Information-Theoretic Representation Learning for Positive-Unlabeled Classification

Tomoya Sakai
t.sakai.kure@gmail.com
University of Tokyo, Kashiwa, Chiba 277-8561, Japan

Gang Niu
gang.niu@riken.jp
RIKEN Center for Advanced Intelligence Project, Chuo-ku, Tokyo 103-0027, Japan

Masashi Sugiyama
sugi@k.u-tokyo.ac.jp
RIKEN Center for Advanced Intelligence Project, Chuo-ku, Tokyo 103-0027, Japan, and University of Tokyo, Kashiwa, Chiba 277-8561, Japan

Recent advances in weakly supervised classification allow us to train a classifier from only positive and unlabeled (PU) data. However, existing PU classification methods typically require an accurate estimate of the class-prior probability, a critical bottleneck particularly for high-dimensional data. This problem has been commonly addressed by applying principal component analysis in advance, but such unsupervised dimension reduction can collapse the underlying class structure. In this letter, we propose a novel representation learning method from PU data based on the information-maximization principle. Our method does not require class-prior estimation and thus can be used as a preprocessing method for PU classification. Through experiments, we demonstrate that our method, combined with deep neural networks, highly improves the accuracy of PU class-prior estimation, leading to state-of-the-art PU classification performance.

1 Introduction

In real-world applications, it is conceivable that only positive and unlabeled (PU) data are available for training a classifier. For instance, in land-cover image classification, images of urban regions can be easily labeled, while images of nonurban regions are difficult to annotate due to the high diversity of nonurban regions containing, for example, forest, seas, grasses, and grasslands.
soil (Li, Guo, & Elkan, 2011). To cope with such situations, PU classification has been actively studied (Letouzey, Denis, & Gilleron, 2000; Elkan & Noto, 2008; du Plessis, Niu, & Sugiyama, 2015), and the state-of-the-art method allows us to systematically train deep neural networks only from PU data (Kiryo, Niu, du Plessis, & Sugiyama, 2017).

However, existing PU classification methods typically require an estimate of the class-prior probability, and their performance is sensitive to the quality of class-prior estimation (Kiryo et al., 2017). Although various class-prior estimation methods from PU data have been proposed so far (du Plessis & Sugiyama, 2014; Ramaswamy, Scott, & Tewari, 2016; Jain, White, & Radivojac, 2016; du Plessis, Niu, & Sugiyama, 2017; Northcutt, Wu, & Chuang, 2017), accurate estimation of the class-prior is still highly challenging, particularly for high-dimensional data.

In practice, principal component analysis is commonly used to reduce the data dimensionality in advance (Ramaswamy et al., 2016; du Plessis et al., 2017). However, such unsupervised dimension reduction completely abandons label information, and thus the underlying class structure may be smashed. As a result, class-prior estimation often becomes even more difficult after dimension reduction.

The goal of this letter is to cope with this problem by proposing a representation learning method that can be executed only from PU data. Our method is developed within the framework of information maximization (Linsker, 1988).

Mutual information (MI) (Cover & Thomas, 2006) is a statistical dependency measure between random variables that is popularly used in information-theoretic machine learning (Torkkola, 2003; Krause, Perona, & Gomes, 2010). However, empirically approximating MI from continuous-valued training data is not straightforward (Moon, Rajagopalan, & Lall, 1995; Kraskov, Stögbauer, & Grassberger, 2004; Khan, Bandyopadhyay, Ganguly, & Saigal, 2007; Van Hulle, 2005; Suzuki, Sugiyama, Sese, & Kanamori, 2008) and is often sensitive to outliers (Basu, Harris, Hjort, & Jones, 1998; Sugiyama, Suzuki, & Kanamori, 2012b). For this reason, we employ a squared-loss variant of mutual information (SMI) (Suzuki, Sugiyama, Kanamori, & Sese, 2009; Sugiyama, 2013), whose empirical estimator is known to be robust to outliers and possess superior numerical properties (Kanamori, Suzuki, & Sugiyama, 2012).

Our contributions are summarized as follows:

• In section 3, we develop a novel estimator of SMI that can be computed only from PU data and prove its convergence to the optimal estimate of SMI in the optimal parametric rate when the linear-in-parameter model is used.

• Based on this PU-SMI estimator, in section 4, we propose a representation learning method that can be executed without estimating the class-prior probabilities of unlabeled data.
Finally, in section 5, we experimentally demonstrate that our PU representation learning method combined with deep neural networks highly improves the accuracy of PU class-prior estimation; consequently, the accuracy of PU classification can also be boosted significantly.

2 SMI

In this section, we review the definition of ordinary MI and its variant, SMI. Let $x \in \mathbb{R}^d$ be an input pattern, $y \in \{\pm 1\}$ be a corresponding class label, and $p(x, y)$ be the underlying joint density, where $d$ is a positive integer.

Mutual information (MI) (Cover & Thomas, 2006) is a statistical dependency measure defined as

$$MI := \sum_{y=\pm 1} \int p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right) dx,$$

where $p(x)$ is the marginal density of $x$ and $p(y)$ is the probability mass of $y$. MI can be regarded as the Kullback-Leibler divergence from $p(x, y)$ to $p(x)p(y)$, and therefore MI is nonnegative and takes zero if and only if $p(x, y) = p(x)p(y)$, that is, $x$ and $y$ are statistically independent. This property allows us to evaluate the dependency between $x$ and $y$. However, empirically approximating MI from continuous data is not straightforward (Moon et al., 1995; Kraskov et al., 2004; Khan et al., 2007; Van Hulle, 2005; Suzuki et al., 2008) and is often sensitive to outliers (Basu et al., 1998; Sugiyama et al., 2012b).

To cope with this problem, squared-loss MI (SMI) has been proposed (Suzuki et al., 2009). It is a squared-loss variant of MI defined as

$$SMI := \sum_{y=\pm 1} \frac{p(y)}{2} \int \left( \frac{p(x, y)}{p(x)p(y)} - 1 \right)^2 p(x) dx. \quad (2.1)$$

SMI can be regarded as the Pearson divergence (Pearson, 1990) from $p(x, y)$ to $p(x)p(y)$. SMI is also nonnegative and takes zero if and only if $x$ and $y$ are independent.

So far, methods for estimating SMI from positive and negative samples and SMI-based machine learning algorithms have been explored extensively, and their effectiveness has been demonstrated (Sugiyama, 2013).

3 SMI Estimation from PU Data

The goal of this letter is to develop a representation learning method from PU data. To this end, we propose an estimator of SMI that can be computed only from PU data in this section.
3.1 SMI with PU Data. Suppose that we are given PU data (Ward, Hastie, Barry, Elith, & Leathwick, 2009):

\[
\{x_i^p\}_{i=1}^n \sim p(x | y = +1),
\]

\[
\{x_i^u\}_{i=1}^n \sim p(x) = \theta_p p(x | y = +1) + \theta_n p(x | y = -1),
\]

where \(\theta_p := p(y = +1)\) and \(\theta_n := p(y = -1)\) are the class-prior probabilities.

First, we express SMI in equation 2.1 in terms of only the densities of PU data, without negative data (see appendix A for its proof):

**Theorem 1.** Let

\[
\text{PU-SMI} := \frac{\theta_p}{2\theta_n} \int \left( \frac{p(x | y = +1)}{p(x)} - 1 \right)^2 p(x) dx.
\]

Then we have PU-SMI = SMI.

If PU densities \(p(x | y = +1)\) and \(p(x)\) are estimated from PU data, the above PU-SMI allows us to approximate SMI only from PU data. However, such a naive approach works poorly due to hardness of density estimation, and computing the ratio of estimated densities further magnifies the estimation error (Sugiyama, Suzuki, & Kanamori, 2012a).

3.2 PU-SMI Estimation. Here, we propose a more sophisticated approach to estimating PU-SMI from PU data.

First, the following theorem gives a lower-bound of PU-SMI (see appendix B for its proof):

**Theorem 2.** For any function \(f(x)\),

\[
\text{PU-SMI} \geq \frac{\theta_p}{\theta_n} \left( - J_{\text{PU}}(f) - \frac{1}{2} \right),
\]

where

\[
J_{\text{PU}}(f) := \frac{1}{2} \int f^2(x)p(x)dx - \int f(x)p(x | y = +1)dx,
\]

and the equality holds if and only if

\[
f(x) = \frac{p(x | y = +1)}{p(x)}.
\]

While PU-SMI itself contains \(p(x | y = +1)\) and \(p(x)\) in a complicated way, the lower bound consists only of the expectations over \(p(x | y = +1)\) and \(p(x)\). Thus, the lower bound can be immediately approximated empirically.
Based on this theorem, we maximize an empirical approximation to the lower bound, equation 3.2, which is expressed as

$$\hat{w} := \arg\min_{w \in \mathcal{W}} \hat{J}_{PU}(w),$$

where

$$\hat{J}_{PU}(w) := \frac{1}{2n_U} \sum_{k=1}^{n_U} w(x_k^U) - \frac{1}{2n_P} \sum_{i=1}^{n_P} w(x_i^P)$$

(3.3)

and $\mathcal{W}$ is a (user-defined) function class such as linear-in-parameter models, kernel models, and neural networks. In this optimization, we can drop the unknown class-prior ratio $\theta_P/\theta_N$, which is difficult to estimate accurately (du Plessis & Sugiyama, 2014; Ramaswamy et al., 2016; Jain et al., 2016; du Plessis et al., 2017).

Finally, our PU-SMI estimator is given as

$$\hat{\text{PU-SMI}} = \frac{\theta_P}{\theta_N} \left( -\hat{J}_{PU}(\hat{w}) - \frac{1}{2} \right).$$

(3.4)

PU-SMI includes the class-prior ratio $\theta_P/\theta_N$ only as a proportional constant. Therefore, class-prior estimation is not needed when we just want to maximize or minimize PU-SMI. We utilize this excellent property in section 4 when we develop a representation learning method.

Note that if $W$ contains the true density-ratio function $p(x \mid y = +1)/p(x)$, $\text{PU-SMI} \rightarrow \text{PU-SMI} (q_P, n_U \rightarrow \infty)$ with some regularity condition. On the other hand, if the function class does not contain the true density-ratio function, there is a gap between PU-SMI and PU-SMI even if $n_P, n_U \rightarrow \infty$. Such a gap often arises in real-world applications because a function class does not always include the true density-ratio function. However, the gap may not be a critical issue in practice as long as a reasonably flexible function class is chosen, as demonstrated by the experiments in section 5. Although the gap may exist in practical implementation, we show that the classification performance can be improved by our proposed representation learning method.

3.3 Analytic Solution for Linear-in-Parameter Models. Our SMI estimator is applicable to any density-ratio model $w$.

If a neural network is used as $w$, the solution may be obtained by a stochastic gradient method (Goodfellow, Bengio, & Courville, 2016; Abadi et al., 2015; Jia et al., 2014).
Another candidate of the density-ratio model is a linear-in-parameter model,

\[ w(x) = \sum_{\ell=1}^{b} \beta_{\ell} \phi_{\ell}(x) = \beta^\top \phi(x), \]  

(3.5)

where \( \beta := (\beta_1, \ldots, \beta_b)^\top \in \mathbb{R}^b \) is a vector of parameters, \( b \) is the number of parameters, and \( \phi(x) := (\phi_1(x), \ldots, \phi_b(x))^\top \in \mathbb{R}^b \) is a vector of basis functions. This model allows us to obtain an analytic-form PU-SMI estimator. Furthermore, the optimal convergence is theoretically guaranteed, as shown in section 3.4.

When the \( \ell_2 \)-regularizer is included, the optimization problem yields

\[ \hat{\beta} := \arg\min_{\beta} \frac{1}{2} \beta^\top \hat{H}^U \beta - \beta^\top \hat{h}^P + \frac{\lambda_{\text{PU}}}{2} \| \beta \|_2^2, \]

where \( \lambda_{\text{PU}} \geq 0 \) is the regularization parameter, \( \| \cdot \|_2 \) denotes the \( \ell_2 \)-norm, and

\[ \hat{H}^U_{\ell,\ell'} := \frac{1}{n_U} \sum_{k=1}^{n_U} \phi_{\ell}(x_U^k) \phi_{\ell'}(x_U^{k}), \]

\[ \hat{h}^P_{\ell} := \frac{1}{n_P} \sum_{i=1}^{n_P} \phi_{\ell}(x_P^i). \]

Note that \( \hat{H}^U_{\ell,\ell'} \) is the \((\ell, \ell')\)th element of \( \hat{H}^U \) and \( \hat{h}^P_{\ell} \) is the \( \ell \)th element of \( \hat{h}^P \). The solution can be obtained analytically by differentiating the objective function with respect to \( \beta \) and setting it to zero: \( \hat{\beta} = (\hat{H}^U + \lambda_{\text{PU}} I_b)^{-1} \hat{h}^P \).

Finally, with the obtained estimator, we can compute an SMI approximator only from positive and unlabeled data:

\[ \text{PU-SMI} = \theta_N \left( \hat{\beta}^\top \hat{h}^P - \frac{1}{2} \beta^\top \hat{H}^U \beta - \frac{1}{2} \right). \]

Note that all hyperparameters such as the regularization parameter can be tuned by the value of \( I_{\text{PU}} \) approximated by (cross-)validation samples.

### 3.4 Convergence Analysis.

Here we analyze the convergence rate of learned parameters of the density-ratio model and the PU-SMI approximator based on the perturbation analysis of optimization problems (Bonnans & Cominetti, 1996; Bonnans & Shapiro, 1998).

In our theoretical analysis, we focus on the linear-in-parameter model in equation 3.5. We first define \( \beta^* \phi(x) \) as the minimizer of the expected error,
\( \beta^* := \arg\min_{\beta \in \mathbb{R}^b} J_{PU}(\beta) \), and denote its estimator by \( \hat{\beta} = \arg\min_{\beta \in \mathbb{R}^b} \hat{J}_{PU}(\beta) \) in this analysis. Note that the linear-in-parameter model is assumed as a simple baseline for theoretical analysis.

For the linear-in-parameter model, we assume that the basis functions satisfy \( 0 \leq \phi_\ell(x) \leq 1 \) for all \( \ell = 1, \ldots, b \), and \( \hat{H}^U \) and \( H^U \) are positive-definite matrices.

Let \( \text{PU-SMI}^* := \frac{\theta_p}{\theta_N} \left( -J_{PU}(\beta^*) - \frac{1}{2} \right) \) be the PU-SMI with \( \beta^* \). Similarly, \( \hat{\text{PU-SMI}} \) is the estimate of the PU-SMI with \( \hat{\beta} \). Let \( O_p \) denote the order in probability. Then we have the following convergence results (its proof is given in appendix C):

**Theorem 3.** As \( n_p, n_U \to \infty \), we have

\[
\| \hat{\beta} - \beta^* \|_2 = O_p(1/\sqrt{n_p} + 1/\sqrt{n_U}),
\]

\[
|\text{PU-SMI}^* - \hat{\text{PU-SMI}}| = O_p(1/\sqrt{n_p} + 1/\sqrt{n_U}).
\]

Theorem 3 guarantees the convergence of the density-ratio estimator and the PU-SMI approximator. In our setting, since \( n_p \) and \( n_U \) can increase independently, this is the optimal convergence rate without any additional assumption (Kanamori, Hido, & Sugiyama, 2009; Kanamori et al., 2012).

Theorem 3 shows that both positive and unlabeled data contribute to convergence. This implies that unlabeled data are directly used in the estimation rather than extracting the information of a data structure, such as the cluster structure frequently assumed in semisupervised learning (Chapelle, Schölkopf, & Zien, 2006). The theorem also shows that the convergence rate of our method is dominated by the smaller size of positive or unlabeled data.

Note that since this analysis focuses on the linear-in-parameter model, there might be a gap between PU-SMI and \( \text{PU-SMI}^* \), implying that \( \text{PU-SMI} < \text{PU-SMI}^* \). The convergence analysis guarantees that PU-SMI with the linear-in-parameter model converges to \( \text{PU-SMI}^* \), but there might be an approximation error (Mohri, Rostamizadeh, & Talwalkar, 2012), as discussed in section 3.2.

4 **PU Representation Learning**

In this section, we propose a representation learning method based on PU-SMI maximization. We extend the existing SMI-based dimension reduction (Suzuki & Sugiyama, 2013), called least-squares dimension reduction (LSDR), to PU representation learning. While LSDR considers only linear
dimension reduction, we extend it to nonlinear dimension reduction by neural networks.

Let \( v : \mathbb{R}^d \rightarrow \mathbb{R}^m \), where \( m < d \), be a mapping from an input vector to its low-dimensional representation. If the mapping function satisfies

\[
p(y \mid x) = p(y \mid v(x)),
\]

the obtained low-dimensional representation can be used as the new input instead of the original input vector. Finding the mapping function satisfying condition 4.1 is known as sufficient dimension reduction (Li, 1991). Let SMI be SMI between \( v(x) \) and \( y \). Suzuki and Sugiyama (2013) proved SMI \( \geq \) SMI, and equality holds when condition 4.1 is satisfied. That is, maximizing SMI is finding sufficient representation for the output \( y \).

Following the information-maximization principle (Linsker, 1988), we maximize PU-SMI with respect to the mapping to find low-dimensional representation that maximally preserves dependency between input and output.

More specifically, since \( \widehat{P}_U-SMI = -\theta_p/\theta_n \cdot \min_{w \in W} \widehat{J}_P(w) - \theta_p/(2\theta_n) \), we minimize \( \widehat{J}_P(w) \) with respect to \( w \). Furthermore, inspired by the alternative optimization algorithm for the SMI-based dimension-reduction method (Suzuki & Sugiyama, 2013), we decompose \( w \) into \( g \) and \( v \) such that \( w = g \circ v \) and minimize \( \widehat{J}_P(g \circ v) \) by optimizing \( g \) and \( v \) alternatively, where \( g : \mathbb{R}^m \rightarrow \mathbb{R} \) and \( \circ \) denotes the function composition, \( (g \circ v)(x) = g(v(x)) \). In this decomposition, \( v \) can be regarded as a mapping function extracting features from input patterns and \( g \) a density ratio function \( p(x \mid y = +1)/p(x) \). First, we approximate SMI by minimizing equation 3.3 with respect to density ratio \( g \) with current mapping \( \widehat{v} \) fixed:

\[
\widehat{g} = \arg\min_{g} \widehat{J}_P(g \circ \widehat{v}).
\]

Then we update mapping \( \widehat{v} \) to increase the estimated PU-SMI with current density ratio \( \widehat{g} \) fixed:

\[
\widehat{v} \leftarrow \widehat{v} - \varepsilon \nabla_\widehat{v} \widehat{J}_P(\widehat{g} \circ \widehat{v}),
\]

where \( \varepsilon \) is the step size. This process is repeated until convergence. In practice, we may alternately optimize \( g \) and \( v \) as described in algorithm 1 to simplify the implementation.\(^1\) We refer to our representation learning method for PU data as positive-unlabeled representation learning (PURL).

\(^1\)We also tried to optimize \( w \) and \( v \) simultaneously. That is, \( J_P \) is minimized with respect to \( g \) without decomposing \( g \) into \( w \) and \( v \), but it did not work well in our preliminary experiments.
Algorithm 1: PU Representation Learning.

 Require: \((x_i^P)^{n_P}_{i=1}\) and \((x_k^U)^{n_U}_{k=1}\).

 1: Initialize \(\hat{g}\) and \(\hat{v}\)

 2: repeat

 3: Update \(\hat{g} \leftarrow \hat{g} - \epsilon_g \nabla_U \hat{J}_{PU}(g \circ \hat{v})|_{g=\hat{g}}\)

 4: Update \(\hat{v} \leftarrow \hat{v} - \epsilon_v \nabla_u \hat{J}_{PU}(\hat{g} \circ v)|_{v=\hat{v}}\)

 5: until stopping conditions meet

 6: return \(\hat{g}\) and \(\hat{v}\).

Note again that in the above optimization process, unknown class-prior ratio \(\theta_P/\theta_N\) does not need to be estimated in advance, a significant advantage of the proposed method.

5 Experiments

In this section, we experimentally investigate the behavior of the proposed PU-SMI estimator and evaluate the performance of the proposed representation learning method on various benchmark data sets.

5.1 Accuracy of PU-SMI Estimation. First, we investigate the estimation accuracy of the proposed PU-SMI estimator on data sets obtained from the LIBSVM web page (Chang & Lin, 2011).

As the model \(w\), we use the linear-in-parameter model with the gaussian basis functions \(\phi_\ell(x) := \exp(-\|x - x_\ell\|^2/(2\sigma^2))\) for \(\ell = 1, \ldots, b\), where \(\sigma > 0\) is the bandwidth and \((x_\ell)^{b}_{\ell=1}\) are the centers of the gaussian functions randomly sampled from \((x_k^U)^{n_U}_{k=1}\). The gaussian bandwidth and the \(\ell_2\)-regularization parameter are determined by five-fold cross-validation. We vary the number of positive/unlabeled samples from 10 to 200, with the number of unlabeled/positive samples fixed. The class-prior was assumed to be known in this illustrative experiment and set at \(\theta_P = 0.5\).

Figure 1 summarizes the average and standard error of the squared estimation error of PU-SMI over 50 trials.\(^2\) This shows that the mean squared error decreases both when the number of positive samples is increased and the number of unlabeled samples is increased. Therefore, both positive and

\(^2\)We compute the squared error between the estimated PU-SMI and the supervised SMI estimator (Suzuki et al., 2009) with a sufficiently large number of positive and negative samples.
unlabeled data contribute to improving the estimation accuracy of SMI, which agrees well with our theoretical analysis in section 3.4.

5.2 Representation Learning. Next, we evaluate the performance of the proposed representation learning method, PURL.

5.2.1 Illustration. We first illustrate how our proposed method works on an artificial data set. We generate samples from the following densities:

\[
p(x \mid y = +1) = N\left(x; \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0.25 & 0 \\ 0 & 4 \end{pmatrix}\right),
\]

\[
p(x \mid y = -1) = \frac{1}{2} N\left(x; \begin{pmatrix} 3 \\ 0 \end{pmatrix}, \begin{pmatrix} 0.25 & 0 \\ 0 & 4 \end{pmatrix}\right) + \frac{1}{2} N\left(x; \begin{pmatrix} -3 \\ 0 \end{pmatrix}, \begin{pmatrix} 0.25 & 0 \\ 0 & 4 \end{pmatrix}\right),
\]

where \(N(x; \mu, \Sigma)\) is the normal density with the mean vector \(\mu\) and the covariance matrix \(\Sigma\). The class-prior is set at \(\theta_p = 0.7\). From the densities, we draw \(n_p = 400\) positive and \(n_U = 1000\) unlabeled samples. For comparison, we apply PCA, Fisher’s discriminant analysis (FDA), and PNRL\(^3\) (the supervised counterpart of PURL) to the data. As the label information for FDA and PNRL, U data are simply regarded as N data even though U data are

\(^3\)The details of PNRL are described in appendix E.
Figure 2: (a) Positive and unlabeled data. The estimated subspaces obtained by PCA, FDA, PNRL, and our proposed method. (b) Unlabeled data with the true labels projected onto the subspaces obtained by PCA, FDA, and our method, respectively. The results indicate that PCA, FDA, and PNRL smash underlying class structure, while the positive and negative labels are visibly separated in the subspace obtained by our method.

We plot the subspaces obtained by PCA, FDA, PNRL, and our proposed method in Figure 2a. Since the data are distributed vertically, the subspace obtained by PCA is almost parallel to the vertical axis (the dashed line). FDA and PNRL return diagonal lines (the dash-dotted and dotted lines), showing that regarding U data as N data is not an appropriate way. On the other hand, the subspace obtained by our method is almost parallel to the horizontal axis (the solid line). Figure 2b plots projected labeled data onto those subspaces. This shows that the labels of the data projected by PCA, FDA, and PNRL are barely distinguishable due to significant overlap, which makes class-prior estimation very hard. In contrast, we can easily separate the classes of samples projected by the proposed method, which eases class-prior estimation.
5.2.2 Benchmark Data. Next we apply the PURL method to benchmark data sets. To obtain low-dimensional representation, we set $m = 20$ and use a fully connected neural network with four layers ($d$-60-20-1; $v$ is $d$-60-20 and $g$ is 20-1) for $w$ except the text classification data set. For text classification data sets, we use another fully connected neural network with four layers ($d$-30-10-1) for $w$; $m = 10$. ReLU is used as activation functions for hidden layers, and batch normalization (Ioffe & Szegedy, 2015) is applied to all hidden layers. Stochastic gradient descent is used for optimization with learning rate 0.001. Also, weight decay with 0.0005 and gradient noise with 0.01 are applied. We iteratively update $w$ with four mini-batches and $v$ with one mini-batch.

We compare the accuracy of class-prior estimation with and without dimension reduction. For comparison, we also consider PCA, FDA, and PNRL. For PCA, we vary the numbers of components as follows: $\lfloor d/4 \rfloor$, $\lfloor d/2 \rfloor$, and $\lfloor 3d/4 \rfloor$, where $\lfloor \cdot \rfloor$ is the floor function. For FDA, the reduced dimension is 1 due to the property of FDA (Hastie, Tibshirani, & Friedman, 2009) in which the reduced dimension becomes the minimum of $m$ or (the number of classes − 1). The neural network for PNRL is the same as the one for the proposed method.

As a class-prior estimation method, we use the method based on the kernel mean embedding (KM) method proposed by Ramaswamy et al. (2016). With the estimated class-prior, we then train a fully connected neural network with five layers ($m$-300-300-300-1). ReLU is used as activation functions for hidden layers, and batch normalization is applied to all hidden layers. Except for text classification data sets, we train the neural networks by Adam (Kingma & Ba, 2015) until 200 epochs. For text classification data sets, we use AdaGrad (Duchi, Hazan, & Singer, 2011) and set the number of epochs to 300. For nonnegative PU learning (Kiryo, Niu, du Plessis, & Sugiyama, 2017), we use the sigmoid loss function and set $\beta$ and $\gamma$ in the letter to 0 and 1, respectively.

We use the ijcnn1, phishing, mushrooms, and a9a data sets taken from the LIBSVM web page (Chang & Lin, 2011). Also, we use the MNIST (LeCun, Bottou, Bengio, & Haffner, 1998), Fashion-MNIST (F-MNIST) (Xiao, Rasul, & Vollgraf, 2017), and 20 Newsgroups (Lang, 1995) data sets. For the MNIST and F-MNIST data sets, we divide the whole classes into two groups to make binary classification tasks. For the 20 Newsgroups data set, we use the “com” topic as the positive class and the “sci” topic as the negative class, and make 2000-dimensional $tf-idf$ vector. From the data sets, we draw $n_P = 1000$ positive and $n_U = 2000$ unlabeled samples. For validation, we use $n_P = 50$ and $n_U = 200$ samples.

---

4 See http://qwone.com/~jason/20Newsgroups/ for the details of topics.
Table 1 lists the average absolute error between the estimated class-prior and the true value. Overall, our proposed dimension-reduction method tends to outperform other methods, meaning that our method provides useful low-dimensional representation. Except for the icml1 data set, the error of FDA tends to be larger than the other methods, implying that regarding U data as N data does not help in class-prior estimation. For the mushrooms and a9a data sets, applying the unsupervised dimension-reduction method, PCA, does not improve the estimation accuracy, while our method reduces the error of class-prior estimation. In particular, for the 20 News-groups data set, the existing approaches (PCA, FDA, and PNRL) perform poorly. In contrast, applying our method significantly reduces the error of class-prior estimation.

Then we summarize the average misclassification rates in Table 2. Since the accuracy of class-prior estimation is improved on the mushrooms and a9a data sets, the classification accuracy is also improved. In particular, the classification results on the 20 Newsgroups data set with $\theta_p = 0.7$ are improved substantially. Overall, our proposed method tends to give the lower or comparable misclassification rates compared with the other methods.

6 Conclusion

In this letter, we have proposed an information-theoretic representation learning method from positive and unlabeled (PU) data. Our method is based on the information maximization principle, and we find low-dimensional representation maximally preserving a squared-loss variant of mutual information (SMI) between inputs and labels. Unlike the existing PU learning methods, since our representation learning method can be executed without knowing an estimate of the class-prior in advance, our method can also be used as preprocessing for the class-prior estimation method. Through numerical experiments, we demonstrated the effectiveness of our method.

Appendix A: Proof of Theorem 1

We express SMI in equation 2.1 as

$$\text{SMI} = \frac{\theta_p}{2} \int \left( \frac{p(x \mid y = +1)}{p(x)} - 1 \right)^2 p(x) dx$$

$$+ \frac{\theta_N}{2} \int \left( \frac{p(x \mid y = -1)}{p(x)} - 1 \right)^2 p(x) dx.$$  (A.1)
Table 1: Average Absolute Error (with Standard Error) between the Estimated Class-Prior and the True Value on Benchmark Data Sets over 20 Trials.

| Data Set | $\theta_p$ | None | $[d/4]$ | $[d/2]$ | $[3d/4]$ | FDA | PNRL | PURL |
|----------|-----------|------|---------|---------|---------|-----|------|------|
| ijcnn1   | 0.3       | 0.23 (0.02) | 0.26 (0.11) | 0.26 (0.11) | 0.28 (0.04) | 0.03 (0.01) | 0.26 (0.08) | 0.21 (0.07) |
|          | 0.5       | 0.18 (0.05) | 0.14 (0.09) | 0.14 (0.09) | 0.17 (0.06) | 0.04 (0.01) | 0.21 (0.08) | 0.19 (0.07) |
|          | 0.7       | 0.08 (0.01) | 0.11 (0.05) | 0.11 (0.05) | 0.10 (0.04) | 0.07 (0.01) | 0.11 (0.05) | 0.10 (0.01) |
| phishing | 0.3       | 0.02 (0.00) | 0.02 (0.00) | 0.02 (0.00) | 0.02 (0.00) | 0.04 (0.02) | 0.03 (0.01) | 0.02 (0.00) |
|          | 0.5       | 0.01 (0.00) | 0.01 (0.00) | 0.01 (0.00) | 0.01 (0.00) | 0.07 (0.03) | 0.04 (0.02) | 0.03 (0.02) |
|          | 0.7       | 0.02 (0.00) | 0.02 (0.00) | 0.02 (0.00) | 0.02 (0.00) | 0.11 (0.04) | 0.05 (0.03) | 0.02 (0.00) |
| mushrooms| 0.3       | 0.05 (0.01) | 0.05 (0.01) | 0.05 (0.01) | 0.05 (0.01) | 0.09 (0.03) | 0.03 (0.00) | 0.03 (0.00) |
|          | 0.5       | 0.05 (0.01) | 0.05 (0.01) | 0.05 (0.01) | 0.05 (0.01) | 0.16 (0.03) | 0.04 (0.01) | 0.04 (0.00) |
|          | 0.7       | 0.03 (0.01) | 0.03 (0.00) | 0.03 (0.00) | 0.03 (0.01) | 0.20 (0.06) | 0.03 (0.00) | 0.04 (0.03) |
| a9a      | 0.3       | 0.11 (0.02) | 0.11 (0.02) | 0.11 (0.02) | 0.11 (0.02) | 0.05 (0.01) | 0.08 (0.03) | 0.04 (0.00) |
|          | 0.5       | 0.10 (0.02) | 0.10 (0.02) | 0.10 (0.02) | 0.10 (0.02) | 0.09 (0.04) | 0.09 (0.03) | 0.04 (0.01) |
|          | 0.7       | 0.08 (0.03) | 0.08 (0.03) | 0.08 (0.03) | 0.08 (0.03) | 0.15 (0.06) | 0.08 (0.03) | 0.04 (0.01) |
| MNIST    | 0.3       | 0.09 (0.02) | 0.09 (0.02) | 0.09 (0.02) | 0.09 (0.02) | 0.27 (0.01) | 0.01 (0.00) | 0.05 (0.02) |
|          | 0.5       | 0.15 (0.11) | 0.15 (0.11) | 0.15 (0.11) | 0.15 (0.11) | 0.46 (0.01) | 0.03 (0.00) | 0.06 (0.03) |
|          | 0.7       | 0.60 (0.21) | 0.60 (0.21) | 0.60 (0.21) | 0.60 (0.21) | 0.65 (0.02) | 0.06 (0.01) | 0.07 (0.01) |
| F-MNIST  | 0.3       | 0.02 (0.00) | 0.02 (0.00) | 0.02 (0.00) | 0.02 (0.00) | 0.25 (0.01) | 0.03 (0.00) | 0.03 (0.00) |
|          | 0.5       | 0.03 (0.00) | 0.03 (0.00) | 0.03 (0.00) | 0.03 (0.00) | 0.45 (0.01) | 0.02 (0.00) | 0.04 (0.03) |
|          | 0.7       | 0.03 (0.00) | 0.03 (0.00) | 0.03 (0.00) | 0.03 (0.00) | 0.66 (0.02) | 0.02 (0.00) | 0.07 (0.03) |
| 20 News  | 0.3       | 0.04 (0.00) | 0.04 (0.01) | 0.04 (0.00) | 0.04 (0.00) | 0.29 (0.00) | 0.29 (0.09) | 0.03 (0.01) |
|          | 0.5       | 0.08 (0.03) | 0.06 (0.01) | 0.07 (0.01) | 0.08 (0.03) | 0.49 (0.00) | 0.25 (0.07) | 0.05 (0.01) |
|          | 0.7       | 0.69 (0.00) | 0.69 (0.00) | 0.69 (0.00) | 0.69 (0.00) | 0.13 (0.03) | 0.07 (0.01) | 0.05 (0.01) |

Note: “None” means that the class-prior is estimated without dimension-reduction methods, PCA is the principal component analysis, FDA is Fisher’s discriminant analysis, and PNRL is the supervised counterpart of the proposed method. The class-prior is estimated by the method based on kernel mean embedding. The boldface denotes the best and comparable approaches in terms of the average absolute error according to the t-test at the significance level 5%.
Table 2: Average Misclassification Rates (with Standard Error) on Benchmark Data Sets over 20 Trials.

| Data Set    | $\theta$ | None $d/4$ | None $d/2$ | None $3d/4$ | FDA $d/4$ | FDA $d/2$ | FDA $3d/4$ | PNRL $d/4$ | PNRL $d/2$ | PNRL $3d/4$ | PURL $d/4$ | PURL $d/2$ | PURL $3d/4$ |
|-------------|----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| ijcnn1      | 0.3      | 25.32     | 22.79     | 27.79     | 29.21     | 7.00      | 28.57     | 25.92     | 25.92     | 25.92     | 29.21     | 25.92     | 25.92     |
|             | 0.5      | 21.43     | 17.88     | 17.88     | 20.75     | 8.25      | 26.52     | 21.52     | 21.52     | 21.52     | 26.52     | 21.52     | 21.52     |
|             | 0.7      | 12.07     | 14.94     | 14.94     | 14.61     | 11.23     | 17.34     | 13.70     | 13.70     | 13.70     | 17.34     | 13.70     | 13.70     |
| phishing    | 0.3      | 7.41      | 7.46      | 7.46      | 7.57      | 10.30     | 11.09     | 7.62      | 7.62      | 7.62      | 11.09     | 7.62      | 7.62      |
|             | 0.5      | 12.85     | 9.75      | 9.75      | 9.82      | 24.43     | 32.02     | 10.05     | 10.05     | 10.05     | 32.02     | 10.05     | 10.05     |
|             | 0.7      | 8.07      | 8.85      | 8.85      | 7.63      | 25.62     | 29.04     | 8.02      | 8.02      | 8.02      | 29.04     | 8.02      | 8.02      |
| mushrooms   | 0.3      | 0.73      | 1.15      | 0.57      | 0.49      | 1.52      | 0.24      | 0.43      | 0.43      | 0.24      | 0.43      | 0.43      | 0.43      |
|             | 0.5      | 0.57      | 0.57      | 0.78      | 0.57      | 3.40      | 1.10      | 3.40      | 3.40      | 1.10      | 3.40      | 3.40      | 3.40      |
|             | 0.7      | 1.42      | 1.42      | 1.50      | 1.42      | 6.38      | 1.40      | 1.61      | 1.61      | 1.61      | 1.61      | 1.61      | 1.61      |
| a9a         | 0.3      | 24.93     | 26.49     | 26.49     | 26.20     | 21.09     | 26.31     | 22.32     | 22.32     | 22.32     | 26.31     | 22.32     | 22.32     |
|             | 0.5      | 30.35     | 26.07     | 26.07     | 29.52     | 22.70     | 27.48     | 23.70     | 23.70     | 23.70     | 27.48     | 23.70     | 23.70     |
|             | 0.7      | 20.35     | 20.54     | 20.54     | 19.94     | 20.59     | 20.59     | 19.39     | 19.39     | 19.39     | 20.59     | 19.39     | 19.39     |
| MNIST       | 0.3      | 24.58     | 17.99     | 18.02     | 21.18     | 20.92     | 11.76     | 13.54     | 13.54     | 13.54     | 11.76     | 13.54     | 13.54     |
|             | 0.5      | 23.00     | 22.35     | 22.35     | 23.55     | 42.10     | 15.35     | 18.64     | 18.64     | 18.64     | 15.35     | 18.64     | 18.64     |
|             | 0.7      | 53.34     | 54.42     | 54.42     | 53.39     | 42.10     | 15.35     | 18.64     | 18.64     | 18.64     | 42.10     | 15.35     | 18.64     |
| F-MNIST     | 0.3      | 14.88     | 17.99     | 17.99     | 22.18     | 20.54     | 12.74     | 13.22     | 13.22     | 13.22     | 12.74     | 13.22     | 13.22     |
|             | 0.5      | 13.40     | 22.35     | 22.35     | 23.55     | 42.10     | 15.35     | 18.64     | 18.64     | 18.64     | 42.10     | 15.35     | 18.64     |
|             | 0.7      | 9.94      | 52.19     | 54.42     | 52.19     | 53.39     | 55.65     | 16.38     | 16.38     | 16.38     | 53.39     | 16.38     | 16.38     |
| 20News      | 0.3      | 38.89     | 42.48     | 42.48     | 38.70     | 18.66     | 66.62     | 36.81     | 36.81     | 36.81     | 66.62     | 36.81     | 36.81     |
|             | 0.5      | 44.48     | 46.67     | 46.67     | 47.77     | 34.73     | 50.00     | 45.88     | 45.88     | 45.88     | 50.00     | 45.88     | 45.88     |
|             | 0.7      | 50.69     | 53.61     | 53.61     | 50.36     | 50.69     | 30.61     | 29.85     | 29.85     | 29.85     | 30.61     | 29.85     | 29.85     |

Note: The boldface denotes the best and comparable approaches in terms of the average absolute error according to the t-test at the significance level 5%.
From the marginal density $p(x)$, we have

\[
\theta_N \frac{p(x \mid y = -1)}{p(x)} = 1 - \theta_P \frac{p(x \mid y = +1)}{p(x)},
\]

\[
\theta_N \left( \frac{p(x \mid y = -1)}{p(x)} - 1 \right) = \theta_P \left( 1 - \frac{p(x \mid y = +1)}{p(x)} \right),
\]

\[
\left( \frac{p(x \mid y = -1)}{p(x)} - 1 \right)^2 = \frac{\theta_P}{\theta_N} \left( \frac{p(x \mid y = +1)}{p(x)} - 1 \right)^2,
\]

where the equality between the first and second equations can be confirmed by using $\theta_P + \theta_N = 1$. Plugging the last equation into the second term of equation A.1, we then obtain an expression of SMI only with positive and unlabeled data (PU-SMI) as

\[
\text{SMI} = \frac{\theta_P}{2\theta_N} \int \left( \frac{p(x \mid y = +1)}{p(x)} - 1 \right)^2 p(x) dx =: \text{PU-SMI}.
\]

\[\square\]

**Appendix B: Proof of Theorem 2**

Let

\[
s(x) := \frac{p(x \mid y = +1)}{p(x)}
\]

be the density ratio. Then PU-SMI can be expressed as

\[
\text{PU-SMI} = \frac{\theta_P}{2\theta_N} \int (s(x) - 1)^2 p(x) dx
\]

\[
= \frac{\theta_P}{\theta_N} \left( \frac{1}{2} \int s^2(x) p(x) dx - \frac{1}{2} \right),
\]

where $s(x)p(x) = p(x \mid y = +1)$ is used. Based on the Fenchel inequality (Boyd & Vandenberghe, 2004), for any function $f(x)$ in a function class $\mathcal{F}$, we have

\[
\frac{1}{2} s^2(x) \geq f(x)s(x) - \frac{1}{2} f^2(x).
\]
Then we obtain the lower bound of the PU-SMI by

\[
\text{PU-SMI} \geq \theta_P \left( - J_{\text{PU}}(f) - \frac{1}{2} \right),
\]

where

\[
J_{\text{PU}}(f) := \frac{1}{2} \int f^2(x)p(x)dx - \int f(x)p(x | y = +1)dx.
\]

Thus, from the Fenchel duality (Keziou, 2003; Nguyen, Wainwright, & Jordan, 2007), we have

\[
\text{PU-SMI} = \sup_{f \in \mathcal{F}} \theta_P \left( - J_{\text{PU}}(f) - \frac{1}{2} \right),
\]

where equality in the supremum is attained when \( f(x) = s(x) = p(x | y = +1)/p(x) \) and \( s \in \mathcal{F} \). □

Appendix C: Proof of Theorem 3

The idea of the proof is to view the approximated squared error as a perturbed optimization of expected one. In the analysis, we focus on the linear-in-parameter model \( w(x) = \sum_{\ell=1}^b \beta_\ell \phi_\ell(x) = \beta^\top \phi(x) \). We assume that \( 0 \leq \phi_\ell \leq 1 \) for all \( \ell = 1, \ldots, b \) and \( x \in \mathbb{R}^d \), and \( \hat{H}^U \) and \( H^U \) are positive-definite matrices. Recall

\[
\beta^* = \arg\min_{\beta \in \mathbb{R}^b} J_{\text{PU}}(\beta),
\]

where

\[
J_{\text{PU}}(\beta) = \frac{1}{2} \beta^\top H^U \beta^\top - \beta^\top h^P,
\]

\[
H^U = \int \phi(x)\phi(x)^\top p(x)dx,
\]

\[
h^P = \int \phi(x)p(x | y = +1)dx.
\]

Similarly,

\[
\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^b} \hat{J}_{\text{PU}}(\beta),
\]

where

\[
\hat{J}_{\text{PU}}(\beta) = \frac{1}{2} \beta^\top \hat{H}^U \beta^\top - \beta^\top \hat{h}^P,
\]

\[
\hat{H}^U = \int \phi(x)\phi(x)^\top \hat{p}(x)dx,
\]

\[
\hat{h}^P = \int \phi(x)p(x | y = +1)dx.
\]
where
\[
\begin{align*}
\hat{J}_{PU}(\beta) &= \frac{1}{2} \beta^\top \hat{H}^U \beta^\top - \beta^\top \hat{h}^P, \\
\hat{H}^U &= \frac{1}{n_U} \sum_{k=1}^{n_U} \phi(x_U^k) \phi(x_U^k)^\top, \\
\hat{h}^P &= \frac{1}{n_P} \sum_{i=1}^{n_P} \phi(x_P^i).
\end{align*}
\]

First we have the following lemma:

**Lemma 1.** Let \( \epsilon \) be the smallest eigenvalue of \( H^U \). We have
\[
J_{PU}(\beta) \geq J_{PU}(\beta^*) + \epsilon \| \beta - \beta^* \|^2_2.
\]
That is, \( J_{PU} \) satisfies the second-order growth condition (Bonnans & Shapiro, 1998).

**Proof.** Since \( H^U \) is positive definite, \( J_{PU}(\beta) \) is strongly convex with parameter at least \( \epsilon \). Then, we have
\[
\begin{align*}
J_{PU}(\beta) &\geq J_{PU}(\beta^*) + \nabla J_{PU}(\beta^*)^\top (\beta - \beta^*) + \epsilon \| \beta - \beta^* \|^2_2 \\
&= J_{PU}(\beta^*) + \epsilon \| \beta - \beta^* \|^2_2,
\end{align*}
\]
where the optimality condition \( \nabla J_{PU}(\beta^*) = 0 \) is used. \( \square \)

Let us define a set of perturbation parameters as
\[
\mathcal{U} := \{ (U^U, u^P) \mid U^U \in \mathbb{S}^b, \ u^P \in \mathbb{R}^b \},
\]
where \( \mathbb{S}^b \) is the set of symmetric \( b \times b \) matrices. With these perturbation parameters, we express \( \hat{H}^U \) and \( \hat{h}^P \) as \( U^U = \hat{H}^U - H^U \) and \( u^P = \hat{h}^P - h^P \), respectively. Let \( u \in \mathcal{U} \). Our perturbed objective function and the solution are given by
\[
\begin{align*}
J_{PU}(\beta, u) &:= \frac{1}{2} \beta^\top (H^U + U^U) \beta - \beta^\top (h^P + u^P), \\
\beta(u) &:= \arg\min_{\beta \in \mathbb{R}^b} J_{PU}(\beta, u).
\end{align*}
\]

Apparently \( J_{PU}(\beta) = J_{PU}(\beta, 0) \). Also, \( \hat{J}_{PU}(\beta) = J_{PU}(\beta, u) \) and \( \hat{\beta} = \beta(u) \) for \( u \neq 0 \). We then have the following lemma:

**Lemma 2.** \( J_{PU}(\cdot, u) - J_{PU}(\cdot) \) is Lipschitz continuous modulus \( \omega(u) = O(\| U^U \|_{\text{Fro}} + \| u^P \|^2_2), \) where \( \| \cdot \|_{\text{Fro}} \) is the Frobenius norm.
Proof. First, we have

$$J_{PU}(\beta, u) - J_{PU}(\beta) = \frac{1}{2} \beta^\top U U \beta - \beta^\top u^p.$$

The partial gradient is given by

$$\frac{\partial}{\partial \beta} (J_{PU}(\beta, u) - J_{PU}(\beta)) = U U \beta - u^p.$$

Let us define the $\delta$-ball of $\beta^*$ as $B_\delta(\beta^*) := \{ \beta : \| \beta - \beta^* \|_2 \leq \delta \}$ and $M = \| \beta^* \|_2$. For any $\beta \in B_\delta(\beta^*)$, we can easily show

$$\| \beta \|_2 \leq \| \beta - \beta^* \|_2 + \| \beta^* \|_2 \leq \delta + M,$$

where we first used the triangle inequality and then $\| \beta - \beta^* \|_2 \leq \delta$ and $M = \| \beta^* \|_2$. Thus,

$$\left\| \frac{\partial}{\partial \beta} (J_{PU}(\beta, u) - J_{PU}(\beta)) \right\|_2 \leq (\delta + M) \| U U \|_{Fro} + \| u^p \|_2.$$

This means that $J_{PU}(\cdot, u) - J_{PU}(\cdot)$ is Lipschitz continuous on $B_\delta(\beta^*)$ with a Lipschitz constant of order $O(\| U U \|_{Fro} + \| u^p \|_2)$.

Finally, we prove theorem 3.

Proof. According to the central limit theorem, we have

$$\| U U \|_{Fro} = O_p(1/\sqrt{n_U}), \quad \| u^p \|_2 = O_p(1/\sqrt{n_U})$$

as $n_p, n_u \to \infty$. Since we proved that $J_{PU}$ satisfies the second-order growth condition (see lemma 1) and $J_{PU}(\cdot, u) - J_{PU}(\cdot)$ is Lipschitz continuous modulus $\omega(u)$ (see lemma 2), we can use proposition 6.1 in Bonnans and Shapiro (1998) and have the first half of theorem 3:

$$\| \hat{\beta} - \beta^* \|_2 \leq \epsilon^{-1} \omega(u)$$

$$= O((\| U U \|_{Fro} + \| u^p \|_2))$$

$$= O_p(1/\sqrt{n_p} + 1/\sqrt{n_U}).$$

Next, we prove the latter half of theorem 3. For the squared errors, we have

$$| \widehat{J}_{PU}(\hat{\beta}) - J_{PU}(\beta^*) | \leq | \widehat{J}_{PU}(\hat{\beta}) - J_{PU}(\beta^*) | + | J_{PU}(\beta^*) - J_{PU}(\beta^*) |.$$
Here, we have

\[ \hat{J}_{PU}(\hat{\beta}) - J_{PU}(\beta^*) = \frac{1}{2} (\hat{\beta}^* - \beta^*)^\top \hat{H}_{U} (\hat{\beta} - \beta^*) - (\hat{\beta} - \beta^*)^\top \hat{H}_{U}. \]

\[ \hat{J}_{PU}(\beta^*) - J_{PU}(\beta^*) = \frac{1}{2} \beta^* \hat{U}^\top \beta^* - u_P^\top. \]

Since \(0 \leq \phi_\ell(x) \leq 1\) and \(M = \|\beta^*\|_2\), it leads to

\[ |\hat{J}_{PU}(\hat{\beta}) - J_{PU}(\beta^*)| \leq |\hat{J}_{PU}(\hat{\beta}) - \hat{J}_{PU}(\beta^*)| + |\hat{J}_{PU}(\beta^*) - J_{PU}(\beta^*)| \]
\[ \leq O_p(\|\hat{\beta} - \beta^*\|_2) + O_p(\|U\|_\text{Fro} + \|u\|_2) \]
\[ = O_p(1/\sqrt{n_P} + 1/\sqrt{n_U}). \]

Recall

\[ PU-\text{SMI}^* = \frac{\theta_P}{\theta_N} \left( - J_{PU}(\beta^*) - \frac{1}{2} \right). \]

\[ PU-\text{SMI} = \frac{\theta_P}{\theta_N} \left( - \hat{J}_{PU}(\hat{\beta}) - \frac{1}{2} \right). \]

We thus have

\[ |PU-\text{SMI}^* - PU-\text{SMI}| = \frac{\theta_P}{\theta_N} |\hat{J}_{PU}(\hat{\beta}) - J_{PU}(\beta^*)| \]
\[ = O_p(1/\sqrt{n_P} + 1/\sqrt{n_U}). \]

\[ \square \]

Appendix D: Effect of Dimension Reduction

In this appendix, we illustrate the effect of dimension reduction and how the number of samples affects class-prior estimation.

We use the artificial data set used in section 5.2 and vary both \(n_P\) and \(n_U\) from 500 to 5000. We set the true class-prior \(\theta_P\) as 0.5. The class-prior is estimated by the method based on kernel mean embedding (KM) (Ramawamy et al., 2016). To evaluate the performance with and without dimension reduction, we use the one-dimensional samples obtained by \(b^\top x\), where \(b = (1, 0)^\top\), and the original two-dimensional samples.

Figure 3a shows the mean absolute error (with its standard error) between the true and estimated class-priors over 10 trials. The error of KM without dimension reduction decreases the number of samples until around \(n_P = n_U = 2000\), but the error is not reduced even if we increase \(n_P = n_U = 5000\). In contrast, at \(n_P = n_U = 500\), the error of KM with dimension reduction is already smaller than that without dimension reduction.
Figure 3: Although the error of KM without dimension reduction decreases the number of samples until around \( n_P = n_U = 2,000 \), the error of KM with dimension reduction is smaller than that without dimension reduction. For the computation time, it grows with the number of samples. Since the short computation time and low absolute error are desirable, this result shows the effectiveness of dimension reduction.

Figure 3b shows the mean computation time (with its standard error) over 10 trials. The computation time grows with the number of samples. Since the short computation time and low absolute error are desirable, this result shows the effectiveness of dimension reduction.

Appendix E: Supervised Counterpart of the Proposed Method

In this appendix, we review the SMI estimation method (Suzuki et al., 2009; Sugiyama, 2013; Sakai & Sugiyama, 2014).

According to Suzuki and Sugiyama (2013), SMI can be expressed as

\[
SMI = \frac{1}{2} \sum_{y \in \{\pm 1\}} \int r^2(x, y)p(x)p(y)dx - \frac{1}{2},
\]

(E.1)

where

\[
r(x, y) := \frac{p(x, y)}{p(x)p(y)}
\]

Based on the Fenchel inequality (Boyd & Vandenberghe, 2004), for any function \( h : \mathbb{R}^d \times \{\pm 1\} \to \mathbb{R} \), we have

\[
\frac{1}{2} r^2(x, y) \geq h(x, y)r(x, y) - \frac{1}{2} h^2(x, y).
\]

(E.2)
where the equality condition is \( h(x, y) = r(x, y) \). We thus obtain the lower bound of SMI by

\[
\text{SMI} \geq L(h) - \frac{1}{2},
\]  

where

\[
L(h) := \frac{1}{2} \sum_{y \in \{\pm 1\}} \int h(x, y)p(x, y)dx - \frac{1}{2} \sum_{y \in \{\pm 1\}} \int h^2(x, y)p(x)p(y)dx.
\]  

To obtain an SMI estimate, we first train \( h \) with PN data by solving the following optimization problem:

\[
\max_{h \in \mathcal{H}} \tilde{L}(h),
\]  

where \( \mathcal{H} \) is a user-specified function class and \( \tilde{L} \) is sample approximation of \( L \):

\[
\tilde{L}(h) := \frac{1}{n} \sum_{i=1}^{n} h(x_i, y_i) - \frac{1}{2n} \sum_{y \in \{\pm 1\}} \hat{p}(y) \sum_{i=1}^{n} h^2(x_i, y) - \frac{1}{2}.
\]  

A simple approach to approximation of \( \hat{p}(y) \) is to use the number of labeled samples, that is, \( \hat{p}(y = +1) \) is approximated by the number of positive samples divided by that of all labeled samples. The SMI approximator from PN data is then given by

\[
\text{PN-SMI} := \tilde{L}(\hat{h}) - \frac{1}{2},
\]  

where \( \hat{h} := \arg\max_{h \in \mathcal{H}} \tilde{L}(h) \).

Similar to the proposed PURL method, we maximize PN-SMI to learn low-dimensional representation.

Acknowledgments

T.S. was supported by KAKENHI 15J09111. G.N. was supported by the JST CREST JPMJCR1403. M.S. was supported by KAKENHI 17H01760. We thank Ikko Yamane, Ryuichi Kiryo, and Takeshi Teshima for their comments.
References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/.

Basu, A., Harris, I. R., Hjort, N. L., & Jones, M. C. (1998). Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3), 549–559.

Bonnans, J. F., & Cominetti, R. (1996). Perturbed optimization in Banach spaces I: A general theory based on a weak directional constraint qualification; II: A theory based on a strong directional qualification condition; III: Semi-infinite optimization. *SIAM Journal on Control and Optimization*, 34(4), 1151–1171, 1172–1189, 1555–1567.

Bonnans, J. F., & Shapiro, A. (1998). Optimization problems with perturbations: A guided tour. *SIAM Review*, 40(2), 228–264.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2, 1–27. http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Chapelle, O., Schölkopf, B., & Zien, A. (Eds.). (2006). *Semi-supervised learning*. Cambridge, MA: MIT Press.

Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory* (2nd ed.). Hoboken, NJ: Wiley.

du Plessis, M. C., Niu, G., & Sugiyama, M. (2015). Convex formulation for learning from positive & unlabeled data. In *Proceedings of 32nd International Conference on Machine Learning*, 37 (pp. 1386–1394). PMLR.

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

du Plessis, M. C., & Sugiyama, M. (2014). Class prior estimation from positive and unlabeled data. *IEICE Transactions on Information and Systems*, E97-D(5), 1358–1362.

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12, 2121–2159.

Elkan, C., & Noto, K. (2008). Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 213–220). New York: ACM.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (pp. 315–323). PMLR.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge, MA: MIT Press.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning* (2nd ed.). New York: Springer-Verlag.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning* (pp. 448–456). PMLR.
Jain, S., White, M., & Radivojac, P. (2016). Estimating the class prior and posterior from noisy positives and unlabeled data. In D. D. Lee, M. Sugiyama, U. V. Luxburg, L. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29. Red Hook, NY: Curran.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., . . . Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM International Conference on Multimedia* (pp. 675–678). New York: ACM.

Kanamori, T., Hido, S., & Sugiyama, M. (2009). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10, 1391–1445.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2012). Statistical analysis of kernel-based least-squares density-ratio estimation. *Machine Learning*, 86(3), 335–367.

Keziou, A. (2003). Dual representation of φ-divergences and applications. *Comptes Rendus Mathématique*, 336(10), 857–862.

Khan, S., Bandyopadhyay, S., Ganguly, A., & Saigal, S. (2007). Relative performance of mutual information estimation methods for quantifying the dependence among short and noisy data. *Physical Review E*, 76, 026209.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the Third International Conference on Learning Representations*. OpenReview.net.

Kiryo, R., Niu, G., du Plessis, M. C., & Sugiyama, M. (2017). Positive-unlabeled learning with non-negative risk estimator. In L. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*, 30 (pp. 1674–1684). Red Hook, NY: Curran.

Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical Review E*, 69(6), 066138.

Krause, A., Perona, P., & Gomes, R. G. (2010). Discriminative clustering by regularized information maximization. In J. D. Lafferty, C. K. I. Williams, J. Shaw-Taylor, R. S. Zemel, & A. Culotta (Eds.), *Advances in neural information processing systems*, 23 (pp. 775–783). Red Hook, NY: Curran.

Lang, K. (1995). Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning* (pp. 331–339). San Mateo, CA: Morgan Kaufman.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.

Letouzey, F., Denis, F., & Gilleron, R. (2000). Learning from positive and unlabeled examples. In *Proceedings of the 11th International Conference on Algorithmic Learning Theory* (pp. 71–85). Berlin: Springer.

Li, K.-C. (1991). Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association*, 86(414), 316–327.

Li, W., Guo, Q., & Elkan, C. (2011). A positive and unlabeled learning algorithm for one-class classification of remote-sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 49(2), 717–725.

Linsker, R. (1988). Self-organization in a perceptual network. *Computer*, 21(3), 105–117.

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2012). *Foundations of machine learning*. Cambridge, MA: MIT Press.
Information-Theoretic Representation Learning for PU Classification

Moon, Y.-I., Rajagopalan, B., & Lall, U. (1995). Estimation of mutual information using kernel density estimators. *Physical Review E*, 52(3), 2318–2321.

Nguyen, X. L., Wainwright, M. J., & Jordan, M. I. (2007). Nonparametric estimation of the likelihood ratio and divergence functionals. In *Proceedings of the IEEE International Symposium on Information Theory* (pp. 2016–2020). Piscataway, NJ: IEEE.

Northcutt, C. G., Wu, T., & Chuang, I. L. (2017). Learning with confident examples: Rank pruning for robust classification with noisy labels. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence*. AUAI Press.

Pearson, K. (1990). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arise from random sampling. *Philosophical Magazine Series 5*, 50(302), 157–175.

Ramaswamy, H. G., Scott, C., & Tewari, A. (2016). Mixture proportion estimation via kernel embedding of distributions. In *Proceedings of the 33rd International Conference on Machine Learning*. PMLR.

Sakai, T., & Sugiyama, M. (2014). Computationally efficient estimation of squared-loss mutual information with multiplicative kernel models. *IEICE Transactions on Information and Systems*, E97-D(4), 968–971.

Sugiyama, M. (2013). Machine learning with squared-loss mutual information. *Entropy*, 15, 80–112.

Sugiyama, M., Suzuki, T., & Kanamori, T. (2012a). *Density ratio estimation in machine learning*. Cambridge: Cambridge University Press.

Sugiyama, M., Suzuki, T., & Kanamori, T. (2012b). Density ratio matching under the Bregman divergence: A unified framework of density ratio estimation. *Annals of the Institute of Statistical Mathematics*, 64(5), 1009–1044.

Suzuki, T., & Sugiyama, M. (2013). Sufficient dimension reduction via squared-loss mutual information. *Neural Computation*, 25(3), 725–758.

Suzuki, T., Sugiyama, M., Kanamori, T., & Sese, J. (2009). Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, 10, S2, 1–12.

Suzuki, T., Sugiyama, M., Sese, J., & Kanamori, T. (2008). Approximating mutual information by maximum likelihood density ratio estimation. In *Proceedings of ECML-PKDD 2008 Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery*, vol. 4 (pp. 5–20). PMLR.

Torkkola, K. (2003). Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research*, 3, 1415–1438.

Van Hulle, M. M. (2005). Edgeworth approximation of multivariate differential entropy. *Neural Computation*, 17(9), 1903–1910.

Ward, G., Hastie, T., Barry, S., Elith, J., & Leathwick, J. R. (2009). Presence-only data and the EM algorithm. *Biometrics*, 65(2), 554–563.

Xiao, H., Rasul, K., & Vollgraf, R. (2017). *Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms*. arXiv:1708.07747.

Received July 1, 2019; accepted August 5, 2020.