Positive Distribution Pollution: Rethinking Positive Unlabeled Learning from a Unified Perspective

Qianqiao Liang[1], Mengying Zhu[1]*, Yan Wang[2], Xiuyuan Wang[1], Wanjia Zhao[1], Mengyuan Yang[1], Hua Wei[3], Bing Han[3], Xiaolin Zheng[1]

[1] College of Computer Science, Zhejiang University, China
[2] School of Computing, Macquarie University, Australia
[3] MYbank, Ant Group, China
{liangqq, mengyingzhu, xiuyuanwang, wanjiaZhao, yangmy412, xlzheng}@zju.edu.cn
yan.wang@mq.edu.au, {shuhu.wh, hanbing.hanbing}@antgroup.com

Abstract

Positive Unlabeled (PU) learning, which has a wide range of applications, is becoming increasingly prevalent. However, it suffers from problems such as data imbalance, selection bias, and prior agnostic in real scenarios. Existing studies focus on addressing part of these problems, which fail to provide a unified perspective to understand these problems. In this paper, we first rethink these problems by analyzing a typical PU scenario and come up with an insightful point of view that all these problems are inherently connected to one problem, i.e., *positive distribution pollution*, which refers to the inaccuracy in estimating positive data distribution under very little labeled data. Then, inspired by this insight, we devise a variational model named `CoVPU`, which addresses all three problems in a unified perspective by targeting the positive distribution pollution problem. `CoVPU` not only accurately separates the positive data from the unlabeled data based on discrete normalizing flows, but also effectively approximates the positive distribution based on our derived unbiased rebalanced risk estimator and supervises the approximation based on a novel prior-free variational loss. Rigorous theoretical analysis proves the convergence of `CoVPU` to an optimal Bayesian classifier. Extensive experiments demonstrate the superiority of `CoVPU` over the state-of-the-art PU learning methods under these problems.

Introduction

Positive Unlabeled (PU) learning has attracted great attention in recent years, which is widely used in scenarios such as information retrieval (Dupret and Piwowarski 2008), anomaly detection (Pang, Shen, and van den Hengel 2019) and disease diagnosis (Chen et al. 2020b). PU learning refers to a binary classification task that only part of the samples from the positive class is labeled while the remaining hard-distinguished positive samples and the samples from the negative class are unlabeled. In real practice, some challenging problems, e.g., data imbalance, selection bias, and prior agnostic, remain unsolved, which reduces the performance of PU learning. Therefore, it is necessary to address these challenging problems for effective PU learning in practice.

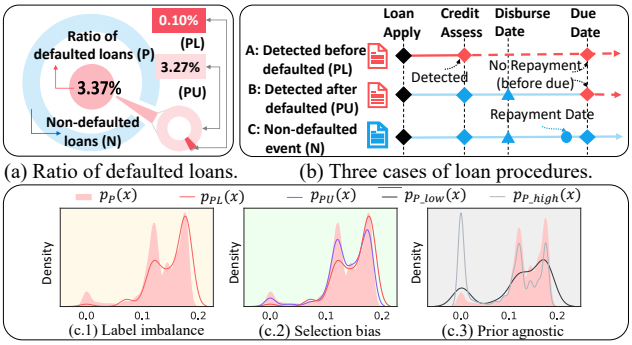
*Corresponding author

Problem Definition. (**P1**) *Data imbalance* refers to the phenomenon that the available labeled data is far less than the unlabeled data. Note that the very few labels of positive data often originate from an extremely imbalanced class ratio, so class imbalance should also be considered in this problem. (**P2**) *Selection bias* refers to the phenomenon that the selected labeled data is biased and can not represent all positive data. (**P3**) *Prior agnostic* refers to the phenomenon that the class prior (Jain et al. 2020) in hindsight is unknown.

Existing PU learning studies address these problems from different perspectives. Firstly, from the *perspective of investigating data features*, existing methods extract additional positive data (Hu et al. 2021) or its features (Na et al. 2020) used for mitigating the selection bias or data imbalance problems. In addition, from the *perspective of samples selection mechanisms*, existing methods address the data imbalance problem by oversampling the positive data (Su, Chen, and Xu 2021; Hu et al. 2021), which, however, becomes unnecessary data repetitions when the data itself is selected with bias. Both the above two perspectives rely on the input of class prior, which is agnostic in real practice. To address this problem, some methods look into the *perspective of data's intrinsic properties* for class prior estimation (Bekker and Davis 2018; Jain et al. 2020). However, their estimations tend to be collapsed unless the selection bias and data imbalance problems are well addressed. Overall, each of these perspectives addresses partial problems and lacks a unified understanding of all the three problems.

In this paper, we rethink the above problems and raise a question: *Whether there exists a unified perspective for addressing these challenging problems simultaneously?*

Motivating Example. To answer this question, we take a real loan scenario in a bank as an example, aiming to gain deeper insights into the generality of these challenging problems. Figures 1(a) and (b) depict the characteristics and procedure of the loan scenario. Obviously, this is a typical PU learning scenario. In the loan procedures, the bank experts first detect high-risk loan requests before they become defaulted (Positive Labeled, PL) according to their expertise, and only offer loans to the remaining requests (Unlabeled, U). Subsequently, some of the loans will be detected after they have defaulted (Positive Unlabeled, PU), and some will maintain benign (Negative, N). In Figure 1(c), we depict the



(a) Ratio of defaulted loans. (b) Three cases of loan procedures.

(c) Positive distributions under three challenges. Note that (c.3) depicts positive distribution estimation under higher or lower deviated class prior.

Figure 1: A real-world motivating example from a bank.

distributions of positive data in this loan scenario under the three problems, from which we have the following three observations. Firstly, the positive labeled distribution deviates from the positive distribution. This is caused by the data imbalance problem, which is common in loan scenarios where defaulters are in the minority and only a tiny fraction of the defaulters are labeled for not disturbing normal customers. This deviation phenomenon makes sense because fitting a distribution with very little data is highly unstable, in which very small changes in data can lead to large perturbations in the estimation. Secondly, the positive labeled distribution deviates from the positive unlabeled distribution. The positive labeled data is selected with bias because of experts' limited knowledge in recognizing all defaulters. Completely trusting all labeled data with selection bias inevitably leads to inaccuracy in distribution estimation. Thirdly, the positive distribution is estimated inaccurately if the inaccurate class prior, i.e., higher deviated or lower deviated value, is provided. In real practice, the underlying defaulted class ratio, i.e., positive class prior, is agnostic to the bank, and the deviated class priors will lead to inaccurate estimations of the positive distribution. Theoretically, given a mixture distribution without class priors, there will be a variety of component separations, and the positive distribution can not be accurately estimated unless the correct class prior is provided, which explains this observation.

In conclusion, all the three challenging problems can be unified as the problem of inaccurate distribution estimation of positive data under very little labeled data, which we term as *Positive Distribution Pollution*.

In this paper, to address the positive distribution pollution problem, i.e., the unified perspective of the challenging data imbalance, selection bias, and prior agnostic problems, we propose a novel PU learning model named `CoVPU`, which includes three key modules, i.e., VPU-Calibrator, VPU-Classifier, and VPU-Collaborator. Firstly, according to our aforementioned analysis, only utilizing the very little and biased labeled data will lead to pollution in positive distribution estimation, so we design VPU-Calibrator to additionally exploit the unlabeled data, i.e., mining positive unlabeled samples and their distributions, for improving the accuracy of the estimation. Specifically, the VPU-Calibrator separates the two-component mixture distribution of the unlabeled data based on discrete normalizing flows, which is used for fitting of the underlying diverse positive distribution (*for addressing* **P1** *and* **P2**). Secondly, to supervise the learning of the positive distribution estimation, we design VPU-Classifier with a novel prior-free variational classifier learning mechanism, which not only has a relaxed data selection assumption (*for addressing* **P2**), but also accurately measures the accuracy of the estimated positive distribution under the data imbalance and prior agnostic problems (*for addressing* **P1** *and* **P3**). Specifically, the VPU-Classifier first inferences the positive distribution based on a Bayesian classifier, then leverages our devised variational loss to estimate the distance between the estimated positive distribution and an ideal one for supervising the learning of the positive distribution. Thirdly, because VPU-Calibrator and VPU-Classifier are unable to process individually and are in need of the knowledge from each other for final decisions, we designed VPU-Collaborator, which fully exploits the two modules' positive consensus, i.e., samples whose component assignment and class assignment are both positive, on the unlabeled data to further boost the learning of both the VPU-Calibrator and VPU-Classifier.

To the best of our knowledge, this is the first work in the literature to provide a unified perspective, i.e., positive distribution pollution, for understanding and addressing the challenging data imbalance, selection bias, and prior agnostic problems in PU learning. Our main contributions are as follows: (1) *Model*: we propose a novel variational model named `CoVPU` for positive distribution modeling, which simultaneously addresses all the three challenging problems from a unified perspective; (2) *Theory*: we theoretically prove that `CoVPU` converges to the optimal classifier. (3) *Experiment*: we have conducted extensive experiments, which demonstrate the superiority of `CoVPU` over the state-of-the-art PU learning methods.

Related Work

PU learning has attracted increasing attention in recent years (Bekker and Davis 2018; Jaskie and Spanias 2019). Early works employed two-step strategies (Li and Liu 2003; Liu et al. 2002; Yu, Han, and Chang 2002), which lack rigorous guarantee on convergence. Risk estimation based methods enhance theory guidance (Plessis, Niu, and Sugiyama 2014; Kiryo et al. 2017). To enhance the feasibility, recent studies target three challenging problems, i.e., data imbalance, selection bias, and prior agnostic, separately.

For the data imbalance problem, most existing studies focus on enlarging the positive sample size. Resampling methods repeat the existing data without any modification, leading to the lack of data diversity (Su, Chen, and Xu 2021). Generative methods learns data distribution for increasing the diversity (Hou et al. 2018). However, the quality of the generated data cannot be guaranteed, which introduces uncertainty in classifiers.

For the selection bias problem, most existing studies exploit diverse data based on various assumptions such as order invariance (Kato, Teshima, and Honda 2019) and negative sample invariance (Hammoudeh and Lowd 2020). How-

ever, their performance can not be guaranteed once their assumptions contradict reality. Another attempt (Na et al. 2020) generates positive samples based on a given class prior that which may be agnostic in reality.

For the class prior agnostic problem, methods can be divided into two kinds. The first kind (Zeiberg, Jain, and Radivojac 2020; Perini, Vercruyssen, and Davis 2020; Akujuobi et al. 2020) estimates class priors (Jain et al. 2020) separately without designing much on the downstream classifier. The second kind (Chen et al. 2020a; Zhang, Hou, and Zhang 2020; Hu et al. 2021; Ivanov 2020) unifies the class prior estimation and the classifier, which regard the classifier's expectation as class prior for further adjusting the classifier. However, these methods are designed based on the selected completely at random assumption, which may perform poorly under selection bias scenarios.

In general, each of the above studies targets partial problem only based on various assumptions or techniques. To the best of our knowledge, no existing study can provide a unified perspective to simultaneously address the above three challenging problems in PU learning.

Problem Settings and Preliminaries

Problem Settings

In PU learning, a dataset $\mathcal{X}$ of size n is composed of two sets, i.e., $\mathcal{X}_{\mathrm{PL}}$ of size n_{PL} containing positive labeled data and $\mathcal{X}_{\mathrm{U}}$ of size n_{U} containing unlabeled data. In hindsight, $\mathcal{X}_{\mathrm{U}}$ is composed of a positive unlabeled dataset $\mathcal{X}_{\mathrm{PU}}$ of size n_{PU} and a negative dataset $\mathcal{X}_{\mathrm{N}}$ of size n_{N}. Therefore, $|\mathcal{X}_{\mathrm{P}}| = n_{\mathrm{P}} = n_{\mathrm{PL}}+n_{\mathrm{PU}}$ and $|\mathcal{X}| = n = n_{\mathrm{P}}+n_{\mathrm{N}}$. We denote the main notations related to three challenging problems as follows.

Definition 1 (Imbalance Ratio for **P1**). *Given a PU dataset, the imbalance ratio is defined as IR* $= \frac{n_{PL}}{n_U} \in [0,1]$. *Data imbalance refers to the phenomenon that IR* $\ll 0.5$.

Definition 2 (Selection Bias for **P2**). *Let* $y \in \{-1,+1\}$ *be the negative or positive label of a sample* $\mathbf{x} \sim p(\mathbf{x})$ *and* $o \in \{-1,+1\}$ *be its unlabeled or labeled indicator. The labeled and unlabeled samples are drawn independently as*

$$\begin{aligned}\mathcal{X}_{PL} &= \{\mathbf{x}_i\}_{i=1}^{n_{PL}} \overset{i.i.d}{\sim} p_{PL}(\mathbf{x}) \triangleq \mathbb{P}(\mathbf{x} \mid y=+1, o=+1),\\ \mathcal{X}_{PU} &= \{\mathbf{x}_i\}_{i=1}^{n_{PU}} \overset{i.i.d}{\sim} p_{PU}(\mathbf{x}) \triangleq \mathbb{P}(\mathbf{x} \mid y=+1, o=-1),\\ \mathcal{X}_{N} &= \{\mathbf{x}_i\}_{i=1}^{n_{N}} \overset{i.i.d}{\sim} p_{N}(\mathbf{x}) \triangleq \mathbb{P}(\mathbf{x} \mid y=-1, o=-1).\end{aligned} \tag{1}$$

Selection bias refers to the phenomenon that $p_{PL} \neq p_{PU}$.

Definition 3 (Class Prior for **P3**). *Class prior is* $\pi_P = \mathbb{P}(y=+1) = \frac{n_P}{n}$. *Additionally, we define* $\pi_{PL} = \frac{n_{PL}}{n} = \frac{IR}{IR+1}$, $\pi_{PU} = \frac{n_{PU}}{n}$, $\pi_N = \frac{n_N}{n}$ *and* $\pi_U = \frac{n_U}{n}$. *Prior agnostic means that the underlying* π_P, π_{PU}, *and* π_N *are unavailable.*

In this paper, we aim to learn a Bayesian classifier $\Phi(\mathbf{x})$ to predict each sample's label y with the goal of letting $\Phi(\mathbf{x})$ close to the optimal classifier $\Phi^*(\mathbf{x}) \triangleq \mathbb{P}(y=+1|\mathbf{x})$.

Preliminaries

Risk estimation based methods are effective PU learning methods with rigorous theory to guarantee the performance of classifiers. They are also important preliminaries for addressing the positive distribution pollution.

Traditional Risk Estimator. In traditional risk estimation based methods, a risk estimator is used to calculate the loss $\ell(\cdot,\cdot)$ induced in the positive label set and unlabeled set weighted by the samples' class prior as follows (Plessis, Niu, and Sugiyama 2014):

$$\begin{aligned}\mathcal{R}_{\mathrm{PU}}(\Phi) =& \pi_{\mathrm{P}}\mathbb{E}_{p_{\mathrm{PL}}}[\ell_+(\Phi(\mathbf{x}))] + \mathbb{E}_{p_{\mathrm{U}}}[\ell_-(\Phi(\mathbf{x}))]\\ &- \pi_{\mathrm{P}}\mathbb{E}_{p_{\mathrm{PL}}}[\ell_-(\Phi(\mathbf{x}))],\end{aligned} \tag{2}$$

where $\ell(\cdot)$ is any trainable loss function.

Such risk estimator is inapplicable when selection bias or data imbalance exists, which pollutes the positive expectation estimation terms.

Selection Unbiased Risk Estimator. For the selection bias problem, the above traditional risk estimator can be separated according to $\mathcal{X}_{\mathrm{PL}}$ and $\mathcal{X}_{\mathrm{PU}}$ as follows (Na et al. 2020):

$$\begin{aligned}\mathcal{R}_{\mathrm{SU}}(\Phi) =& \pi_{\mathrm{PL}}\mathbb{E}_{p_{\mathrm{PL}}}[\ell_+(\Phi(\mathbf{x}))] + \pi_{\mathrm{U}}\mathbb{E}_{p_{\mathrm{U}}}[\ell_-(\Phi(\mathbf{x}))]\\ &+ \pi_{\mathrm{PU}}\mathbb{E}_{p_{\mathrm{PU}}}[\ell_+(\Phi(\mathbf{x})) - \ell_-(\Phi(\mathbf{x}))],\end{aligned} \tag{3}$$

which distinguishes between labeled and unlabeled positive samples to effectively approximate the positive expectations.

Rebalance Risk Estimator. For the data imbalance problem, pseudo-oversampling technique (Su, Chen, and Xu 2021) can be applied to increase the proportion of positive samples for rebalancing. Specifically, the pseudo-oversampling technique increases the small class prior π_{P} to a larger π' by modifying the risk only without involving an actual oversampling procedure as follows:

$$\begin{aligned}\mathcal{R}_{\text{balancePN}}(\Phi) =& \pi'\mathbb{E}_{p_{\mathrm{P}}}[\ell_+(\Phi(\mathbf{x}))] + \left(1-\pi'\right)\mathbb{E}_{p_{\mathrm{N}}}[\ell_-(\Phi(\mathbf{x}))].\end{aligned} \tag{4}$$

Method

Overview

The architecture of `CoVPU` is presented in Figure 2, which contains three key modules, i.e., VPU-Calibrator, VPU-Classifier, and VPU-Collaborator. Specifically, VPU-Calibrator separates the latent two-component mixture distribution of unlabeled samples based on discrete normalizing flows within a variational auto-encoder framework. VPU-Classifier leverages a novel prior-free variational classifier learning mechanism to minimize the distance between our learned Φ classifier and an optimal one Φ^*, thereby boosting the inference of positive distribution. Because both VPU-Calibrator and VPU-Classifier are not self-contained and rely on knowledge from each other, VPU-Collaborator leverages our devised consensus learning mechanism to fully exploit the two modules' positive consensus, i.e., samples whose component assignment and class assignment are both positive, on the unlabeled data to further boost the learning of both VPU-Calibrator and VPU-Classifier.

VPU-Calibrator: Variational PU Calibrator

Positive data buried in the unlabeled data is relatively rich and diverse to calibrate the polluted positive distribution and address **P1**, i.e., data imbalance, and **P2**, i.e., selection bias. Because there is no supervision in unlabeled data, we design an unsupervised module, named VPU-Calibrator, based

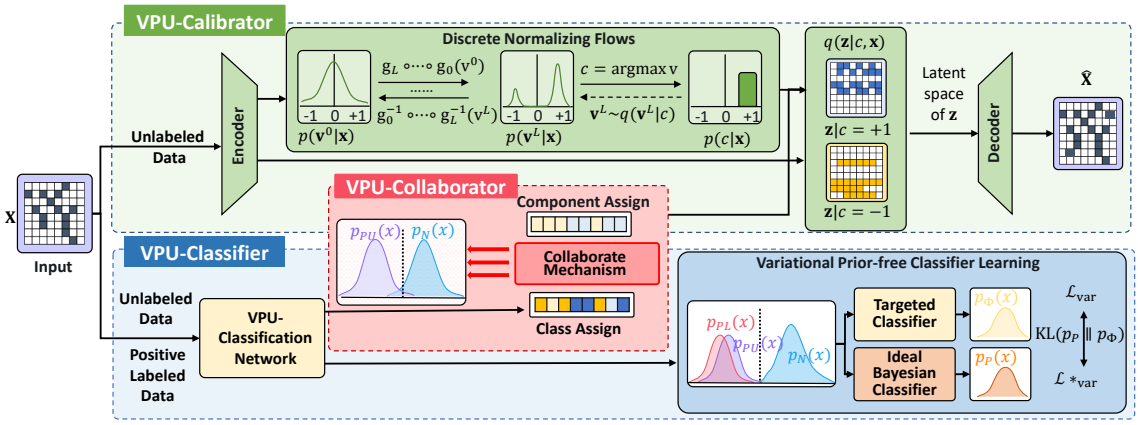


Figure 2: The architecture of CoVPU, which includes the VPU-Calibrator, VPU-Classifier, and VPU-Collaborator modules.

on the encoder-decoder architecture to separate the complex two-component, i.e., positive component and negative component, mixture distribution in the unlabeled data. We elaborate on the architecture and the optimization process.

Variational Assumption. We assume the latent feature $\mathbf{z}$ of unlabeled data obeys a two-component Gaussian mixture distribution and denote the component indicator as $c \in \{-1, +1\}$. Given a sample $\mathbf{x} \in \mathcal{X}_{\mathrm{U}}$, the joint probability is:

$$p(\mathbf{x}, \mathbf{z}, c) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z}|c)p(c), \tag{5}$$

where $p(c) = \mathrm{Cat}(\boldsymbol{\pi}_c)$ is a categorical distribution parameterized by $\boldsymbol{\pi}_c = [\frac{\pi_{\mathrm{PU}}}{\pi_{\mathrm{U}}}, \frac{\pi_{\mathrm{U}} - \pi_{\mathrm{PU}}}{\pi_{\mathrm{U}}}]$, $p(\mathbf{z}|c) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{z}}(c), \boldsymbol{\sigma}^2_{\mathbf{z}}(c)\mathbf{I})$ is a two-component Gaussian mixture distribution controlled by c, and $p(\mathbf{x}|\mathbf{z})$ can be multivariate Bernoulli, i.e., $\mathrm{Ber}(\boldsymbol{\mu}_{\mathbf{x}}(\mathbf{z}))$ or Gaussian, i.e., $\mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}(\mathbf{z}), \boldsymbol{\sigma}^2_{\mathbf{x}}(\mathbf{z})\mathbf{I})$ distribution controlled by $\mathbf{z}$ according to data's characteristic.

The main challenge of approximating the distribution in unlabeled data lies in the discrete separation of components, i.e., c obeys a categorical distribution, which is very hard to be optimized. Therefore, we approximate it by discrete normalizing flows, which bridges the gap between a complex categorical distribution and a continuous base distribution for modeling the complicated data distribution based on a stack of several planar transformations followed by an argmax transformation.

Encoder. We can take an inference model as an encoder, which approximates the distribution $p(\mathbf{z}, c)$ using a posterior distribution $q_\phi(\mathbf{z}, c|\mathbf{x}) = q_{\phi_c}(c|\mathbf{x})q_{\phi_{\mathbf{z}}}(\mathbf{z}|c, \mathbf{x})$ parameterized by trainable parameters ϕ_c, and $\phi_{\mathbf{z}}$.

Firstly, to increase the expressiveness of $q_{\phi_c}(c|\mathbf{x})$, we adopt normalizing flows to transform $\mathbf{v}^0 \in \mathbb{R}^2$, i.e., the latent features obeying a simple Gaussian distribution, to a complex $\mathbf{v}^L \in \mathbb{R}^2$ that obeys a complex distribution. Then, due to c's discrete property, we apply an argmax transformation to get $c = \arg\max_k \mathbf{v}^L_k$. Specifically, we first encode the variable $\mathbf{v}^0$ sampled from a simple Gaussian distribution $p(\mathbf{v}^0|\mathbf{x})$ by $[\boldsymbol{\mu}_{\mathbf{v}^0}, \boldsymbol{\sigma}_{\mathbf{v}^0}] = f_{\mathbf{v}}(\mathbf{x}; \phi_{\mathbf{v}})$, where $f_{\mathbf{v}}$ denotes a neural network with parameters $\phi_{\mathbf{v}}$. Then, we adopt planar transformations (Rezende and Mohamed 2015) to generate the complicated $\mathbf{v}^L$ via several layers, each of which is $\mathbf{v}^{i+1} = g_i(\mathbf{v}^i) = \mathbf{v}^i + \mathbf{u}^i\sigma(\mathbf{w}^{i\top}\mathbf{v}^i + a^i)$ and $\sigma(\cdot)$ denotes the tanh activation function. After that, we apply the argmax transformation to map the continuous $\mathbf{v}^L$ to a discrete c using $c = \arg\max_k \mathbf{v}^L_k$.

Secondly, the variable $\mathbf{z}$ can be encoded as $q_{\phi_{\mathbf{z}}}(\mathbf{z}|c, \mathbf{x}) = f_{\mathbf{z}}(c, \mathbf{x}; \phi_{\mathbf{z}})$, where $f_{\mathbf{z}}$ denote neural networks with parameters $\phi_{\mathbf{z}}$ for learning the distributions of the latent features from training samples.

Decoder. We apply a generative model as the decoder to reconstruct the data. More specifically, for a sample $\mathbf{x}$, we model its generative process as follows:

$$\begin{aligned} &\mathbf{v}^0 \sim p(\mathbf{v}^0), \qquad \mathbf{v}^L = g_L \circ g_{L-1} \circ \cdots \circ g_0(\mathbf{v}^0), \\ &c = \arg\max(\mathbf{v}^L), \quad \mathbf{z} \sim p(\mathbf{z}|c), \quad p(\mathbf{x}|\mathbf{z}) = f_{\mathbf{x}}(\mathbf{z}; \theta_{\mathbf{x}}), \end{aligned} \tag{6}$$

where $f_{\mathbf{x}}$ are networks parameterized by $\theta_{\mathbf{x}}$, and L is the number of the planar flows layers.

Variational Lower Bound. The objective of the VPU-Calibrator is maximizing the given data's log likelihood $\log p(\mathbf{x})$. Given the generative process in Eq. (6), by using Jensen's inequality, $\log p(\mathbf{x})$ can be written as:

$$\begin{aligned} \log p(\mathbf{x}) &= \log \int_{\mathbf{z}} \sum_c p(\mathbf{x}, \mathbf{z}, c) \mathrm{d}\mathbf{z} \\ &\geq E_{q(\mathbf{z},c|\mathbf{x})} \log \frac{p(\mathbf{x}, \mathbf{z}, c)}{q(\mathbf{z}, c \mid \mathbf{x})} = \mathcal{L}'_{\mathrm{ELBO}}(\mathbf{x}), \end{aligned} \tag{7}$$

where $\mathcal{L}'_{\mathrm{ELBO}}(\mathbf{x})$ is the evidence lower bound (ELBO).

According to Eq. (7), $\mathcal{L}'_{\mathrm{ELBO}}(\mathbf{x})$ can be decomposed as:

$$\begin{aligned} \mathcal{L}'_{\mathrm{ELBO}}(\mathbf{x}) &= E_{q(\mathbf{z},c|\mathbf{x})}[\log p(\mathbf{x}, \mathbf{z}, c) - \log q(\mathbf{z}, c|\mathbf{x})] \\ &= E_{q(\mathbf{z},c|\mathbf{x})}[\log \frac{p(c)}{q(c|\mathbf{x})} + \log \frac{p(\mathbf{z}|c)}{q(\mathbf{z}|c, \mathbf{x})} + \log p(\mathbf{x}|\mathbf{z})]. \end{aligned} \tag{8}$$

While calculating the last two terms is straightforward, the main difficulty lies in optimizing $q(c|\mathbf{x})$ in the first term. It can be decomposed as:

$$\begin{aligned} q(c \mid \mathbf{x}) &= \int_p (c \mid \mathbf{v}^L) q(\mathbf{v}^L \mid \mathbf{x}) \mathrm{d}\mathbf{v}^L, \\ p(c \mid \mathbf{v}^L) &= \delta(c = \arg\max(\mathbf{v}^L)), \end{aligned} \tag{9}$$

which is intractable, so we resort to variational inference and specify a variational distribution $q(\mathbf{v}^L|c)$. Due to the

$\arg\max$ constraint (Hoogeboom et al. 2021), the variational distribution $q(\mathbf{v}^L|c)$ should have support limited to $\mathcal{S}(c) = \{\mathbf{v}^L \in \mathbb{R}^2 : c = \arg\max \mathbf{v}^L\}$. Then, we have

$$\log q(c \mid \mathbf{x}) \geq \log q(\mathbf{v}^L \mid \mathbf{x}) + E_{q(\mathbf{v}^L|c)}[-\log q(\mathbf{v}^L \mid c)], \quad (10)$$

where $\log q(\mathbf{v}^L|\mathbf{x}) = \log q(\mathbf{v}^0|\mathbf{x}) - \sum_{i=0}^{L-1} \log|\det \frac{\mathrm{d}\mathbf{v}^{i+1}}{\mathrm{d}\mathbf{v}^i}|$. When calculating the expectation in terms of $q(\mathbf{v}^L \mid c)$, we first adopt a surjective flow layer (Nielsen et al. 2020) with Gumbel distribution (Kool, Van Hoof, and Welling 2019) parameterized by $\boldsymbol{\pi}_c$ to obtain the variational distribution $q(\mathbf{v}^L|c)$, then sample $\mathbf{v}^L$ by a Gumbel sampling algorithm (see Alg. 4 in (Hoogeboom et al. 2021)).

By substituting the term in Eq. (8) with Eq. (5), Eq. (10) and using the reparameterization trick, the $\mathcal{L}'_{\text{ELBO}}(x)$ can be upper bounded by:

$$\begin{aligned}
&\mathcal{L}_{\text{ELBO}}(\mathbf{x}) = \mathbb{I}(q(c|\mathbf{x}) = 1) \log \frac{\pi_{\text{PU}}}{\pi_{\text{U}}} + \mathbb{I}(q(c|\mathbf{x}) = 0) \log \frac{\pi_{\text{U}} - \pi_{\text{PU}}}{\pi_{\text{U}}} \\
&- \sum_c q(c|\mathbf{x})[\log q(\mathbf{v}^0|\mathbf{x}) - \sum_{i=0}^{L-1} \log|\det \frac{\mathrm{d}\mathbf{v}^{i+1}}{\mathrm{d}\mathbf{v}^i}| + \mathbb{E}_{q(\mathbf{v}^L|c)}[-\log q(\mathbf{v}^L|c)]] \\
&- \sum_c q(c|\mathbf{x}) \,\mathrm{KL}(q(\mathbf{z}|c,\mathbf{x}))||p(\mathbf{z}|c)) + \mathbb{E}_{q(c,\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z})],
\end{aligned} \quad (11)$$

where KL refers to Kullback–Leibler divergence. The detailed deviation of Eq. (11) is presented in our longer version.

VPU-Classifier: Variational PU Classifier

To be able to leverage the component separation learned from VPU-Calibrator for addressing **P1** and **P2**, we first derive an unbiased and rebalanced risk estimator. Then, we analyze the disadvantage of this risk estimator, and design a prior-free variational loss based on this risk estimator, which is used to train our Bayesian classifier under the three challenging problems **P1**, **P2**, and **P3** more effectively.

Unbiased and Rebalanced Risk Estimator. Combining Eq. (3) and Eq. (4), the adjusted risk is:

$$\begin{aligned}
\mathcal{R}(\Phi) = &\underbrace{\frac{1}{2}(\pi' + \pi_{\text{PL}})\mathbb{E}_{p_{\text{PL}}}[\ell_+(\Phi(\mathbf{x}))]}_{\text{classify PL samples}} + \underbrace{\frac{1-\pi'}{1-\pi_{\text{P}}}\pi_{\text{U}}\mathbb{E}_{p_{\text{U}}}[\ell_-(\Phi(\mathbf{x}))]}_{\text{classify U samples}} \\
&+ \underbrace{\frac{1}{2}(\pi' - \pi_{\text{PL}})\,\mathbb{E}_{p_{\text{PU}}}[\ell_+(\Phi(\mathbf{x}))] - \frac{1-\pi'}{1-\pi_{\text{P}}}\pi_{\text{PU}}\mathbb{E}_{p_{\text{PU}}}[\ell_-(\Phi(\mathbf{x}))]}_{\text{classify PU samples}}.
\end{aligned} \quad (12)$$

The detailed derivation of Eq. (12) can be found in our longer version.

Obviously, this unbiased and rebalanced risk estimator derived based on Eq. (3) and Eq. (4) is able to address the data imbalance (**P1**) and selection bias (**P2**) problems. However, it highly depends on class priors that are often unavailable in reality.

Prior-Free Variational Classifier Learning. In order to overcome the disadvantage of the unbiased and rebalanced risk estimator, we first recall our insight that the three challenging problems can be unified as the positive distribution pollution problem. Then, we devise a prior-free variational classifier learning mechanism to supervise the classifie's learning based on a variational loss, which measures the difference between the polluted positive distribution and the underlying positive distribution instead of measuring the expection of estimation error. Such variational loss on top of distribution is much more practical because it is prior-free.

Firstly, because $\Phi(\cdot)$ is a Bayesian classifier, we choose a trainable loss function $\ell_+(\Phi(\mathbf{x})) = -\log(\Phi(\mathbf{x}))$ and $\ell_-(\Phi(\mathbf{x})) = -\log(1-\Phi(\mathbf{x}))$ in Eq. (12), and infer the polluted positive distribution and the underlying positive distribution based on Eq. (12) as:

$$p_{\text{P}}(\mathbf{x}) = \frac{\Phi^*(\mathbf{x})\big(1-\pi'\big)}{\pi'(1-\Phi^*(\mathbf{x}))} p_{\text{N}}(\mathbf{x}), \quad p_\Phi(\mathbf{x}) = \frac{\Phi(\mathbf{x})p(\mathbf{x})}{\mathbb{E}_p[\Phi(\mathbf{x})]}. \quad (13)$$

Then, the approximation quality of the classifier Φ can be evaluated by the KL divergence between p_{P} and p_Φ, i.e., $\mathrm{KL}\left(p_{\text{P}} \| p_\Phi\right)$. The above analysis leads to Theorem 1.

Theorem 1. *For all $\Phi : \mathbb{R}^d \mapsto [0,1]$ with $\mathbb{E}_P[\Phi(\mathbf{x})] > 0$,*

$$\mathrm{KL}\left(p_P \| p_\Phi\right) = \mathcal{L}_{var}(\Phi) - \mathcal{L}_{var}(\Phi^*), \quad (14)$$

where $\mathcal{L}_{var}(\Phi)$ is the prior-free variational loss derived based on Eq.(12) as

$$\begin{aligned}
\mathcal{L}_{var}(\Phi) = &\log((\frac{\pi' + \pi_{PL}}{2})\mathbb{E}_{p_{PL}}[\Phi(\mathbf{x})] \\
&+ (\frac{\pi' - \pi_{PL}}{2})\mathbb{E}_{p_{PU}}[\Phi(\mathbf{x})] + (1-\pi')\mathbb{E}_{p_N}[\Phi(\mathbf{x})]) \\
&- (\frac{\pi' + \pi_{PL}}{2\pi'})\mathbb{E}_{p_{PL}}[\log(\Phi(\mathbf{x}))] - (\frac{\pi' - \pi_{PL}}{2\pi'})\mathbb{E}_{p_{PU}}[\log(\Phi(\mathbf{x}))].
\end{aligned} \quad (15)$$

Since the KL divergence is always nonnegative, $\mathcal{L}_{\text{RVI}}(\Phi)$ provides a variational upper bound of $\mathcal{L}_{\text{RVI}}(\Phi^*)$, and minimizing $\mathcal{L}_{\text{RVI}}(\Phi)$ equals narrowing p_{P} and p_Φ.

VPU-Collaborator

Despite the effectiveness of VPU-Calibrator and VPU-Classifier, they are unable to process individually. Specifically, VPU-Calibrator requires a prior distribution with parameters $\boldsymbol{\pi}_c$ that can be estimated from VPU-Classifier, whereas VPU-Classifier needs the two components' separation from VPU-Calibrator to complete the calculation of the prior-free variational loss, i.e., Eq. (14). VPU-Collaborator bridges the two modules based on our devised consensus learning mechanism, which leverages the two modules' positive consensus on the unlabeled data to calculate the necessary information for the two modules and completes CoVPU's training process.

The overall training process is presented in Algorithm 1. Specifically, VPU-Calibrator first separates the unlabeled set into two components (Line 3) and constructs a positive class predicted set $\widetilde{X}_{\text{PU}}$ based on the Bayesian Classifier from the unlabeled set $\mathcal{X}_{\text{U}}$ (Line 4). Then, it selects the top J highest confident samples, i.e., samples with the highest predicted $\Phi(\mathbf{x})$, from $\widetilde{X}_{\text{PU}}$ (Line 5). The component with more high-confident samples is set to be the positive component. Then, it constructs a pseudo positive unlabeled set $\hat{\mathcal{X}}_{\text{PU}}$ with a consensus, i.e., samples that have positive class assignments and positive component assignments simultaneously is supposed to be positive unlabeled samples (Lines 6-10). After that, CoVPU's loss is calculated based on the sets $\mathcal{X}$, $\mathcal{X}_{\text{PL}}$, and $\hat{\mathcal{X}}_{\text{PU}}$ (Lines 11-12) for model update (Line 13). Finally, the priors are estimated based on the Bayesian classifier according to (Chen et al. 2020a) (Line 14). Note that we adopt the same MixUp based regularization technique in (Chen et al. 2020a) to avoid overfitting.

Algorithm 1: Training process of CoVPU

Input: datasets $\mathcal{X}_{\mathrm{PL}}$ and $\mathcal{X}_{\mathrm{U}}$, number of query nodes J, number of pretrain epochs T, rebalanced class prior π', imbalance ratio IR.

Output: VPU-Classifier $\Phi(x)$ for PU task.

Initial class prior $\pi_{\mathrm{U}} = 1 - \frac{\mathrm{IR}}{1+\mathrm{IR}}$, $\hat{\pi}_{\mathrm{PU}} = \frac{\hat{\pi}_{\mathrm{U}}}{2}$.

while *not converged* **do**

  $\mathcal{X}_{\mathrm{U}}^1, \mathcal{X}_{\mathrm{U}}^2 \leftarrow \mathrm{VPU\_Calibrator}(\hat{\pi}_{\mathrm{PU}}, \pi_{\mathrm{U}}, \mathcal{X}_{\mathrm{U}})$.

  $\widetilde{\mathcal{X}}_{\mathrm{PU}} \leftarrow \mathrm{VPU\_Classifier}(\mathcal{X}_{\mathrm{U}})$.

  $\widetilde{\mathcal{X}}_{\mathrm{PU}}^{\text{high\_c}} \leftarrow \mathrm{Top\_k}(\widetilde{\mathcal{X}}_{\mathrm{PU}}, J)$.

  $\widetilde{\mathcal{X}}_{\mathrm{PU}}^{1,\text{high\_c}} \leftarrow \mathcal{X}_{\mathrm{U}}^1 \cap \widetilde{\mathcal{X}}_{\mathrm{PU}}^{\text{high\_c}}, \quad \widetilde{\mathcal{X}}_{\mathrm{PU}}^{2,\text{high\_c}} \leftarrow \mathcal{X}_{\mathrm{U}}^2 \cap \widetilde{\mathcal{X}}_{\mathrm{PU}}^{\text{high\_c}}$.

  if $|\widetilde{\mathcal{X}}_{PU}^{1,high\_c}| > |\widetilde{\mathcal{X}}_{PU}^{2,high\_c}|$ **then**

    $\hat{\mathcal{X}}_{\mathrm{PU}} \leftarrow \mathcal{X}_{\mathrm{U}}^1 \cap \widetilde{\mathcal{X}}_{\mathrm{PU}}$.

  else

    $\hat{\mathcal{X}}_{\mathrm{PU}} \leftarrow \mathcal{X}_{\mathrm{U}}^2 \cap \widetilde{\mathcal{X}}_{\mathrm{PU}}$.

  Approximate $\mathcal{L}_{\mathrm{var}}$ by

$$\mathcal{L}'_{\mathrm{var}} = \log\Big(\frac{(\pi'+\pi_{\mathrm{PL}})\sum_{\mathbf{x}\in\hat{\mathcal{X}}_{\mathrm{PL}}}\Phi(\mathbf{x})}{2|\hat{\mathcal{X}}_{\mathrm{PL}}|} + \frac{(\pi'-\pi_{\mathrm{P}})\sum_{\mathbf{x}\in\hat{\mathcal{X}}_{\mathrm{PU}}}\Phi(\mathbf{x})}{2\hat{\mathcal{X}}_{\mathrm{PU}}} + \frac{(1-\pi')\sum_{\mathbf{x}\in\mathcal{X}_{\mathrm{U}}-\hat{\mathcal{X}}_{\mathrm{PU}}}\Phi(\mathbf{x})}{|\mathcal{X}_{\mathrm{U}}-\hat{\mathcal{X}}_{\mathrm{PU}}|}\Big) - \frac{\pi_{\mathrm{PL}}\sum_{\mathbf{x}\in\hat{\mathcal{X}}_{\mathrm{PL}}}\log(\Phi(\mathbf{x}))}{\pi'|\hat{\mathcal{X}}_{\mathrm{PL}}|} - \frac{(\pi'-\pi_{\mathrm{PL}})\sum_{\mathbf{x}\in\hat{\mathcal{X}}_{\mathrm{PU}}}\log(\Phi(\mathbf{x}))}{\pi'|\hat{\mathcal{X}}_{\mathrm{PU}}|}.$$

  Calculate $\mathcal{L}_{\mathrm{ELBO}}$ according to Eq.(11).

  Update CoVPU with loss $\mathcal{L} = -\mathcal{L}_{\mathrm{ELBO}} + \mathcal{L}'_{\mathrm{var}}$.

  Re-estimate $\hat{\pi}_P = \frac{\sum_{\mathbf{x}\in\mathcal{X}}\Phi(\mathbf{x})}{|\mathcal{X}|}$ and $\hat{\pi}_{\mathrm{PU}} = \hat{\pi}_P - \frac{\mathrm{IR}}{1+\mathrm{IR}}$.

Theoretical Analysis: Estimation Error Bound

CoVPU's effective variational mechanisms for addressing the positive distribution pollution contributes to its asymptotic correctness as shown in Theorem 2, which ensures that a classifier trained by the variational loss function in Eq. (15) can approximate the optimal Bayesian classifier. Detailed proof can be found in our longer version.

Theorem 2. *Provided that (1) there exists a set $\mathcal{D} \subset \mathbb{R}^d$ s.t. $\int_{\mathcal{D}} p_P(\mathbf{x})\mathrm{d}\mathbf{x} > 0$ and $\Phi^*(\mathbf{x}) = 1, \quad \forall \mathbf{x} \in \mathcal{D}$, (2) a classifier is modeled as $\Phi(\mathbf{x})$ with parameters θ, and (3) there exists θ^* for the optimal Bayesian classifier $\Phi^*(\mathbf{x})$, then we can conclude that $\Phi : \mathbb{R}^d \mapsto [0,1]$ with $\sup_x \Phi(x) = 1$ satisfies $\mathcal{L}_{\mathrm{var}}(\Phi) = \mathcal{L}_{\mathrm{var}}(\Phi^*)$ iff $\Phi = \Phi^*$.*

Experiment

In this section, we present the extensive experiments to answer the following question: **Q1:** How does CoVPU perform when compared to the state-of-the-art PU learning methods? **Q2:** How do CoVPU's key components contribute to its performance? **Q3:** How does CoVPU perform in addressing the three challenging problems?

Experimental Settings

Dataset Descriptions. In experiments, we adopt two widely used PU learning datasets, i.e., MNIST and Cifar10 (Kiryo et al. 2017; Su, Chen, and Xu 2021; Hu et al. 2021). Additionally, we collect an industrial dataset, i.e., BankLoan, which contains time series of loan events with disburse dates between 2021/07/01 to 2021/08/31. The dataset statistics are presented in Table 1. For data preprocessing, we chronologically split the time series in BankLoan with ratio 0.6/0.1/0.3 as train/validation/test dataset, and adopt the train/validation/test dataset splits as provided by the original datasets for MNIST and Cifar10, which have been widely adopted in PU leering studies (Kiryo et al. 2017; Su, Chen, and Xu 2021; Chen et al. 2020a). To create the PU learning datasets, we select part of the positive data as positive labeled data (see more details in our longer version).

Dataset	Class Prior			Imbalance Ratio	Size
	π_{P}	π_{PL}	π_{PU}		
MNIST	0.1000	0.0020	0.0980	0.0020	70,000
Cifar10	0.1002	0.0020	0.0982	0.0020	60,000
BankLoan	0.0337	0.0010	0.0327	0.0010	238,275

Table 1: Dataset descriptions.

Comparison Methods. Nine methods classified into four groups and three ablation models are compared:

(1) *Traditional PU:* **uPU** (Plessis, Niu, and Sugiyama 2014) and **nnPU** (Kiryo et al. 2017) are two conventional risk estimators based PU learning methods.

(2) *PU for Data Imbalance Problem:* **ImbalancednnPU** (Su, Chen, and Xu 2021) and **GenPU** (Hou et al. 2018) are two state-of-the-art rebalanced methods by oversampling and generating data.

(3) *PU for Selection Bias Problem:* **PUSB** (Kato, Teshima, and Honda 2019) and **APU** (Hammoudeh and Lowd 2020) are two state-of-the-art methods to approximate the selection biased positive distribution. **VAE-PU** (Na et al. 2020) is a state-of-the-art method to separate unbiased positive data from unlabeled one.

(4) *PU for Prior Unavailability Problem:* **VPU** (Chen et al. 2020a) is a state-of-the-art method, which estimate class prior based on a Bayesian classifier. **PAN** (Hu et al. 2021) is a state-of-the-art generative method without class prior estimation.

(5) *Ablation models:* To investigate the effectiveness of CoVPU's key components, we compare CoVPU with its simplified variants. CoVPU**-w/o-Rebalance** refers to CoVPU without the rebalanced risk estimation mechanism, which means that the positive distribution is estimated based on the very few labeled data merely. CoVPU**-w/o-Calibrator** refers to CoVPU without VPU-Calibrator, which means that it is unable to leverage the positive unlabeled data to approximate the positive distribution. CoVPU**-w/o-RVI** refers to CoVPU without the prior-free variational classifier learning mechanism, which is provided with the ground-truth prior values to estimate the true risk instead of the variational loss.

Evaluation Metrics. In our experiments, we adopt the accuracy (Acc) and F1 score (F1) as evaluation metrics, which are two popular metrics in PU learning (Na et al. 2020; Su, Chen, and Xu 2021; Hu et al. 2021). By default, the reported results are evaluated on test sets, and the mean and standard

Categories	Strategies	MNIST		Cifar10		BankLoan	
		F1	Acc	F1	Acc	F1	Acc
Traditional PU	uPU	0.915 ± 0.005	0.983 ± 0.001	0.454 ± 0.038	0.887 ± 0.017	0.419 ± 0.012	0.945 ± 0.005
	nnPU	0.947 ± 0.002	0.990 ± 0.000	0.480 ± 0.046	0.910 ± 0.006	0.422 ± 0.005*	0.954 ± 0.004
PU for Data Imbalance Problem	ImbalancednnPU	0.955 ± 0.004*	0.991 ± 0.001	0.517 ± 0.007*	0.866 ± 0.022	0.421 ± 0.007	0.950 ± 0.003
	GenPU	0.948 ± 0.027	0.992 ± 0.000*	0.482 ± 0.015	0.885 ± 0.016	0.393 ± 0.010	0.929 ± 0.006
PU for Selection Bias Problem	PUSB	0.945 ± 0.009	0.989 ± 0.002	0.445 ± 0.016	0.813 ± 0.016	0.405 ± 0.006	0.959 ± 0.001
	VAE-PU	0.950 ± 0.005	0.990 ± 0.001	0.448 ± 0.012	0.802 ± 0.038	0.401 ± 0.002	0.959 ± 0.001
	APU	0.935 ± 0.004	0.987 ± 0.001	0.450 ± 0.020	0.911 ± 0.002*	0.401 ± 0.200	0.908 ± 0.002
PU for Prior Agnostic Problem	VPU	0.914 ± 0.006	0.983 ± 0.001	0.464 ± 0.024	0.876 ± 0.018	0.408 ± 0.020	0.957 ± 0.009
	PAN	0.939 ± 0.008	0.988 ± 0.002	0.451 ± 0.015	0.881 ± 0.012	0.410 ± 0.011	0.961 ± 0.002*
Ours	CoVPU	**0.985 ± 0.008**	**0.993 ± 0.000**	**0.527 ± 0.005**	**0.916 ± 0.001**	**0.447 ± 0.001**	**0.987 ± 0.006**
	CoVPU-w/o-Rebalance	0.971 ± 0.003	0.987 ± 0.004	0.520 ± 0.003	0.825 ± 0.005	0.416 ± 0.046	0.942 ± 0.003
	CoVPU-w/o-Calibrator	0.950 ± 0.003	0.989 ± 0.005	0.520 ± 0.014	0.819 ± 0.013	0.392 ± 0.002	0.931 ± 0.010
	CoVPU-w/o-RVI	0.966 ± 0.005	0.990 ± 0.002	0.522 ± 0.007	0.825 ± 0.003	0.446 ± 0.005	0.988 ± 0.008
Improvement [1]		3.141%	0.101%	1.934%	0.549%	5.924%	2.706%
p-value[2]		0.000	0.000	0.003	0.012	0.000	0.000

[1] Improvement of CoVPU over the best-performing comparison methods.

[2] Statistically not different from the best-performing comparison methods if p-value < 0.05 (p-value with paired t-test).

* Bold values & Star marked values represent the best & second-best results.

Table 2: Results of all methods (mean ± standard deviation, computed across 10 runs).

deviation values are computed from 10 independent runs. On each of the 10 independent runs, we re-create the PU learning datasets according to our data preprocessing.

Implementation Details. We implement CoVPU's three parts, i.e., VPU-Calibrator, VPU-Classifier, and the VPU-Collaborator as follows. In the VPU-Calibrator, we use the embedding network following (Kiryo et al. 2017) for MNIST and Cifar10, and use LSTM with hidden dimension equals 32 as the embedding model for BankLoan as the encoder. We use 1-layer fully connected neural networks to encode the $\boldsymbol{\mu}_{\mathbf{v}^0}$, $\boldsymbol{\sigma}_{\mathbf{v}^0}$, $\boldsymbol{\mu}_{\mathbf{z}}$, and $\boldsymbol{\sigma}_{\mathbf{z}}$, respectively. We use a 1-layer fully connected neural network with hidden dimension equals 2 as a planar transformation layer and stack $L = 3$ planar transformation layers to transform $\mathbf{v}^0$ to $\mathbf{v}^L$. The argmax transformation layer is based on the model proposed in (Hoogeboom et al. 2021). We use a 4-layers fully connected neural network (more specifically, 300-300-300-d for MNIST, 500-500-500-d for Cifar10, and 32-32-32-d for BankLoan, where d is the input feature dimension of each dataset) as the decoder. In the VPU-Classifier, we use the embedding network with a 1-layer fully connected neural network as the VPU-Classifier network.

For hyperparameter settings. In our proposed CoVPU, we set π' to be 0.5 and J to be 20, and we tune all other hyperparameters through grid search. We also carefully tune the hyper-parameters of all comparison methods through cross-validation to achieve their best performance. We provide the ground-truth class prior values to all the comparison methods that require the input of class prior.

For training details, we adopt Adam optimizer (Kingma and Ba 2015) and tune the learning rate and weight decay by a grid select in $[10^{-2}, 10^{-5}]$ for all methods. We set the batch size to 1024 for batch training. For the surrogate loss, we utilize the logistic loss as the surrogate loss of the PU classifier for uPU, NNPU, ImbalancednnPU, and APU, sigmoid loss for VAE-PU, and logarithmic loss for PUSB, VPU and CoVPU as recommended in their publications.

More implementation details are in our longer version.

Comparison with Baselines (for Q1)

We compare our proposed CoVPU to the comparison methods. The Acc and F1 results are presented in Table 2. In general, Co-VPU achieves the highest Acc and F1 values with average improvements of 1.118% and 3.666%, respectively, over the best-performing comparison methods. Given that CoVPU does not need class prior unlike most of the baselines, this is even more significant.

From the results, we can draw three conclusions. Firstly, comparing CoVPU with imbalancednnPU and GenPU, CoVPU achieves the highest Acc and F1 with average improvements of 3.286% and 6.372%, respectively, because CoVPU adopts the VPU-Calibrator and the rebalance risk estimation mechanism to effectively approximate positive distribution. Secondly, comparing CoVPU with PUSB, VAEPU, and APU, CoVPU outperforms these methods, especially on the typical selection biased dataset BankLoan with average improvements of 4.847% and 11.104%, respectively, on Acc and F1. This is because CoVPU has a more relaxed assumption and quality assurance on the exploited positive data. Thirdly, comparing CoVPU with VPU and PAN, CoVPU outperforms VPU and PAN with average improvements of 2.650% and 10.280%, respectively, on Acc and F1, because CoVPU pays attention to the selection bias problem and performs better than VPU and PAN in modeling positive distributions.

Ablation Studies (for Q2)

We conduct ablation experiments with three variants of CoVPU. Results in Table 2 demonstrates three conclusions.

Firstly, CoVPU outperforms CoVPU-w/o-Rebalance with average improvements of 5.472% and 25.639% on Acc and F1, respectively. With the rebalanced risk estimator, CoVPU is able to reweight the positive data reasonably for approximating the positive distribution effectively.

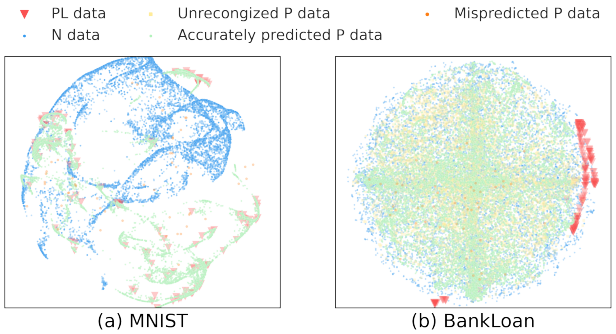


Figure 3: t-SNE on (a) MNIST and (b) BankLoan.

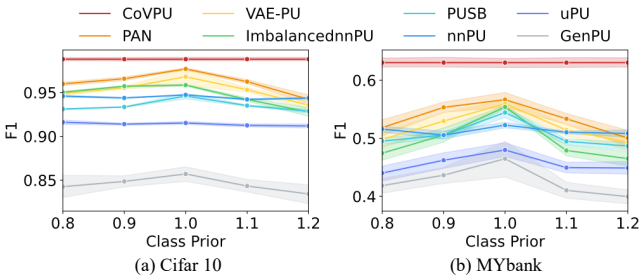


Figure 4: F1 under varying class prior π_P. Dense lines are the average of 10 runs, and bars denote standard errors.

Secondly, CoVPU outperforms CoVPU-w/o-Calibrator with average improvements of 6.251% and 29.234% on Acc and F1, respectively. Without VPU-Calibrator for calibrating positive distribution and extracting diverse positive data, CoVPU-w/o-Calibrator is unable to rectify the distribution deviation, thereby leading to poor performance.

Thirdly, CoVPU achieves comparable performance to CoVPU-w/o-RVI, which replaces the prior-free variational loss with prior-based risk estimator. The reasons for such results lie in two aspects: (1) CoVPU-w/o-RVI with the risk estimator only cares about the "mean" of the estimation error. By contrast, CoVPU with variational loss considers a different but more comprehensive perspective on top of the polluted distribution, leading to a better guide for Bayesian classifier learning; (2) CoVPU with variational loss delivers better performance on handling datasets that satisfy the irreducibility assumption, which has been widely adopted in PU learning studies, i.e., the two-class distributions can be fully separated. In the MNIST and Cifar10 datasets, positive samples contain only one class, which satisfies the irreducibility assumption, resulting in CoVPU's outperformance over CoVPU-w/o-RVI. As for the BankLoan dataset, although it is naturally a binary classification dataset, it contains some cases that dissatisfy the irreducibility assumption. For example, in loan scenarios, there exist some well-camouflaged defaulters with hard-distinguished distributions from benign users. Therefore, CoVPU only achieves comparable performance to CoVPU-w/o-RVI on this dataset.

Performance on Addressing Problems (for Q3)

For Data Imbalance Problem. Recall that CoVPU achieves the highest Acc and F1 values on the three datasets over all comparison methods as presented in Table 2. Combing the data imbalance ratio presented in Table 1 and CoVPU's superiority on all the three datasets, especially on BankLoan, i.e., the most data imbalanced dataset with π_{PL} being merely 0.1%, we can conclude that CoVPU can adapt to the data imbalance scenario well.

For Selection Bias Problem. To verify the performance of CoVPU in the selection bias scenario, we perform t-SNE on the three datasets. Results on MNIST and BankLoan are presented in Figure 3, and results on Cifar10 are presented in our longer version, in which markers with red, light blue, green, yellow, and dark blue colors represent positive labeled (PL), positive unlabeled (PU), accurately predicted positive (P), unrecognized positive (P), and mispredicted positive (P) data, in sequence. From Figure 3, we can draw two conclusions. Firstly, the red circles do not necessarily cover the entire positive data, especially on BankLoan, which is consistent to our analysis that the labeled data are selected with bias. Secondly, CoVPU can correctly identify potential positive samples, especially on BankLoan, demonstrating its effectiveness in addressing selection bias.

For Prior Agnostic Problem. To verify the performance of CoVPU in the prior agnostic scenario, we compare the F1 achieved by CoVPU and the class prior based methods, i.e., uPU, nnPU, ImbalancednnPU, GenPU, PUSB, VAEPU, PAN, given class priors varying from $\{0.8, 0.9, 1.0, 1.1, 1.2\} \times \pi_{\text{P}}$, where π_{P} is the ground-truth class prior. The results presented in Figure 4 demonstrate two conclusions. Firstly, the prior-based PU learning methods are highly sensitive to the given class priors. Their performances decay greatly even with a small deviation from the real class prior. Secondly, CoVPU outperforms the PU learning method even though they are provided with the accurate class prior, which demonstrates the effectiveness of CoVPU in addressing prior agnostic problem.

Conclusions and Future Work

In this paper, we first present an interesting insight that the data imbalance, selection bias, and prior agnostic problems in PU learning inherently originate from positive distribution pollution. Inspired by this insight, we then devise a model named CoVPU, which addresses these problems simultaneously from a unified perspective. Rigorous theoretical analysis proves the convergence of CoVPU. Extensive experiments demonstrate the superiority of CoVPU over the state-of-the-art methods. In the future, we plan to fully exploit CoVPU's potential in more practical applications such as disease diagnosis and cyberattacks detection.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No.62172362), Leading Expert of "Ten Thousands Talent Program" of Zhejiang Province (No.2021R52001), and the cooperation project of MYbank, Ant Group.

References

Akujuobi, U.; Chen, J.; Elhoseiny, M.; Spranger, M.; and Zhang, X. 2020. Temporal Positive-unlabeled Learning for Biomedical Hypothesis Generation via Risk Estimation. In *Proc. of NeurIPS*, 4597–4609.

Bekker, J.; and Davis, J. 2018. Estimating the Class Prior in Positive and Unlabeled Data Through Decision Tree Induction. In *Proc. of AAAI*, 2712–2719.

Chen, H.; Liu, F.; Wang, Y.; Zhao, L.; and Wu, H. 2020a. A Variational Approach for Learning from Positive and Unlabeled Data. In *Proc. of NeurIPS*, 14844–14854.

Chen, X.; Chen, W.; Chen, T.; Yuan, Y.; Gong, C.; Chen, K.; and Wang, Z. 2020b. Self-pu: Self boosted and calibrated positive-unlabeled training. In *Proc. of ICML*, 1510–1519.

Dupret, G. E.; and Piwowarski, B. 2008. A user browsing model to predict search engine click data from past observations. In *Proc. of SIGIR*, 331–338.

Hammoudeh, Z.; and Lowd, D. 2020. Learning from positive and unlabeled data with arbitrary positive shift. In *Proc. of NeurIPS*, 13088–13099.

Hoogeboom, E.; Nielsen, D.; Jaini, P.; Forré, P.; and Welling, M. 2021. Argmax Flows and Multinomial Diffusion: Learning Categorical Distributions. In *Proc. of NeurIPS*, 12454–12465.

Hou, M.; Chaib-draa, B.; Li, C.; and Zhao, Q. 2018. Generative Adversarial Positive-Unlabelled Learning. In *Proc. of IJCAI*, 2255–2261.

Hu, W.; Le, R.; Liu, B.; Ji, F.; Ma, J.; Zhao, D.; and Yan, R. 2021. Predictive Adversarial Learning from Positive and Unlabeled Data. In *Proc. of AAAI*, 7806–7814.

Ivanov, D. 2020. DEDPUL: Difference-of-Estimated-Densities-based Positive-Unlabeled Learning. In *Proc. of ICMLA*, 782–790.

Jain, S.; Delano, J.; Sharma, H.; and Radivojac, P. 2020. Class Prior Estimation with Biased Positives and Unlabeled Examples. In *Proc. of AAAI*, 4255–4263.

Jaskie, K.; and Spanias, A. 2019. Positive and unlabeled learning algorithms and applications: A survey. In *Proc. of IISA*, 1–8.

Kato, M.; Teshima, T.; and Honda, J. 2019. Learning from positive and unlabeled data with a selection bias. In *Proc. of ICLR*, 1–10.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *Proc. of ICLR*, 1–10.

Kiryo, R.; Niu, G.; du Plessis, M. C.; and Sugiyama, M. 2017. Positive-Unlabeled Learning with Non-Negative Risk Estimator. In *Proc. of NeurIPS*, 1675–1685.

Kool, W.; Van Hoof, H.; and Welling, M. 2019. Stochastic beams and where to find them: The gumbel-top-k trick for sampling sequences without replacement. In *Proc. of ICML*, 3499–3508.

Li, X.; and Liu, B. 2003. Learning to Classify Texts Using Positive and Unlabeled Data. In *Proc. of IJCAI*, 587–594.

Liu, B.; Lee, W. S.; Yu, P. S.; and Li, X. 2002. Partially Supervised Classification of Text Documents. In *Proc. of ICML*, 387–394.

Na, B.; Kim, H.; Song, K.; Joo, W.; Kim, Y.; and Moon, I. 2020. Deep generative positive-unlabeled learning under selection bias. In *Proc. of CIKM*, 1155–1164.

Nielsen, D.; Jaini, P.; Hoogeboom, E.; Winther, O.; and Welling, M. 2020. Survae flows: Surjections to bridge the gap between vaes and flows. In *Proc. of NeurIPS*, 12685–12696.

Pang, G.; Shen, C.; and van den Hengel, A. 2019. Deep anomaly detection with deviation networks. In *Proc. of KDD*, 353–362.

Perini, L.; Vercruyssen, V.; and Davis, J. 2020. Class Prior Estimation in Active Positive and Unlabeled Learning. In *Proc. of IJCAI*, 2915–2921.

Plessis, M. C. d.; Niu, G.; and Sugiyama, M. 2014. Analysis of learning from positive and unlabeled data. In *Proc. of NeurIPS*, 703–711.

Rezende, D.; and Mohamed, S. 2015. Variational inference with normalizing flows. In *Proc. of ICML*, 1530–1538.

Su, G.; Chen, W.; and Xu, M. 2021. Positive-Unlabeled Learning from Imbalanced Data. In *Proc. of IJCAI*, 2995–3001.

Yu, H.; Han, J.; and Chang, K. C.-C. 2002. PEBL: positive example based learning for Web page classification using SVM. In *Proc. of SIGKDD*, 239–248.

Zeiberg, D.; Jain, S.; and Radivojac, P. 2020. Fast Nonparametric Estimation of Class Proportions in the Positive-Unlabeled Classification Setting. In *Proc. of AAAI*, 6729–6736.

Zhang, C.; Hou, Y.; and Zhang, Y. 2020. Learning from Positive and Unlabeled Data without Explicit Estimation of Class Prior. In *Proc. of AAAI*, 6762–6769.