Classification From Positive and Biased Negative Data
With Skewed Labeled Posterior Probability

Shotaro Watanabe
s6021142@st.shiga-u.ac.jp
Graduate School of Data Science, Shiga University, Shiga 522-8522, Japan

Hidetoshi Matsui
hmatsui@biwako.shiga-u.ac.jp
Faculty of Data Science, Shiga University, Shiga 522-8522, Japan

The binary classification problem has a situation where only biased data are observed in one of the classes. In this letter, we propose a new method to approach the positive and biased negative (PbN) classification problem, which is a weakly supervised learning method to learn a binary classifier from positive data and negative data with biased observations. We incorporate a method to correct the negative influence due to a skewed confidence, which is represented by the posterior probability that the observed data are positive. This reduces the distortion of the posterior probability that the data are labeled, which is necessary for the empirical risk minimization of the PbN classification problem. We verified the effectiveness of the proposed method by synthetic and benchmark data experiments.

1 Introduction

The purpose of binary classification is to identify whether a sample belongs to a positive or a negative class, and this classification is applicable to various fields, such as life science, materials science, and marketing. In binary classification problems, conventional supervised classification learns a classifier using fully positive (P) data and fully negative (N) data. We call this problem PN classification. In contrast, weakly supervised machine learning has been widely studied in recent years, including positive and unlabeled (PU) classification, which learns a classifier using P data and unlabeled (U) data (Elkan & Noto, 2008; du Plessis et al., 2014, 2015; Kiryo et al., 2017), semisupervised classification (Chapelle et al., 2006; Sakai et al., 2017), and noisy-label learning (Natarajan et al., 2013; Shi et al., 2018; Zhang et al., 2019). Kato et al. (2019) approached the PU learning with selection bias for P data.

In this letter, we consider the situation where the majority of the N data are not observed. In this situation, it is difficult to collect complete N data,
but it may be easier to collect only a small set of biased N data. We call these data biased negative (bN) data. We explain an example for real data through the wireless indoor localization data set (Bhatt, 2017), available from the UCI Machine Learning Repository. The data set consists of strengths of seven WiFi signals at four rooms, and we want to identify a certain room (P data) from the other three rooms (N data). Although the strengths of signals differ for each room, those for one of the other three rooms may not be observed, or the observations for other three rooms are biased. Even for such a situation, we want to correctly classify the P room and the N room.

As applications for classifying P and bN data, Li et al. (2010) and Fei and Liu (2015) both tackled the problem in the context of text classification. Li et al. (2010) mentioned that bN data may have a negative impact on classification and therefore constructed a classifier using only P and U data. Fei and Liu (2015) considered the situation in which collecting unbiased U data would be difficult and trained a classifier using only P and bN data. However, their methods are specialized for text classification because it relies on the domain knowledge; it uses an effective similarity measure to evaluate the similarity between documents. In contrast to these two studies, Hsieh et al. (2019) proposed PUbN classification, a method for learning from P, U, and bN data without requiring specific domain knowledge for fitting arbitrary classifiers (ranging from linear to deep models). However, their method assumes the use of U data, and it is difficult to learn from only P and bN data. When U data are not available, it is difficult to estimate the distribution of the entire feature set, which makes it difficult to learn the classifier.

We extend PUbN classification and propose PbN classification, which trains a classifier using only P and bN data even when U data and domain knowledge are not available. One of the critical issues of PbN classification is that we cannot obtain accurately the posterior probability that each instance is observed. As an approach to this problem, we use the idea of positive confidence (Pconf) classification (Ishida et al., 2018). Pconf classification is a learning method in which a binary classifier is learned from only P data with a confidence that represents the positive posterior probability, without N or U data. Pconf classification gives a skew due to the fact that the confidence cannot be obtained explicitly. Here we follow Shinoda et al. (2020) and refer to the deformation of confidence and other parameters by bias as skew. We incorporate a measure that corrects the negative influence of the skew of the confidence of Pconf classification in PbN classification. We call this method the adjusted PbN classification and that without the adjustment, the naive PbN classification. We verified the effectiveness of the proposed method by applying it to the analysis of synthetic data and benchmark data.

This letter is organized as follows. In section 2, we review the problem setup for PN, PUbN, and Pconf classification. In section 3, we first present the problem setup of PbN classification and then show how to adjust the
posterior probability that each instance is the skewed observation. In section 4, we compare the performance of adjusted PbN classification with that of PN and naive PbN classification using synthetic data and benchmark data and then examine how we verified the effectiveness of the proposed method. We present our conclusions in section 5.

2 Problem Formulation and Existing Method

In this section, we review the problem setup for PN, PuBn, and Pconf classification.

2.1 PN Classification. Let \( x \in \mathbb{R}^d \), \( y \in \{ +1, -1 \} \) be a \( d \)-dimensional feature vector and a label, respectively, and suppose they follow an unknown probability density function \( p(x, y) \). In addition, let \( p_P(x) = p(x|y = +1) \) and \( p_N(x) = p(x|y = -1) \) be densities of features with positive and negative labels, respectively, and \( \pi = p(y = +1) \) be a class prior. Furthermore, let \( g : \mathbb{R}^d \rightarrow \mathbb{R} \) be a binary classifier that returns positive or negative values that represent labels and \( \ell : \mathbb{R} \rightarrow \mathbb{R}_+ \) be a loss function that returns positive value if the classifier gives the wrong result and zero if the classifier gives the correct result. We denote \( \mathbb{E}_{x \sim p_P(x)}[\ell(g(x))] \) and \( \mathbb{E}_{x \sim p_N(x)}[\ell(-g(x))] \) as an expected false negative rate and an expected false positive rate, respectively. Then the PN classification risk is expressed as

\[
R(g) = \pi R^+_P(g) + (1 - \pi) R^-_N(g).
\]

We determine the classifier \( g \) by minimizing the classification risk, equation 2.1, under the situation that \( g \) and \( \ell \) are specified. In practice, the minimizer of that classification risk cannot be calculated directly because it includes the expectation for the loss function. Instead, if the P data \( X_P = \{x_i^P\}_{i=1}^{n_P} \) and the N data \( X_N = \{x_i^N\}_{i=1}^{n_N} \) are observed independently from \( p_P(x) \) and \( p_N(x) \), respectively, the classification risk, equation 2.1, can be approximated as

\[
\hat{R}(g) = \pi \hat{R}^+_P(g) + (1 - \pi) \hat{R}^-_N(g),
\]

where \( \hat{R}^+_P(g) = \frac{1}{n_P} \sum_{i=1}^{n_P} \ell(g(x_i^P)) \) and \( \hat{R}^-_N(g) = \frac{1}{n_N} \sum_{i=1}^{n_N} \ell(-g(x_i^N)) \) are empirical losses for false negative and false positive rates, respectively.

2.2 PuBn Classification. PuBn classification is an approach for binary classification from P, U, and bN data. To obtain the classifier, Hsieh et al. (2019) introduce a latent variable \( s \) that returns +1 for the observed P, bN, and U data and −1 for unobserved data. Let \( \rho = p(y = -1, s = +1) \), \( p_{bN}(x) = p(x|y = -1, s = +1) \), and \( p_{s=-1}(x) = p(x|s = -1) \). Furthermore, let
\[ R_{bN}(g) = E_{x \sim p_{bN}(x)}[\ell(-g(x))] \] and \[ R_{s=-1}(g) = E_{x \sim p_{s=-1}(x)}[\ell(-g(x))] \]. Then the PUbN classification risk is expressed as
\[ R(g) = \pi R_{P}^+(g) + \rho R_{bN}^-(g) + (1 - \pi - \rho) R_{s=-1}(g). \] (2.2)

Hsieh et al. (2019) decompose \( R_{s=-1}(g) \) in equation 2.2 into three terms to focus on how important U data are against P and bN data.

Let us denote \( X_{bN} = \{x_{i}^{bN}\}_{i=1}^{n_{bN}} \) bN data; then the empirical version of equation 2.2 can be approximated as
\[ \hat{R}(g) = \pi \hat{R}_{P}^+(g) + \rho \hat{R}_{bN}^-(g) + (1 - \pi - \rho) \hat{R}_{s=-1}(g), \]
where \( \hat{R}_{bN}^-(g) \) and \( \hat{R}_{s=-1}(g) \) are empirical versions of \( R_{bN}(g) \) and \( R_{s=-1}(g) \), respectively.

2.3 Pconf Classification. Pconf classification is an approach for binary classification from only P data under the condition that neither N nor U data are observed. In this situation, it is difficult to evaluate the expected false positive rate \( R_{N}^- \) in the classification risk. Ishida et al. (2018) expressed the Pconf classification risk as
\[ R(g) = \pi \left[ R_{P}^+(g) + R_{P}^- \left( \frac{1 - r}{r} g \right) \right], \] (2.3)
where \( R_{P}^+(g) = E_{x \sim p_{P}(x)}[\ell(-g(x))] \) and \( r = p(y = +1|x) \) confirms that the observed data are P. In Pconf classification, the classifier can be trained from only the positive data \( X_{P} = \{x_{i}^{P}\}_{i=1}^{n_{P}} \), and then the empirical classification risk can be approximated as
\[ \hat{R}(g) = \pi \left[ \hat{R}_{P}^+(g) + \hat{R}_{P}^- \left( \frac{1 - r}{r} g \right) \right], \]
where \( \hat{R}_{P}^- \) is an empirical version of \( R_{P}^- \).

To calculate the confidence \( r \), it is necessary to evaluate the marginal density \( p(x) \) or \( p(x|y = -1) \) directly or to apply the logistic regression using fully P and N data as a classifier. However, it is difficult to consider the situation where they are known in real-world applications. Therefore, we need to estimate the confidence \( r \) with the help of the domain knowledge by experts, but this may cause bias and skew of the confidence. Shinoda et al. (2020) assumed a false negative rate of the classification as prior knowledge and modified the confidence by minimizing the squared difference between the false negative rate and the empirical false negative rate.
3 Proposed Method

In this section, we first propose a classification risk for PbN classification and then a method for correcting the skew of the posterior probability that each instance is observed in PbN classification.

3.1 PbN Classification. Unlike PN classification, PbN classification considers the situation where only some biased negative data rather than all negative data are observed. To solve this problem, we introduce a latent variable \( s \) that returns +1 for the observed P and bN data and −1 for the unobserved N data. The class prior probability \( \pi = P(y = +1) \) is assumed to be known in this letter. A similar assumption is considered in Hsieh et al. (2019). In addition, Chen et al. (2020) and Kiryo et al. (2017) approached the PU classification by assuming that \( \pi \) is known. Du Plessis and Sugiyama (2014), du Plessis et al. (2016), and Yao et al. (2021) considered the method for estimating \( \pi \) for PU classification. Let \( \rho = p(y = -1, s = +1) \), \( p_{bN}(x) = p(x|y = -1, s = +1) \), and \( p_{s=+1}(x) = p(x|s = +1) \). Furthermore, let \( \sigma = p(s = +1|x) \) be a probability that the data are observed. Since \( \pi \) is assumed to be known, then \( \rho \) and \( p(s = +1) \) can be estimated respectively by

\[
\hat{\rho} = \frac{n_{bN}}{n_p} \pi, \quad \hat{p}(s = +1) = \frac{n_p + n_{bN}}{n_p} \pi.
\]

Let \( R_{bN}^- (g) = E_{x \sim p_{bN}(x)}[\ell(-g(x))] \) and \( R_{s=+1}^- (g) = E_{x \sim p_{s=+1}(x)}[\ell(-g(x))] \). Then if \( \sigma \neq 0 \), the PbN classification risk is expressed as

\[
R(g) = \pi R_p^+(g) + \rho R_{bN}^-(g) + (\pi + \rho) R_{s=+1}^- \left( \frac{1 - \sigma}{\sigma} g \right). \tag{3.1}
\]

The derivation of equation 3.1 is given in appendix A.

Suppose that \( \mathcal{X}_{bN} = \{x_{i_{bN}}^{n_{bN}}\}_{i_{bN}=1}^{n_{bN}} \), \( \mathcal{X}_{s=+1} = \{x_{i_{s=+1}}^{n_p+n_{bN}}\}_{i_{s=+1}=1}^{n_p+n_{bN}} \) are samples observed independently from \( p_{bN}(x) \) and \( p_{s=+1}(x) \), respectively. Then the classification risk equation 3.1, can be approximated as

\[
\hat{R}(g) = \pi \hat{R}_p^+(g) + \rho \hat{R}_{bN}^-(g) + (\pi + \rho) \hat{R}_{s=+1}^- \left( \frac{1 - \sigma}{\sigma} g \right),
\]

where \( \hat{R}_{bN}^- (g) = \frac{1}{n_{bN}} \sum_{i_{bN}=1}^{n_{bN}} \ell(-g(x_{i_{bN}})) \) and \( \hat{R}_{s=+1}^- = \frac{1}{n_p+n_{bN}} \sum_{i_{s=+1}=1}^{n_p+n_{bN}} \ell(-g(x_{i_{s=+1}})) \). In this case, as in the case of Pconf classification, the probability \( \sigma \) cannot be obtained explicitly because it depends on unknown \( p(x) \). Instead, if we use bN data, we can obtain a pseudo \( p(x) \) as \( p_{\text{bias}}(x) = \pi p(x|y = +1) + (1 - \pi) p(x|y = -1, s = +1) \). Then we can obtain a skewed \( \sigma \) directly as follows:

\[
\tilde{\sigma} = \frac{p(s = +1)p(x|s = +1)}{p_{\text{bias}}(x)}. \tag{3.2}
\]
In Pconf classification, we need to estimate the confidence $r$ in equation 2.3 using domain knowledge. On the other hand, we can estimate $\sigma$ in equation 3.1 by estimating the probability densities $p(x|s = +1)$, $p(x|y = +1)$ and $p(x|y = -1, s = +1)$. Note that, however, unlike PUbN classification (Hsieh et al., 2019), it may be difficult to provide learning with the kernel density estimation on high-dimensional data. PUbN classification treats P data and bN data as P data and then solves the PU classification problem by applying a deep neural network with the sigmoid function for the output layer to estimate the probability that the data are observed. However, it is difficult to apply it in our method because we cannot use the U data. In benchmark data experiments given in section 4.3, we apply the kernel density estimation to estimate the above densities.

### 3.2 Adjusted PbN Classification.

The probability $\tilde{\sigma}$ given in equation 3.2 is skewed due to the bias of the bN data, which has a negative influence on prediction accuracy. To correct the skew of the probability $\tilde{\sigma}$, we incorporate a measure to correct the skew of the confidence level in Pconf classification proposed by Shinoda et al. (2020). Using this idea, we consider the following empirical classification risk with a hyperparameter $k \in (0, \infty)$:

$$\hat{R}_{\text{bias}}(g) = \pi \hat{R}_P^+(g) + \rho \hat{R}_{\text{bN}}^-(g) + (\pi + \rho) \hat{R}_{s=+1}^- \left( \frac{1 - \hat{\sigma}_k}{\hat{\sigma}_k} g \right).$$

If there were a validation set containing P and N data, the hyperparameter $k$ could be selected by cross-validation. However, it is difficult to use cross-validation because there are only P and bN data. Therefore, we assume that the following false negative rate for classification is known as prior knowledge:

$$\phi = \int_{\{x: g(x) < 0\}} p(x|y = +1) dx.$$

Under this assumption, we obtain the optimal hyperparameter $k^*$ by minimizing the error between the known false negative rate $\phi$ and the empirical false negative rate:

$$k^* = \arg \min_k \left\{ \frac{1}{n_p} \sum_{i=1}^{n_p} \ell_{01}(g(x_i)) - \phi \right\}^2,$$

where $\ell_{01}(z) = (1 - \text{sign}(z))/2$ is the 0-1 loss. The selected $k^*$ is used to express the adjusted risk for the PbN classification as

$$\hat{R}_{\text{bias}}(g) = \pi \hat{R}_P^+(g) + \rho \hat{R}_{\text{bN}}^-(g) + (\pi + \rho) \hat{R}_{s=+1}^- \left( \frac{1 - \hat{\sigma}_{k^*}}{\hat{\sigma}_{k^*}} g \right).$$
In practice, the assumption that the false negative rate \( \phi \) is known is not realistic. However, as shown in the numerical experiments in section 4.1.3, the classification accuracy of the proposed method is not sensitive to the choice of \( \phi \).

4 Experiments

This section describes how we validated the performance of our proposed method with experiments using two settings of synthetic data and two benchmark data sets.

4.1 Synthetic Data Experiment 1.

4.1.1 Setup. In the first synthetic data experiment, we generated a two-dimensional training data set, a validation data set, a test data set, and a data set for false negative rate estimation from \( P \) and \( N \) data generated independently. The training data set contains 500 \( P \) samples and 100 \( bN \) samples, and the validation data set contains 500 \( P \) samples. Furthermore, both the test data set and the data set for false negative rate estimation contain 500 \( P \) samples and 500 \( N \) samples. We compared the classification accuracy of the adjusted \( PbN \) classification with that of the naive \( PbN \) classification and the ordinary \( PN \) classification. For the naive \( PbN \) classification and \( PN \) classification, the training data set and the validation data set were combined as the training data set.

For all methods, we used the linear model \( g(x) = a^\top x + \beta \) for the binary classifier and the logistic loss \( \ell_L(z) = \log(1 + e^{-z}) \) for the loss function. Furthermore, the stochastic gradient descent was applied for the optimization to construct the classifier. The candidate of the hyperparameters \( k \) was selected from \( \{0.3, 0.5, 0.7, 1.0, 1.5, 2.0, 4.0\} \). The \( \sigma \) is calculated as equation 3.1, where \( p(x|s = +1), p(x|y = +1), \) and \( p(x|y = -1, s = +1) \) are supposed to follow a gaussian distribution, and their means and variance covariance matrices are estimated by sample means and sample variance covariance matrices for them, respectively. Note that \( \sigma \) less than 0.01 was rounded up to 0.01 for optimization stability.

The \( P \) data were generated from a gaussian distribution with a mean vector \( [0, 0]^\top \) and an identity covariance matrix. In contrast, the \( N \) data were assumed to be generated from a mixture of four gaussian distributions whose means were different from each other and whose variance covariance matrices were all identity matrices. We considered two patterns according to the degree of overlap for the \( P \) and \( N \) data. For the pattern with the smaller overlap, the \( N \) data were distributed by the mixture of gaussian distributions with mean vectors \( [1.0, 1.0]^\top, [1.5, 1.5]^\top, [2.0, 2.0]^\top, \) and \( [2.5, 2.5]^\top \); conversely, for the pattern with the larger overlap, the \( N \) data were distributed by a mixture of distributions with mean vectors \( [2.0, 2.0]^\top, [3.0, 3.0]^\top, [4.0, 4.0]^\top, \) and \( [5.0, 5.0]^\top \). By applying the proposed method to
the analysis of these data sets, we examined the difference in accuracy depending on the degree of overlap of the classes. We also considered two cases depending on which data were actually observed in a biased manner from the N data; specifically, the N data were observed from only one of the four components, or the N data were observed partially from each of four components with probabilities $[0.25, 0.25, 0.25, 0.25], [0.40, 0.10, 0.35, 0.15], [0.15, 0.40, 0.10, 0.35]$, and $[0.35, 0.15, 0.40, 0.10]$.

To summarize the above, we conducted synthetic data experiment 1 under the following four situations:

- **Situation 1.** The degree of the class overlap is large, and one of the four components of the N data is observed.
- **Situation 2.** The degree of the class overlap is small, and one of the four components of the N data is observed.
- **Situation 3.** The degree of the class overlap is large, and the observed N data are partially observed from each of the four components of the N data.
- **Situation 4.** The degree of the class overlap is small, and the observed N data are partially observed from each of the four components of the N data.

In each of these settings, we calculated the classification accuracy by naive PbN classification, adjusted PbN classification, and PN classification.

### 4.1.2 Experiments with Four Situations

We report the means and standard deviations of the classification accuracy for the 10 trials of situations 1 and 2 in Table 1, where $\mu_{bN}$ is the mean of the bN data, and $\phi$ is the mean and
Standard deviation of the estimated false negative rate over 10 trials. A.PbN and N.PbN represent the adjusted PbN classification and the naive PbN classification, respectively. The adjusted PbN classification is significantly better than the PN classification when the class overlap is larger and significantly better than the naive PbN classification when the class overlap is smaller.

Next, the means and standard deviations of the classification accuracy in 10 trials with situations 3 and 4 are shown in Table 2. The adjusted PbN classification is significantly better than the PN classification when the class overlap is larger and significantly better than the naive PbN classification when the class overlap is smaller. These results show that the classification accuracy of the proposed method is stable regardless of the degree of the class overlap and the shape of the bN data.

The decision boundaries in situations 1 to 4 are shown in Figures 1a and 1d, respectively. The blue, green, and orange points represent the P data, observed bN data, and unobserved N data, respectively. The decision boundaries for naive PbN classification, adjusted PbN classification, and PN classification are shown in yellow, red, and purple, respectively. From these figures, we can see that the naive PbN classification is strongly influenced by the P data and the PN classification is strongly influenced by the bN data compared to the adjusted PbN classification.

4.1.3 Effect of Estimation Error of $\phi$. In the experiments described in section 4.1.2, we assumed that the false negative rate $\phi$ can be estimated accurately, but in practice, this assumption does not necessarily hold in practical situations. To examine how much the estimation error of $\phi$ affects the

Table 2: Means and Standard Deviations of the Classification Accuracies for Situations 3 and 4 in Synthetic Data Experiment 1.

| Situation 3 | Probability | A.PbN     | N.PbN     | PN        | $\phi$   |
|------------|-------------|-----------|-----------|-----------|----------|
| [0.25, 0.25, 0.25, 0.25] | 86.18 ± 1.05 | 85.92 ± 0.85 | 82.50 ± 1.01 | 10.58 ± 1.34 |
| [0.40, 0.10, 0.35, 0.15] | 86.06 ± 0.99 | 86.08 ± 1.11 | 82.79 ± 1.35 |
| [0.15, 0.40, 0.10, 0.35] | 86.64 ± 1.05 | 86.24 ± 1.30 | 82.49 ± 1.21 |
| [0.35, 0.15, 0.40, 0.10] | 86.04 ± 1.03 | 85.94 ± 1.11 | 83.26 ± 1.50 |

| Situation 4 | Probability | A.PbN     | N.PbN     | PN        | $\phi$   |
|------------|-------------|-----------|-----------|-----------|----------|
| [0.25, 0.25, 0.25, 0.25] | 96.21 ± 0.79 | 86.93 ± 1.45 | 95.19 ± 0.44 | 2.52 ± 0.78 |
| [0.40, 0.10, 0.35, 0.15] | 96.38 ± 0.38 | 88.80 ± 1.12 | 95.74 ± 0.55 |
| [0.15, 0.40, 0.10, 0.35] | 95.46 ± 0.55 | 86.89 ± 1.96 | 94.13 ± 0.72 |
| [0.35, 0.15, 0.40, 0.10] | 95.82 ± 0.65 | 89.66 ± 1.83 | 95.06 ± 0.52 |

Note: The best methods based on the 5% $t$-test and equivalent methods are in bold.
accuracy of the adjusted PbN classification, we conducted some experiments with deviations for $\hat{\phi}$.

We applied the adjusted PbN classification in the same setting as situations 1 and 2 described in section 4.1.2, but with the correctly estimated values of $\hat{\phi}$ multiplied by $c \in \{0.5, 0.7, 1.3, 1.5\}$, which causes a deviation
Table 3: Means and Standard Deviations of the Classification Accuracies of the Adjusted PbN Classification Using \( \hat{\phi} \) with the Estimation Error for Situations 1 and 2 in Synthetic Data Experiment 1.

| Situation 1 | \( \mu_{bN} \) | \( c = 0.5 \) | \( c = 0.7 \) | \( c = 1.3 \) | \( c = 1.5 \) |
|-------------|----------------|--------------|--------------|--------------|--------------|
| [1.0, 1.0] \( ^T \) | 79.23 ± 2.65 | 83.24 ± 2.19 | 85.53 ± 0.89 | 85.66 ± 0.78 |
| [1.5, 1.5] \( ^T \) | 81.44 ± 1.67 | 83.68 ± 1.29 | 85.21 ± 0.52 | 85.14 ± 0.47 |
| [2.0, 2.0] \( ^T \) | 81.02 ± 2.32 | 84.04 ± 1.57 | 85.80 ± 1.74 | 85.00 ± 1.82 |
| [2.5, 2.5] \( ^T \) | 79.74 ± 2.76 | 83.14 ± 2.27 | 84.27 ± 1.53 | 82.74 ± 1.69 |

| Situation 2 | \( \mu_{bN} \) | \( c = 0.5 \) | \( c = 0.7 \) | \( c = 1.3 \) | \( c = 1.5 \) |
|-------------|----------------|--------------|--------------|--------------|--------------|
| [2.0, 2.0] \( ^T \) | 89.67 ± 1.79 | 94.32 ± 1.04 | 96.24 ± 0.53 | 96.24 ± 0.53 |
| [3.0, 3.0] \( ^T \) | 90.61 ± 1.38 | 95.14 ± 1.48 | 95.76 ± 0.45 | 95.76 ± 0.45 |
| [4.0, 4.0] \( ^T \) | 91.60 ± 3.95 | 94.74 ± 1.60 | 95.44 ± 0.64 | 95.30 ± 0.77 |
| [5.0, 5.0] \( ^T \) | 89.91 ± 2.71 | 95.67 ± 0.70 | 95.44 ± 0.48 | 95.44 ± 0.48 |

Note: Based on a 5% t-test, results equivalent to those for \( c = 1.0 \) are shown in bold. \( \mu_{bN} \) is the mean of the bN data.

to the estimation. The means and standard deviations of the classification accuracies in 10 trials are shown in Table 3. When the degree of class overlap is larger, the effect of estimation error is larger for the case of \( c = 0.5, 0.7 \), whereas when the degree of class overlap is smaller, the effect of the estimation error is negligible unless \( \hat{\phi} \) is underestimated. This result indicates that the proposed method is effective even in practical situations where \( \phi \) cannot be estimated accurately, as long as \( \phi \) is estimated larger than the true value or the degree of class overlap is small.

### 4.2 Synthetic Data Experiment 2

In section 4.1, we examined simulation studies for two-dimensional data to verify the effectiveness of the proposed method numerically and visually. This section examines how the accuracy of the proposed PbN classification is affected for high-dimensional data as the second synthetic data experiment.

In this experiment, we generated the P data from a gaussian distribution with a \( d \)-dimensional mean vector \( [0, \ldots, 0] ^T \) and an identity covariance matrix. The N data are generated from a mixture of four gaussian distributions with different mean vectors and identity covariance matrices. We set the mean vectors for N data as follows:

\[
\begin{align*}
\mu_{bN}^1 &= (3/\sqrt{d}) 1_d, \\
\mu_{bN}^2 &= \mu_{bN}^1 + u_1, \\
\mu_{bN}^3 &= \mu_{bN}^2 + u_2, \\
\mu_{bN}^4 &= \mu_{bN}^3 + u_3,
\end{align*}
\]
Table 4: Means and Standard Deviations of the Classification Accuracies in Synthetic Data Experiment 2.

| \(\mu_{bN}^d\) | A.PbN | N.PbN | PN | \(\hat{\phi}\) |
|----------------|--------|--------|----|----------|
| \(\mu_{bN}^1\) | 95.64 ± 0.90 | 95.22 ± 0.88 | 95.67 ± 1.00 | 2.58 ± 0.98 |
| \(\mu_{bN}^2\) | 90.86 ± 1.08 | 91.10 ± 0.69 | 89.83 ± 0.58 |
| \(\mu_{bN}^3\) | 87.30 ± 2.61 | 86.78 ± 0.95 | 84.92 ± 1.10 |
| \(\mu_{bN}^4\) | 80.80 ± 1.26 | 81.23 ± 1.14 | 79.70 ± 0.81 |

\(d = 50\)

| \(\mu_{bN}^d\) | A.PbN | N.PbN | PN | \(\hat{\phi}\) |
|----------------|--------|--------|----|----------|
| \(\mu_{bN}^1\) | 89.43 ± 2.68 | 89.10 ± 3.46 | 89.40 ± 2.79 | 3.61 ± 0.97 |
| \(\mu_{bN}^2\) | 89.73 ± 0.96 | 89.24 ± 0.26 | 88.98 ± 0.13 |
| \(\mu_{bN}^3\) | 88.48 ± 0.81 | 88.17 ± 0.47 | 87.20 ± 0.83 |
| \(\mu_{bN}^4\) | 87.39 ± 2.05 | 85.76 ± 1.19 | 84.00 ± 1.34 |

\(d = 100\)

| \(\mu_{bN}^d\) | A.PbN | N.PbN | PN | \(\hat{\phi}\) |
|----------------|--------|--------|----|----------|
| \(\mu_{bN}^1\) | 88.69 ± 3.65 | 88.90 ± 3.23 | 88.74 ± 3.30 | 6.25 ± 1.33 |
| \(\mu_{bN}^2\) | 88.52 ± 0.82 | 88.71 ± 0.93 | 87.79 ± 1.33 |
| \(\mu_{bN}^3\) | 87.54 ± 1.24 | 86.83 ± 1.05 | 85.97 ± 1.19 |
| \(\mu_{bN}^4\) | 84.37 ± 3.28 | 79.33 ± 1.30 | 77.63 ± 1.06 |

Note: Based on a 5% t-test, results-equivalent methods are in bold.

where \(1_d\) is a \(d\)-dimensional vector whose elements are all 1 and \(u_k = (u_{k1}, \ldots, u_{kd})^\top (k = 1, 2, 3)\) are \(d\)-dimensional vectors whose elements follow uniform distributions \(U(-10/\sqrt{d}, 10/\sqrt{d})\). In section 4.1, we estimated \(\sigma\) by assuming that the probabilities \(p(x|s = +1), p(x|y = +1), p(x|y = -1, s = +1)\) that are needed for estimating \(\sigma\) follow the gaussian distribution, whereas it is difficult to apply this method for high-dimensional data since their probabilities are shrunken toward zeroes. Therefore, we apply the kernel density estimation with a gaussian kernel with bandwidth 0.1 for estimating these probabilities. In this simulation, one of the four gaussian components of the N data is supposed to be observed, similar to situations 1 and 2 in section 4.1, and we performed the simulation replacing the observed component for the N data. Other settings such as the model \(g(x)\), loss function \(\ell(z)\) are the same as those for section 4.1.

The results for the 10 trials for dimensions \(d = 10, 50,\) and 100 are given in Table 4, where \(\mu_{bN}\) is the mean of the observed bN data. The proposed method gives superior accuracy when the observed bN data are more distant from P data than unobserved N data for all three settings of dimensions. In addition, the proposed method gives competitive results when the observed bN data are closer than unobserved N data. These results show that the proposed method gives stable prediction accuracy for high-dimensional data.
4.3 Benchmark Data Experiments.

4.3.1 Wireless Indoor Localization Data Set. We used the wireless indoor localization data set available from the UCI Machine Learning Repository. This data set was collected for an experiment to locate indoor positions using wireless LAN signal strength. The data set consists of four rooms numbered 1 to 4, each with a sample size of 500 consisting of seven-dimensional feature vectors. Here, we created a data set for binary classification with room 2 as a positive class and the other rooms as a negative class. The bN data were obtained only from rooms 1, 3, and 4, respectively. We also obtained N data randomly from each room, which were not biased. This data set was then divided into a training data set, a validation data set, a test data set, and a data set for false negative rate estimation. The training data set contained 200 P samples and 100 bN samples, the validation data set contained 100 P samples, and the test data set and the data set for false negative rate estimation respectively contained 100 P samples and 300 N samples.

Since the numbers of P and N samples are unbalanced, we compared the \( F \)-scores as well as the classification accuracies of the adjusted PbN classification with that of the naive PbN classification and the ordinary PN classification. For all methods, the linear model \( g(x) = a^T x + \beta \) was used as the binary classifier, the logistic loss \( \ell_L(z) = \log(1 + e^{-z}) \) was used as the loss function, and the stochastic gradient descent method was applied for optimization to construct the classifier. For the estimation of \( p(x|s = +1) \), \( p(x|y = +1) \), and \( p(x|y = -1, s = +1) \), which are necessary for the estimation of the probability \( \sigma \) that the data are observed, we used the kernel density estimation with gaussian kernels with a bandwidth of 0.1. The candidates of the hyperparameter \( k \) in equation 3.3 were \{0.5, 0.7, 0.9, 1.0, 1.2, 1.5, 2.0\} and estimated \( \tilde{\sigma} \) less than 0.01 was rounded up to 0.01 for optimization stability.

The means and standard deviations of the classification accuracies and \( F \)-scores in 100 trials are shown in Table 5. The \( \hat{\phi} \) is the mean and standard deviation of the estimated false negative rate over 100 trials. The proposed adjusted PbN classification performed as well as or better than the existing methods in all cases.

4.3.2 MNIST Data Set. The MNIST data set (LeCun & Cortes, 1998) consists of 70,000 grayscale images with handwritten numbers from 0 to 9. Each image has a pixel size of 28 \( \times \) 28 (784 dimensions) and is associated with a label of 0 to 9. In this experiment, we treated the even numbers as a positive class and the odd numbers as a negative class to do binary classifications. Then we divided the data set into a training, validation, and test data set and a data set for estimating false negative rates. Since a large number of samples makes the computation of the proposed method difficult, we used one-tenth of the all samples of the data set. Then we have 2000 P samples and 400 bN samples as the training set, 500 P samples as the validation set,
Table 5: Means and Standard Deviations of the Percentage of Classification Accuracies and $F$-Scores for the Wireless Indoor Localization Data.

| Y | A.PbN | N.PbN | PN | $\phi$ |
|---|---|---|---|---|
| **Accuracy** | | | | |
| Room 1 | 86.28 ± 4.76 | 83.76 ± 3.42 | 82.45 ± 3.09 | 3.34 ± 2.17 |
| Room 3 | 97.87 ± 1.72 | 96.60 ± 4.99 | 94.61 ± 5.61 | |
| Room 4 | 89.96 ± 6.39 | 87.83 ± 6.76 | 77.18 ± 8.23 | |
| Random | 98.05 ± 0.87 | 97.97 ± 0.98 | 96.25 ± 3.32 | |
| **F-score ($\times 10^2$)** | | | | |
| Room 1 | 78.88 ± 5.98 | 75.67 ± 3.85 | 74.16 ± 3.33 | |
| Room 3 | 95.85 ± 2.61 | 90.97 ± 18.60 | 86.40 ± 17.84 | |
| Room 4 | 84.05 ± 7.62 | 81.22 ± 7.59 | 69.58 ± 8.20 | |
| Random | 96.13 ± 1.70 | 95.98 ± 1.90 | 93.21 ± 4.92 | |

Note: The best method based on the 5% $t$-test and equivalent methods are in bold.

Means and standard deviations of the estimated false negative rates in 10 trials are shown in Table 6. The proposed method performs as well as or better than the existing and naive PbN methods in all cases, but failed to show enough advantages over PN classification compared to the result of the experiments of the wireless data set given in section 4.3.1.

5 Conclusion

In this letter, we have proposed a new approach for the PbN classification problem under the condition that only positive data and some biased negative data are observed. In the proposed method, the risk cannot be estimated properly due to the skew of the probability that indicates whether each instance is actually observed. The skew is reduced by minimizing the squared
Table 6: Means and Standard Deviations of the Percentage of Classification Accuracies for the MNIST Data.

| Probability          | A.PbN      | B.PbN      | PN         | $\phi$     |
|----------------------|------------|------------|------------|------------|
| [0.40, 0.30, 0.20, 0.05, 0.05]$^T$ | 76.78 ± 3.77 | 69.53 ± 5.77 | 75.25 ± 2.92 | 6.47 ± 1.38 |
| [0.05, 0.40, 0.30, 0.20, 0.05]$^T$ | 77.41 ± 4.24 | 71.08 ± 3.50 | 74.71 ± 2.64 |            |
| [0.05, 0.05, 0.40, 0.30, 0.20]$^T$ | 77.51 ± 2.80 | 72.55 ± 7.10 | 73.62 ± 2.99 |            |
| [0.20, 0.05, 0.05, 0.40, 0.30]$^T$ | 77.30 ± 4.13 | 69.90 ± 6.40 | 75.14 ± 2.47 |            |
| [0.30, 0.20, 0.05, 0.05, 0.40]$^T$ | 77.20 ± 5.25 | 72.59 ± 1.93 | 75.71 ± 3.25 |            |

Note: The best methods based on the 5% $t$-test and equivalent methods are in bold.

error of the false negative rate and the empirical false negative rate. We confirmed that our method is superior to the existing classification methods through analyses of synthetic and benchmark data sets. We also found that under certain circumstances, even if there is a deviation in the estimation of the false negative rate, the effect on the classification accuracy is small through the synthetic data experiment.

Our proposed method can perform classification using only P and bN data without U data. As described in section 3.1, however, it is difficult to estimate the probability $\sigma$ without U data, especially for high-dimensional data. In section 4.3.2, we applied the kernel density estimation that is applicable for high dimension, but it sometimes gives unstable results. Future work is to consider a stable estimation method for high-dimensional data. In addition, we have to explore another method for hyperparameter selection that does not require the assumption that the false negative rate is known.

Appendix: Derivation of PbN Classification Risk

The second term of the classification risk, equation 2.1, can be decomposed as follows:

\[
(1 - \pi)R_{\tilde{N}}(g) = (1 - \pi) \int \ell(-g(x))p(x|y = -1)dx \\
= \int p(y = -1)p(x|y = -1)\ell(-g(x))dx \\
= \int p(x, y = -1)\ell(-g(x))dx \\
= \int \left[p(x, y = -1, s = +1) + p(x, s = -1)\right] \ell(-g(x))dx \\
= \int p(y = -1, s = +1)p(x|y = -1, s = +1)\ell(-g(x))dx
\]
\[ + \int p(x, s = -1)\ell(-g(x))dx \]
\[ = \rho R_{\text{bn}}^-(g) + \int p(x, s = -1)\ell(-g(x))dx. \tag{A.1} \]

because \( p(x, s = -1, y = +1) = 0 \) and \( p(x, s = -1) = p(x, s = -1, y = -1) \).

Under the assumption that \( \sigma = p(s = +1|x) \neq 0 \), we obtain
\[ p(x, s = -1) + p(x, s = +1) = p(x) \]
\[ = \frac{p(x, s = +1)}{p(s = +1|x)}. \]

Then,
\[ p(x, s = -1) = \frac{p(x, s = +1) \left[ 1 - p(s = +1|x) \right]}{p(s = +1|x)} \]
\[ = \frac{1 - \sigma}{\sigma} p(x, s = +1). \]

Therefore, the second term in equation 3.1 can be expressed as
\[ \int p(x, s = -1)\ell(-g(x))dx = \int \frac{1 - \sigma}{\sigma} p(x, s = +1)\ell(-g(x))dx \]
\[ = \int \frac{1 - \sigma}{\sigma} p(s = +1)p(x|s = +1)\ell(-g(x))dx \]
\[ = (\pi + \rho)R_{s=+1}^\left( \frac{1 - \sigma}{\sigma} g \right). \]

Acknowledgments

We are grateful to reviewers, whose comments have greatly contributed to improve the quality of this letter. This work was supported by JSPS KAKENHI grant 19K11858.

References

Backurs, A., Indyk, P., & Wagner, T. (2019). Space and time efficient kernel density estimation in high dimensions. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), Advances in neural information processing systems, 32 (pp. 115773–15782). Curran.

Bhatt, R. (2017). Wireless indoor localization. UCI Machine Learning Repository.

Chapelle, O., Schölkopf, B., & Zien, A. (2006). Semi-supervised learning. MIT Press.
Chen, X., Chen, W., Chen, T., Yuan, Y., Gong, C., & Chen, K. (2020). Self-PU: Self-boosted and calibrated positive-unlabeled training. In Proceedings of the International Conference on Machine Learning (pp. 1510–1519).

du Plessis, M. C., Niu, G., & Sugiyama, M. (2014). Analysis of learning from positive and unlabeled data. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger (Eds.), Advances in neural information processing systems, 27 (pp. 703–711). Curran.

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

du Plessis, M. C., Niu, G., & Sugiyama, M. (2016). Class-prior estimation for learning from positive and unlabeled data. In Proceedings of the Asian Conference on Machine Learning (pp. 1386–1394).

du Plessis, M. C., & Sugiyama, M. (2014). Class prior estimation from positive and unlabeled data. IEICE Trans. Inform. Syst., 97(5), 1358–1362. 10.1587/transinf.E97.D.1358

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

Fei, G., & Liu, B. (2015). Social media text classification under negative covariate shift. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 2347–2356).

Hsieh, Y. G., Niu, G., & Sugiyama, M. (2019). Classification from positive, unlabeled and biased negative data. In Proceedings of the International Conference on Learning Representations (pp. 2820–2829).

Ishida, T., Niu, G., & Sugiyama, M. (2018). Binary classification from positive-confidence data. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), Advances in neural information processing systems, 31 (pp. 5921–5032). Curran.

Kato, M., Teshima, T., & Honda, J. (2019). Learning from positive and unlabeled data with a selection bias. In Proceedings of the International Conference on Learning Representations.

Kiryo, R., Niu, G., du Plessis, M. C., & Sugiyama, M. (2017). Positive-unlabeled learning with non-negative risk estimator. In I. Guyon, Y. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), Advances in neural information processing systems, 30 (pp. 1673–1685). Curran.

LeCun, Y., & Cortes, C. (1998). The MNIST database of handwritten digits.

Li, X.-L., Liu, B., & Ng, S.-K. (2010). Negative training data can be harmful to text classification. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 218–228).

Natarajan, N., S. Dhillon, I., Ravikumar, P., & Tewari, A. (2013). Learning with noisy labels. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), Advances in neural information processing systems, 26 (pp. 1196–1204). Curran.

Sakai, T., du Plessis, M. C. d., Niu, G., & Sugiyama, M. (2017). Semi-supervised classification based on classification from positive and unlabeled data. In Proceedings of the Conference on Machine Learning (pp. 2998–3006).
Shi, H., Pan, S., Yang, J., & Gong, C. (2018). Positive and unlabeled learning via loss decomposition and centroid estimation. In Proceedings of the International Joint Conference on Artificial Intelligence (pp. 2689–2695).

Shinoda, K., Kaji, H., & Sugiyama, M. (2020). Binary classification from positive data with skewed confidence. In Proceedings of the International Joint Conference on Artificial Intelligence (pp. 3328–3334).

Yao, Y., Liu, T., Han, B., Gong, M., Niu, G., Sugiyama, M., & Tao, D. (2021). Rethinking class-prior estimation for positive-unlabeled learning. In Proceedings of the International Conference on Learning Representations.

Zhang, Z., Zhao, P., Jiang, Y., & Zhou, Z. (2019). Learning from incomplete and inaccurate supervision. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 1017–1025).

Received March 29, 2022; accepted December 30, 2022.