Classification from Positive, Unlabeled and Biased Negative Data: Supplementary Material

Appendix

A. Proofs

A.1. Proof of Theorem 1

We notice that $(1-\pi-\rho)p(\boldsymbol{x} \mid s=-1) = p(\boldsymbol{x}, s=-1)$ and that when $h(\boldsymbol{x}) > \eta$, we have $p(s=+1 \mid \boldsymbol{x}) = \sigma(\boldsymbol{x}) > 0$, which allows us to write $p(s=-1 \mid \boldsymbol{x}) = (p(s=-1 \mid \boldsymbol{x})/p(s=+1 \mid \boldsymbol{x}))p(s=+1 \mid \boldsymbol{x})$. We can thus decompose $\bar{R}^{-}_{s=-1}(g)$ as following:

$$\begin{aligned}
\bar{R}^{-}_{s=-1}(g) &= \int \ell(-g(\boldsymbol{x}))p(\boldsymbol{x}, s=-1)\,dx \\
&= \int \mathbb{1}_{h(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))p(\boldsymbol{x}, s=-1)\,dx \\
&\quad + \int \mathbb{1}_{h(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))p(\boldsymbol{x}, s=-1)\,dx \\
&= \int \mathbb{1}_{h(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))\frac{p(\boldsymbol{x}, s=-1)}{p(\boldsymbol{x})}p(\boldsymbol{x})\,dx \\
&\quad + \int \mathbb{1}_{h(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))\frac{p(\boldsymbol{x}, s=-1)}{p(\boldsymbol{x}, s=+1)}p(\boldsymbol{x}, s=+1)\,dx.
\end{aligned}$$

By writing $p(\boldsymbol{x}, s=-1) = p(s=-1 \mid \boldsymbol{x})p(\boldsymbol{x}) = (1-\sigma(\boldsymbol{x}))p(\boldsymbol{x})$ and $p(\boldsymbol{x}, s=+1) = p(s=+1 \mid \boldsymbol{x})p(\boldsymbol{x}) = \sigma(\boldsymbol{x})p(\boldsymbol{x})$, we have

$$\begin{aligned}
\bar{R}^{-}_{s=-1}(g) = &\int \mathbb{1}_{h(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))(1-\sigma(\boldsymbol{x}))p(\boldsymbol{x})\,dx \\
&+ \int \mathbb{1}_{h(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))\frac{1-\sigma(\boldsymbol{x})}{\sigma(\boldsymbol{x})}p(\boldsymbol{x}, s=+1)\,dx.
\end{aligned}$$

We obtain Equation (6) after replacing $p(\boldsymbol{x}, s=+1)$ by $\pi p(x \mid y=+1) + \rho p(x \mid y=-1, s=+1)$.

A.2. Proof of Theorem 2

For $\hat{\sigma}$ and η given, let us define

$$R_{\mathrm{PUbN},\eta,\hat{\sigma}}(g) = \pi R^{+}_{\mathrm{P}}(g) + \rho R^{-}_{\mathrm{bN}}(g) + \bar{R}^{-}_{s=-1,\eta,\hat{\sigma}}(g).$$

The following lemma establishes the uniform deviation bound from $\hat{R}_{\mathrm{PUbN},\eta,\hat{\sigma}}$ to $R_{\mathrm{PUbN},\eta,\hat{\sigma}}$.

Lemma 1. *Let $\hat{\sigma}: \mathbb{R}^d \to [0,1]$ be a fixed function independent of data used to compute $\hat{R}_{\mathrm{PUbN},\eta,\hat{\sigma}}$ and $\eta \in (0,1]$. For any $\delta > 0$, with probability at least $1-\delta$,*

$$\begin{aligned}
&\sup_{g\in\mathcal{G}} |\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g) - R_{\mathrm{PUbN},\eta,\hat{\sigma}}(g)| \\
&\quad \leq 2L_\ell \mathfrak{R}_{n_{\mathrm{U}},p}(\mathcal{G}) + \frac{2\pi L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{P}},p_{\mathrm{P}}}(\mathcal{G}) + \frac{2\rho L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{bN}},p_{\mathrm{bN}}}(\mathcal{G}) + C_\ell\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{U}}}} + \frac{\pi C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{P}}}} + \frac{\rho C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{bN}}}}.
\end{aligned}$$

Proof. For ease of notation, let

$$R_{\mathrm{P}}(g) = \mathbb{E}_{\boldsymbol{x}\sim p_{\mathrm{P}}(\boldsymbol{x})}\left[\ell(g(\boldsymbol{x})) + \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))\frac{1-\hat{\sigma}(\boldsymbol{x})}{\hat{\sigma}(\boldsymbol{x})}\right],$$

$$R_{\mathrm{bN}}(g) = \mathbb{E}_{\boldsymbol{x}\sim p_{\mathrm{bN}}(\boldsymbol{x})}\left[\ell(-g(\boldsymbol{x}))\left(1+\mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\frac{1-\hat{\sigma}(\boldsymbol{x})}{\hat{\sigma}(\boldsymbol{x})}\right)\right],$$

$$R_{\mathrm{U}}(g) = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}\left[\mathbb{1}_{\hat{\sigma}(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))(1-\hat{\sigma}(\boldsymbol{x}))\right],$$

$$\hat{R}_{\mathrm{P}}(g) = \frac{1}{n_{\mathrm{P}}}\sum_{i=1}^{n_{\mathrm{P}}}\left[\ell(g(\boldsymbol{x}_i^{\mathrm{P}})) + \mathbb{1}_{\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{P}})>\eta}\,\ell(-g(\boldsymbol{x}_i^{\mathrm{P}}))\frac{1-\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{P}})}{\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{P}})}\right],$$

$$\hat{R}_{\mathrm{bN}}(g) = \frac{1}{n_{\mathrm{bN}}}\sum_{i=1}^{n_{\mathrm{bN}}}\left[\ell(-g(\boldsymbol{x}_i^{\mathrm{bN}}))\left(1+\mathbb{1}_{\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{bN}})>\eta}\frac{1-\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{bN}})}{\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{bN}})}\right)\right],$$

$$\hat{R}_{\mathrm{U}}(g) = \frac{1}{n_{\mathrm{U}}}\sum_{i=1}^{n_{\mathrm{U}}}\left[\mathbb{1}_{\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{U}})\leq\eta}\,\ell(-g(\boldsymbol{x}_i^{\mathrm{U}}))(1-\hat{\sigma}(\boldsymbol{x}_i^{\mathrm{U}}))\right].$$

From the sub-additivity of the supremum operator, we have

$$\begin{aligned}&\sup_{g\in\mathcal{G}}|\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g) - R_{\mathrm{PUbN},\eta,\hat{\sigma}}(g)|\\ &\quad\leq \pi\sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{P}}(g)-R_{\mathrm{P}}(g)| + \rho\sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{bN}}(g)-R_{\mathrm{bN}}(g)| + \sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{U}}(g)-R_{\mathrm{U}}(g)|.\end{aligned}$$

As a consequence, to conclude the proof, it suffices to prove that with probability at least $1-\delta/3$, the following bounds hold separately:

$$\sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{P}}(g)-R_{\mathrm{P}}(g)| \leq \frac{2L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{P}},p_{\mathrm{P}}}(\mathcal{G}) + \frac{C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{P}}}}, \tag{1}$$

$$\sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{bN}}(g)-R_{\mathrm{bN}}(g)| \leq \frac{2L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{bN}},p_{\mathrm{bN}}}(\mathcal{G}) + \frac{C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{bN}}}}, \tag{2}$$

$$\sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{U}}(g)-R_{\mathrm{U}}(g)| \leq 2L_\ell\mathfrak{R}_{n_{\mathrm{U}},p}(\mathcal{G}) + C_\ell\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{U}}}}. \tag{3}$$

Below we prove (1). (2) and (3) are proven similarly.

Let $\phi_{\boldsymbol{x}}:\mathbb{R}\to\mathbb{R}_+$ be the function defined by $\phi_{\boldsymbol{x}}: z\mapsto \ell(z)+\mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\,\ell(-z)((1-\hat{\sigma}(\boldsymbol{x}))/\hat{\sigma}(\boldsymbol{x}))$. For $\boldsymbol{x}\in\mathbb{R}^d, g\in\mathcal{G}$, since $\ell(g(\boldsymbol{x}))\in[0,C_\ell]$, $\ell(-g(\boldsymbol{x}))\in[0,C_\ell]$ and $\mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}((1-\hat{\sigma}(\boldsymbol{x}))/\hat{\sigma}(\boldsymbol{x}))\in[0,(1-\eta)/\eta]$, we always have $\phi_{\boldsymbol{x}}(g(\boldsymbol{x}))\in[0,C_\ell/\eta]$. Following the proof of Theorem 3.1 in (Mohri et al., 2012), it is then straightforward to show that with probability at least $1-\delta/3$, it holds that

$$\sup_{g\in\mathcal{G}}|\hat{R}_{\mathrm{P}}(g)-R_{\mathrm{P}}(g)| \leq 2\,\mathbb{E}_{\mathcal{X}_{\mathrm{P}}\sim p_{\mathrm{P}}^{n_{\mathrm{P}}}}\mathbb{E}_\theta\left[\sup_{g\in\mathcal{G}}\frac{1}{n_{\mathrm{P}}}\sum_{i=1}^{n_{\mathrm{P}}}\theta_i\phi_{\boldsymbol{x}_i}(g(\boldsymbol{x}_i))\right] + \frac{C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{P}}}},$$

where $\theta=\{\theta_1,\ldots,\theta_{n_{\mathrm{P}}}\}$ and each θ_i is a Rademacher variable.

Also notice that for all $\boldsymbol{x}$, $\phi_{\boldsymbol{x}}$ is a (L_ℓ/η)-Lipschitz function on the interval $[-C_g,C_g]$. By using a modified version of Talagrad's concentration lemma (specifically, Lemma 26.9 in (Shalev-Shwartz & Ben-David, 2014)), we can show that, when the set $\mathcal{X}_{\mathrm{P}}$ is fixed, we have

$$\mathbb{E}_\theta\left[\sup_{g\in\mathcal{G}}\frac{1}{n_{\mathrm{P}}}\sum_{i=1}^{n_{\mathrm{P}}}\theta_i\phi_{\boldsymbol{x}_i}(g(\boldsymbol{x}_i))\right] \leq \frac{L_\ell}{\eta}\mathbb{E}_\theta\left[\sup_{g\in\mathcal{G}}\frac{1}{n_{\mathrm{P}}}\sum_{i=1}^{n_{\mathrm{P}}}\theta_i g(\boldsymbol{x}_i)\right].$$

In particular, as the inequality deals with empirical Rademacher complexity, the dependence of $\phi_{\boldsymbol{x}}$ on $\boldsymbol{x}$ would not be an issue. In fact, with $\boldsymbol{x}$ being fixed, the indicator function $\mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}$ is nothing but a constant and its discontinuity has nothing to do with the Lipschitz continuity of $\phi_{\boldsymbol{x}}$. We obtain Equation (1) After taking expectation over $\mathcal{X}_{\mathrm{P}} \sim p_{\mathrm{P}}^{n_{\mathrm{P}}}$. □

However, what we really want to minimize is the true risk $R(g)$. Therefore, we also need to bound the difference between $R_{\mathrm{PUbN},\eta,\hat{\sigma}}(g)$ and $R(g)$, or equivalently, the difference between $\bar{R}^-_{s=-1,\eta,\hat{\sigma}}(g)$ and $\bar{R}^-_{s=-1}(g)$.

Lemma 2. *Let $\hat{\sigma} : \mathbb{R}^d \to [0,1]$, $\eta \in (0,1]$, $\zeta = p(\hat{\sigma} \leq \eta)$ and $\epsilon = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[|\hat{\sigma}(\boldsymbol{x}) - \sigma(\boldsymbol{x})|^2]$. For all $g \in \mathcal{G}$, it holds that*

$$|\bar{R}^-_{s=-1,\eta,\hat{\sigma}}(g) - \bar{R}^-_{s=-1}(g)| \leq C_\ell\sqrt{\zeta\epsilon} + \frac{C_\ell}{\eta}\sqrt{(1-\zeta)\epsilon}.$$

Proof. One one hand, we have

$$\begin{aligned}\bar{R}^-_{s=-1}(g) = &\underbrace{\int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))(1-\sigma(\boldsymbol{x}))p(\boldsymbol{x})\,dx}_{A_1} \\ &+ \underbrace{\int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))(1-\sigma(\boldsymbol{x}))p(\boldsymbol{x})\,dx}_{B_1}.\end{aligned}$$

On the other hand, we can express $\bar{R}^-_{s=-1,\eta,\hat{\sigma}}(g)$ as

$$\begin{aligned}\bar{R}^-_{s=-1,\eta,\hat{\sigma}}(g) &= \int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))(1-\hat{\sigma}(\boldsymbol{x}))p(\boldsymbol{x})\,dx \\ &\quad + \int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))\frac{1-\hat{\sigma}(\boldsymbol{x})}{\hat{\sigma}(x)}p(\boldsymbol{x}, s=+1)\,dx. \\ &= \underbrace{\int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})\leq\eta}\,\ell(-g(\boldsymbol{x}))(1-\hat{\sigma}(\boldsymbol{x}))p(\boldsymbol{x})\,dx}_{A_2} \\ &\quad + \underbrace{\int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))(1-\hat{\sigma}(\boldsymbol{x}))\frac{\sigma(\boldsymbol{x})}{\hat{\sigma}(\boldsymbol{x})}p(\boldsymbol{x})\,dx}_{B_2}.\end{aligned}$$

The last equality follows from $p(\boldsymbol{x}, s=+1) = \sigma(\boldsymbol{x})p(\boldsymbol{x})$. As $|\bar{R}^-_{s=-1,\eta,\hat{\sigma}}(g) - \bar{R}^-_{s=-1}(g)| \leq |A_1 - A_2| + |B_1 - B_2|$, it is sufficient to derive bounds for $|A_1 - A_2|$ and $|B_1 - B_2|$ separately. For $|B_1 - B_2|$, we write

$$\begin{aligned}|B_1 - B_2| &\leq \int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}\,\ell(-g(\boldsymbol{x}))\frac{|\hat{\sigma}(\boldsymbol{x})-\sigma(\boldsymbol{x})|}{\hat{\sigma}(\boldsymbol{x})}p(\boldsymbol{x})\,dx \\ &\leq \frac{C_\ell}{\eta}\int \mathbb{1}_{\hat{\sigma}(\boldsymbol{x})>\eta}|\hat{\sigma}(\boldsymbol{x})-\sigma(\boldsymbol{x})|p(\boldsymbol{x})\,dx \\ &\leq \frac{C_\ell}{\eta}\left(\int \mathbb{1}^2_{\hat{\sigma}(\boldsymbol{x})>\eta}p(\boldsymbol{x})\,dx\right)^{\frac{1}{2}}\left(\int |\hat{\sigma}(\boldsymbol{x})-\sigma(\boldsymbol{x})|^2p(\boldsymbol{x})\,dx\right)^{\frac{1}{2}} \\ &= \frac{C_\ell}{\eta}\sqrt{(1-\zeta)\epsilon}\end{aligned}$$

From the second to the third line we use the Cauchy-Schwarz inequality. $|A_1 - A_2| \leq C_\ell\sqrt{\zeta\epsilon}$ can be proven similarly, which concludes the proof. □

Combining lemma 1 and lemma 2, we know that with probability at least $1-\delta$, the following holds:

$$\begin{aligned}
&\sup_{g\in\mathcal{G}} |\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g) - R(g)| \\
&\quad\leq 2L_\ell \mathfrak{R}_{n_{\mathrm{U}},p}(\mathcal{G}) + \frac{2\pi L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{P}},p_{\mathrm{P}}}(\mathcal{G}) + \frac{2\rho L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{bN}},p_{\mathrm{bN}}}(\mathcal{G}) \\
&\qquad + C_\ell\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{U}}}} + \frac{\pi C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{P}}}} + \frac{\rho C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{bN}}}} + C_\ell\sqrt{\zeta\epsilon} + \frac{C_\ell}{\eta}\sqrt{(1-\zeta)\epsilon}.
\end{aligned}$$

Finally, with probability at least $1-\delta$,

$$\begin{aligned}
&R(\hat{g}_{\mathrm{PUbN},\eta,\hat{\sigma}}) - R(g^*) \\
&\quad= (R(\hat{g}_{\mathrm{PUbN},\eta,\hat{\sigma}}) - \hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(\hat{g}_{\mathrm{PUbN},\eta,\hat{\sigma}})) \\
&\qquad + (\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(\hat{g}_{\mathrm{PUbN},\eta,\hat{\sigma}}) - \hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g^*)) + (\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g^*) - R(g^*)) \\
&\quad\leq \sup_{g\in\mathcal{G}} |\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g) - R(g)| + 0 + \sup_{g\in\mathcal{G}} |\hat{R}^{-}_{\mathrm{PUbN},\eta,\hat{\sigma}}(g) - R(g)| \\
&\quad\leq 4L_\ell \mathfrak{R}_{n_{\mathrm{U}},p}(\mathcal{G}) + \frac{4\pi L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{P}},p_{\mathrm{P}}}(\mathcal{G}) + \frac{4\rho L_\ell}{\eta}\mathfrak{R}_{n_{\mathrm{bN}},p_{\mathrm{bN}}}(\mathcal{G}) \\
&\qquad + 2C_\ell\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{U}}}} + \frac{2\pi C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{P}}}} + \frac{2\rho C_\ell}{\eta}\sqrt{\frac{\ln(6/\delta)}{2n_{\mathrm{bN}}}} + 2C_\ell\sqrt{\zeta\epsilon} + \frac{2C_\ell}{\eta}\sqrt{(1-\zeta)\epsilon}.
\end{aligned}$$

The first inequality uses the definition of $\hat{g}_{\mathrm{PUbN},\eta,\hat{\sigma}}$.

B. Validation Loss for Estimation of σ

In terms of validation we want to choose the model for $\hat{\sigma}$ such that $J_0(\hat{\sigma}) = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[|\hat{\sigma}(\boldsymbol{x}) - \sigma(\boldsymbol{x})|^2]$ is minimized. Since $\sigma(\boldsymbol{x})p(\boldsymbol{x}) = p(\boldsymbol{x}, s=+1)$, we have

$$\begin{aligned}
J_0(\hat{\sigma}) &= \int (\hat{\sigma}(\boldsymbol{x}) - \sigma(\boldsymbol{x}))^2 p(\boldsymbol{x})\,dx \\
&= \int \hat{\sigma}(\boldsymbol{x})^2 p(\boldsymbol{x})\,dx - 2\int \hat{\sigma}(\boldsymbol{x})p(\boldsymbol{x}, s=+1)\,dx + \int \sigma(\boldsymbol{x})^2 p(\boldsymbol{x})\,dx.
\end{aligned}$$

The last term does not depend on $\hat{\sigma}$ and can be ignored if we want to identify $\hat{\sigma}$ achieving the smallest $J(\hat{\sigma})$. We denote by $J(\hat{\sigma})$ the sum of the first two terms. The middle term can be further expanded using

$$\int \hat{\sigma}(\boldsymbol{x})p(\boldsymbol{x}, s=+1)\,dx = \pi\int \hat{\sigma}(\boldsymbol{x})p(\boldsymbol{x} \mid y=+1)\,dx + \rho\int \hat{\sigma}(\boldsymbol{x})p(\boldsymbol{x} \mid y=-1, s=+1)\,dx.$$

The validation loss of an estimation $\hat{\sigma}$ is then defined as

$$\hat{J}(\hat{\sigma}) = \frac{1}{n_{\mathrm{U}}}\sum_{i=1}^{n_{\mathrm{U}}} \hat{\sigma}(\boldsymbol{x}_i^{\mathrm{U}})^2 - \frac{2\pi}{n_{\mathrm{P}}}\sum_{i=1}^{n_{\mathrm{P}}} \hat{\sigma}(\boldsymbol{x}_i^{\mathrm{P}}) - \frac{2\rho}{n_{\mathrm{bN}}}\sum_{i=1}^{n_{\mathrm{bN}}} \hat{\sigma}(\boldsymbol{x}_i^{\mathrm{bN}}).$$

It is also possible to minimize this value directly to acquire $\hat{\sigma}$. In our experiments we decide to learn $\hat{\sigma}$ by nnPU for a better comparison between different methods.

C. Detailed Experimental Setting

C.1. From Multiclass to Binary Class

In the experiments we work on multiclass classification datasets. Therefore it is necessary to define the P and N classes ourselves. MNIST is processed in such a way that pair numbers 0, 2, 4, 6, 8 form the P class and impair numbers 1, 3, 5, 7, 9 form the N class. Accordingly, $\pi = 0.49$. For CIFAR-10, we consider two definitions of the P class. The first one corresponds to a quite natural task that aims to distinguish vehicles from animals. Airplane, automobile, ship and truck are therefore defined to be the P class while the N class is formed by bird, cat, deer, dog, frog and horse. For the sake of diversity, we also study another task in which we attempt to distinguish the mammals from the non-mammals. The P class is then formed by cat, deer, dog, and horse while the N class consists of the other six categories. We have $\pi = 0.4$ in the two cases. As for 20 Newsgroups, alt., comp., misc. and rec. make up the P class whereas sci., soc. and talk. make up the N class. This gives $\pi = 0.56$.

C.2. Training, Validation and Test Set

For the three datasets, we use the standard test examples as a held-out test set. The test set size is thus of 10000 for MNIST and CIFAR-10, and 7528 for 20 Newsgroups. Regarding the training set, we sample 500, 500 and 6000 P, bN and U training examples for MNIST and 20 Newsgroups, and 1000, 1000 and 10000 P, bN and U training examples for CIFAR-10. The validation set is always five times smaller than the training set.

C.3. 20 Newsgroups Preprocessing

The original 20 Newsgroups dataset contains raw text data and needs to be preprocessed into text feature vectors for classification. In our experiments we borrow the pre-trained ELMo word embedding (Peters et al., 2018) from https://allennlp.org/elmo. The used 5.5B model was, according to the website, trained on a dataset of 5.5B tokens consisting of Wikipedia (1.9B) and all of the monolingual news crawl data from WMT 2008-2012 (3.6B). For each word, we concatenate the features from the three layers of the ELMo model, and for each document, as suggested by Rücklé et al. (2018), we concatenate the average, minimum, and maximum computed along the word dimension. This results in a 9216-dimensional feature vector for a single document.

C.4. Models and Hyperparameters

Shared The nnPU threshold parameter β and the weight decay are respectively fixed at 0 and 10^{-4}. Other hyperparameters including $\tau \in \{0.5, 0.7, 0.9\}$, $\gamma \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ and learning rate are selected with validation data.

MNIST For MNIST, we use a standard ConvNet with ReLU. This model contains two 5x5 convolutional layers and one fully-connected layer, with each convolutional layer followed by a 2x2 max pooling. The channel sizes are 5-10-40. The model is trained for 100 epochs with each minibatch made up of 10 P, 10 bN (if available) and 120 U samples. The learning rate is selected from the range $\alpha \in \{10^{-2}, 10^{-3}\}$.

CIFAR-10 For CIFAR-10, we train PreAct ResNet-18 (He et al., 2016) for 200 epochs and the learning rate is divided by 10 after 80 epochs and 120 epochs. This is a common practice and similar adjustment can be found in (He et al., 2016). The minibatch size is 1/100 of the number of training samples, and the initial learning rate is chosen from $\{10^{-2}, 10^{-3}\}$.

20 Newsgroups For 20 Newsgroups, with the extracted features, we simply train a multilayer perceptron with two hidden layers of 300 neurons for 50 epochs. We use basically the same hyperparameters as for MNIST except that the learning rate α is selected from $\{5 \cdot 10^{-3}, 10^{-3}, 5 \cdot 10^{-4}\}$.

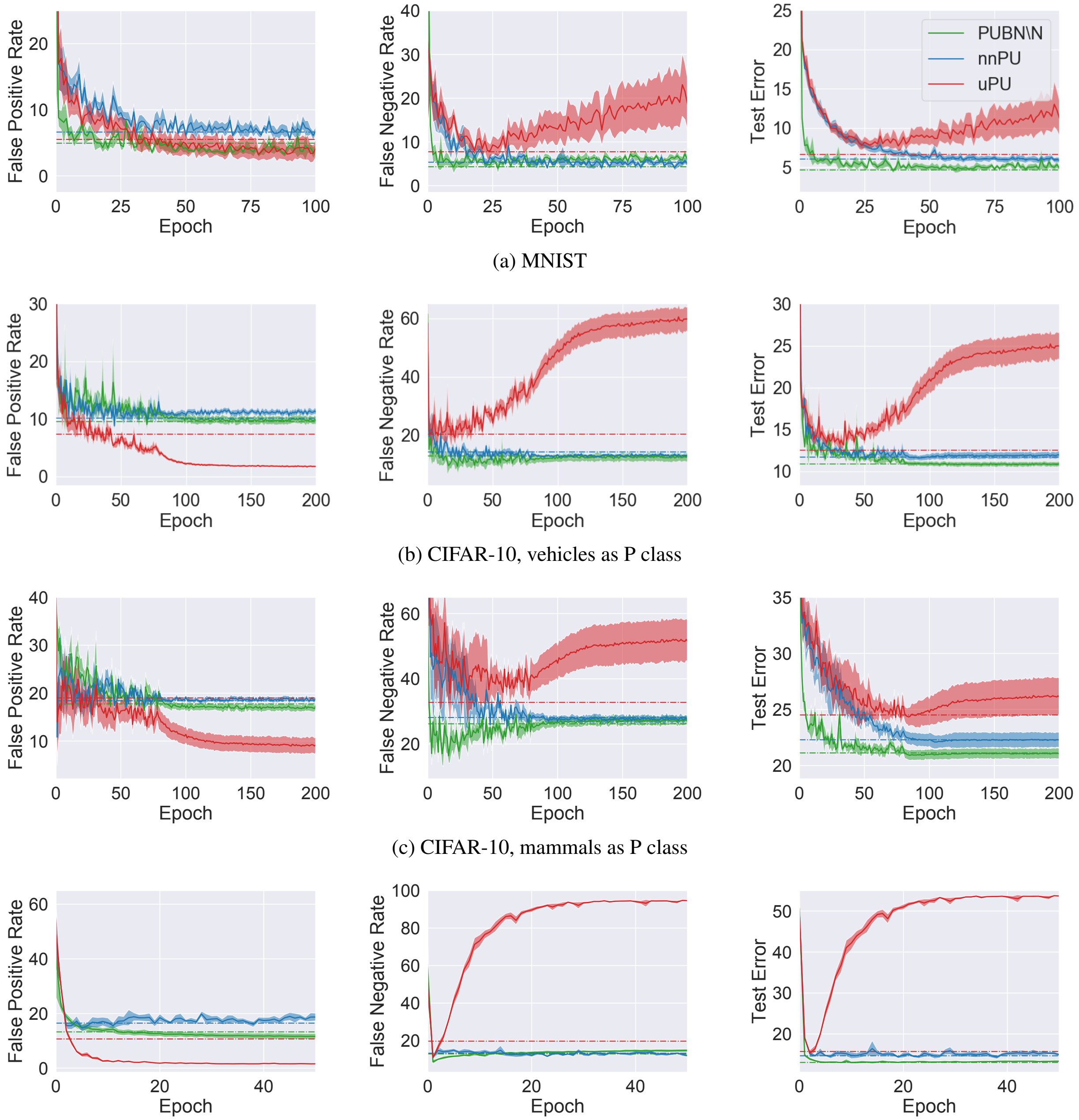
D. Additional Experiments

D.1. Why Does PUbN\N Outperform nnPU ?

Here we complete the results presented in Section 4.4 with the plots on the other two PU learning tasks (Figure 1). We recall that we compare between PUbN\N, nnPU and uPU learning, and that both uPU and nnPU are learned with the sigmoid loss, learning rate 10^{-3} for MNIST and initial learning rate 10^{-4} for CIFAR-10. The learning rate is 10^{-4} for 20 Newsgroups.

D.2. Influence of η and ρ

In the proposed algorithm we introduce η to control how $\bar{R}_{s=-1}(g)$ is approximated from data and assume that $\rho = p(y = -1, s = +1)$ is given. Here we conduct experiments to see how our method is affected by these two factors. To assess the influence of η, from Table 1 we pick four learning tasks and we choose τ from $\{0.5, 0.7, 0.9, 2\}$ while all the other hyperparameters are fixed. Similarly, to simulate the case where ρ is misspecified, we replace it by $\rho' \in \{0.8\rho, \rho, 1.2\rho\}$ in



(a) MNIST

(b) CIFAR-10, vehicles as P class

(c) CIFAR-10, mammals as P class

(d) 20 Newsgroups

Figure 1. Comparison of uPU, nnPU and PUbN\N over the four PU learning tasks. For each task, means and standard deviations are computed based on the same 10 random samplings. Dashed lines indicate the corresponding values of the final classifiers (recall that at the end we select the model with the lowest validation loss out of all epochs).

our learning method and run experiments with all hyperparameters being fixed to a certain value. However, we still use the true ρ to compute η from τ to ensure that we always use the same number of U samples in the second step of the algorithm independent of the choice of ρ'.

The results are reported in Table 1 and Table 2. We can see that the performance of the algorithm is sensitive to the choice of τ. With larger value of τ, more U data are treated as N data in PUbN learning, and consequently it often leads to higher false negative rate and lower false positive rate. The trade-off between these two measures is a classic problem in binary classification. In particular, when $\tau = 2$, a lot more U samples are involved in the computation of the PUbN risk (7), but this

Table 1. Results on four different PUbN learning tasks when we vary the value of τ (and accordingly, η). Reported are means of false positive rates (FPR), false negative rates (FNR), misclassification rates (Error), and validation losses (VLoss) over 10 trials.

Dataset	P	biased N	τ	FPR	FNR	Error	VLoss
MNIST	0, 2, 4, 6, 8	1, 3, 5	0.5	4.79	4.32	4.56	10.11
			0.7	3.32	4.81	4.05	**9.15**
			0.9	3.29	4.40	**3.83**	9.30
			2	3.38	5.32	4.33	10.68
CIFAR-10	Airplane, automobile, ship, truck	Horse > deer = frog > others	0.5	8.31	12.35	**9.92**	**12.50**
			0.7	8.23	13.15	10.20	12.62
			0.9	7.54	14.68	10.40	13.08
			2	6.23	20.29	11.85	13.64
CIFAR-10	Cat, deer, dog, horse	Bird, frog	0.5	14.45	27.57	19.70	22.08
			0.7	13.20	27.27	**18.83**	**20.72**
			0.9	13.00	32.61	20.84	23.78
			2	11.67	31.49	19.60	22.52
20 Newsgroups	alt., comp., misc., rec.	soc. > talk. > sci.	0.5	11.28	12.90	**12.18**	**16.04**
			0.7	11.40	13.58	12.62	16.64
			0.9	10.09	16.70	13.79	16.90
			2	10.34	20.55	16.06	20.99

Table 2. Mean and standard deviation of misclassification rates over 10 trials on different PUbN learning tasks when we replace ρ by $\rho' \in \{0.8\rho, \rho, 1.2\rho\}$. Underlines indicate significant degradation of performance according to the 5% t-test.

Dataset	P	biased N	ρ'/ρ		
			0.8	1	1.2
MNIST	0, 2, 4, 6, 8	1, 3, 5	4.10 ± 0.39	4.05 ± 0.27	4.14 ± 0.45
		9 > 5 > others	3.85 ± 0.55	3.91 ± 0.66	3.94 ± 0.54
CIFAR-10	Airplane, automobile, ship, truck	Cat, dog, horse	10.23 ± 0.59	9.71 ± 0.51	$\underline{10.32 \pm 0.57}$
		Horse > deer = frog > others	10.18 ± 0.40	9.92 ± 0.42	10.05 ± 0.59
CIFAR-10	Cat, deer, dog, horse	Bird, frog	18.94 ± 0.50	18.83 ± 0.71	19.06 ± 0.80
		Car, truck	20.39 ± 1.24	20.19 ± 1.06	19.92 ± 0.89
20 Newsgroups	alt., comp., misc., rec.	sci.	13.49 ± 0.61	13.10 ± 0.90	13.31 ± 1.05
		talk.	12.64 ± 0.69	12.61 ± 0.75	$\underline{13.77 \pm 0.85}$
		soc. > talk. > sci.	$\underline{12.90 \pm 0.79}$	12.18 ± 0.59	$\underline{12.74 \pm 0.35}$

does not allow the classifier to achieve a better performance. We also observe that there is a positive correlation between the misclassification rate and the validation loss, which confirms that the optimal value of η can be chosen without need of unbiased N data.

Table 2 shows that in general slight misspecification of ρ does not cause obvious degradation of the classification performance. In fact, misspecification of ρ mainly affect the weights of each sample when we compute $\hat{R}_{\mathrm{PUbN},\eta,\hat{\sigma}}$ (due to the direct presence of ρ in (7) and influence on estimating σ). However, as long as the variation of these weights remain in a reasonable range, the learning algorithm should yield classifiers with similar performances.

Table 3. Mean and standard deviation of misclassification rates over 10 trials on different PUbN learning tasks with $\hat{\sigma}$ and g trained using either the same or different sets of data.

Dataset	P	biased N	Data for $\hat{\sigma}$ and g	
			Same	Different
MNIST	0, 2, 4, 6, 8	1, 3, 5	4.05 ± 0.27	3.71 ± 0.45
		$9 > 5 >$ others	3.91 ± 0.66	4.06 ± 0.36
CIFAR-10	Airplane, automobile, ship, truck	Cat, dog, horse	9.71 ± 0.51	10.00 ± 0.51
		Horse $>$ deer $=$ frog $>$ others	9.92 ± 0.42	9.66 ± 0.46
CIFAR-10	Cat, deer, dog, horse	Bird, frog	18.83 ± 0.71	18.52 ± 0.70
		Car, truck	20.19 ± 1.06	19.98 ± 0.93
20 Newsgroups	alt., comp., misc., rec.	sci.	15.61 ± 1.50	16.60 ± 2.38
		talk.	17.14 ± 1.87	15.80 ± 0.95
		soc. $>$ talk. $>$ sci.	15.93 ± 1.88	15.80 ± 1.91

D.3. Estimating σ from Separate Data

Theorem 2 suggests that $\hat{\sigma}$ should be independent from the data used to compute $\hat{R}_{\mathrm{PUbN},\eta,\hat{\sigma}}$. Therefore, here we investigate the performance of our algorithm when $\hat{\sigma}$ and g are optimized using different sets of data. We sample two training sets and two validation sets in such a way that they are all disjoint. The size of a single training set and a single validation set is as indicated in Appendix C.2, except for 20 Newsgroups we reduce the number of examples in a single set by half. We then use different pairs of training and validation sets to learn $\hat{\sigma}$ and g. For 20 Newsgroups we also conduct standard experiments where $\hat{\sigma}$ and g are learned on the same data, whereas for MNIST and CIFAR-10 we resort to Table 1.

The results are presented in Table 3. Estimating σ from separate data does not seem to benefit much the final classification performance, despite the fact that it requires collecting twice more samples. In fact, $\hat{\bar{R}}^{-}_{s=-1,\eta,\hat{\sigma}}(g)$ is a good approximation of $\bar{R}^{-}_{s=-1,\eta,\hat{\sigma}}(g)$ as long as the function $\hat{\sigma}$ is smooth enough and does not possess abrupt changes between data points. With the use of non-negative correction, validation data and L2 regularization, the resulting $\hat{\sigma}$ does not overfit training data so this should always be the case. As a consequence, even if $\hat{\sigma}$ and g are learned on the same data, we are still able to achieve small generalization error with sufficient number of samples.

D.4. Alternative Definition of nnPNU

In subsection 2.3, we define the nnPNU algorithm by forcing the estimator of the whole N partial risk to be positive. However, notice that the term $\gamma(1-\pi)\hat{R}^{-}_{\mathrm{N}}(g)$ is always positive and the chances are that including it simply makes non-negative correction weaker and is thus harmful to the final classification performance. Therefore, here we consider an alternative definition of nnPNU where we only force the term $(1-\gamma)(\hat{R}^{-}_{\mathrm{U}}(g) - \pi\hat{R}^{-}_{\mathrm{P}}(g))$ to be positive. We plug the resulting algorithm in the experiments of subsection 4.2 and summarize the results in Table 4 in which we denote the alternative version of nnPNU by nnPU+PN since it uses the same non-negative correction as nnPU. The table indicates that neither of the two definitions of nnPNU consistently outperforms the other. It also ensures that there is always a clear superiority of our proposed PUbN algorithm compared to nnPNU despite its possible variant that is considered here.

D.5. More on Text Classification

Fei & Liu (2015) introduced CBS learning in the context of text classification. The idea is to transform document representation from the traditional n-gram feature space to a center-based similarity (CBS) space, in hope that this could mitigate the adverse effect of N data being biased. They conducted experiments with SVMs and showed that the transformation could effectively help improving the classification performance. However, this process largely reduces the number of input features, and for us it is unclear whether CBS transformation would still be beneficial when it is possible

Table 4. Mean and standard deviation of misclassification rates over 10 trials on different PUbN learning tasks for the two possible definitions of the nnPNU algorithm.

Dataset	P	biased N	nnPNU	nnPU + PN
MNIST	0, 2, 4, 6, 8	1, 3, 5	5.33 ± 0.97	5.68 ± 0.78
		$9 > 5 >$ others	4.60 ± 0.65	5.10 ± 1.54
CIFAR-10	Airplane, automobile, ship, truck	Cat, dog, horse	10.25 ± 0.38	10.87 ± 0.62
		Horse $>$ deer $=$ frog $>$ others	9.98 ± 0.53	10.77 ± 0.65
CIFAR-10	Cat, deer, dog, horse	Bird, frog	22.00 ± 0.53	21.41 ± 1.01
		Car, truck	22.00 ± 0.74	21.80 ± 0.74
20 Newsgroups	alt., comp., misc., rec.	sci.	14.69 ± 0.46	14.50 ± 1.32
		talk.	14.38 ± 0.74	14.71 ± 1.01
		soc. $>$ talk. $>$ sci.	14.41 ± 0.70	13.66 ± 0.72

to use other kinds of text features or more sophisticated models. On the contrary, we propose a training strategy that is a priori compatible with any extracted features and models. As mentioned in the introduction, another important difference between our work and CBS learning is that the latter does not assume availability of U data while the presence of U data is indispensable in our case.

Experimental Setup. Below we compare PUbN learning with CBS learning on 20 Newsgroups text classification experiments. The numbers of different types of examples that are used by each learning method are summarized in Table 5, where PbN denotes the case where only P and bN data are available. Notice that here we consider two different PbN learning settings, depending on the number of used bN samples. If we compare the two PbN settings with the PUbN setting, for one we add extra U samples and for the other we replace a part of bN samples by U samples. A second point to notice is that no validation data are used in the PbN settings. In fact, although we empirically observe that the performance of CBS learning is greatly influenced by the choice of the hyperparameters, from (Fei & Liu, 2015) it is unclear how validation data can be used for hyperparameter tuning. As a result, for CBS learning we simply report the best results that were achieved in our experiments. The reported values can be regarded as an upper bound on the performance of this method. By the way, SVM training itself usually does not require the use of validation data.

To make PUbN and CBS learning comparable, we use linear model and take normalized tf-idf vectors as input variables in PUbN learning. We also include another PbN baseline that directly trains a SVM on the normalized tf-idf feature vectors. Regarding CBS learning, we use Chi-Square feature selection as it empirically produces the best results. The number of retained features is considered as a hyperparameter and as mentioned above its value can in effect have great impact on the final result. Although Fei & Liu (2015) suggested using simultaneously unigram, bigram and trigram representations of each document, the use of bigrams and trigrams does not appear to provide any benefits in our experiments. Therefore for the results that are presented here we only use the unigram representation of a document.

Results. Table 6 affirms the superiority of PUbN learning, even in the case where the PbN and PUbN settings share the same total number of samples. Only in the third learning task with 1000 bN samples and without CBS transformation PbN learning slightly outperforms PUbN learning. This can be explained by the fact that in this learning task, though the collected N samples are biased, they still cover all the possible topics appearing in the N distribution.

CBS transformation does sometimes improve the classification accuracy, but the improvement is not consistent. We conjecture that CBS learning is not so effective here both because our P class contains several distinct topics, and because our N class is not so diverse compared with (Fei & Liu, 2015). Having multiple topics in P implies that there may not be a meaningful center for the P class in the feature space. On the other hand, in the results reported by Fei & Liu (2015) we can see that the benefit of CBS learning becomes less significant when a large proportion of N topics can be found in the bN set. In particular, in the last learning task, CBS transformation even turns out to be harmful. We also observe that after CBS transformation, the classifier becomes less sensitive to both how bN data are sampled and the size of the bN set. This seems

Table 5. Number of samples in each set under different learning settings for supplementary 20 Newsgroups experiments that compare PUbN learning with PbN learning and (Fei & Liu, 2015).

	P train	bN train	U train	P validation	bN validation	U validation	Total
PUbN	500	320	500	100	80	100	1600
PbN 400	600	400	NA	NA	NA	NA	1000
PbN 1000	600	1000	NA	NA	NA	NA	1600

Table 6. Mean and standard deviation of misclassification rates over 10 trials for text classification tasks on 20 newsgroups to compare PUbN learning with CBS learning and a PbN baseline.

biased N	PUbN	PbN 400		PbN 1000	
		tf-idf	CBS	tf-idf	CBS
sci.	$\mathbf{13.06 \pm 0.69}$	25.31 ± 0.72	21.15 ± 0.73	19.68 ± 0.94	17.72 ± 0.74
talk.	$\mathbf{14.88 \pm 0.63}$	23.42 ± 0.44	22.73 ± 0.81	19.41 ± 0.50	19.59 ± 0.47
soc. > talk. > sci.	$\mathbf{13.96 \pm 0.61}$	19.82 ± 0.86	21.68 ± 0.84	$\mathbf{13.68 \pm 0.43}$	17.43 ± 0.53

to suggest that CBS learning is the most beneficial when both the number of topics appeared in the bN set and the amount of bN data are very limited.