Semi-Supervised Classification Based on Classification from Positive and Unlabeled Data

Tomoya Sakai [1 2] **Marthinus Christoffel du Plessis** **Gang Niu** [1] **Masashi Sugiyama** [2 1]

Abstract

Most of the semi-supervised classification methods developed so far use unlabeled data for regularization purposes under particular distributional assumptions such as the cluster assumption. In contrast, recently developed methods of *classification from positive and unlabeled data* (PU classification) use unlabeled data for risk evaluation, i.e., label information is directly extracted from unlabeled data. In this paper, we extend PU classification to also incorporate negative data and propose a novel semi-supervised classification approach. We establish generalization error bounds for our novel methods and show that the bounds decrease with respect to the number of unlabeled data *without* the distributional assumptions that are required in existing semi-supervised classification methods. Through experiments, we demonstrate the usefulness of the proposed methods.

1. Introduction

Collecting a large amount of labeled data is a critical bottleneck in real-world machine learning applications due to the laborious manual annotation. In contrast, unlabeled data can often be collected automatically and abundantly, e.g., by a web crawler. This has led to the development of various semi-supervised classification algorithms over the past decades.

To leverage unlabeled data in training, most of the existing semi-supervised classification methods rely on particular assumptions on the data distribution (Chapelle et al., 2006). For example, the *manifold assumption* supposes that samples are distributed on a low-dimensional manifold in the data space (Belkin et al., 2006). In the existing framework, such a distributional assumption is encoded as a regularizer for training a classifier and *biases* the classifier toward a better one under the assumption. However, if such a distributional assumption contradicts the data distribution, the bias behaves adversely, and the performance of the obtained classifier becomes worse than the one obtained with supervised classification (Cozman et al., 2003; Sokolovska et al., 2008; Li & Zhou, 2015; Krijthe & Loog, 2017).

Recently, *classification from positive and unlabeled data* (PU classification) has been gathering growing attention (Elkan & Noto, 2008; du Plessis et al., 2014; 2015; Jain et al., 2016), which trains a classifier only from positive and unlabeled data without negative data. In PU classification, the *unbiased* risk estimators proposed in du Plessis et al. (2014; 2015) utilize unlabeled data for *risk evaluation*, implying that label information is directly extracted from unlabeled data without restrictive distributional assumptions, unlike existing semi-supervised classification methods that utilize unlabeled data for *regularization*. Furthermore, theoretical analysis (Niu et al., 2016) showed that PU classification (or its counterpart, *NU classification*, classification from negative and unlabeled data) is likely to outperform classification from positive and negative data (*PN classification*, i.e., ordinary supervised classification) depending on the number of positive, negative, and unlabeled samples. It is thus naturally expected that combining PN, PU, and NU classification can be a promising approach to semi-supervised classification without restrictive distributional assumptions.

In this paper, we propose a novel semi-supervised classification approach by considering convex combinations of the risk functions of PN, PU, and NU classification. Without any distributional assumption, we theoretically show that the confidence term of the generalization error bounds decreases at the optimal parametric rate with respect to the number of positive, negative, and unlabeled samples, and the variance of the proposed risk estimator is almost always smaller than the plain PN risk function given an infinite number of unlabeled samples. Through experiments, we analyze the behavior of the proposed approach and demonstrate the usefulness of the proposed semi-supervised classification methods.

[1]The University of Tokyo, Japan [2]RIKEN, Japan. Correspondence to: Tomoya Sakai <sakai@ms.k.u-tokyo.ac.jp>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

2. Background

In this section, we first introduce the notation commonly used in this paper and review the formulations of PN, PU, and NU classification.

2.1. Notation

Let random variables $\boldsymbol{x} \in \mathbb{R}^d$ and $y \in \{+1, -1\}$ be equipped with probability density $p(\boldsymbol{x}, y)$, where d is a positive integer. Let us consider a binary classification problem from $\boldsymbol{x}$ to y, given three sets of samples called the *positive* (P), *negative* (N), and *unlabeled* (U) data:

$$\mathcal{X}_{\mathrm{P}} := \{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}} \overset{\text{i.i.d.}}{\sim} p_{\mathrm{P}}(\boldsymbol{x}) := p(\boldsymbol{x} \mid y = +1),$$
$$\mathcal{X}_{\mathrm{N}} := \{\boldsymbol{x}_i^{\mathrm{N}}\}_{i=1}^{n_{\mathrm{N}}} \overset{\text{i.i.d.}}{\sim} p_{\mathrm{N}}(\boldsymbol{x}) := p(\boldsymbol{x} \mid y = -1),$$
$$\mathcal{X}_{\mathrm{U}} := \{\boldsymbol{x}_i^{\mathrm{U}}\}_{i=1}^{n_{\mathrm{U}}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}) := \theta_{\mathrm{P}} p_{\mathrm{P}}(\boldsymbol{x}) + \theta_{\mathrm{N}} p_{\mathrm{N}}(\boldsymbol{x}),$$

where

$$\theta_{\mathrm{P}} := p(y = +1), \quad \theta_{\mathrm{N}} := p(y = -1)$$

are the class-prior probabilities for the positive and negative classes such that $\theta_{\mathrm{P}} + \theta_{\mathrm{N}} = 1$.

Let $g\colon \mathbb{R}^d \to \mathbb{R}$ be an arbitrary real-valued decision function for binary classification, and classification is performed based on its sign. Let $\ell\colon \mathbb{R} \to \mathbb{R}$ be a loss function such that $\ell(m)$ generally takes a small value for large margin $m = yg(\boldsymbol{x})$. Let $R_{\mathrm{P}}(g)$, $R_{\mathrm{N}}(g)$, $R_{\mathrm{U,P}}(g)$, and $R_{\mathrm{U,N}}(g)$ be the risks of classifier g under loss ℓ:

$$R_{\mathrm{P}}(g) := \mathrm{E}_{\mathrm{P}}[\ell(g(\boldsymbol{x}))], \quad R_{\mathrm{N}}(g) := \mathrm{E}_{\mathrm{N}}[\ell(-g(\boldsymbol{x}))],$$
$$R_{\mathrm{U,P}}(g) := \mathrm{E}_{\mathrm{U}}[\ell(g(\boldsymbol{x}))], \quad R_{\mathrm{U,N}}(g) := \mathrm{E}_{\mathrm{U}}[\ell(-g(\boldsymbol{x}))],$$

where E_{P}, E_{N}, and E_{U} denote the expectations over $p_{\mathrm{P}}(\boldsymbol{x})$, $p_{\mathrm{N}}(\boldsymbol{x})$, and $p(\boldsymbol{x})$, respectively. Since we do not have any samples from $p(\boldsymbol{x}, y)$, the true risk $R(g) = \mathrm{E}_{p(\boldsymbol{x},y)}[\ell(yg(\boldsymbol{x}))]$, which we want to minimize, should be recovered without using $p(\boldsymbol{x}, y)$ as shown below.

2.2. PN Classification

In standard supervised classification (PN classification), we have both positive and negative data, i.e., fully labeled data. The goal of PN classification is to train a classifier using labeled data.

The risk in PN classification (the PN risk) is defined as

$$\begin{aligned} R_{\mathrm{PN}}(g) &:= \theta_{\mathrm{P}} \mathrm{E}_{\mathrm{P}}[\ell(g(\boldsymbol{x}))] + \theta_{\mathrm{N}} \mathrm{E}_{\mathrm{N}}[\ell(-g(\boldsymbol{x}))] \\ &= \theta_{\mathrm{P}} R_{\mathrm{P}}(g) + \theta_{\mathrm{N}} R_{\mathrm{N}}(g), \end{aligned} \tag{1}$$

which is equal to $R(g)$, but $p(\boldsymbol{x}, y)$ is not included. If we use the hinge loss function $\ell_{\mathrm{H}}(m) := \max(0, 1 - m)$, the PN risk coincides with the risk of the support vector machine (Vapnik, 1995).

2.3. PU Classification

In PU classification, we do not have labeled data for the negative class, but we can use unlabeled data drawn from marginal density $p(\boldsymbol{x})$. The goal of PU classification is to train a classifier using only positive and unlabeled data. The basic approach to PU classification is to discriminate P and U data (Elkan & Noto, 2008). However, naively classifying P and U data causes a bias.

To address this problem, du Plessis et al. (2014; 2015) proposed a risk equivalent to the PN risk but where $p_{\mathrm{N}}(\boldsymbol{x})$ is not included. The key idea is to utilize unlabeled data to evaluate the risk for negative samples in the PN risk. Replacing the second term in Eq. (1) with[1]

$$\theta_{\mathrm{N}} \mathrm{E}_{\mathrm{N}}[\ell(-g(\boldsymbol{x}))] = \mathrm{E}_{\mathrm{U}}[\ell(-g(\boldsymbol{x}))] - \theta_{\mathrm{P}} \mathrm{E}_{\mathrm{P}}[\ell(-g(\boldsymbol{x}))],$$

we obtain the risk in PU classification (the PU risk) as

$$\begin{aligned} R_{\mathrm{PU}}(g) &:= \theta_{\mathrm{P}} \mathrm{E}_{\mathrm{P}}[\widetilde{\ell}(g(\boldsymbol{x}))] + \mathrm{E}_{\mathrm{U}}[\ell(-g(\boldsymbol{x}))] \\ &= \theta_{\mathrm{P}} R_{\mathrm{P}}^{\mathrm{C}}(g) + R_{\mathrm{U,N}}(g), \end{aligned} \tag{2}$$

where $R_{\mathrm{P}}^{\mathrm{C}}(g) := \mathrm{E}_{\mathrm{P}}[\widetilde{\ell}(g(\boldsymbol{x}))]$ and $\widetilde{\ell}(m) = \ell(m) - \ell(-m)$ is a composite loss function.

Non-Convex Approach: If the loss function satisfies

$$\ell(m) + \ell(-m) = 1, \tag{3}$$

the composite loss function becomes $\widetilde{\ell}(m) = 2\ell(m) - 1$. We thus obtain the *non-convex* PU risk as

$$R_{\text{N-PU}}(g) := 2\theta_{\mathrm{P}} R_{\mathrm{P}}(g) + R_{\mathrm{U,N}}(g) - \theta_{\mathrm{P}}. \tag{4}$$

This formulation can be seen as cost-sensitive classification of P and U data with weight $2\theta_{\mathrm{P}}$ (du Plessis et al., 2014).

The ramp loss used in the robust support vector machine (Collobert et al., 2006),

$$\ell_{\mathrm{R}}(m) := \frac{1}{2} \max(0, \min(2, 1 - m)), \tag{5}$$

satisfies the condition (3). However, the use of the ramp loss (and any other losses that satisfy the condition (3)) yields a non-convex optimization problem, which may be solved locally by the *concave-convex procedure* (CCCP) (Yuille & Rangarajan, 2002; Collobert et al., 2006; du Plessis et al., 2014).

Convex Approach: If a convex surrogate loss function satisfies

$$\ell(m) - \ell(-m) = -m, \tag{6}$$

[1]The equation comes from the definition of the marginal density $p(\boldsymbol{x}) = \theta_{\mathrm{P}} p_{\mathrm{P}}(\boldsymbol{x}) + \theta_{\mathrm{N}} p_{\mathrm{N}}(\boldsymbol{x})$.

the composite loss function becomes a linear function $\widetilde{\ell}(m) = -m$ (see Table 1 in du Plessis et al., 2015). We thus obtain the *convex* PU risk as

$$R_{\text{C-PU}}(g) := \theta_{\text{P}} R_{\text{P}}^{\text{L}}(g) + R_{\text{U,N}}(g),$$

where $R_{\text{P}}^{\text{L}}(g) := \text{E}_{\text{P}}[-g(\boldsymbol{x})]$ is the risk with the linear loss $\ell_{\text{Lin}}(m) := -m$. This formulation yields the convex optimization problem that can be solved efficiently.

2.4. NU Classification

As a mirror of PU classification, we can consider NU classification. The risk in NU classification (the NU risk) is given by

$$\begin{aligned} R_{\text{NU}}(g) &:= \theta_{\text{N}} \text{E}_{\text{N}}[\widetilde{\ell}(-g(\boldsymbol{x}))] + \text{E}_{\text{U}}[\ell(g(\boldsymbol{x}))] \\ &= \theta_{\text{N}} R_{\text{N}}^{\text{C}}(g) + R_{\text{U,P}}(g), \end{aligned}$$

where $R_{\text{N}}^{\text{C}}(g) := \text{E}_{\text{N}}[\widetilde{\ell}(-g(\boldsymbol{x}))]$ is the risk function with the composite loss. Similarly to PU classification, the non-convex and convex NU risks are expressed as

$$R_{\text{N-NU}}(g) := 2\theta_{\text{N}} R_{\text{N}}(g) + R_{\text{U,P}}(g) - \theta_{\text{N}}, \quad (7)$$

$$R_{\text{C-NU}}(g) := \theta_{\text{N}} R_{\text{N}}^{\text{L}}(g) + R_{\text{U,P}}(g), \quad (8)$$

where $R_{\text{N}}^{\text{L}}(g) := \text{E}_{\text{N}}[g(\boldsymbol{x})]$ is the risk with the linear loss.

3. Semi-Supervised Classification Based on PN, PU, and NU Classification

In this section, we propose semi-supervised classification methods based on PN, PU, and NU classification.

3.1. PUNU Classification

A naive idea to build a semi-supervised classifier is to combine the PU and NU risks. For $\gamma \in [0, 1]$, let us consider a linear combination of the PU and NU risks:

$$R_{\text{PUNU}}^{\gamma}(g) := (1-\gamma) R_{\text{PU}}(g) + \gamma R_{\text{NU}}(g).$$

We refer to this combined method as *PUNU classification*.

If we use a loss function satisfying the condition (3), the non-convex PUNU risk $R_{\text{N-PUNU}}^{\gamma}(g)$ can be expressed as

$$\begin{aligned} R_{\text{N-PUNU}}^{\gamma}(g) = {} & 2(1-\gamma)\theta_{\text{P}} R_{\text{P}}(g) + 2\gamma\theta_{\text{N}} R_{\text{N}}(g) \\ & + \text{E}_{\text{U}}[(1-\gamma)\ell(-g(\boldsymbol{x})) + \gamma\ell(g(\boldsymbol{x}))] \\ & - (1-\gamma)\theta_{\text{P}} - \gamma\theta_{\text{N}}. \end{aligned}$$

Here, $R_{\text{N-PUNU}}^{1/2}(g)$ agrees with $R_{\text{PN}}(g)$ due to the condition (3). Thus, when $\gamma = 1/2$, PUNU classification is reduced to ordinary PN classification.

On the other hand, $\gamma = 1/2$ is still effective when the condition (6) is satisfied. Its risk $R_{\text{C-PUNU}}^{\gamma}(g)$ can be expressed as

$$\begin{aligned} R_{\text{C-PUNU}}^{\gamma}(g) = {} & (1-\gamma)\theta_{\text{P}} R_{\text{P}}^{\text{L}}(g) + \gamma\theta_{\text{N}} R_{\text{N}}^{\text{L}}(g) \\ & + \text{E}_{\text{U}}[(1-\gamma)\ell(g(\boldsymbol{x})) + \gamma\ell(-g(\boldsymbol{x}))]. \end{aligned}$$

Here, $(1-\gamma)\ell(g(\boldsymbol{x})) + \gamma\ell(-g(\boldsymbol{x}))$ can be regarded as a loss function for unlabeled samples with weight γ.

When $\gamma = 1/2$, unlabeled samples incur the same loss for the positive and negative classes. On the other hand, when $0 < \gamma < 1/2$, a smaller loss is incurred for the negative class than the positive class. Thus, unlabeled samples tend to be classified into the negative class. The opposite is true when $1/2 < \gamma < 1$.

3.2. PNU Classification

Another possibility of using PU and NU classification in semi-supervised classification is to combine the PN and PU/NU risks. For $\gamma \in [0, 1]$, let us consider linear combinations of the PN and PU/NU risks:

$$\begin{aligned} R_{\text{PNPU}}^{\gamma}(g) &:= (1-\gamma) R_{\text{PN}}(g) + \gamma R_{\text{PU}}(g), \\ R_{\text{PNNU}}^{\gamma}(g) &:= (1-\gamma) R_{\text{PN}}(g) + \gamma R_{\text{NU}}(g). \end{aligned}$$

In practice, we combine PNPU and PNNU classification and adaptively choose one of them with a new trade-off parameter $\eta \in [-1, 1]$ as

$$R_{\text{PNU}}^{\eta}(g) := \begin{cases} R_{\text{PNPU}}^{\eta}(g) & (\eta \geq 0), \\ R_{\text{PNNU}}^{-\eta}(g) & (\eta < 0). \end{cases}$$

We refer to the combined method as *PNU classification*. Clearly, PNU classification with $\eta = -1, 0, +1$ corresponds to NU, PN, and PU classification. As η gets large/small, the effect of the positive/negative classes is more emphasized.

In the theoretical analyses in Section 4, we denote the combinations of the PN risk with the non-convex PU/NU risks by $R_{\text{N-PNPU}}^{\gamma}$ and $R_{\text{N-PNNU}}^{\gamma}$, and that with the convex PU/NU risks by $R_{\text{C-PNPU}}^{\gamma}$ and $R_{\text{C-PNNU}}^{\gamma}$.

3.3. Practical Implementation

We have so far only considered the true risks R (with respect to the expectations over true data distributions). When a classifier is trained from samples in practice, we use the empirical risks $\widehat{R}$ where the expectations are replaced with corresponding sample averages.

More specifically, in the theoretical analysis in Section 4 and experiments in Section 5, we use a linear-in-parameter model given by $g(\boldsymbol{x}) = \sum_{j=1}^{b} w_j \phi_j(\boldsymbol{x}) = \boldsymbol{w}^{\top} \boldsymbol{\phi}(\boldsymbol{x})$, where $^{\top}$ denotes the transpose, b is the number of basis

functions, $\boldsymbol{w} = (w_1, \ldots, w_b)^\top$ is a parameter vector, and $\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_1(\boldsymbol{x}), \ldots, \phi_b(\boldsymbol{x}))^\top$ is a basis function vector. The parameter vector $\boldsymbol{w}$ is learned in order to minimize the ℓ_2-regularized empirical risk:

$$\min_{\boldsymbol{w}} \widehat{R}(g) + \lambda \boldsymbol{w}^\top \boldsymbol{w},$$

where $\lambda \geq 0$ is the regularization parameter.

4. Theoretical Analyses

In this section, we theoretically analyze the behavior of the empirical versions of the proposed semi-supervised classification methods. We first derive generalization error bounds and then discuss variance reduction. Finally, we discuss whether PUNU or PNU classification is more promising. All proofs can be found in Appendix A.

4.1. Generalization Error Bounds

Let $\mathcal{G}$ be a function class of bounded hyperplanes:

$$\mathcal{G} = \{g(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}) \rangle \mid \|\boldsymbol{w}\| \leq C_w, \|\boldsymbol{\phi}(\boldsymbol{x})\| \leq C_\phi\},$$

where C_w and C_ϕ are certain positive constants. Since ℓ_2-regularization is always included, we can naturally assume that the empirical risk minimizer g belongs to a certain $\mathcal{G}$. Denote by $\ell_{0\text{-}1}(m) = (1 - \mathrm{sign}(m))/2$ the *zero-one loss* and $I(g) = \mathrm{E}_{p(\boldsymbol{x},y)}[\ell_{0\text{-}1}(yg(\boldsymbol{x}))]$ the risk of g for binary classification, i.e., the generalization error of g. In the following, we study upper bounds of $I(g)$ holding uniformly for all $g \in \mathcal{G}$. We respectively focus on the *(scaled) ramp and squared losses* for the non-convex and convex methods due to limited space. Similar results can be obtained with a little more effort if other eligible losses are used. For convenience, we define a function as

$$\chi(c_\mathrm{P}, c_\mathrm{N}, c_\mathrm{U}) = c_\mathrm{P}\theta_\mathrm{P}/\sqrt{n_\mathrm{P}} + c_\mathrm{N}\theta_\mathrm{N}/\sqrt{n_\mathrm{N}} + c_\mathrm{U}/\sqrt{n_\mathrm{U}}.$$

Non-Convex Methods: A key observation is that $\ell_{0\text{-}1}(m) \leq 2\ell_\mathrm{R}(m)$, and consequently $I(g) \leq 2R(g)$. Note that by definition we have

$$R^\gamma_{\text{N-PUNU}}(g) = R^\gamma_{\text{N-PNPU}}(g) = R^\gamma_{\text{N-PNNU}}(g) = R(g).$$

The theorem below can be proven using the Rademacher analysis (see, for example, Mohri et al., 2012; Ledoux & Talagrand, 1991).

Theorem 1 *Let $\ell_\mathrm{R}(m)$ be the loss for defining the empirical risks. For any $\delta > 0$, the following inequalities hold separately with probability at least $1 - \delta$ for all $g \in \mathcal{G}$:*

$$I(g) \leq 2\widehat{R}^\gamma_{\text{N-PUNU}}(g) + C_{w,\phi,\delta} \cdot \chi(2 - 2\gamma, 2\gamma, |2\gamma - 1|),$$
$$I(g) \leq 2\widehat{R}^\gamma_{\text{N-PNPU}}(g) + C_{w,\phi,\delta} \cdot \chi(1 + \gamma, 1 - \gamma, \gamma),$$
$$I(g) \leq 2\widehat{R}^\gamma_{\text{N-PNNU}}(g) + C_{w,\phi,\delta} \cdot \chi(1 - \gamma, 1 + \gamma, \gamma),$$

where $C_{w,\phi,\delta} = 2C_wC_\phi + \sqrt{2\ln(3/\delta)}$.

Theorem 1 guarantees that when $\ell_\mathrm{R}(m)$ is used, $I(g)$ can be bounded from above by two times the empirical risks, i.e., $2\widehat{R}^\gamma_{\text{N-PUNU}}(g)$, $2\widehat{R}^\gamma_{\text{N-PNPU}}(g)$, and $2\widehat{R}^\gamma_{\text{N-PNNU}}(g)$, plus the corresponding confidence terms of order

$$\mathcal{O}_p(1/\sqrt{n_\mathrm{P}} + 1/\sqrt{n_\mathrm{N}} + 1/\sqrt{n_\mathrm{U}}).$$

Since n_P, n_N, and n_U can increase independently, this is already the optimal convergence rate without any additional assumption (Vapnik, 1998; Mendelson, 2008).

Convex Methods: Analogously, we have $\ell_{0\text{-}1}(m) \leq 4\ell_\mathrm{S}(m)$ for the squared loss. However, it is too loose when $|m| \gg 0$. Fortunately, we do not have to use $\ell_\mathrm{S}(m)$ if we work on the generalization error rather than the estimation error. To this end, we define the *truncated (scaled) squared loss* $\ell_\mathrm{TS}(m)$ as

$$\ell_\mathrm{TS}(m) = \begin{cases} \ell_\mathrm{S}(m) & 0 < m \leq 1, \\ \ell_{0\text{-}1}(m)/4 & \text{otherwise}, \end{cases}$$

so that $\ell_{0\text{-}1}(m) \leq 4\ell_\mathrm{TS}(m)$ is much tighter. For $\ell_\mathrm{TS}(m)$, $R_{\text{C-PU}}(g)$ and $R_{\text{C-NU}}(g)$ need to be redefined as follows (see du Plessis et al., 2015):

$$R_{\text{C-PU}}(g) := \theta_\mathrm{P} R'_\mathrm{P}(g) + R_{\mathrm{U,N}}(g),$$
$$R_{\text{C-NU}}(g) := \theta_\mathrm{N} R'_\mathrm{N}(g) + R_{\mathrm{U,P}}(g),$$

where $R'_\mathrm{P}(g)$ and $R'_\mathrm{N}(g)$ are simply $R_\mathrm{P}(g)$ and $R_\mathrm{N}(g)$ w.r.t. the composite loss $\widetilde{\ell}_\mathrm{TS}(m) = \ell_\mathrm{TS}(m) - \ell_\mathrm{TS}(-m)$. The condition $\widetilde{\ell}_\mathrm{TS}(m) \neq -m$ means the loss of convexity, but the equivalence is not lost; indeed, we still have

$$R^\gamma_{\text{C-PUNU}}(g) = R^\gamma_{\text{C-PNPU}}(g) = R^\gamma_{\text{C-PNNU}}(g) = R(g).$$

Theorem 2 *Let $\ell_\mathrm{TS}(m)$ be the loss for defining the empirical risks (where $R_{\text{C-PU}}(g)$ and $R_{\text{C-NU}}(g)$ are redefined). For any $\delta > 0$, the following inequalities hold separately with probability at least $1 - \delta$ for all $g \in \mathcal{G}$:*

$$I(g) \leq 4\widehat{R}^\gamma_{\text{C-PUNU}}(g) + C'_{w,\phi,\delta} \cdot \chi(1 - \gamma, \gamma, 1),$$
$$I(g) \leq 4\widehat{R}^\gamma_{\text{C-PNPU}}(g) + C'_{w,\phi,\delta} \cdot \chi(1, 1 - \gamma, \gamma),$$
$$I(g) \leq 4\widehat{R}^\gamma_{\text{C-PNNU}}(g) + C'_{w,\phi,\delta} \cdot \chi(1 - \gamma, 1, \gamma),$$

where $C'_{w,\phi,\delta} = 4C_wC_\phi + \sqrt{2\ln(4/\delta)}$.

Theorem 2 ensures that when $\ell_\mathrm{TS}(m)$ is used (for evaluating the empirical risks rather than learning the empirical risk minimizers), $I(g)$ can be bounded from above by four times the empirical risks plus confidence terms in the optimal parametric rate. As $\ell_\mathrm{TS}(m) \leq \ell_\mathrm{S}(m)$, Theorem 2 is valid (but weaker) if all empirical risks are w.r.t. $\ell_\mathrm{S}(m)$.

4.2. Variance Reduction

Our empirical risk estimators proposed in Section 3 are all unbiased. The next question is whether their variance can be smaller than that of $\widehat{R}_{\mathrm{PN}}(g)$, i.e., whether $\mathcal{X}_{\mathrm{U}}$ can help reduce the variance in estimating $R(g)$. To answer this question, pick any g of interest. For simplicity, we assume that $n_{\mathrm{U}} \to \infty$, to illustrate the maximum variance reduction that could be achieved. Due to limited space, we only focus on the non-convex methods.

Similarly to $R_{\mathrm{P}}(g)$ and $R_{\mathrm{N}}(g)$, let $\sigma_{\mathrm{P}}^2(g)$ and $\sigma_{\mathrm{N}}^2(g)$ be the corresponding variance:

$$\sigma_{\mathrm{P}}^2(g) := \mathrm{Var}_{\mathrm{P}}[\ell(g(\boldsymbol{x}))], \quad \sigma_{\mathrm{N}}^2(g) := \mathrm{Var}_{\mathrm{N}}[\ell(-g(\boldsymbol{x}))],$$

where $\mathrm{Var}_{\mathrm{P}}$ and $\mathrm{Var}_{\mathrm{N}}$ denote the variance over $p_{\mathrm{P}}(\boldsymbol{x})$ and $p_{\mathrm{N}}(\boldsymbol{x})$. Moreover, denote by $\psi_{\mathrm{P}} = \theta_{\mathrm{P}}^2\sigma_{\mathrm{P}}^2(g)/n_{\mathrm{P}}$ and $\psi_{\mathrm{N}} = \theta_{\mathrm{N}}^2\sigma_{\mathrm{N}}^2(g)/n_{\mathrm{N}}$ for short, and let Var be the variance over $p_{\mathrm{P}}(\boldsymbol{x}_1^{\mathrm{P}}) \cdots p_{\mathrm{P}}(\boldsymbol{x}_{n_{\mathrm{P}}}^{\mathrm{P}}) \cdot p_{\mathrm{N}}(\boldsymbol{x}_1^{\mathrm{N}}) \cdots p_{\mathrm{N}}(\boldsymbol{x}_{n_{\mathrm{N}}}^{\mathrm{N}}) \cdot p(\boldsymbol{x}_1^{\mathrm{U}}) \cdots p(\boldsymbol{x}_{n_{\mathrm{U}}}^{\mathrm{U}})$.

Theorem 3 *Assume $n_{\mathrm{U}} \to \infty$. For any fixed g, let*

$$\gamma_{\text{N-PUNU}} = \operatorname*{argmin}_{\gamma} \mathrm{Var}[\widehat{R}^{\gamma}_{\text{N-PUNU}}(g)] = \frac{\psi_{\mathrm{P}}}{\psi_{\mathrm{P}} + \psi_{\mathrm{N}}}. \quad (9)$$

Then, we have $\gamma_{\text{N-PUNU}} \in [0, 1]$. Further, $\mathrm{Var}[\widehat{R}^{\gamma}_{\text{N-PUNU}}(g)] < \mathrm{Var}[\widehat{R}_{\mathrm{PN}}(g)]$ for all $\gamma \in (2\gamma_{\text{N-PUNU}} - 1/2, 1/2)$ if $\psi_{\mathrm{P}} < \psi_{\mathrm{N}}$, or for all $\gamma \in (1/2, 2\gamma_{\text{N-PUNU}} - 1/2)$ if $\psi_{\mathrm{P}} > \psi_{\mathrm{N}}$.[2]

Theorem 3 guarantees that the variance is always reduced by $\widehat{R}^{\gamma}_{\text{N-PUNU}}(g)$ if γ is close to $\gamma_{\text{N-PUNU}}$, which is optimal for variance reduction. The interval of such good γ values has the length $\min\{|\psi_{\mathrm{P}} - \psi_{\mathrm{N}}|/(\psi_{\mathrm{P}} + \psi_{\mathrm{N}}), 1/2\}$. In particular, if $3\psi_{\mathrm{P}} \leq \psi_{\mathrm{N}}$ or $\psi_{\mathrm{P}} \geq 3\psi_{\mathrm{N}}$, the length is $1/2$.

Theorem 4 *Assume $n_{\mathrm{U}} \to \infty$. For any fixed g, let*

$$\gamma_{\text{N-PNPU}} = \operatorname*{argmin}_{\gamma} \mathrm{Var}[\widehat{R}^{\gamma}_{\text{N-PNPU}}(g)] = \frac{\psi_{\mathrm{N}} - \psi_{\mathrm{P}}}{\psi_{\mathrm{P}} + \psi_{\mathrm{N}}}, \quad (10)$$

$$\gamma_{\text{N-PNNU}} = \operatorname*{argmin}_{\gamma} \mathrm{Var}[\widehat{R}^{\gamma}_{\text{N-PNNU}}(g)] = \frac{\psi_{\mathrm{P}} - \psi_{\mathrm{N}}}{\psi_{\mathrm{P}} + \psi_{\mathrm{N}}}. \quad (11)$$

Then, we have $\gamma_{\text{N-PNPU}} \in [0, 1]$ if $\psi_{\mathrm{P}} \leq \psi_{\mathrm{N}}$ or $\gamma_{\text{N-PNNU}} \in [0, 1]$ if $\psi_{\mathrm{P}} \geq \psi_{\mathrm{N}}$. Additionally, $\mathrm{Var}[\widehat{R}^{\gamma}_{\text{N-PNPU}}(g)] < \mathrm{Var}[\widehat{R}_{\mathrm{PN}}(g)]$ for all $\gamma \in (0, 2\gamma_{\text{N-PNPU}})$ if $\psi_{\mathrm{P}} < \psi_{\mathrm{N}}$, or $\mathrm{Var}[\widehat{R}^{\gamma}_{\text{N-PNNU}}(g)] < \mathrm{Var}[\widehat{R}_{\mathrm{PN}}(g)]$ for all $\gamma \in (0, 2\gamma_{\text{N-PNNU}})$ if $\psi_{\mathrm{P}} > \psi_{\mathrm{N}}$.

Theorem 4 implies that the variance of $\widehat{R}_{\mathrm{PN}}(g)$ is reduced by either $\widehat{R}^{\gamma}_{\text{N-PNPU}}(g)$ if $\psi_{\mathrm{P}} \leq \psi_{\mathrm{N}}$ or $\widehat{R}^{\gamma}_{\text{N-PNNU}}(g)$ if $\psi_{\mathrm{P}} \geq \psi_{\mathrm{N}}$, where γ should be close to $\gamma_{\text{N-PNPU}}$ or $\gamma_{\text{N-PNNU}}$. The range of such good γ values is of length $\min\{2|\psi_{\mathrm{P}} - \psi_{\mathrm{N}}|/(\psi_{\mathrm{P}} + \psi_{\mathrm{N}}), 1\}$. In particular, if $3\psi_{\mathrm{P}} \leq \psi_{\mathrm{N}}$, $\widehat{R}^{\gamma}_{\text{N-PNPU}}(g)$ given any $\gamma \in (0, 1)$ can reduce the variance, and if $\psi_{\mathrm{P}} \geq 3\psi_{\mathrm{N}}$, $\widehat{R}^{\gamma}_{\text{N-PNNU}}(g)$ given any $\gamma \in (0, 1)$ can reduce the variance.

As a corollary of Theorems 3 and 4, the minimum variance achievable by $\widehat{R}^{\gamma}_{\text{N-PUNU}}(g)$, $\widehat{R}^{\gamma}_{\text{N-PNPU}}(g)$, and $\widehat{R}^{\gamma}_{\text{N-PNNU}}(g)$ at their optimal $\gamma_{\text{N-PUNU}}$, $\gamma_{\text{N-PNPU}}$, and $\gamma_{\text{N-PNNU}}$ is exactly the same, namely, $4\psi_{\mathrm{P}}\psi_{\mathrm{N}}/(\psi_{\mathrm{P}} + \psi_{\mathrm{N}})$. Nevertheless, $\widehat{R}^{\gamma}_{\text{N-PNPU}}(g)$ and $\widehat{R}^{\gamma}_{\text{N-PNNU}}(g)$ have a much wider range of nice γ values than $\widehat{R}^{\gamma}_{\text{N-PUNU}}(g)$.

If we further assume that $\sigma_{\mathrm{P}}(g) = \sigma_{\mathrm{N}}(g)$, the condition in Theorems 3 and 4 as to whether $\psi_{\mathrm{P}} \leq \psi_{\mathrm{N}}$ or $\psi_{\mathrm{P}} \geq \psi_{\mathrm{N}}$ will be independent of g. Also, it will coincide with the condition in *Theorem 7* in Niu et al. (2016) where the minimizers of $\widehat{R}_{\mathrm{PN}}(g)$, $\widehat{R}_{\mathrm{PU}}(g)$ and $\widehat{R}_{\mathrm{NU}}(g)$ are compared.

A final remark is that learning is uninvolved in Theorems 3 and 4, such that $\ell(m)$ can be any loss that satisfies $\ell(m) + \ell(-m) = 1$, and g can be any fixed decision function. For instance, we may adopt $\ell_{0\text{-}1}(m)$ and pick some g resulted from some other learning methods. As a consequence, the variance of $\widehat{I}_{\mathrm{PN}}(g)$ over the validation data can be reduced, and then the cross-validation should be more stable, given that n_{U} is sufficiently large. Therefore, even without being minimized, our proposed risk estimators are themselves of practical importance.

4.3. PUNU vs. PNU Classification

We discuss here which approach, PUNU or PNU classification, is more promising according to state-of-the-art theoretical comparisons (Niu et al., 2016), which are based on estimation error bounds.

Let $\widehat{g}_{\mathrm{PN}}$, $\widehat{g}_{\mathrm{PU}}$, and $\widehat{g}_{\mathrm{NU}}$ be the minimizers of $\widehat{R}_{\mathrm{PN}}(g)$, $\widehat{R}_{\mathrm{PU}}(g)$, and $\widehat{R}_{\mathrm{NU}}(g)$, respectively. Let $\alpha_{\mathrm{PU,PN}} := (\theta_{\mathrm{P}}/\sqrt{n_{\mathrm{P}}} + 1/\sqrt{n_{\mathrm{U}}})/(\theta_{\mathrm{N}}/\sqrt{n_{\mathrm{N}}})$ and $\alpha_{\mathrm{NU,PN}} := (\theta_{\mathrm{N}}/\sqrt{n_{\mathrm{N}}} + 1/\sqrt{n_{\mathrm{U}}})/(\theta_{\mathrm{P}}/\sqrt{n_{\mathrm{P}}})$. The finite-sample comparisons state that if $\alpha_{\mathrm{PU,PN}} > 1$ ($\alpha_{\mathrm{NU,PN}} > 1$), PN classification is more promising than PU (NU) classification, i.e., $R(\widehat{g}_{\mathrm{PN}}) < R(\widehat{g}_{\mathrm{PU}})$ ($R(\widehat{g}_{\mathrm{PN}}) < R(\widehat{g}_{\mathrm{NU}})$); otherwise PU (NU) classification is more promising than PN classification (cf. Section 3.2 in Niu et al., 2016).

Suppose that n_{U} is not sufficiently large against n_{P} and n_{N}. According to the finite-sample comparisons, PN classification is most promising, and either PU or NU classification is the second best, i.e., $R(\widehat{g}_{\mathrm{PN}}) < R(\widehat{g}_{\mathrm{PU}}) < R(\widehat{g}_{\mathrm{NU}})$ or $R(\widehat{g}_{\mathrm{PN}}) < R(\widehat{g}_{\mathrm{NU}}) < R(\widehat{g}_{\mathrm{PU}})$. On the other hand, if n_{U} is sufficiently large ($n_{\mathrm{U}} \to \infty$, which is faster than $n_{\mathrm{P}}, n_{\mathrm{N}} \to \infty$), we have the asymptotic comparisons: $\alpha^*_{\mathrm{PU,PN}} = \lim_{n_{\mathrm{P}}, n_{\mathrm{N}}, n_{\mathrm{U}} \to \infty} \alpha_{\mathrm{PU,PN}}$, $\alpha^*_{\mathrm{NU,PN}} =$

[2] Being fixed means g is determined before seeing the data for evaluating the empirical risk. For example, if g is trained by some learning method, and the empirical risk is subsequently evaluated on the validation/test data, g is regarded as fixed in the evaluation.

$\lim_{n_{\mathrm{P}},n_{\mathrm{N}},n_{\mathrm{U}}\to\infty}\alpha_{\mathrm{NU,PN}}$, and $\alpha^*_{\mathrm{PU,PN}}\cdot\alpha^*_{\mathrm{NU,PN}}=1$. From the last equation, if $\alpha^*_{\mathrm{PU,PN}}<1$, then $\alpha^*_{\mathrm{NU,PN}}>1$, implying that PU (PN) classification is more promising than PN (NU) classification, i.e., $R(\widehat{g}_{\mathrm{PU}})<R(\widehat{g}_{\mathrm{PN}})<R(\widehat{g}_{\mathrm{NU}})$. Similarly, when $\alpha^*_{\mathrm{PU,PN}}>1$ and $\alpha^*_{\mathrm{NU,PN}}<1$, $R(\widehat{g}_{\mathrm{NU}})<R(\widehat{g}_{\mathrm{PN}})<R(\widehat{g}_{\mathrm{PU}})$ (cf. Section 3.3 in Niu et al., 2016).

In real-world applications, since we do not know whether the number of unlabeled samples is sufficiently large or not, a practical approach is to combine the best methods in both the finite-sample and asymptotic cases. PNU classification is the combination of the best methods in both cases, but PUNU classification is not. In addition, PUNU classification includes the worst one in its combination in both cases. From this viewpoint, PNU classification would be more promising than PUNU classification, as demonstrated in the experiments shown in the next section.

5. Experiments

In this section, we first numerically analyze the proposed approach and then compare the proposed semi-supervised classification methods against existing methods. All experiments were carried out using a PC equipped with two 2.60GHz Intel® Xeon® E5-2640 v3 CPUs.

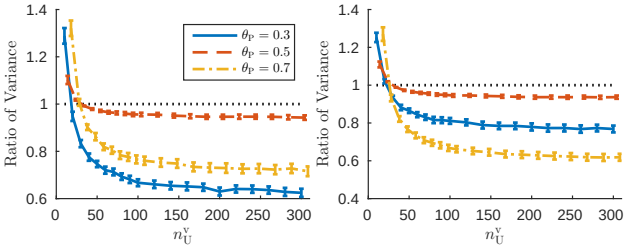
5.1. Experimental Analyses

Here, we numerically analyze the behavior of our proposed approach. Due to limited space, we show results on two out of six data sets and move the rest to Appendix C.

Common Setup: As a classifier, we use the Gaussian kernel model: $g(\boldsymbol{x})=\sum_{i=1}^{n}w_i\exp(-\|\boldsymbol{x}-\boldsymbol{x}_i\|^2/(2\sigma^2))$, where $n=n_{\mathrm{P}}+n_{\mathrm{N}}$, $\{w_i\}_{i=1}^{n}$ are the parameters, $\{\boldsymbol{x}_i\}_{i=1}^{n}=\mathcal{X}_{\mathrm{P}}\cup\mathcal{X}_{\mathrm{N}}$, and $\sigma>0$ is the Gaussian bandwidth. The bandwidth candidates are $\{1/8,1/4,1/2,1,3/2,2\}\times\mathrm{median}(\|\boldsymbol{x}_i-\boldsymbol{x}_j\|_{i,j=1}^{n})$. The classifier trained by minimizing the empirical PN risk is denoted by $\widehat{g}_{\mathrm{PN}}$. The number of labeled samples for training is 20, where the class-prior was 0.5. In all experiments, we used the squared loss for training. We note that the class-prior of test data was the same as that of unlabeled data.

Variance Reduction in Practice: Here, we numerically investigate how many unlabeled samples are sufficient in practice such that the variance of the empirical PNU risk is smaller than that of the PN risk: $\mathrm{Var}[\widehat{R}^{\eta}_{\mathrm{PNU}}(g)]<\mathrm{Var}[\widehat{R}_{\mathrm{PN}}(g)]$ given a fixed classifier g.

As the fixed classifier, we used the classifier $\widehat{g}_{\mathrm{PN}}$, where the hyperparameters were determined by five-fold cross-validation. To compute the variance of the empirical PN and PNU risks, $\mathrm{Var}[\widehat{R}_{\mathrm{PN}}(\widehat{g}_{\mathrm{PN}})]$ and $\mathrm{Var}[\widehat{R}^{\eta}_{\mathrm{PNU}}(\widehat{g}_{\mathrm{PN}})]$, we repeatedly drew additional $n^{\mathrm{V}}_{\mathrm{P}}=10$ positive, $n^{\mathrm{V}}_{\mathrm{N}}=10$ negative, and $n^{\mathrm{V}}_{\mathrm{U}}$ unlabeled samples from the rest of the data set. The additional samples were also used for approximating $\widehat{\sigma}_{\mathrm{P}}(\widehat{g}_{\mathrm{PN}})$ and $\widehat{\sigma}_{\mathrm{N}}(\widehat{g}_{\mathrm{PN}})$ to compute η, i.e., γ in Eqs.(10) and (11).



(a) Phoneme ($d=5$) (b) Magic ($d=10$)

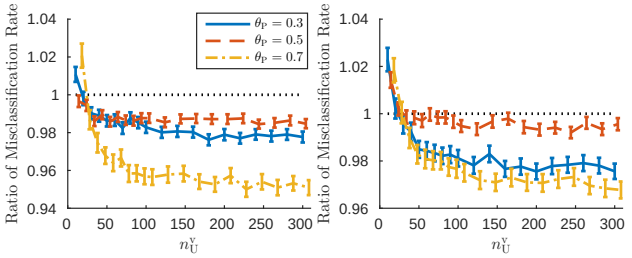
Figure 1. Average and standard error of the ratio between the variance of empirical PNU risk and that of PN risk, $\mathrm{Var}[\widehat{R}^{\eta}_{\mathrm{PNU}}(\widehat{g}_{\mathrm{PN}})]/\mathrm{Var}[\widehat{R}_{\mathrm{PN}}(\widehat{g}_{\mathrm{PN}})]$, as a function of the number of unlabeled samples over 100 trials. Although the variance reduction is proved for an infinite number of samples, it can be observed with a finite number of samples.

Figure 1 shows the ratio between the variance of the empirical PNU risk and that of the PN risk, $\mathrm{Var}[\widehat{R}^{\eta}_{\mathrm{PNU}}(\widehat{g}_{\mathrm{PN}})]/\mathrm{Var}[\widehat{R}_{\mathrm{PN}}(\widehat{g}_{\mathrm{PN}})]$. The number of unlabeled samples for validation $n^{\mathrm{V}}_{\mathrm{U}}$ increases from 10 to 300. We see that with a rather small number of unlabeled samples, the ratio becomes less than 1. That is, the variance of the empirical PNU risk becomes smaller than that of the PN risk. This implies that although the variance reduction is proved for an infinite number of unlabeled samples, it can be observed under a finite number of samples in practice.

Compared to when $\theta_{\mathrm{P}}=0.3$ and 0.7, the effect of variance reduction is small when $\theta_{\mathrm{P}}=0.5$. This is because if we assume $\sigma_{\mathrm{P}}(g)\approx\sigma_{\mathrm{N}}(g)$, when $n_{\mathrm{P}}\approx n_{\mathrm{N}}$ and $\theta_{\mathrm{P}}=0.5$, we have $\gamma_{\text{N-PNPU}}\approx\gamma_{\text{N-PNNU}}\approx 0$ (because $\psi_{\mathrm{P}}\approx\psi_{\mathrm{N}}$. See Theorem 4). That is, the PNU risk is dominated by the PN risk, implying that $\mathrm{Var}[\widehat{R}^{\eta}_{\mathrm{PNU}}(g)]\approx\mathrm{Var}[\widehat{R}_{\mathrm{PN}}(g)]$. Note that the class-prior is not the only factor for variance reduction; for example, if $\theta_{\mathrm{P}}=0.5$, $n_{\mathrm{P}}\gg n_{\mathrm{N}}$, and $\sigma_{\mathrm{P}}(g)\approx\sigma_{\mathrm{N}}(g)$, then $\gamma_{\text{N-PNPU}}\not\approx 0$ (because $\psi_{\mathrm{P}}\ll\psi_{\mathrm{N}}$) and the variance reduction will be large.

PNU Risk in Validation: As discussed in Section 4, the empirical PNU risk will be a reliable validation score due to its having smaller variance than the empirical PN risk. We show here that the empirical PNU risk is a promising alternative to a validation score.

To focus on the effect of validation scores only, we trained two classifiers by using the same risk, e.g, the empirical PN risk. We then tune the classifiers with the empirical PN and PNU risks denoted by $\widehat{g}^{\mathrm{PN}}_{\mathrm{PN}}$ and $\widehat{g}^{\mathrm{PNU}}_{\mathrm{PN}}$, respectively. The number of validation samples was the same as in the previous experiment.



(a) Phoneme ($d=5$) (b) Magic ($d=10$)

Figure 2. Average and standard error of the ratio between the misclassification rates of $\widehat{g}_{\mathrm{PN}}^{\mathrm{PNU}}$ and $\widehat{g}_{\mathrm{PN}}^{\mathrm{PN}}$ as a function of unlabeled samples over 1000 trials. In many cases, the ratio becomes less than 1, implying that the PNU risk is a promising alternative to the standard PN risk in validation if unlabeled data are available.

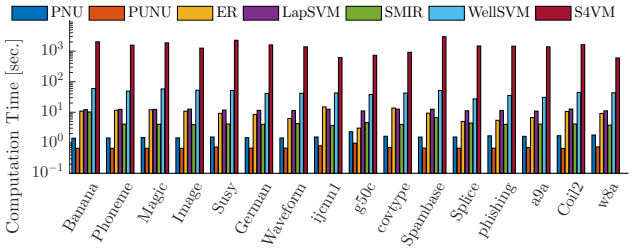


Figure 3. Average computation time over 50 trials for benchmark data sets when $n_{\mathrm{L}} = 50$.

Figure 2 shows the ratio between the misclassification rate of $\widehat{g}_{\mathrm{PN}}^{\mathrm{PNU}}$ and that of $\widehat{g}_{\mathrm{PN}}^{\mathrm{PN}}$. The number of unlabeled samples for validation increases from 10 to 300. With a rather small number of unlabeled samples, the ratio becomes less than 1, i.e., $\widehat{g}_{\mathrm{PN}}^{\mathrm{PNU}}$ achieves better performance than $\widehat{g}_{\mathrm{PN}}^{\mathrm{PN}}$. In particular, when $\theta_{\mathrm{P}} = 0.3$ and 0.7, $\widehat{g}_{\mathrm{PN}}^{\mathrm{PNU}}$ improved substantially; the large improvement tends to give the large variance reduction (cf. Figure 1). This result shows that the use of the empirical PNU risk for validation improved the classification performance given a relatively large size of unlabeled data.

5.2. Comparison with Existing Methods

Next, we numerically compare the proposed methods against existing semi-supervised classification methods.

Common Setup: We compare our methods against five conventional semi-supervised classification methods: *entropy regularization* (ER) (Grandvalet & Bengio, 2004), the *Laplacian support vector machine* (LapSVM) (Belkin et al., 2006; Melacci & Belkin, 2011), *squared-loss mutual information regularization* (SMIR) (Niu et al., 2013), the *weakly labeled support vector machine* (WellSVM) (Li et al., 2013), and the *safe semi-supervised support vector machine* (S4VM) (Li & Zhou, 2015).

Among the proposed methods, PNU classification and PUNU classification with the squared loss were tested.[3]

Table 1. Average and standard error of the misclassification rates of each method over 50 trials for benchmark data sets. Boldface numbers denote the best and comparable methods in terms of average misclassifications rate according to a t-test at a significance level of 5%. The bottom row gives the number of best/comparable cases of each method.

Data set	n_{L}	PNU	PUNU	ER	LapSVM	SMIR	WellSVM	S4VM
Banana	10	**30.1 (1.0)**	**32.1 (1.1)**	35.8 (1.0)	36.9 (1.0)	37.7 (1.1)	41.8 (0.6)	45.3 (1.0)
$d=2$	50	**19.0 (0.6)**	26.4 (1.2)	20.6 (0.7)	21.3 (0.7)	21.1 (1.0)	42.6 (0.5)	38.7 (0.9)
Phoneme	10	32.5 (0.8)	33.5 (1.0)	33.4 (1.2)	36.5 (1.5)	36.4 (1.2)	**28.4 (0.6)**	33.7 (1.4)
$d=5$	50	28.1 (0.5)	32.8 (0.9)	27.8 (0.6)	27.0 (0.8)	28.6 (1.0)	26.8 (0.4)	**25.1 (0.2)**
Magic	10	**31.7 (0.8)**	34.1 (0.9)	34.2 (1.1)	37.9 (1.3)	36.0 (1.2)	**30.1 (0.8)**	33.3 (0.9)
$d=10$	50	**29.9 (0.8)**	33.4 (0.9)	30.9 (0.5)	31.0 (0.9)	**30.8 (0.9)**	**28.8 (0.8)**	**29.2 (0.4)**
Image	10	**29.8 (0.9)**	**31.7 (0.8)**	33.7 (1.1)	36.6 (1.2)	36.7 (1.2)	34.7 (1.1)	35.9 (1.0)
$d=18$	50	**20.7 (0.8)**	26.6 (1.1)	**20.8 (0.8)**	**20.3 (1.0)**	**20.9 (0.9)**	27.2 (1.0)	23.2 (0.7)
Susy	10	**44.6 (0.6)**	**45.0 (0.6)**	47.7 (0.4)	48.2 (0.4)	**45.1 (0.7)**	48.0 (0.3)	46.8 (0.3)
$d=18$	50	**38.9 (0.6)**	41.5 (0.6)	**37.9 (0.7)**	43.1 (0.6)	43.9 (0.8)	43.8 (0.7)	42.1 (0.4)
German	10	**40.8 (0.9)**	**42.4 (0.7)**	43.6 (0.9)	45.9 (0.7)	46.2 (0.8)	**42.4 (0.8)**	**42.0 (0.7)**
$d=20$	50	**36.2 (0.8)**	39.0 (0.8)	38.9 (0.6)	40.6 (0.6)	38.4 (1.1)	38.5 (1.0)	**34.9 (0.5)**
Waveform	10	**17.4 (0.6)**	**18.0 (0.9)**	18.5 (0.6)	24.9 (1.4)	**18.0 (1.0)**	**16.7 (0.6)**	20.8 (0.8)
$d=21$	50	16.3 (0.6)	23.7 (1.2)	**14.2 (0.4)**	18.1 (0.8)	**15.4 (0.6)**	15.5 (0.5)	15.3 (0.3)
ijcnn1	10	43.6 (0.6)	**40.3 (1.0)**	49.7 (0.1)	49.2 (0.3)	44.0 (1.0)	45.9 (0.7)	49.3 (0.8)
$d=22$	50	**34.5 (0.8)**	37.1 (0.9)	**35.5 (0.8)**	**33.4 (1.1)**	49.4 (0.3)	46.2 (0.8)	48.6 (0.4)
g50c	10	11.4 (0.6)	12.5 (0.6)	23.3 (2.3)	39.8 (1.6)	21.9 (1.3)	**6.6 (0.4)**	27.0 (1.4)
$d=50$	50	12.5 (1.1)	10.1 (0.6)	8.7 (0.4)	22.5 (1.5)	10.6 (0.6)	**7.4 (0.4)**	12.1 (0.5)
covtype	10	**46.2 (0.4)**	**46.0 (0.4)**	**46.0 (0.5)**	47.1 (0.5)	47.9 (0.5)	**46.9 (0.6)**	**46.4 (0.4)**
$d=54$	50	**41.3 (0.5)**	42.3 (0.5)	**41.0 (0.4)**	**41.5 (0.5)**	46.2 (0.8)	43.6 (0.6)	**40.8 (0.4)**
Spambase	10	27.2 (0.9)	28.1 (1.1)	31.8 (1.4)	39.7 (1.4)	30.9 (1.3)	**23.8 (0.8)**	36.1 (1.5)
$d=57$	50	23.4 (1.0)	26.6 (1.0)	22.1 (0.7)	28.5 (1.3)	20.9 (0.5)	**19.1 (0.4)**	24.5 (0.9)
Splice	10	**38.3 (0.8)**	**39.3 (0.8)**	43.9 (0.8)	47.9 (0.5)	41.6 (0.7)	42.0 (1.0)	42.4 (0.6)
$d=60$	50	**30.6 (0.8)**	34.7 (0.9)	**30.9 (0.8)**	38.8 (1.0)	**30.6 (0.9)**	40.9 (0.8)	35.9 (0.7)
phishing	10	**24.2 (1.2)**	**25.8 (1.0)**	**27.3 (1.6)**	37.2 (1.6)	27.6 (1.6)	27.5 (1.4)	31.7 (1.3)
$d=68$	50	**15.8 (0.6)**	18.3 (0.8)	**15.4 (0.5)**	21.1 (1.3)	**14.7 (0.8)**	17.2 (0.7)	16.7 (0.8)
a9a	10	**31.4 (0.9)**	**31.3 (1.0)**	34.3 (1.2)	41.0 (1.1)	37.3 (1.3)	**33.1 (1.2)**	34.3 (1.2)
$d=83$	50	27.9 (0.6)	29.9 (0.8)	28.6 (0.7)	33.3 (1.0)	**26.9 (0.7)**	28.9 (0.8)	**26.2 (0.4)**
Coil2	10	**38.7 (0.8)**	**40.1 (0.8)**	42.8 (0.7)	43.9 (0.8)	43.2 (0.8)	**39.1 (0.9)**	44.0 (0.8)
$d=241$	50	**23.2 (0.6)**	30.5 (0.9)	**23.6 (0.9)**	**22.8 (0.9)**	25.1 (0.9)	**22.6 (0.8)**	25.4 (0.8)
w8a	10	**35.9 (0.9)**	**33.6 (1.0)**	41.6 (1.0)	46.6 (0.8)	39.4 (0.9)	42.1 (0.8)	43.0 (0.8)
$d=300$	50	**28.1 (0.7)**	**27.6 (0.6)**	**27.0 (0.9)**	38.7 (0.8)	**28.0 (0.9)**	33.7 (0.8)	35.2 (1.0)
#Best/Comp.		23	13	11	4	9	13	7

Data Sets: We used sixteen benchmark data sets taken from the *UCI Machine Learning Repository* (Lichman, 2013), the *Semi-Supervised Learning book* (Chapelle et al., 2006), the *LIBSVM* (Chang & Lin, 2011), the *ELENA Project*,[4] and a paper by Chapelle & Zien (2005).[5] Each feature was scaled to $[0, 1]$. Similarly to the setting in Section 5.1, we used the Gaussian kernel model for all methods. The training data is $\{\boldsymbol{x}_i\}_{i=1}^{n} = \mathcal{X}_{\mathrm{P}} \cup \mathcal{X}_{\mathrm{N}} \cup \mathcal{X}_{\mathrm{U}}$, where $n = n_{\mathrm{P}} + n_{\mathrm{N}} + n_{\mathrm{U}}$. We selected all hyper-parameters with validation samples of size 20 ($n_{\mathrm{P}}^{\mathrm{V}} = n_{\mathrm{N}}^{\mathrm{V}} = 10$). For training, we drew n_{L} labeled and $n_{\mathrm{U}} = 300$ unlabeled samples. The class-prior of labeled data was set at 0.7 and that of unlabeled samples was set at $\theta_{\mathrm{P}} = 0.5$ that were assumed to be known. In practice, the class-prior, θ_{P}, can be estimated

[3] In preliminary experiments, we tested other loss functions such as the ramp and logistic losses and concluded that the difference in loss functions did not provide noticeable difference.

[4] https://www.elen.ucl.ac.be/neural-nets/Research/Projects/ELENA/elena.htm

[5] http://olivier.chapelle.cc/lds/

Table 2. Average and standard error of misclassification rates over 30 trials for the Places 205 data set. Boldface numbers denote the best and comparable methods in terms of the average misclassification rate according to a t-test at a significance level of 5%.

Data set	n_{U}	θ_{P}	$\widehat{\theta}_{\mathrm{P}}$	PNU	ER	LapSVM	SMIR	WellSVM
Arts	1000	0.50	0.49 (0.01)	**27.4 (1.3)**	**26.6 (0.5)**	**26.1 (0.7)**	40.1 (3.9)	**27.5 (0.5)**
	5000	0.50	0.50 (0.01)	**24.8 (0.6)**	26.1 (0.5)	26.1 (0.4)	30.1 (1.6)	N/A
	10000	0.50	0.52 (0.01)	**25.6 (0.7)**	**25.4 (0.5)**	**25.5 (0.6)**	N/A	N/A
Deserts	1000	0.73	0.67 (0.01)	**13.0 (0.5)**	15.3 (0.6)	16.7 (0.8)	17.2 (0.8)	18.2 (0.7)
	5000	0.73	0.67 (0.01)	**13.4 (0.4)**	**13.3 (0.5)**	16.6 (0.6)	24.4 (0.6)	N/A
	10000	0.73	0.68 (0.01)	**13.3 (0.5)**	**13.7 (0.6)**	16.8 (0.8)	N/A	N/A
Fields	1000	0.65	0.57 (0.01)	**22.4 (1.0)**	26.2 (1.0)	26.6 (1.3)	28.2 (1.1)	26.6 (0.8)
	5000	0.65	0.57 (0.01)	**20.6 (0.5)**	22.6 (0.6)	24.7 (0.8)	29.6 (1.2)	N/A
	10000	0.65	0.57 (0.01)	**21.6 (0.6)**	**22.5 (0.6)**	25.0 (0.9)	N/A	N/A
Stadiums	1000	0.50	0.50 (0.01)	**11.4 (0.4)**	**11.5 (0.5)**	12.5 (0.5)	**17.4 (3.6)**	**11.7 (0.4)**
	5000	0.50	0.50 (0.01)	**11.0 (0.5)**	**10.9 (0.3)**	**11.1 (0.3)**	13.4 (0.7)	N/A
	10000	0.50	0.51 (0.00)	**10.7 (0.3)**	**10.9 (0.3)**	**11.2 (0.2)**	N/A	N/A
Platforms	1000	0.27	0.33 (0.01)	**21.8 (0.5)**	23.9 (0.6)	24.1 (0.5)	30.1 (2.3)	26.2 (0.8)
	5000	0.27	0.34 (0.01)	**23.3 (0.8)**	**24.4 (0.7)**	**24.9 (0.7)**	26.6 (0.3)	N/A
	10000	0.27	0.34 (0.01)	**21.4 (0.5)**	24.3 (0.6)	24.8 (0.5)	N/A	N/A
Temples	1000	0.55	0.51 (0.01)	**43.9 (0.7)**	**43.9 (0.6)**	**43.4 (0.6)**	50.7 (1.6)	**44.3 (0.5)**
	5000	0.55	0.54 (0.01)	**43.4 (0.9)**	**43.0 (0.6)**	**43.1 (1.0)**	**43.6 (0.7)**	N/A
	10000	0.55	0.50 (0.01)	**45.2 (0.8)**	**44.4 (0.8)**	**44.2 (0.7)**	N/A	N/A

by methods proposed, e.g., by Blanchard et al. (2010), Ramaswamy et al. (2016), or Kawakubo et al. (2016).

Table 1 lists the average and standard error of the misclassification rates over 50 trials and the number of best/comparable performances of each method in the bottom row. The superior performance of PNU classification over PUNU classification agrees well with the discussion in Section 4.3. With the g50c data set, which well satisfies the low-density separation principle, the WellSVM achieved the best performance. However, in the Banana data set, where the two classes are highly overlapped, the performance of WellSVM was worse than the other methods. In contrast, PNU classification achieved consistently better/comparable performance and its performance did not degenerate considerably across data sets. These results show that the idea of using PU classification in semi-supervised classification is promising.

Figure 3 plots the computation time, which shows that the fastest computation was achieved using the proposed methods with the square loss.

Image Classification: Finally, we used the *Places* 205 *data set* (Zhou et al., 2014), which contains 2.5 million images in 205 scene classes. We used a 4096-dimensional feature vector extracted from each image by *AlexNet* under the framework of *Caffe*,[6] which is available on the project website[7]. We chose two similar scenes to construct binary classification tasks (see the description of data sets in Appendix B.3). We drew 100 labeled and n_{U} unlabeled samples from each task; the class-prior of labeled and unlabeled data were respectively set at 0.5 and $\theta_{\mathrm{P}} = m_{\mathrm{P}}/(m_{\mathrm{P}} + m_{\mathrm{N}})$, where m_{P} and m_{N} respectively denote the number of total samples in positive and negative scenes. We used a linear classifier $g(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + w_0$, where $\boldsymbol{w}$ is the weight vector and w_0 is the offset (in the SMIR, the linear kernel model is used; see Niu et al. (2013) for details).

[6] http://caffe.berkeleyvision.org/

[7] http://places.csail.mit.edu/

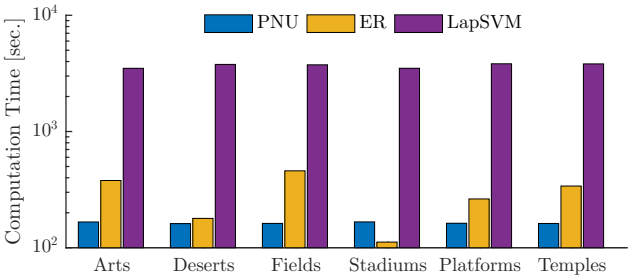


Figure 4. Average computation time over 30 trials for the Places 205 data set when $n_{\mathrm{U}} = 10000$.

We selected hyper-parameters in PNU classification by applying five-fold cross-validation with respect to $R^{\bar{\eta}}_{\mathrm{PNU}}(g)$ with the zero-one loss, where $\bar{\eta}$ was set at Eq.(10) or Eq.(11) with $\sigma_{\mathrm{P}}(g) = \sigma_{\mathrm{N}}(g)$. The class-prior $p(y = +1) = \theta_{\mathrm{P}}$ was estimated using the method based on energy distance minimization (Kawakubo et al., 2016).

Table 2 lists the average and standard error of the misclassification rates over 30 trials, where methods taking more than 2 hours were omitted and indicated as *N/A*. The results show that PNU classification was most effective. The average computation times are shown in Figure 4, revealing again that PNU classification was the fastest method.

6. Conclusions

In this paper, we proposed a novel semi-supervised classification approach based on classification from positive and unlabeled data. Unlike most of the conventional methods, our approach does not require strong assumptions on the data distribution such as the cluster assumption. We theoretically analyzed the variance of risk estimators and showed that unlabeled data help reduce the variance without the conventional distributional assumptions. We also established generalization error bounds and showed that the confidence term decreases with respect to the number of positive, negative, and unlabeled samples without the conventional distributional assumptions in the optimal parametric order. We experimentally analyzed the behavior of the proposed methods and demonstrated that one of the proposed methods, termed PNU classification, was most effective in terms of both classification accuracy and computational efficiency. It was recently pointed out that PU classification can behave undesirably for very flexible models and a modified PU risk has been proposed (Kiryo et al., 2017). Our future work is to develop a semi-supervised classification method based on the modified PU classification.

Acknowledgements

TS was supported by JSPS KAKENHI 15J09111. GN was supported by the JST CREST program and Microsoft Research Asia. MCdP and MS were supported by the JST CREST program.

References

Belkin, M., Niyogi, P., and Sindhwani, V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

Blanchard, G., Lee, G., and Scott, C. Semi-supervised novelty detection. *Journal of Machine Learning Research*, 11:2973–3009, 2010.

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

Chapelle, O. and Zien, A. Semi-supervised classification by low density separation. In *AISTATS*, pp. 57–64, 2005.

Chapelle, O., Schölkopf, B., and Zien, A. (eds.). *Semi-Supervised Learning*. MIT Press, 2006.

Collobert, R., Sinz, F., Weston, J., and Bottou, L. Trading convexity for scalability. In *ICML*, pp. 201–208, 2006.

Cozman, F. G., Cohen, I., and Cirelo, M. C. Semi-supervised learning of mixture models. In *ICML*, pp. 99–106, 2003.

du Plessis, M. C., Niu, G., and Sugiyama, M. Analysis of learning from positive and unlabeled data. In *NIPS*, pp. 703–711, 2014.

du Plessis, M. C., Niu, G., and Sugiyama, M. Convex formulation for learning from positive and unlabeled data. In *ICML*, volume 37, pp. 1386–1394, 2015.

Elkan, C. and Noto, K. Learning classifiers from only positive and unlabeled data. In *SIGKDD*, pp. 213–220, 2008.

Grandvalet, Y. and Bengio, Y. Semi-supervised learning by entropy minimization. In *NIPS*, pp. 529–536, 2004.

Jain, S., White, M., and Radivojac, P. Estimating the class prior and posterior from noisy positives and unlabeled data. In *NIPS*, 2016.

Kawakubo, H., du Plessis, M. C., and Sugiyama, M. Computationally efficient class-prior estimation under class balance change using energy distance. *IEICE Transactions on Information and Systems*, E99-D(1):176–186, 2016.

Kiryo, R., Niu, G., du Plessis, M. C., and Sugiyama, M. Positive-unlabeled learning with non-negative risk estimator. *arXiv preprint arXiv:1703.00593*, 2017.

Krijthe, J. H. and Loog, M. Robust semi-supervised least squares classification by implicit constraints. *Pattern Recognition*, 63:115–126, 2017.

Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, 1991.

Li, Y.-F. and Zhou, Z.-H. Towards making unlabeled data never hurt. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):175–188, 2015.

Li, Y.-F., Tsang, I. W., Kwok, J. T., and Zhou, Z.-H. Convex and scalable weakly labeled SVMs. *Journal of Machine Learning Research*, 14(1):2151–2188, 2013.

Lichman, M. UCI machine learning repository, 2013. URL `http://archive.ics.uci.edu/ml`.

Melacci, S. and Belkin, M. Laplacian support vector machines trained in the primal. *Journal of Machine Learning Research*, 12:1149–1184, 2011.

Mendelson, S. Lower bounds for the empirical minimization algorithm. *IEEE Transactions on Information Theory*, 54(8):3797–3803, 2008.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT Press, 2012.

Niu, G., Jitkrittum, W., Dai, B., Hachiya, H., and Sugiyama, M. Squared-loss mutual information regularization: A novel information-theoretic approach to semi-supervised learning. In *ICML*, volume 28, pp. 10–18, 2013.

Niu, G., du Plessis, M. C., Sakai, T., Ma, Y., and Sugiyama, M. Theoretical comparisons of positive-unlabeled learning against positive-negative learning. In *NIPS*, 2016.

Ramaswamy, H. G., Scott, C., and Tewari, A. Mixture proportion estimation via kernel embedding of distributions. In *ICML*, 2016.

Sokolovska, N., Cappé, O., and Yvon, F. The asymptotics of semi-supervised learning in discriminative probabilistic models. In *ICML*, pp. 984–991, 2008.

Vapnik, V. N. *Statistical Learning Theory*. John Wiley & Sons, 1998.

Vapnik, V.N. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, NY, USA, 1995.

Yuille, A. L. and Rangarajan, A. The concave-convex procedure (CCCP). In *NIPS*, pp. 1033–1040, 2002.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. Learning deep features for scene recognition using places database. In *NIPS*, pp. 487–495, 2014.