on our benchmark tasks. We then go on to compare the best variant with its naive counterpart, showing the benefits of applying our proposed decision rule. Each experiment (defined as a combination of dataset, label frequency and method) was performed 10 times, each time initialized with a different random seed (equal to experiment number). All code used for method implementation and performed experiments is publicly available[7].

The result section will also contain a brief comparison of VAE-PU-Bayes (abbrev. VP-B) method with the baseline VAE-PU (abbrev. VP) and two recommended VAE-PU+OCC variants – A^3 (abbrev. VP-$\boldsymbol{A^3}$) and Isolation Forest (abbrev. VP-IF). Those experiments were performed without test label availability, and use accuracy as the main metric. The other experimental settings do not differ from

[3] http://yann.lecun.com/exdb/mnist/
[4] https://www.cs.toronto.edu/~kriz/cifar.html
[5] https://cs.stanford.edu/~acoates/stl10/
[6] https://archive.ics.uci.edu/dataset/891/cdc+diabetes+health+indicators
[7] https://github.com/wawrzenczyka/VP-Bayes-S

Table 1. Accuracy values – VAE-PU-Bayes (traditional PU setting)

c	Method	MNIST 3v5	MNIST OvE	CIFAR CT	CIFAR VA	STL VA	CDC Diabetes
0.02	VP	**79.67 ± 0.90**	70.00 ± 1.76	87.31 ± 0.58	90.51 ± 0.52	81.64 ± 0.44	50.82 ± 0.22
	VP-A^3	79.01 ± 0.70	74.89 ± 1.62	83.67 ± 1.05	90.73 ± 0.27	79.62 ± 0.55	53.77 ± 1.33
	VP-IF	79.07 ± 0.75	**76.87 ± 0.99**	89.98 ± 1.23	89.99 ± 0.36	79.98 ± 1.06	52.38 ± 1.20
	VP-B	78.65 ± 0.87	73.13 ± 1.70	**91.74 ± 0.90**	**93.70 ± 0.17**	**84.68 ± 0.65**	**57.92 ± 1.31**
0.10	VP	83.57 ± 0.59	77.08 ± 0.92	91.22 ± 0.19	91.54 ± 0.31	85.31 ± 0.32	51.37 ± 0.25
	VP-A^3	89.91 ± 0.32	83.14 ± 1.41	90.35 ± 0.47	92.35 ± 0.28	84.91 ± 0.44	59.78 ± 1.36
	VP-IF	90.12 ± 0.30	83.60 ± 1.28	92.26 ± 0.39	90.21 ± 0.57	85.52 ± 0.56	57.24 ± 1.51
	VP-B	**90.76 ± 0.54**	**85.72 ± 1.05**	**93.73 ± 0.17**	**94.39 ± 0.14**	**88.44 ± 0.28**	**63.46 ± 0.83**
0.30	VP	86.77 ± 0.48	83.71 ± 0.27	92.96 ± 0.28	93.72 ± 0.13	88.23 ± 0.27	54.32 ± 0.26
	VP-A^3	92.65 ± 0.22	90.49 ± 0.23	89.88 ± 0.68	93.45 ± 0.12	86.38 ± 0.37	68.32 ± 0.38
	VP-IF	92.73 ± 0.22	90.59 ± 0.23	93.37 ± 0.21	92.02 ± 0.44	87.11 ± 0.51	67.72 ± 0.44
	VP-B	**92.98 ± 0.34**	**90.88 ± 0.27**	**94.22 ± 0.13**	**94.95 ± 0.06**	**89.99 ± 0.26**	**69.87 ± 0.19**
0.50	VP	88.32 ± 0.55	80.87 ± 1.35	92.91 ± 0.31	88.19 ± 0.37	88.57 ± 0.49	60.58 ± 0.37
	VP-A^3	93.28 ± 0.59	91.88 ± 0.42	88.03 ± 1.16	93.44 ± 0.12	87.46 ± 0.24	70.97 ± 0.19
	VP-IF	93.40 ± 0.55	91.59 ± 0.35	93.74 ± 0.13	92.04 ± 0.26	87.91 ± 0.35	70.66 ± 0.22
	VP-B	**93.92 ± 0.38**	**92.10 ± 0.28**	**94.46 ± 0.17**	**94.99 ± 0.05**	**90.57 ± 0.30**	**71.79 ± 0.12**
0.70	VP	91.58 ± 0.60	91.17 ± 0.29	94.20 ± 0.20	94.67 ± 0.08	90.12 ± 0.34	65.91 ± 0.25
	VP-A^3	93.89 ± 0.46	94.10 ± 0.28	88.93 ± 1.41	93.99 ± 0.07	89.06 ± 0.28	72.01 ± 0.07
	VP-IF	94.21 ± 0.39	94.39 ± 0.25	93.99 ± 0.16	93.74 ± 0.10	89.27 ± 0.32	71.93 ± 0.15
	VP-B	**94.59 ± 0.57**	**94.78 ± 0.16**	**94.51 ± 0.18**	**95.28 ± 0.04**	**91.01 ± 0.24**	**72.42 ± 0.07**
0.90	VP	94.63 ± 0.17	93.15 ± 0.25	**94.49 ± 0.14**	94.79 ± 0.13	91.14 ± 0.23	71.02 ± 0.17
	VP-A^3	95.35 ± 0.15	95.90 ± 0.10	91.12 ± 0.34	94.69 ± 0.09	91.03 ± 0.24	72.21 ± 0.07
	VP-IF	**95.70 ± 0.18**	95.80 ± 0.12	94.48 ± 0.19	94.58 ± 0.08	**91.29 ± 0.29**	**72.45 ± 0.13**
	VP-B	95.29 ± 0.16	**95.96 ± 0.11**	94.29 ± 0.22	**95.12 ± 0.13**	91.16 ± 0.27	72.20 ± 0.13

the augmented PU prediction experiments. The code for this method is a modification of the original VAE-PU+OCC code, also publicly available in a separate repository[8].

5.1 *Results of experiments*

VAE-PU-Bayes. First, we show the effectiveness of VAE-PU-Bayes in traditional PU setting. Table 1 presents the accuracy comparison between the newly introduced variant and previously existing VAE-PU and VAE-PU+OCC. In the vast majority of cases it outperforms the other VAE-PU variants, often by a very large margin – up to 5 percentage points (pp.). The only exceptions are the lowest label frequency $c = 0.02$, where is it outperformed on MNIST datasets, and $c = 0.9$, where even in this case it is roughly comparable to the best alternative. As VAE-PU+OCC was shown to achieve state-of-the-art level performance when compared to non-generative alternatives [24], VAE-PU-Bayes can be recommended as an improved variant of this model for traditional PU learning problems.

Augmented PU prediction. The rest of the result section focuses on augmented PU prediction scenario (with available test label). We stress that the aim here is to choose the best performing method among possible proposals for the new scenario. Tables 2 and 3 aggregate experiments performed with d_B^{PU} rule for synthetic and real-world datasets, respectively. The best U-Accuracy is marked in bold for each dataset and label frequency combination. The results for Balanced Accuracy are given in the Appendix C (note that the ratio of positives to negatives among unlabeled equals $\pi(1-c)/(1-\pi)$ and may be small for $c = 0.7, 0.9$). For synthetic datasets, VP-B+S is the top performer in the low frequency region; LBE+S does not work well for low label frequencies, but tends to overtake VP-B+S for $c = 0.7$ – then it levels off and falls off for $c = 0.9$. Even though VP+S is better that VP-B-S for $c = 0.9$, it is outperformed by it for all other label frequencies. For real-world datasets, VP-B+S shows even better performance, dominating in the vast majority of test cases, except for high label frequencies $c = 0.5, 0.7$ in the case of CDC Diabetes (where it is outperformed by LBE-S). Overall, we find that VP-B+S is the empirical variant of rule (2) most suited for general recommendation and use, thus we will use it in the further results' presentation. We also note that the dependence of performance on labeling frequency c is much less pronounced here than in classical PU inference. This is due to the fact that large value of c means in general relatively smaller number of positive observations among unlabeled ones and they are harder to detect.

Table 2. U-Accuracy values – Method comparison – Synthetic datasets

c	Method	Synth. 1	Synth. 2	Synth. 3	Synth. SCAR
0.02	S-Prophet	73.29 ± 0.35	73.24 ± 0.35	71.37 ± 0.35	73.48 ± 0.35
	VP+S	60.55 ± 2.48	59.15 ± 2.62	59.77 ± 2.40	63.19 ± 1.75
	VP-B+S	**61.23 ± 2.35**	**59.35 ± 2.66**	**60.16 ± 2.36**	**63.45 ± 1.82**
	LBE+S	50.32 ± 0.50	50.59 ± 0.50	50.66 ± 0.49	50.29 ± 0.47
0.10	S-Prophet	72.63 ± 0.30	72.16 ± 0.35	70.61 ± 0.30	73.74 ± 0.34
	VP+S	67.18 ± 0.42	65.96 ± 0.58	67.02 ± 0.57	67.64 ± 0.42
	VP-B+S	**67.71 ± 0.49**	**66.63 ± 0.60**	**67.49 ± 0.59**	**68.37 ± 0.50**
	LBE+S	52.72 ± 0.47	53.45 ± 0.50	53.04 ± 0.45	52.39 ± 0.53
0.30	S-Prophet	71.70 ± 0.42	70.83 ± 0.48	69.45 ± 0.39	74.30 ± 0.46
	VP+S	67.77 ± 0.57	65.29 ± 0.64	66.90 ± 0.55	70.20 ± 0.45
	VP-B+S	**68.51 ± 0.54**	**66.41 ± 0.57**	**67.27 ± 0.47**	**71.03 ± 0.42**
	LBE+S	61.05 ± 0.36	60.80 ± 0.43	61.03 ± 0.31	58.80 ± 0.52
0.50	S-Prophet	72.78 ± 0.57	71.96 ± 0.46	70.75 ± 0.59	76.93 ± 0.56
	VP+S	66.87 ± 0.41	65.04 ± 0.47	66.07 ± 0.67	69.78 ± 0.68
	VP-B+S	67.90 ± 0.45	65.57 ± 0.46	67.01 ± 0.51	**72.31 ± 0.38**
	LBE+S	**68.72 ± 0.51**	**67.72 ± 0.50**	**68.45 ± 0.45**	70.86 ± 0.48
0.70	S-Prophet	78.79 ± 0.40	78.37 ± 0.35	77.70 ± 0.50	81.31 ± 0.37
	VP+S	67.28 ± 0.88	66.38 ± 0.88	66.05 ± 0.78	69.34 ± 1.27
	VP-B+S	70.49 ± 0.54	68.91 ± 0.51	69.04 ± 0.48	73.57 ± 0.59
	LBE+S	**74.74 ± 0.42**	**73.50 ± 0.52**	**73.57 ± 0.42**	**81.03 ± 0.38**
0.90	S-Prophet	91.20 ± 0.49	91.26 ± 0.50	91.42 ± 0.44	91.83 ± 0.36
	VP+S	**85.54 ± 0.49**	**86.00 ± 0.76**	**85.64 ± 0.70**	**87.97 ± 0.55**
	VP-B+S	84.00 ± 0.42	84.48 ± 0.69	83.90 ± 0.56	86.50 ± 0.54
	LBE+S	74.76 ± 0.46	74.53 ± 0.47	73.57 ± 0.55	78.14 ± 0.48

Table 3. U-Accuracy values – Method comparison – Real-world datasets

c	Method	MNIST 3v5	MNIST OvE	CIFAR CT	CIFAR VA	STL VA	CDC-Diabetes
0.02	VP+S	77.74 ± 1.10	68.47 ± 1.18	87.19 ± 0.49	90.32 ± 0.25	81.43 ± 0.66	49.76 ± 1.49
	VP-B+S	**78.74 ± 1.53**	**74.91 ± 1.51**	**92.45 ± 0.43**	**94.11 ± 0.09**	**84.54 ± 0.67**	**51.15 ± 1.74**
	LBE+S	47.38 ± 0.32	49.82 ± 0.14	50.50 ± 0.40	60.67 ± 0.22	60.55 ± 0.28	50.47 ± 0.20
0.10	VP+S	80.21 ± 0.61	73.96 ± 1.43	91.42 ± 0.33	91.93 ± 0.33	86.36 ± 0.38	56.57 ± 0.79
	VP-B+S	**84.32 ± 0.76**	**83.12 ± 1.24**	**93.45 ± 0.19**	**94.37 ± 0.12**	**88.74 ± 0.30**	**61.01 ± 0.69**
	LBE+S	49.81 ± 0.34	51.55 ± 0.15	53.19 ± 0.39	62.81 ± 0.29	62.93 ± 0.28	52.55 ± 0.20
0.30	VP+S	80.66 ± 0.73	78.38 ± 0.96	92.95 ± 0.30	93.49 ± 0.16	88.93 ± 0.20	51.76 ± 0.94
	VP-B+S	**86.95 ± 0.52**	**87.98 ± 0.64**	**94.32 ± 0.14**	**95.22 ± 0.07**	**89.90 ± 0.30**	**62.63 ± 0.89**
	LBE+S	56.26 ± 0.34	57.11 ± 0.15	63.07 ± 1.04	74.37 ± 2.36	71.53 ± 1.13	58.72 ± 0.20
0.50	VP+S	82.25 ± 0.78	81.15 ± 0.93	94.39 ± 0.20	94.63 ± 0.26	90.85 ± 0.26	48.01 ± 0.85
	VP-B+S	**89.20 ± 0.81**	**90.25 ± 0.56**	**95.13 ± 0.20**	**95.65 ± 0.12**	**91.44 ± 0.30**	66.88 ± 0.37
	LBE+S	64.23 ± 0.33	64.80 ± 0.23	80.81 ± 1.68	85.35 ± 1.18	84.15 ± 1.00	**72.39 ± 0.81**
0.70	VP+S	86.42 ± 0.65	85.88 ± 0.67	95.32 ± 0.22	95.53 ± 0.18	**93.27 ± 0.29**	42.13 ± 1.05
	VP-B+S	**92.19 ± 0.47**	**92.76 ± 0.43**	**95.73 ± 0.18**	**96.23 ± 0.12**	**93.27 ± 0.21**	71.74 ± 0.49
	LBE+S	74.84 ± 0.40	76.65 ± 0.22	93.86 ± 0.57	95.16 ± 0.25	92.17 ± 0.38	**77.12 ± 0.79**
0.90	VP+S	91.90 ± 0.35	90.91 ± 0.35	**96.65 ± 0.16**	96.76 ± 0.15	**95.83 ± 0.19**	42.56 ± 2.15
	VP-B+S	**94.07 ± 0.25**	**95.73 ± 0.23**	96.47 ± 0.16	**97.11 ± 0.09**	95.37 ± 0.24	**83.53 ± 0.33**
	LBE+S	90.00 ± 0.27	87.03 ± 0.97	94.21 ± 0.48	94.69 ± 1.25	94.01 ± 0.85	71.09 ± 1.63

In Tables 4 and 5, we aim to capture the impact of using d_B^{PU} rule-based VP-B+S instead of its naive counterpart, VP-B. For synthetic datasets, where we have access to true $y(x)$ and $s(x)$ value, we contrast them with the analogous, reference S-Prophet and Y-Prophet methods. In this case, we also introduce the "semi-Prophet" methods – VP-B+S with true $s(x)$ and VP-B+S with true $y(x)$, where the true values replace the corresponding VP-B+S probability estimation. First thing to note is that S-Prophet is nearly equivalent to Y-Prophet in low label frequency setting. When there is a small number of labeled examples, it leads to low in expectation predicted labeling probability $s(x)$. As the rules d_B^{PU} and d_B are equivalent when $s(x) = 0$, for low label frequencies the change in predicted class is relatively infrequent. As the label frequency increases, so does the discrepancy between prophet methods – culminating in the drastic difference of 20 pp. for $c = 0.9$. The differences between VP-B+S and VP-B are not as big, and also tend to increase jointly with label frequency. However, p-value of the binomial test for testing H_0: P(U-acc. of VP-B $>$ U-acc. of VP-B+S)$\geq 1/2$ against the opposite hypothesis, equals to 1.8×10^{-5} (corresponding to 2 wins in 24 trials) for Table 4 and 1.1×10^{-7} in case of Table 5. Using the correct decision rule via VP-B-S we obtain U-Accuracy increase in almost every test scenario, though the margin here is much smaller than in the case of Prophets, and more pronounced for synthetic datasets. Inspecting semi-Prophet results gives us additional insights into the d_B^{PU} components. Note that when using true $y(x)$, the VP-B+S semi-Prophet's accuracy does not deviate significantly from S-Prophet's – even though the estimation of $s(x)$ was fairly crude, it is good enough when combined with accurate $y(x)$ estimations to improve results significantly. The same does not hold true for VP-B+S semi-Prophet

[8] https://github.com/wawrzenczyka/VAE-PU-Bayes

Table 4. U-Accuracy values – Decision rule comparison – Synthetic datasets

c	Method	Synth. 1	Synth. 2	Synth. 3	Synth. SCAR
0.02	S-Prophet	73.29 ± 0.35	73.24 ± 0.35	71.37 ± 0.35	73.48 ± 0.35
	Y-Prophet	73.31 ± 0.36	73.24 ± 0.35	71.40 ± 0.36	73.50 ± 0.35
	VP-B	61.00 ± 2.40	**59.62 ± 2.56**	60.14 ± 2.38	63.37 ± 1.77
	VP-B+S	**61.23 ± 2.35**	59.35 ± 2.66	**60.16 ± 2.36**	**63.45 ± 1.82**
	VP-B+S + true s(x)	60.65 ± 2.41	59.15 ± 2.69	59.98 ± 2.39	63.14 ± 1.76
	VP-B+S + true y(x)	73.29 ± 0.37	73.21 ± 0.35	71.44 ± 0.35	73.46 ± 0.33
0.10	S-Prophet	72.63 ± 0.30	72.16 ± 0.35	70.61 ± 0.30	73.74 ± 0.34
	Y-Prophet	72.63 ± 0.35	72.19 ± 0.37	70.68 ± 0.37	73.66 ± 0.33
	VP-B	**67.81 ± 0.48**	66.42 ± 0.53	67.38 ± 0.60	68.35 ± 0.48
	VP-B+S	67.71 ± 0.49	**66.63 ± 0.60**	**67.49 ± 0.59**	**68.37 ± 0.50**
	VP-B+S + true s(x)	67.64 ± 0.50	66.33 ± 0.51	67.16 ± 0.62	68.07 ± 0.48
	VP-B+S + true y(x)	72.71 ± 0.34	71.92 ± 0.39	70.61 ± 0.35	73.73 ± 0.32
0.30	S-Prophet	71.70 ± 0.42	70.83 ± 0.48	69.45 ± 0.39	74.30 ± 0.46
	Y-Prophet	71.06 ± 0.39	70.08 ± 0.39	69.00 ± 0.37	73.56 ± 0.34
	VP-B	68.25 ± 0.47	66.27 ± 0.62	67.14 ± 0.52	70.56 ± 0.43
	VP-B+S	**68.51 ± 0.54**	**66.41 ± 0.57**	**67.27 ± 0.47**	**71.03 ± 0.42**
	VP-B+S + true s(x)	68.19 ± 0.54	66.02 ± 0.63	67.06 ± 0.51	70.72 ± 0.44
	VP-B+S + true y(x)	71.26 ± 0.46	70.56 ± 0.52	69.19 ± 0.43	74.32 ± 0.48
0.50	S-Prophet	72.78 ± 0.57	71.96 ± 0.46	70.75 ± 0.59	76.93 ± 0.56
	Y-Prophet	69.87 ± 0.40	68.83 ± 0.39	67.81 ± 0.43	73.26 ± 0.35
	VP-B	67.07 ± 0.40	65.18 ± 0.45	66.14 ± 0.65	70.43 ± 0.46
	VP-B+S	**67.90 ± 0.45**	**65.57 ± 0.46**	**67.01 ± 0.51**	**72.31 ± 0.38**
	VP-B+S + true s(x)	67.58 ± 0.38	65.36 ± 0.51	66.67 ± 0.63	71.99 ± 0.46
	VP-B+S + true y(x)	72.06 ± 0.59	71.11 ± 0.61	69.83 ± 0.62	76.34 ± 0.62
0.70	S-Prophet	78.79 ± 0.40	78.37 ± 0.35	77.70 ± 0.50	81.31 ± 0.37
	Y-Prophet	69.39 ± 0.41	68.79 ± 0.43	67.44 ± 0.47	73.42 ± 0.35
	VP-B	66.46 ± 0.59	65.69 ± 0.56	65.25 ± 0.65	68.76 ± 0.71
	VP-B+S	**70.49 ± 0.54**	**68.91 ± 0.51**	**69.04 ± 0.48**	**73.57 ± 0.59**
	VP-B+S + true s(x)	69.95 ± 0.58	69.16 ± 0.49	68.88 ± 0.53	73.09 ± 0.58
	VP-B+S + true y(x)	77.42 ± 0.45	77.11 ± 0.39	75.95 ± 0.61	80.46 ± 0.46
0.90	S-Prophet	91.20 ± 0.49	91.26 ± 0.50	91.42 ± 0.44	91.83 ± 0.36
	Y-Prophet	71.30 ± 0.44	71.17 ± 0.45	69.25 ± 0.47	73.33 ± 0.48
	VP-B	69.71 ± 0.37	69.47 ± 0.41	68.16 ± 0.49	71.76 ± 0.42
	VP-B+S	**84.00 ± 0.42**	**84.48 ± 0.69**	**83.90 ± 0.56**	**86.50 ± 0.54**
	VP-B+S + true s(x)	84.12 ± 0.56	83.89 ± 0.55	83.56 ± 0.65	87.03 ± 0.49
	VP-B+S + true y(x)	90.44 ± 0.41	90.69 ± 0.47	89.72 ± 0.37	90.82 ± 0.24

Table 5. U-Accuracy values – Decision rule comparison – Real-world datasets

c	Method	MNIST 3v5	MNIST OvE	CIFAR CT	CIFAR MA	STL MA	CDC-Diabetes
0.02	VP-B	**78.75 ± 1.44**	74.53 ± 1.49	92.40 ± 0.41	93.94 ± 0.10	84.50 ± 0.66	51.07 ± 1.76
	VP-B+S	78.74 ± 1.53	**74.91 ± 1.51**	**92.45 ± 0.43**	**94.11 ± 0.09**	**84.54 ± 0.67**	**51.15 ± 1.74**
0.10	VP-B	84.14 ± 0.65	82.67 ± 1.30	93.32 ± 0.18	94.29 ± 0.12	88.54 ± 0.30	**61.25 ± 0.80**
	VP-B+S	**84.32 ± 0.76**	**83.12 ± 1.24**	**93.45 ± 0.19**	**94.37 ± 0.12**	**88.74 ± 0.30**	61.01 ± 0.69
0.30	VP-B	86.64 ± 0.56	87.89 ± 0.65	94.18 ± 0.17	95.11 ± 0.07	**90.11 ± 0.26**	**63.44 ± 0.81**
	VP-B+S	**86.95 ± 0.52**	**87.98 ± 0.64**	**94.32 ± 0.14**	**95.22 ± 0.07**	89.90 ± 0.30	62.63 ± 0.89
0.50	VP-B	88.75 ± 0.84	90.19 ± 0.54	94.87 ± 0.22	95.44 ± 0.11	91.35 ± 0.25	66.52 ± 0.31
	VP-B+S	**89.20 ± 0.81**	**90.25 ± 0.56**	**95.13 ± 0.20**	**95.65 ± 0.12**	**91.44 ± 0.30**	**66.88 ± 0.37**
0.70	VP-B	91.84 ± 0.48	92.73 ± 0.39	95.30 ± 0.20	95.94 ± 0.10	92.64 ± 0.24	67.73 ± 0.43
	VP-B+S	**92.19 ± 0.47**	**92.76 ± 0.43**	**95.73 ± 0.18**	**96.23 ± 0.12**	**93.27 ± 0.21**	**71.74 ± 0.49**
0.90	VP-B	93.90 ± 0.25	95.45 ± 0.21	95.68 ± 0.16	96.51 ± 0.05	93.54 ± 0.26	68.13 ± 0.34
	VP-B+S	**94.07 ± 0.25**	**95.73 ± 0.23**	**96.47 ± 0.16**	**97.11 ± 0.09**	**95.37 ± 0.24**	**83.53 ± 0.33**

using true $s(x)$ values, which indicates that $y(x)$ estimation inaccuracy is a major contributor to the performance drop compared with the Prophet methods. This is evident by contrasting the results with the S-Prophet – Y-Prophet pair, where using true $s(x)$ for d_B^{PU} rule proved to increase performance dramatically for high label frequencies. Note that sometimes even a slight variation of $y(x)$ might lead to a change of d_B^{PU} influencing the final example label or leaving it unchanged.

Results above shows that in real-world scenarios, the performance gain obtained by using the proposed decision rule over d_B rule is systematic but relatively small; this is especially apparent when comparing it to Prophets' improvements. Figure 2 aims to illustrate one of the potential causes of that problem. For the sake of this example, we will plot those values only for samples from $S = 0$ stratum. Note that for this stratum, d_B rule is equivalent to $\mathbb{I}\{y(x) > 0.5\}$, whereas d_B^{PU} – to $\mathbb{I}\{y(x) - \frac{s(x)}{2} > 0.5\}$. This formulation provides us with a matching threshold 0.5 for both rules.

The example orders the test samples according to increasing $y(x)$ (blue color, basis of d_B rule). In the figure, we introduce one additional dot for each test instance, which now corresponds to $y(x) - \frac{s(x)}{2}$ (basis of d_B^{PU} rule). Those dots are colored based on their test class (positive examples in green, and negative – in red). As $y(x) - \frac{s(x)}{2}$ is always lower or equal to $y(x)$, in order for the d_B and d_B^{PU} classification rules to differ on the $S = 0$ stratum (i) the blue dot for the given test example must be above black boundary line ($y = 0.5$), and (ii) the other dot (green or red) must be lying below it. The area in the chart where this is possible is shaded gold, and only examples falling there fulfill both conditions. The important thing to note is that the amount of examples falling in the golden area is relatively small, due to approximated $y(x)$ tending to the extremes of 0 and 1 – which might not hold true for the true $y(x)$ distribution. This limits the benefits of applying the d_B^{PU} rule, as even though the examples in the golden area are mostly negative (resulting in decreasing of the number of false signals), and the green, positive unlabeled samples are concentrated in the high $y(x)$ area, the limited number of affected samples by rule's modification lowers the impact of the correction on the metrics such as U-Accuracy.

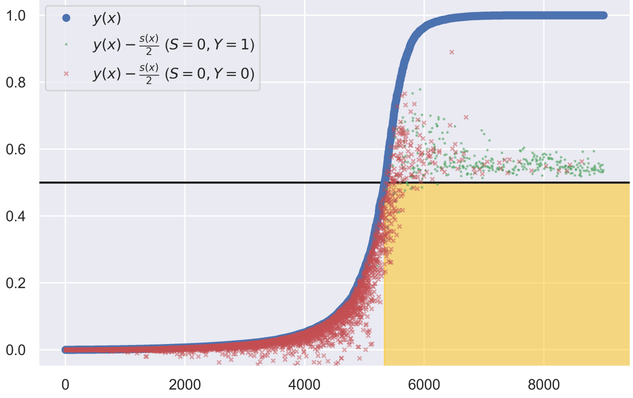

Figure 2. Classification rules for test instances, CIFAR VA, $c = 0.9$, $S = 0$ stratum (colored by test class).

6 Conclusions

The contribution of this paper is twofold: firstly, we highlight a previously unexplored area of PU learning (augmented PU prediction) where samples' labels at prediction time are available. Secondly, we propose a novel d_B^{PU} decision rule tailored for this setting. We study the basic properties of the proposed rule and contrast it with the properties of the usual Bayes rule based solely on samples' features. We also show that d_B^{PU}'s usefulness is not limited to the augmented PU prediction scenario, and it can be employed also in e.g. in traditional PU setting as a part of VAE-PU-Bayes model. The latter half of the paper focuses on the practical experiments, combining d_B^{PU} rule with preexisting PU models. We start off by showing the substantial improvements of VP-B over the VAE-PU+OCC baseline for traditional PU tasks. In augmented PU prediction setting, we identify VP-B+S model as the most promising among the newly constructed methods. By comparing it with its naive counterpart, VP-B, we show that using d_B^{PU} rule systematically improves accuracy on the test dataset. However, results for the two Prophet methods (which use perfect knowledge of $y(x)$ and $s(x)$), as well as semi-Prophets (utilizing the perfect knowledge of only one of those variables) indicate that those improvements could be potentially significantly larger, especially for the high label frequencies. As this paper introduces a new, practical setting for PU data, it naturally presents researchers with a rich opportunities for further work. One of those possibilities involves better modeling of $y(x)$, which currently is not sensitive enough to corrections via d_B^{PU} rule in direct, practical applications. Proposing new classifiers relying on estimators of both $y(x)$ and $s(x)$ directly (similarly to LBE) is an important challenge. Moreover, an excellent performance of VAE-PU-Bayes in traditional setting encourages further work on this model, or incorporating d_B^{PU} rule as a component of more PU models.

References

[1] J. Bekker and J. Davis. Learning from positive and unlabeled data: a survey. *Machine Learning*, 109:719–760, 2020.

[2] J. Bekker, P. Robberechts, and J. Davis. Beyond the selected completely at random assumption for learning from positive and unlabeled data. In *Proceedings of the 2019 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, ECML'19, pages 71–85, 2019.

[3] A. Cabannes, V. Rudi and F. Bach. Structured prediction with partial labelling through the infimum loss. In *International Conference on Machine Learning*, ICML'2020, pages 1230–1239, 2020.

[4] T. Cannings, Y. Fan, and R. Samworth. Classification with imperfect training labels. *Biometrika*, pages 311–330, 2020.

[5] J.-L. Chen, J.-J. Cai, Y. Jiang, and S.-J. Huang. PU active learning for recommender systems. *Neural Processing Letters*, 53:3639–3652, 2021.

[6] O. Coudray, C. Keribin, P. Massart, and P. Pamphile. Risk bounds for positive-unlabeled learning under the selected at random ssumption. *Journal of Machine Learning Research*, pages 1–31, 2023.

[7] W. Gerych, T. Hartvigsen, L. Buquicchio, E. Agu, and E. Rundensteiner. Recovering the propensity score from biased positive unlabeled data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI'22, pages 6694–6702, 2022.

[8] C. Gong, Q. Wang, T. Liu, B. Han, J. You, J. Yang, and D. Tao. Instance-dependent positive and unlabeled learning with labeling bias estimation. *IEEE Trans Pattern Anal Mach Intell*, pages 1–16, 2021.

[9] C. Gong, Q. Wang, T. Liu, B. Han, J. J. You, J. Yang, and D. Tao. Instance-dependent positive and unlabeled learning with labeling bias estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44:4163–4177, 2022.

[10] E. Hüllemeier. Learning from imprecise and fuzzy observations: Data disambiguation through generalised loss minimisation. *International Journall of Approximate Reasoning*, 55:1519–1534, 2014.

[11] M. Kato, T. Teshima, and I. Honda. Learning from positive and unlabeled data with selection bias. In *Proceedings of International Conference on Learning Representations*, 2019.

[12] Q. Liang, M. Zhu, Y. Wang, X. Wang, W. Zhao, M. Yang, H. Wei, B. Han, and X. Zheng. Positive distribution pollution: rethinking positive unlabeled learning from a unified perspective. In *Proceedings of the 37 AAAI Conference on Artificial Intelligence AAAI-23*, 2023.

[13] L. Liu and T. Dietterich. Learnability of the superset label learning problem. In *International Conference on Machine Learning*, ICML'2014, pages 1629–1637, 2014.

[14] Y. Luo, S. Cheng, C. Liu, , and F. Jiang. Pu-learning in payload-based web anomaly detection. In *Proceedings of the Third Conference on Security of Smart Cities, industrial Control Systems and Communications*, SSIC'2018, pages 1–5, 2018.

[15] A. Menon, B. Rooyen, C. S. Ong, and B. Williamson. Learning from corrupted binary labels via class-probability estimation. In *Proceedings of the 32nd International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 125–134, 2015.

[16] A. Menon, B. Rooyen, and N. Natarajan. Learning from binary labels with instant-dependent noise. *Machine Learning*, pages 1561–1595, 2018.

[17] B. Na, H. Kim, K. Song, W. Joo, Y.-Y. Kim, and I. Moon. Deep generative positive-unlabeled learning under selection bias. In *Proceedings of CIKM'20*, CIKM '20, pages 1155–1164, New York, NY, USA, 2020. ACM. ISBN 9781450368599.

[18] M. Naumov, M. Mudigere, H. Shi, J. Huang, N. Sundaraman, J. Park, X. Wang, U. Gupta, C.-J. Wu, A. Azzolini, D. Dzhulgakov, A. Mallevich, I. Cherniavskii, Y. Lu, R. Krishnamoorthi, A. Yu, V. Kondratenko, S. Pereira, X. Chen, W. Chen, V. Rao, B. Jia, L. Xiong, and M. Smelyanskiy. Deep learning recommendation model for personalization and recommendation systems. manuscript, 2019. URL arXiv:1906.00091.

[19] J. W. Park, M. K. Chu, J. M. Kim, S. G. Park, and S. J. Cho. Analysis of trigger factors in episodic migraineurs using a smartphone headache diary applications. *PloS one*, 11(2):1–13, 2016.

[20] T. Schnabel, A. Swaminathan, A. Singh, N. Chandak, and T. Joachims. Recommendation as treatments: debiasing learning and evaluation. *ICML*, 48:1670–1679, 2016.

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

[22] E. Shultheis, R. Babbar, M. Wydmuch, and K. Dembczyński. On missing labels, long-tails and propensities in extreme multi-label classification. In *KDD'22*, pages 1547–1557, 2022.

[23] G. Ward, T. Hastie, S. Barry, J. Elith, and J. Leathwick. Presence-only data and the EM algorithm. *Biometrics*, 65:554–563, 2009.

[24] A. Wawrzeńczyk and J. Mielniczuk. One-class classification approach to variational learning from biased positive unlabelled data. In *Proceedings of the European Conference on Artificial Intelligence*, ECAI'23, pages 1720–1727, 2023.

[25] P. Yang, X.-L. Li, J.-P. Mei, C.-K. Kwoh, and S.-K. Ng. Positive-unlabeled learning for disease gene identification. *Bioinformatics*, 28 (20):2640–2647, 8 2012.