Revisiting Sample Selection Approach to Positive-Unlabeled Learning: Turning Unlabeled Data into Positive rather than Negative

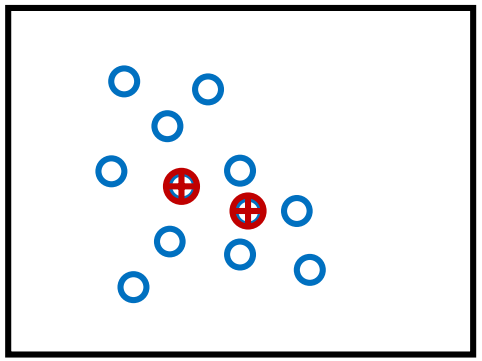
Miao Xu et.al.

RIKEN

arXiv:1901.10155

1 Introduction

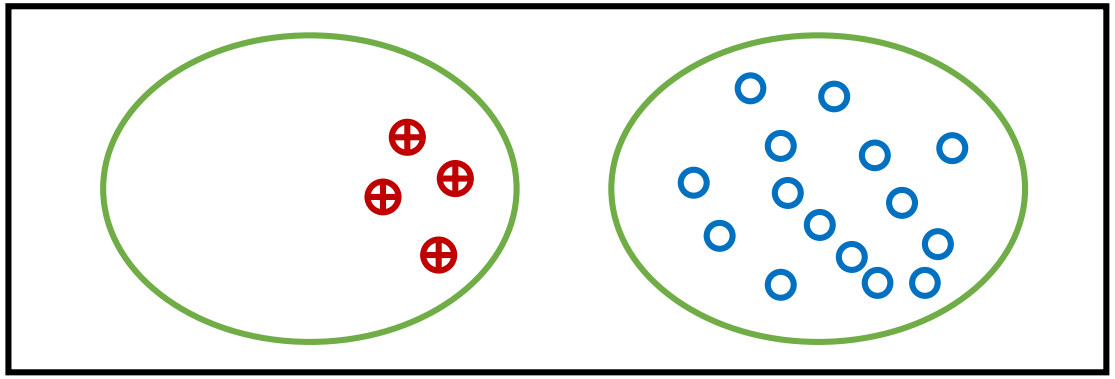
censoring PU learning



U data is collected first and then some of P data in the U data is labeled

e.g. recommendation

case-control PU learning:



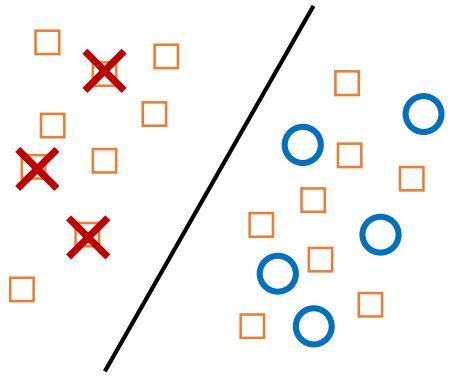
P and U data are collected separately

e.g. anomaly detection

1 Introduction

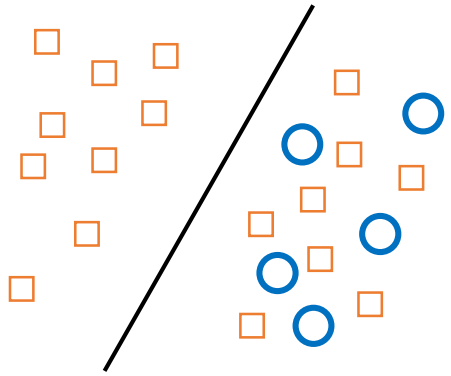
PU Learning Methods

sample selection



Select N data from U data and performs ordinary PN learning afterwards

importance reweighting



Take U data as N data with weights

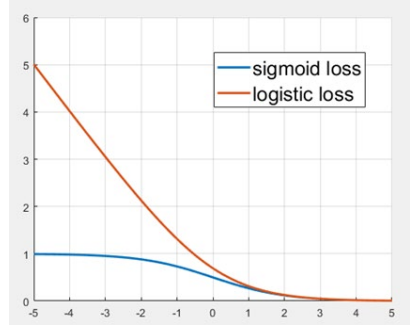
e.g. uPU, nnPU

2 ERM-based PU Learning Review

$$R(g) = \mathbb{E}_{(X,Y)\sim p(x,y)}[\ell(Y g(X;\theta))]$$

$$\ell_{\text{sigmoid}}(ty) = 1/(1+\exp(ty))$$

$$\ell_{\text{logistic}}(ty) = \ln(1+\exp(-ty))$$



P N

$$\widehat{R}_{\text{pn}} = \frac{\pi}{n_{\text{p}}}\sum_{i=1}^{n_{\text{p}}}\ell(g(x_i;\theta)) + \frac{1-\pi}{n_{\text{n}}}\sum_{i=1}^{n_{\text{n}}}\ell(-g(x_i;\theta))$$

$$\widehat{R}_{\text{pu}} = \frac{\pi}{n_{\text{p}}}\sum_{i=1}^{n_{\text{p}}}\ell(g(x_i;\theta)) + \left(\frac{1}{n_{\text{u}}}\sum_{i=1}^{n_{\text{u}}}\ell(-g(x_i;\theta)) - \frac{\pi}{n_{\text{p}}}\sum_{i=1}^{n_{\text{p}}}\ell(-g(x_i;\theta))\right)$$

Take U as N

Take P as N

$$\widehat{R}_{\text{nnPU}} = \frac{\pi}{n_{\text{p}}}\sum_{i=1}^{n_{\text{p}}}\ell(g(x_i;\theta)) + \max\left(0,\ \frac{1}{n_{\text{u}}}\sum_{i=1}^{n_{\text{u}}}\ell(-g(x_i;\theta)) - \frac{\pi}{n_{\text{p}}}\sum_{i=1}^{n_{\text{p}}}\ell(-g(x_i;\theta))\right)$$

3 Proposed Method

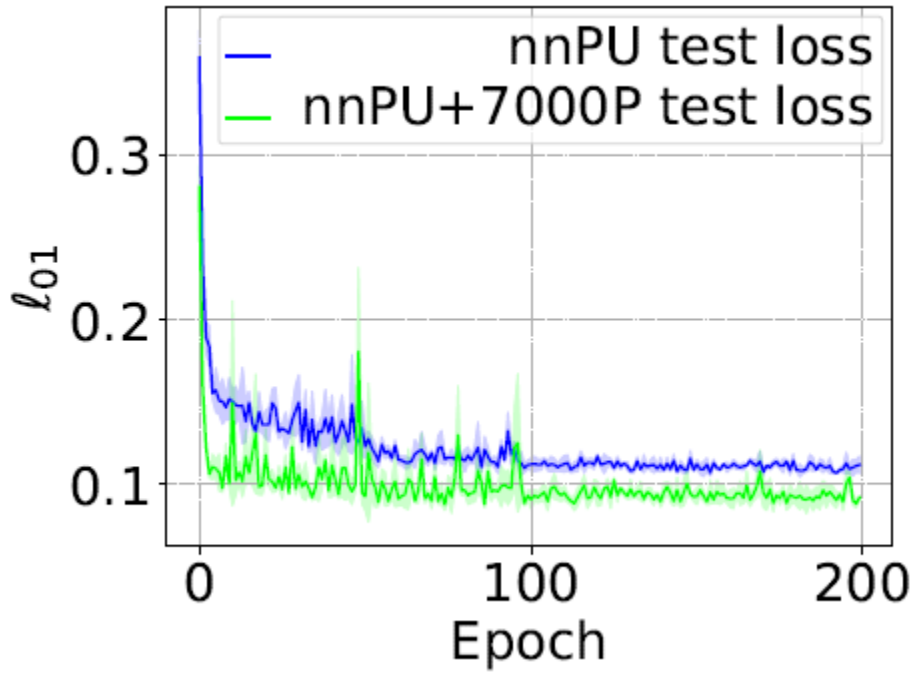
Motivation: promote performance without extra annotation cost – **Self training**

What to select?

How to select?

How to use the selected data?

3.1 What to Select

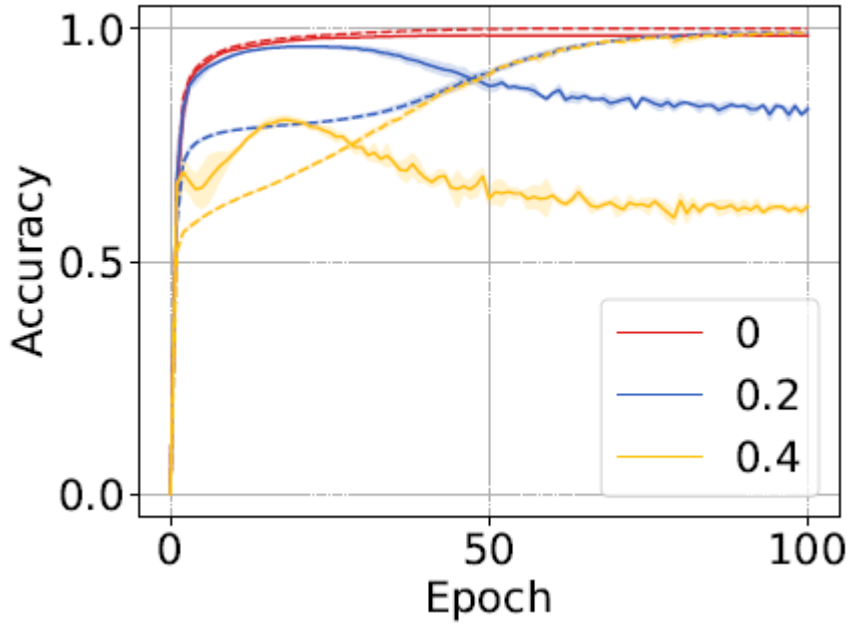


The effect of adding positive data for PU learning on test loss of the CIFAR-10 dataset.

How to select P from U?

3.2 How to Select

Affect of noise



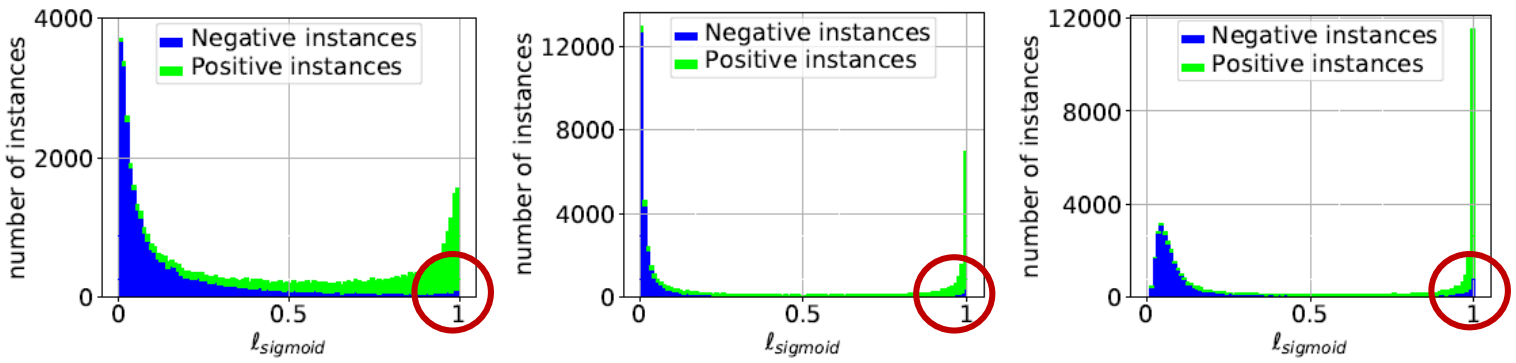
0: no noise
0.2: 20% random label noise
0.4: 40% random label noise

3.2 How to Select

*** how to select P data as *pure* as possible?**

1. feed forward all U data into the neural network and calculate their sigmoid loss when treating them as negative
2. divide the range [0, 1] equally into 100 bins and calculate the number of instances falling into each bin

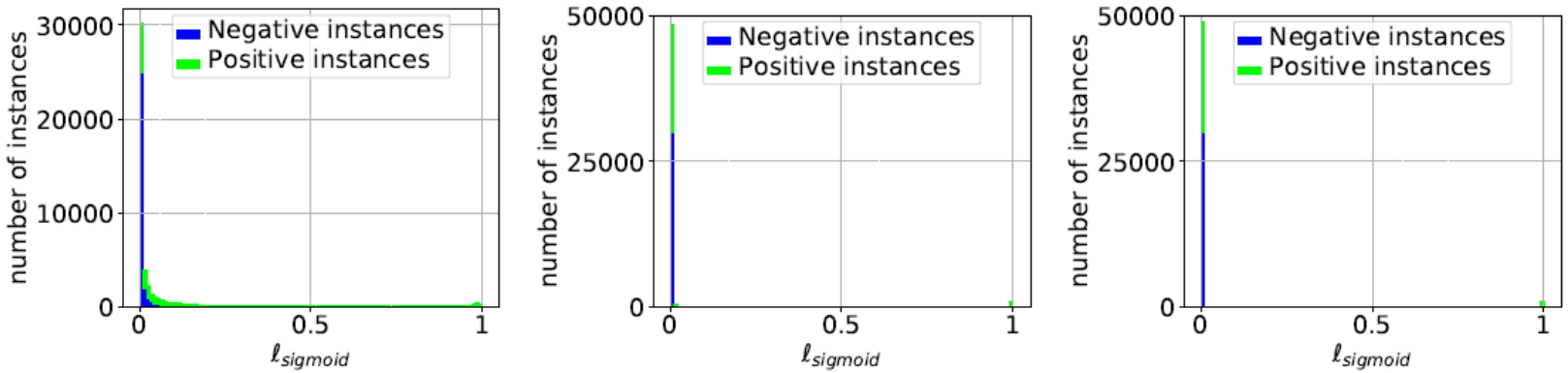
nnPU



(a) Epoch 10 (b) Epoch 50 (c) Epoch 200

3.2 How to Select

How about change a PU learning method?

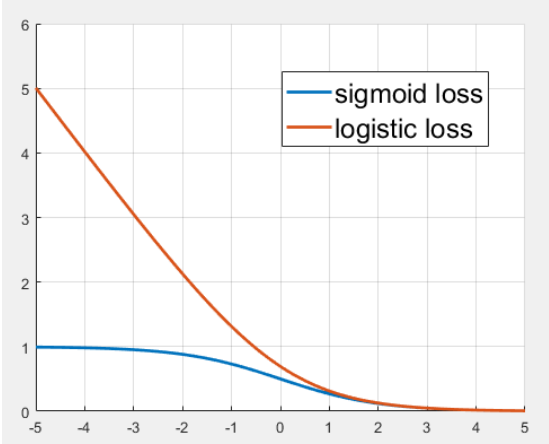


(a) Epoch 10 (b) Epoch 50 (c) Epoch 200

The uPU algorithm's sigmoid loss histogram of U data on CIFAR-10.

3.2 How to Select

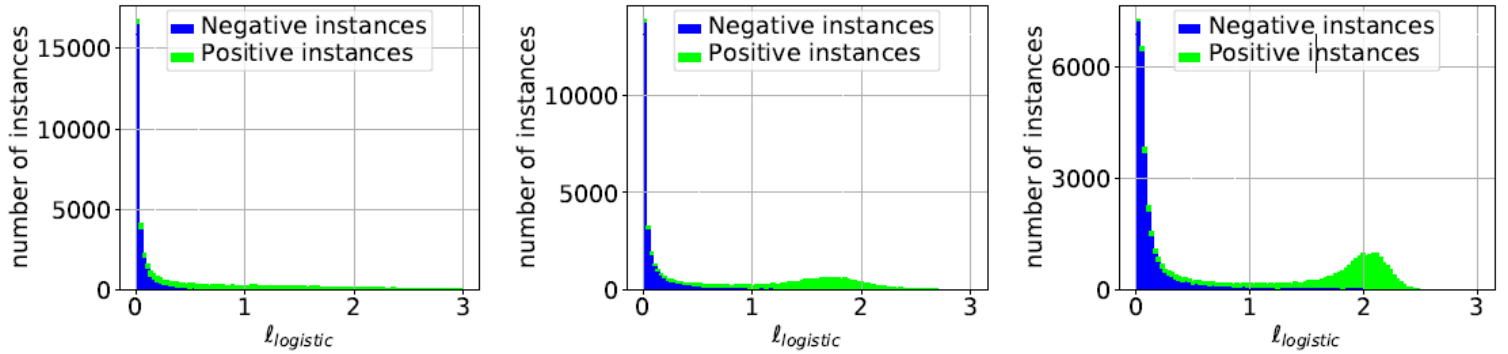
Different Loss Function



$$\ell_{\text{sigmoid}}(ty) = 1/(1+\exp(ty))$$

$$\ell_{\text{logistic}}(ty) = \ln(1+\exp(-ty))$$

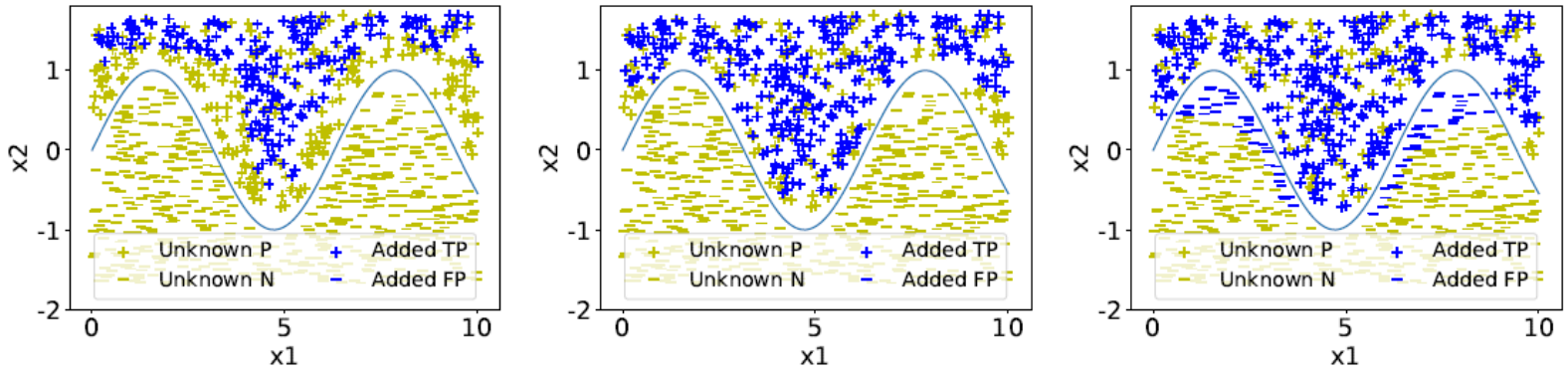
Logistic loss



(a) Epoch 10 (b) Epoch 50 (c) Epoch 200

3.3 How to Use the Selected Data

bias



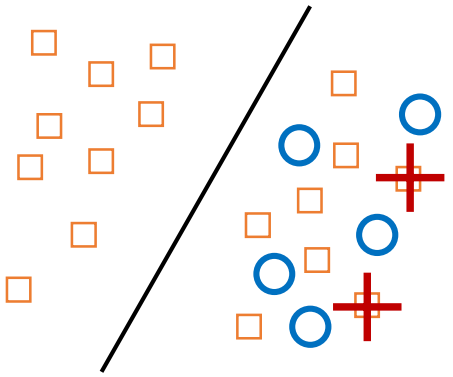
(a) Top 200 large-loss U data (b) Top 300 large-loss U data (c) Top 400 large-loss U data

noisy

Experimental results showing the data with largest loss vs. decision boundary in epoch 200. (a) The top 200 largest loss data. (b) The top 300 largest loss data. (c) The top 400 largest loss data.

3.4 aaPU-*adaptively augmented PU learning*

$$R_{\mathrm{pu}} = \pi R_{\mathrm{p}}^{+} + (R_{\mathrm{u}} - \pi R_{\mathrm{p}}^{-})$$



$$\begin{aligned} R_{\mathrm{p}}^{+} &= \mathbb{E}_{X\sim p(x|Y=1)}[\ell(g(X;\theta))], \\ R_{\mathrm{p}}^{-} &= \mathbb{E}_{X\sim p(x|Y=1)}[\ell(-g(X;\theta))], \\ R_{\mathrm{n}} &= \mathbb{E}_{X\sim p(x|Y=-1)}[\ell(-g(X;\theta))], \\ R_{\mathrm{u}} &= \mathbb{E}_{X\sim p(x)}[\ell(-g(X;\theta))]. \end{aligned}$$

$$\widehat{R}_{\mathrm{aaPU}} = \frac{\pi}{n_{\mathrm{p}}} \sum_{x_i \in \mathcal{X}_{\mathrm{p}} \cup \mathcal{S}} \ell_{\mathrm{logistic}}(g(x_i;\theta))$$

$$+ \max\left(\frac{1}{n_{\mathrm{u}}} \sum_{x_i \in \mathcal{X}_{\mathrm{u}}} \ell_{\mathrm{logistic}}(-g(x_i;\theta)) - \frac{\pi}{n_{\mathrm{p}}} \sum_{x_i \in \mathcal{X}_{\mathrm{p}}} \ell_{\mathrm{logistic}}(-g(x_i;\theta)), 0 \right)$$

Algorithm

Algorithm 1 adaptively augmented PU learning (aaPU)

Input

$\mathcal{X}_{\mathrm{p}}$	positive training data
$\mathcal{X}_{\mathrm{u}}$	unlabeled training data
n_{u}	number of unlabeled training data
T	maximum number of epochs
η	learning rate of stochastic gradient descent
$\mathcal{S}$	selected data

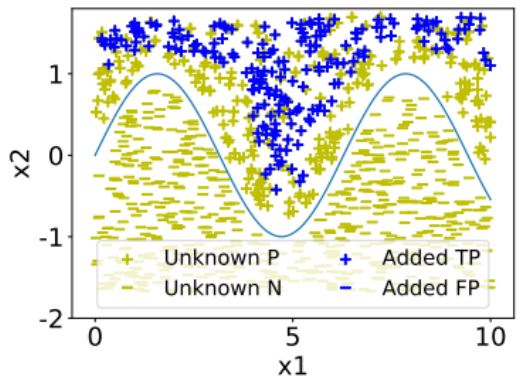
Output

θ	model parameter θ for $g(x;\theta)$

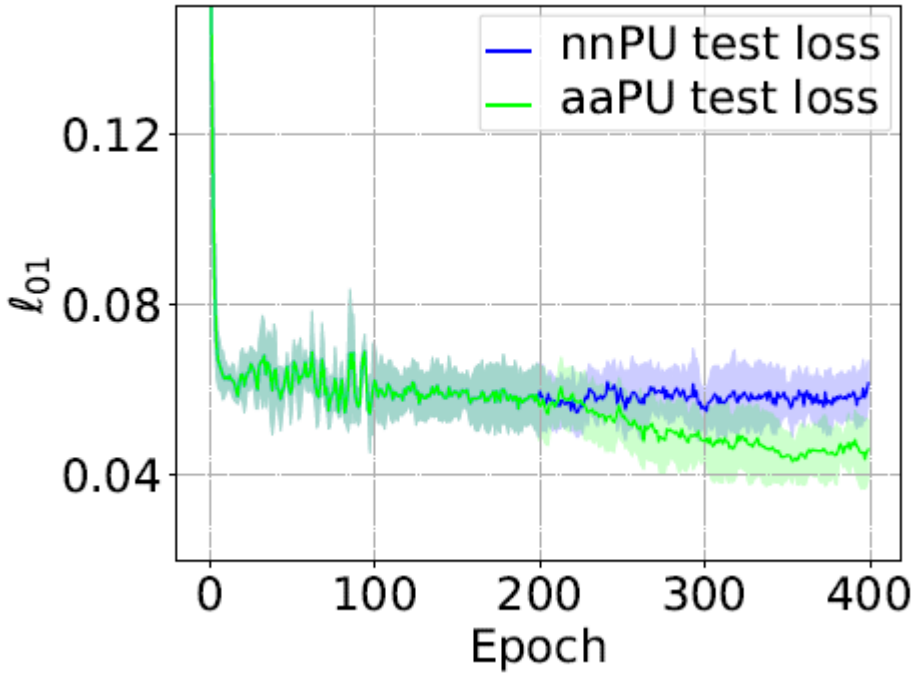
Initialize θ and $\mathcal{S}=\emptyset$
for $t=1,2,\ldots,T$ **do**
  Shuffle $\mathcal{X}_{\mathrm{p}}, \mathcal{X}_{\mathrm{u}}, \mathcal{S}$ into N_t mini-batches
  Denoted by $(\mathcal{X}_{\mathrm{p}}^i, \mathcal{X}_{\mathrm{u}}^i, \mathcal{S}^i)$ the i-th mini-batch
  for $i=1,2,\ldots,N_t$ **do**
    Update θ by $\theta = \theta - \eta \Delta_\theta \widehat{R}_{\mathrm{aaPU}}(g; \mathcal{X}_{\mathrm{p}}^i; \mathcal{X}_{\mathrm{u}}^i; \mathcal{S}^i)$
  Calculate L_u using Eq. (11)
  Select the first n_s largest L_{ui} keeping $x_i \notin \mathcal{X}_{\mathrm{p}}$
  Update $\mathcal{S}$ using corresponding x_i-s

$$L_u = [\ell(-g(x_1;\theta)), \ldots, \ell(-g(x_{n_{\mathrm{u}}};\theta))]$$

4 Experiments - Synthetic Data

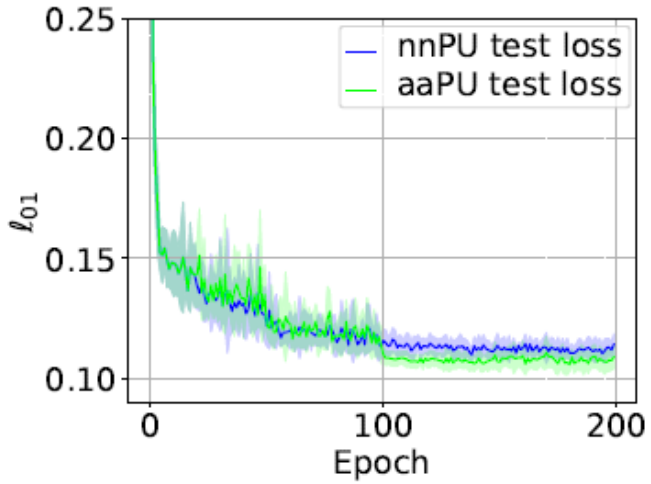


Synthetic Data

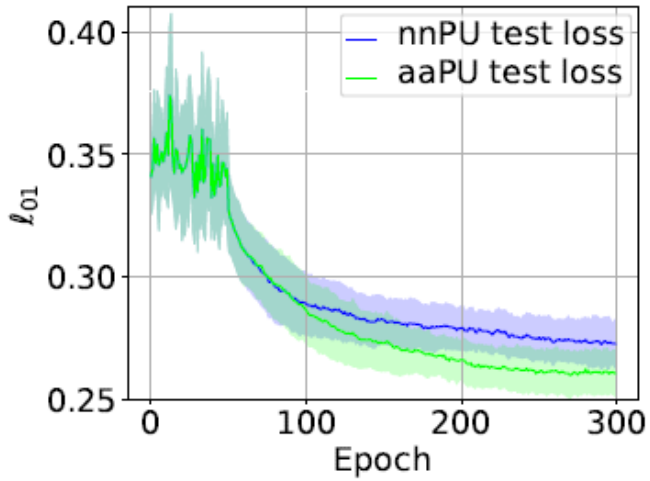


Results on synthetic data comparing aaPU and nnPU

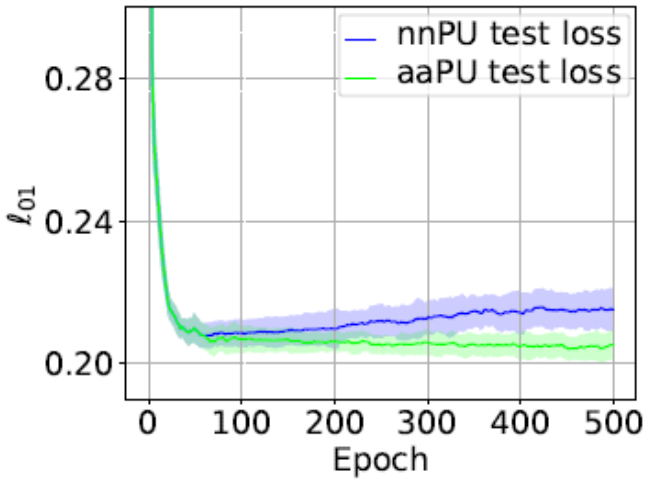
4 Experiments - Real Data



CIFAR-10



CIFAR-100



20News

P: animal
N: non animal

When to add: begin to add P data when the validation loss becomes stable
How much to add: select the number of positive data selected from [40; 80; 160; 320]

5 Conclusion

1) Propose aaPU, a new sample selection method which can boost performance of PU
2) Automatically selects P data from U data during training, and uses these selected data in future epochs
3) Data with large loss are selected to estimate the positive risk
4) Experiments validate that the proposed aaPU can work better than nnPU