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

Tomoya Sakai, Marthinus Christoffel du Plessis,
Gang Niu, Masashi Sugiyama.

ICML, 2017

Two Samples

Traditional Classification

$$\{x_i, y_i\}_{i=1}^{m} \overset{i.i.d.}{\sim} p(x, y)$$

PU Classification

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

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

PN Classification

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

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

PN Classification

$$l(t) = \begin{cases} 0, t > 0 \\ 1, \mathrm{t} \leq 0 \end{cases}$$

- Let g be a **decision function** & ℓ be a **loss function**
- The **risk** of g is

$$R(g) = \mathbb{E}_{(X,Y)\sim p(x,y)}\left[\ell\left(Yg(X)\right)\right]$$
$$= \pi_{\mathrm{p}}\mathbb{E}_{\mathrm{p}}\left[\ell\left(g(X)\right)\right] + \pi_{\mathrm{n}}\mathbb{E}_{\mathrm{n}}\left[\ell\left(-g(X)\right)\right]$$

where $\pi_{\mathrm{n}} = 1 - \pi_{\mathrm{p}}$

- The risk can be approximated directly by

$$\hat{R}_{\mathrm{pn}}(g) = \frac{\pi_{\mathrm{p}}}{n_{\mathrm{p}}}\sum_{x\in\mathcal{X}_{\mathrm{p}}}\ell\left(g(x)\right) + \frac{\pi_{\mathrm{n}}}{n_{\mathrm{n}}}\sum_{x\in\mathcal{X}_{\mathrm{n}}}\ell\left(-g(x)\right)$$

- This doesn't work for PU learning!

PU Classification

- Key observations
  - $\pi_{\mathrm{n}} p_{\mathrm{n}}(x) = p(x) - \pi_{\mathrm{p}} p_{\mathrm{p}}(x)$
  - $\pi_{\mathrm{n}} \mathbb{E}_{\mathrm{n}}[\ell(-g(X))] = \mathbb{E}_X[\ell(-g(X))] - \pi_{\mathrm{p}} \mathbb{E}_{\mathrm{p}}[\ell(-g(X))]$
- Thus the risk can be expressed as

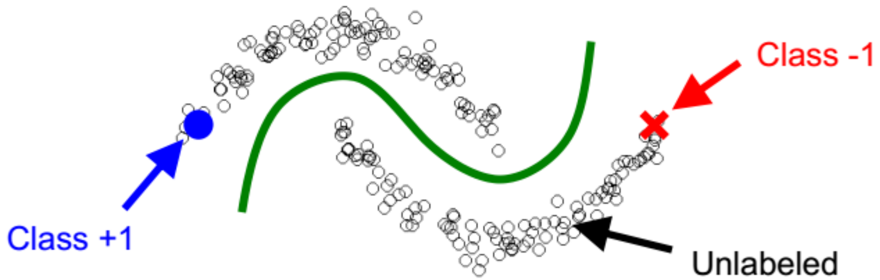
$$R(g) = \pi_{\mathrm{p}} \mathbb{E}_{\mathrm{p}}[\ell(g(X)) - \ell(-g(X))] + \mathbb{E}_X[\ell(-g(X))]$$

- This can be approximated indirectly by

$$\hat{R}_{\mathrm{pu}}(g) = \frac{\pi_{\mathrm{p}}}{n_{\mathrm{p}}} \sum_{x \in \mathcal{X}_{\mathrm{p}}} [\ell(g(x)) - \ell(-g(x))] + \frac{1}{n_{\mathrm{u}}} \sum_{x \in \mathcal{X}_{\mathrm{u}}} \ell(-g(x))$$

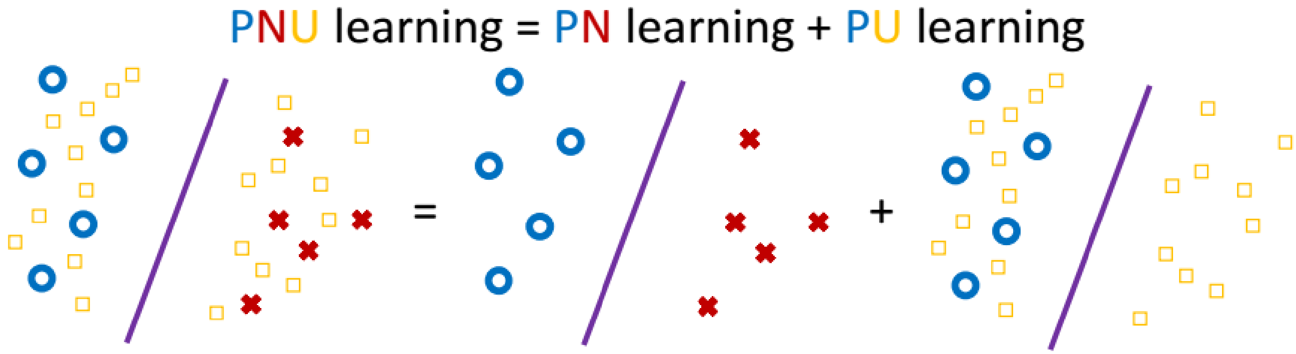
Semi-supervised Classification

- Use a small number of labeled samples and a large number of unlabeled samples:
  $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ $\{\boldsymbol{x}'_i\}_{i=1}^{n'}$
- Find a decision boundary along cluster structure induced by unlabeled samples.
- Cluster assumption rarely holds in practice.



PU learning has no additional distributional assumption

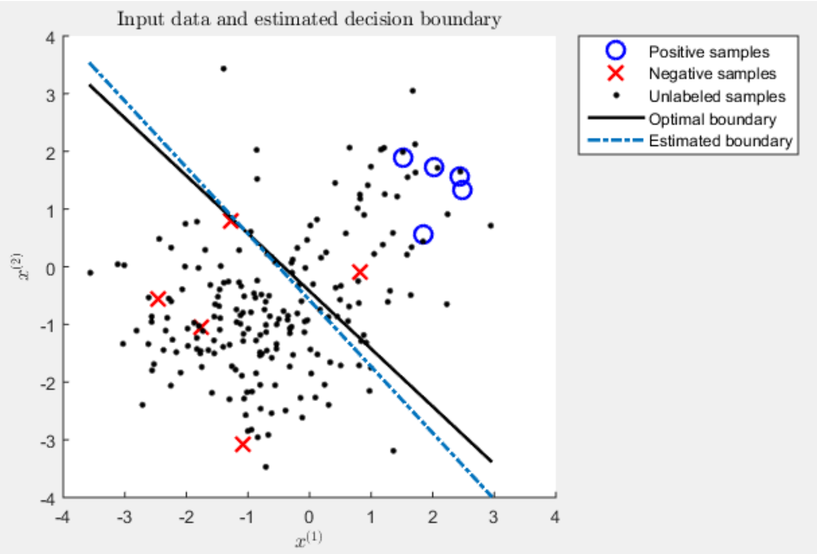
PNU (=PU+PN) Classification



$$R^{\gamma}_{\mathrm{PU+PN}}(f) = \gamma R_{\mathrm{PU}}(f) + (1-\gamma) R_{\mathrm{PN}}(f) \quad 0 \le \gamma \le 1$$

Experiment

Misclassification error rate: Ave (std)



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

ERM in Query Selection

$$\widehat{\mathcal{R}}(f^{+x_k}) = (1-f_k)\,\widehat{\mathcal{R}}(f^{+(x_k,0)}) + f_k\,\widehat{\mathcal{R}}(f^{+(x_k,1)})$$ [Zhu. ICML, 2003]

$$k = \arg\min_{k'} \widehat{\mathcal{R}}(f^{+x_{k'}})$$

$$\hat{R}_{PNU}(f^{+x_k}) = (1-f_k)\hat{R}_{PNU}(f^{+(x_k,-1)}) + f_k\hat{R}_{PNU}(f^{+(x_k,+1)})$$

where

$$\hat{R}_{PNU}(f^{+(x_k,-1)}) = \hat{R}_{PU}(f) + \hat{R}_{PN}(f^{+(x_k,-1)})$$

$$\hat{R}_{PNU}(f^{+(x_k,+1)}) = \hat{R}_{PU}(f^{+(x_k,+1)}) + \hat{R}_{PN}(f^{+(x_k,+1)})$$

Query后只更新集合P或N，而不改变集合U。